



MINISTÉRIO DA EDUCAÇÃO
Secretaria de Educação Profissional e Tecnológica
Instituto Federal de Educação, Ciência e Tecnologia de Santa
Catarina, *Campus* Florianópolis
Mestrado Profissional em Educação Profissional e Tecnológica em
Rede Nacional (ProfEPT)



**OBSERVATÓRIO DA EXTENSÃO NO INSTITUTO FEDERAL DE SANTA
CATARINA**

Dissertação de Mestrado
Paulo Victor Rebouças Soares

Florianópolis, SC
2025

Paulo Victor Rebouças Soares

**OBSERVATÓRIO DA EXTENSÃO NO INSTITUTO FEDERAL DE SANTA
CATARINA**

Dissertação submetida ao Instituto Federal de Santa Catarina (IFSC) como requisito parcial para obtenção de Diploma de Curso de Pós-Graduação *stricto sensu* Mestrado Profissional em Educação Profissional e Tecnológica em Rede Nacional.

Orientador(a): Prof. Igor Thiago Marques Mendonça, Dr.

Co-orientador(a): Profa. Crislaine Gruber, Dra.

Linha de Pesquisa 2 - Organização e Memórias de Espaços Pedagógicos na Educação Profissional e Tecnológica (EPT).

Macroprojeto 6 - Organização de espaços pedagógicos na EPT.

Florianópolis, SC

2025

Ficha de identificação da obra elaborada pelo autor.

Soares, Paulo Victor Rebouças
Observatório da Extensão no Instituto Federal de Santa
Catarina / Paulo Victor Rebouças Soares; orientação de
Igor Thiago Marques Mendonça; coorientação de Crislaine
Gruber. - Florianópolis, SC, 2025.
365 p.

Dissertação (Mestrado) - Instituto Federal de Santa
Catarina, Câmpus Florianópolis. Mestrado Profissional
em Educação Profissional e Tecnológica em Rede
Nacional (ProfEPT). Departamento Acadêmico de Linguagem,
Tecnologia, Educação e Ciência.
Inclui Referências.

1. Extensão. 2. Educação. 3. Governança de dados.
4. Publicização de dados. I. Marques Mendonça, Igor Thiago.
II. Gruber, Crislaine. III. Instituto Federal de
Santa Catarina. IV. Observatório da Extensão no Instituto
Federal de Santa Catarina.



Paulo Victor Rebouças Soares

Observatório da Extensão no Instituto Federal de Santa Catarina

Dissertação apresentada ao Programa de Pós-Graduação em Educação Profissional e Tecnológica (ProfEPT), do Instituto Federal de Santa Catarina – Câmpus Florianópolis, como requisito parcial para obtenção do título de Mestre em Educação Profissional e Tecnológica.

Aprovada em 27 de março de 2025.

COMISSÃO EXAMINADORA



Documento assinado digitalmente
ED WILSON TAVARES FERREIRA
Data: 01/04/2025 18:39:24-0300
verifique em <https://validar.iti.gov.br>

Prof. Dr. Ed Wilson Tavares Ferreira
Instituto Federal do Mato Grosso - Membro Interno



Documento assinado digitalmente
LUIS HENRIQUE LINDNER
Data: 01/04/2025 11:31:23-0300
verifique em <https://validar.iti.gov.br>

Prof. Dr. Luis Henrique Lindner
Instituto Federal de Santa Catarina - Membro Externo



Documento assinado digitalmente
VALTER VANDER DE OLIVEIRA
Data: 07/04/2025 17:11:51-0300
verifique em <https://validar.iti.gov.br>

Prof. Me. Valter Vander de Oliveira
Instituto Federal de Santa Catarina - Membro Externo



Documento assinado digitalmente
IGOR THIAGO MARQUES MENDONÇA
Data: 07/04/2025 17:17:23-0300
Verifique em <https://validar.itl.gov.br>

.....
Prof. Dr. Igor Thiago Marques Mendonça.
Instituto Federal de Santa Catarina - Orientador



Documento assinado digitalmente
CRISLAINE GRUBER
Data: 07/04/2025 20:21:06-0300
Verifique em <https://validar.itl.gov.br>

.....
Prof^a. Dra. Crislaine Gruber
Instituto Federal de Santa Catarina - Co-orientadora



Paulo Victor Rebouças Soares

Observatório da Extensão

Produto Educacional apresentado ao Programa de Pós-Graduação em Educação Profissional e Tecnológica (ProfEPT), do Instituto Federal de Santa Catarina – Câmpus Florianópolis, como requisito parcial para obtenção do título de Mestre em Educação Profissional e Tecnológica.

Aprovado e validado em 27 de março de 2025.

COMISSÃO EXAMINADORA

Documento assinado digitalmente
 **ED WILSON TAVARES FERREIRA**
Data: 01/04/2025 18:38:27-0300
Verifique em <https://validar.iti.gov.br>

.....

Prof. Dr. Ed Wilson Tavares Ferreira
Instituto Federal do Mato Grosso - Membro Interno

Documento assinado digitalmente
 **LUIS HENRIQUE LINDNER**
Data: 01/04/2025 11:27:20-0300
Verifique em <https://validar.iti.gov.br>

.....

Prof. Dr. Luis Henrique Lindner
Instituto Federal de Santa Catarina - Membro Externo

Documento assinado digitalmente
 **VALTER VANDER DE OLIVEIRA**
Data: 07/04/2025 17:09:49-0300
Verifique em <https://validar.iti.gov.br>

.....

Prof. Me. Valter Vander de Oliveira
Instituto Federal de Santa Catarina - Membro Externo



Documento assinado digitalmente
 **IGOR THIAGO MARQUES MENDONÇA**
Data: 07/11/2025 17:16:19 -0100
verifique em <https://validar.iti.gov.br>

.....

Prof. Dr. Igor Thiago Marques Mendonça.
Instituto Federal de Santa Catarina - Orientador

Documento assinado digitalmente
 **CRISLAINE GRUBER**
Data: 07/04/2025 20:18:45 -0300
verifique em <https://validar.iti.gov.br>

.....

Profª. Dra. Crislaine Gruber
Instituto Federal de Santa Catarina - Co-orientadora

RESUMO

Instituída pela lei nº 11.892/2008, a Rede Federal de Educação Profissional, Científica e Tecnológica é composta por autarquias tendo como o pilar a indissociabilidade do ensino, da pesquisa e da extensão para formação do ser humano e preparação para o mundo do trabalho. A extensão como processo educativo construiu uma ponte entre o mundo acadêmico e a sociedade, na qual o discente apreende a ciência de interesse da sociedade no espaço formal da sala de aula e a compartilha no meio cultural, social e político, além de treinar a prática tecnológica transformadora no meio em que vive. Compelida por um estado interessado na governança de dados e soberania dessas, e incentivada por redes de fórum de discussão, a extensão, enquanto vanguarda da sistematização de suas ações, exige dar visibilidade dessa gestão à sociedade. No entanto, a publicização e transparência dos dados referentes às atividades de extensão no Instituto Federal de Santa Catarina (IFSC) ainda enfrentam desafios. Embora o Sistema Integrado de Gestão de Atividades Acadêmicas (SIGAA) registre informações sobre tais atividades, a extração e visualização desses dados não são intuitivas, dificultando sua comunicação para gestores e sociedade civil. Logo, esta pesquisa visa apresentar o observatório das atividades de extensão no IFSC por meio da divulgação de dados através de um produto educacional do tipo *software*, tendo como base para a sua confecção, explorar o cenário extensionista desse membro da rede federal pela revisão documental, por entrevistas junto à pró-reitoria de extensão e relações externas e gestão da extensão nos câmpus, e na análise de dados públicos de acesso à informação, para então extrair os dados de seu sistema, compilá-los em informações de interesse, em seguida exibí-los em painéis de informações de dados para então divulgá-los através do modelo difusionista da informação, por fim espera-se que como resultado a divulgação atenda às demandas de transparência da sociedade e das entidades de controle do estado para o fortalecimento da extensão no instituto.

Palavras-chave: Extensão; Educação; Governança de dados; Publicização de dados.

ABSTRACT

Established by Law No. 11,892/2008, the federal network of professional, scientific and technological education is composed of autarchies whose pillar is the inseparability of schooling, research and extension for the education of human beings and preparation for the world of labor. The extension as an educational process had built a bridge between the academic world and society, in which students learn science of interest to society in the formal space of the classroom and share it in the cultural, social and political environment, in addition to training transformative technological practice in the environment in which they live. Compelled by a state interested in data governance and their sovereignty, and encouraged by discussion forum networks, the extension, as a vanguard of the systematization of its actions, requires giving visibility to this management to society. However, the publication and transparency of data related to extension activities at the Federal Institute of Santa Catarina (IFSC) still face challenges. Although the Integrated Academic Activities Management System (SIGAA) records information about such activities, extracting and viewing this data is not intuitive, making it difficult to communicate it to managers and civil society. Therefore, this research aims to present the observatory of extension activities at the Federal Institute of Santa Catarina through the publication of data through an educational product observatory software, based on its creation, exploring the extension scenario of this member of the federal network through document review, interviews with the pro-rector of extension and external relations and extension management on campuses, and the analysis of public access data to information, to then extract the data from its academic activities management system, compile them into information of interest, then display them on data information panels and then disseminate them through the information diffusionist model. Finally, it is expected that as a result of the dissemination, it will meet the demands of transparency from society and state control entities to strengthen the extension at the institute.

Palavras-chave: Extension; Education; Data governance; Publication of data.

LISTA DE ABREVIATURAS E SIGLAS

ANPD - Autoridade Nacional de Proteção de Dados
APCN - Avaliação de Propostas de Cursos Novos
API - Interface de Programação de Aplicações
AVA - Ambiente Virtual de Aprendizagem
CAPES - Coordenação de Aperfeiçoamento de Pessoal de Nível Superior
CCGD - Comitê Central de Governança de Dados
CCN - Comitê Nacional de Cibersegurança
CEFET - Centro Federal de Educação Tecnológica
CEPSH - Comitê de Ética em Pesquisa com Seres Humanos
CGD - Coordenadoria de Gestão de Dados do IFSC
CGU - Controladoria Geral da União
CMM - *Capability Maturity Model*
CMMI - *Capability Maturity Model Integration*
CNE - Conselho Nacional de Educação
CEPE - Colegiado de Ensino, Pesquisa e Extensão do IFSC
CEP/CONEP - Comitê de Ética em Pesquisa/Comissão Nacional de Ética em Pesquisa
CONIF - Conselho Nacional de Instituições da Rede Federal de Educação Profissional, Científica e Tecnológica
CONSUP - Conselho Superior do IFSC
DAG - *Directed Acyclic Graph*
DAMA-DMBoK - *Data Management Body of Knowledge*
EJA - Educação de Jovens e Adultos
EPE - Ensino, Pesquisa e Extensão
EPT - Educação Profissional e Tecnológica
EPCT - Educação Profissional, Científica e Tecnológica
ETL - *Extract, Transform and Load*
FORPROEXT - Fórum de Pró-reitores de Extensão das Instituições Públicas de Educação Superior
FUCAPI - Fundação e Centro de Análise, Pesquisa e Inovação Tecnológica
IFAM - Instituto Federal do Amazonas

IFSC - Instituto Federal de Santa Catarina
INEP - Instituto
ISO - *International Organization for Standardization*
LAI - Lei de Acesso à Informação
LGPD - Lei Geral de Proteção de Dados Pessoais
MEC - Ministério da Educação
MPS.BR - Melhoria de Processo do Software Brasileiro
ODA - Objeto Digital de Aprendizagem
PaaS - Platform as a Service
PDCA - *Plan Do Check Act*
PDI - Plano de Desenvolvimento Institucional
PNE - Plano Nacional de Educação
PPA - Plano Plurianual
PPC - Projeto Pedagógico de Curso
PROEX - Pró-reitoria de Extensão e Relações Externas
PROFEPT - Mestrado Profissional em Educação Profissional e Tecnológica
RENEX - Rede Nacional de Extensão
SETEC - Secretaria de Educação Profissional e Tecnológica
SIEX - Sistema de Informação de Extensão
SIGAA - Sistema Integrado de Gestão de Atividades Acadêmicas
SIGProj - Sistema de Informação e Gestão de Projetos
TCLE - Termo de Consentimento Livre e Esclarecido
TCU - Tribunal de Contas da União
TI - Tecnologia da Informação
TIC - Tecnologia da Informação e Comunicação
UI - *User Interface*
UX - *User Experience*
WYSIWYG - *What you see is what you get*

LISTA DE QUADROS

Quadro 1 - Etapas da pesquisa.....	43
Quadro 2 - Exemplos de riscos e como preveni-los.....	50
Quadro 3 - Estratificação de indicadores de extensão provenientes de documentos infraconstitucionais e de fóruns da extensão.....	55
Quadro 4 - Estratificação de indicadores de extensão provenientes de documentos infraconstitucionais, de fóruns da extensão, e de documentos do IFSC.....	63
Quadro 5 - Estratificação de indicadores de extensão provenientes de solicitações da sociedade de dados extensionistas.....	73
Quadro 6 - Estratificação de indicadores de extensão provenientes de documentos infraconstitucionais, de fóruns da extensão, de documentos do IFSC, e de solicitações da sociedade de dados extensionistas.....	75
Quadro 7 - Estratificação de indicadores de extensão provenientes das entrevistas semiestruturadas com a gestão da extensão do IFSC.....	88
Quadro 8 - Estratificação de indicadores de extensão provenientes de documentos infraconstitucionais, de fóruns da extensão, de documentos do IFSC, de solicitações da sociedade de dados extensionistas, e de entrevistas semiestruturadas com a gestão da extensão do IFSC.....	97
Quadro 9 - Documentos que apontam a visão temporal dos dados.....	114
Quadro 10 - Estratificação de indicadores de extensão provenientes de documentos infraconstitucionais, de fóruns da extensão, de documentos do IFSC, de solicitações da sociedade de dados extensionistas, de entrevistas semiestruturadas com a gestão da extensão do IFSC, com apontamentos do grupo focal.....	141

LISTA DE FIGURAS

Figura 1 - Indicadores de extensão propostos pelo TCU.....	54
Figura 2 - Lista de câmpus do IFSC por região em ordem de criação.....	83
Figura 3 - Modelo de desenvolvimento incremental em espiral.....	109
Figura 4 - Modelo de desenvolvimento iterativo incremental em cascata.....	110
Figura 5 - Relacionamento entre tabelas no SIGAA.....	112
Figura 6 - Comandos de agregação em linguagem SQL.....	113
Figura 7 - Página de DAGs do Apache Airflow.....	120
Figura 8 - Grafo do fluxo de dados de atividades de extensão.....	129
Figura 9 - Execução do grafo de curricularização da extensão.....	130
Figura 10 - Metadados de fonte de dados no Looker Studio.....	132
Figura 11 - Página de edição do painel de dados de produções da extensão.....	134
Figura 12 - Sub filtro do filtro de ano.....	135
Figura 13 - Tela de dados gerais do Observatório da Extensão.....	136
Figura 14 - Tela listagem de editais com e sem fomento do Observatório da Extensão.....	149

LISTA DE CÓDIGOS

Código 1 - <i>Script</i> Python de seleção de solicitações de acesso à informação direcionadas ao IFSC entre 2020 e 2024.....	70
Código 2 - <i>Script</i> Python de seleção de solicitações de acesso à informação direcionadas à extensão do IFSC entre 2020 e 2024.....	71
Código 3 - <i>Script</i> Python de conversão de formato de arquivo de .csv para .xlsx.....	72
Código 4 - <i>Script</i> SQL com projeção e seleção.....	113
Código 5 - <i>Script</i> SQL de operação de divisão com group by e having.....	114
Código 6 - <i>Script</i> SQL de seleção de quantitativo do macroindicador de áreas temáticas da extensão por câmpus, ano, CEP e área temática.....	115
Código 7 - <i>Script</i> Python com função PostgresOperator do Apache Airflow de conexão com o banco de dados e chamada de consulta SQL.....	117
Código 8 - <i>Script</i> SQL com limpeza dos dados de CEP para obtenção apenas do valor numérico.....	118
Código 9 - <i>Script</i> Python com estrutura básica da sintaxe da linguagem.....	119
Código 10 - <i>Script</i> Python com mapeamento de DAG do Apache Airflow.....	121
Código 11 - <i>Script</i> Python com Apache Airflow de tratamento de dados com dicionário de palavras e dataframes.....	122
Código 12 - <i>Script</i> Python com Apache Airflow de transformação e tratamento de dados percentuais.....	124
Código 13 - <i>Script</i> Python com Apache Airflow de carregamento de dados em arquivo .csv.....	127
Código 14 - <i>Script</i> Apache Airflow de mapeamento de fluxo de processamento de dados.....	128

SUMÁRIO

1 INTRODUÇÃO.....	1
1.1 TEMA E PROBLEMA DE PESQUISA.....	1
1.2 OBJETIVOS.....	3
1.2.1 Objetivo Geral.....	3
1.2.2 Objetivos Específicos.....	3
1.3 JUSTIFICATIVA.....	4
1.3.1 Relevância dos resultados da pesquisa.....	5
1.4 ADERÊNCIA AO PROGRAMA PROFEPT.....	6
1.5 TRAJETÓRIA DO AUTOR E O TEMA.....	8
1.6 ESTRUTURA DO TRABALHO.....	11
2 FUNDAMENTAÇÃO TEÓRICA.....	13
2.1 RELAÇÃO EDUCAÇÃO, TRABALHO E TECNOLOGIA.....	13
2.2 A EXTENSÃO NA EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA.....	16
2.3 PUBLICIZAÇÃO DE DADOS.....	23
2.3.1 Escola e as tecnologias da informação e comunicação.....	23
2.3.2 Democracia e governança de dados.....	28
2.3.3 Modelo difusionista de informação.....	38
3 PROCEDIMENTOS METODOLÓGICOS.....	40
3.1 CARACTERIZAÇÃO DA PESQUISA.....	40
3.2 PARTICIPANTES.....	42
3.3 DETALHAMENTO DAS ETAPAS DA PESQUISA.....	42
3.3.1 Etapa 1 - Identificação das informações da extensão de interesse da gestão do IFSC e da sociedade civil.....	43
3.3.2 Etapa 2 - Recuperação dos dados da extensão e tradução para uma estrutura de dados comunicável.....	44
3.3.3 Etapa 3 - Construção dos painéis de observação em software de publicação dos dados e informações.....	44
3.3.4 Etapa 4 - Divulgação dos painéis de observação em sítio institucional.....	45
3.3.5 Etapa 5 - Avaliação do produto educacional.....	45
3.4 LOCAL E PERÍODO.....	45
3.5 CRITÉRIOS DE INCLUSÃO.....	46
3.6 CRITÉRIOS DE EXCLUSÃO.....	46
3.7 PROCEDIMENTOS E INSTRUMENTOS PARA A COLETA E GERAÇÃO DE DADOS.....	46
3.8 PROCEDIMENTOS PARA A ANÁLISE DOS DADOS.....	48
3.9 ASPECTOS ÉTICOS.....	49
3.9.1 Riscos.....	50
3.9.2 Benefícios.....	51

4 INDICADORES DE EXTENSÃO.....	52
4.1 REVISÃO DOCUMENTAL.....	52
4.2 SOLICITAÇÕES DE ACESSO À INFORMAÇÃO DA SOCIEDADE CIVIL.....	69
4.3 ENTREVISTAS COM A GESTÃO DA EXTENSÃO.....	82
5 PRODUTO EDUCACIONAL.....	106
5.1 CARACTERIZAÇÃO E FINALIDADE DO PRODUTO EDUCACIONAL.....	106
5.2 ELABORAÇÃO DO PRODUTO EDUCACIONAL.....	106
5.2.1 Extração dos indicadores de extensão.....	110
5.2.2 Tratamento de dados.....	117
5.2.3 Carregamento de dados.....	125
5.2.4 Exibição de dados e aplicação do produto educacional.....	130
5.3 AVALIAÇÃO E VALIDAÇÃO DO PRODUTO EDUCACIONAL.....	137
6 CONSIDERAÇÕES FINAIS.....	150
REFERÊNCIAS.....	154
APÊNDICE A – CARTA DE ANUÊNCIA PARA REALIZAÇÃO DA PESQUISA....	168
APÊNDICE B – TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO (TCLE) INDIVIDUAL.....	168
APÊNDICE C – TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO (TCLE) DE GRUPO FOCAL.....	171
APÊNDICE D – TERMO DE CONFIDENCIALIDADE E SIGILO.....	175
APÊNDICE E – ROTEIRO DE ENTREVISTA COM GESTORES DA EXTENSÃO	177
ANEXO A - LISTA DE SCRIPTS SQL PARA EXTRAÇÃO DOS INDICADORES DE EXTENSÃO DO SIGAA.....	178
ANEXO B - LISTA DE SCRIPTS PYTHON COM APACHE AIRFLOW PARA EXTRAÇÃO, TRATAMENTO E CARREGAMENTO DOS DADOS DA EXTENSÃO...	217

1 INTRODUÇÃO

1.1 TEMA E PROBLEMA DE PESQUISA

No Brasil, o início da educação profissional e tecnológica se dá com a necessidade de dar assistência aos mais desfavorecidos, aqueles considerados que não praticavam ações em prol da sociedade. Para isso, D. João VI criou o Colégio das Fábricas em 1809 com foco no ensino das primeiras letras e iniciação em ofícios. Um século depois, sob a mesma herança de atender os mais necessitados, o presidente Nilo Peçanha cria a Escola de Aprendizes Artífices e do Ensino Agrícola com vista na necessidade de mão de obra para os empreendimentos fabris e agrícolas (Ramos, 2014).

Na década de 1930 há um movimento dialético na organização legislativa da educação no Brasil: movimentos de segregação e incorporação da educação profissional e tecnológica nas escolas. Isso resultou em uma cisão entre ensino privado para a elite comandante e público para os operários executores. Apenas em 1959 surge a Lei nº 3.552 que estabelece o surgimento das escolas técnicas federais que futuramente viriam a receber o nome de Centros Federais de Educação Tecnológica (CEFETs) (Brasil, 1978) que após o movimento neoliberal e os embates ideológicos contra-hegemônicos nas décadas de 80 e 90 recebem o nome de Institutos Federais de Educação, Ciência e Tecnologia (IFs) em 2008 pela lei nº 11.892 (Ramos, 2014).

Os Institutos Federais pautam suas atividades no pilar indissociável: ensino, pesquisa e extensão (EPE). O ensino é constituído de várias técnicas pedagógicas onde se dá no espaço formal da sala de aula, na qual os discentes apreendem e refletem sobre o conhecimento científico acumulado pela humanidade ao longo da história. A Pesquisa tem por objetivo a geração de conhecimento; o discente utiliza-se de técnicas mediadas por um orientador para melhorar um processo, uma técnica e/ou um conhecimento científico. Por fim, a extensão promove no discente a externalização do conhecimento gerado para a sociedade e a realidade regional, ela incentiva o discente a conhecer a realidade em que está inserido e em como seu papel como trabalhador pode ser transformador no meio econômico, social e político

em que vive (IFSC, 2020). O pilar se resume que

A educação baseia-se na relação entre estas três áreas: ensino, pesquisa e extensão. No ensino, inter-relacionam-se os diferentes saberes, na pesquisa eleva-se o conhecimento a novos patamares do saber e, na extensão, compartilham-se conhecimentos com a sociedade, contribuindo, dessa forma, para o cumprimento da missão institucional (IFSC, 2020).

A extensão tem o papel de integrar o discente ao mundo do trabalho e intensificar a empatia social no seu entorno. Ela divide-se em várias ações ou atividades: programas, projetos, cursos, eventos, produtos ou prestação de serviços (FORPROEXT, 2012). Também em diversas áreas ou eixos tecnológicos do conhecimento: comunicação, cultura, direitos humanos e justiça, educação, meio ambiente, trabalho, saúde e tecnologia e produção (Filho, F. 2022). Depende de investimentos e insumos para sua realização, todavia, para que a extensão continue a ser materializada é necessário que os resultados já alcançados sejam divulgados. A extensão surge formalmente no cenário de educação profissional e tecnológica nos cursos de nível superior. No âmbito brasileiro esta é regulada pelo conselho nacional de educação (CNE) na sua resolução nº 7, de 18 de dezembro de 2018 que prevê a sua vinculação em planos de desenvolvimento institucionais (PDIs), em planos pedagógicos institucionais (PPIs) e em projetos pedagógicos de cursos (PPCs).

Por vezes, as atividades de extensão materializadas em um câmpus não são percebidas por outro, quanto mais para organizações e pessoas externas ao instituto federal. Para tanto, as tecnologias da informação e comunicação podem se tornar um meio facilitador para preencher essa lacuna. Através dessa, emerge a possibilidade de dar ciência aos interessados sobre as atividades de extensão ao publicar informações e dados em um ambiente informatizado acessível. Assim como as demais formas de comunicação tem sua estrutura lógica e encadeada de palavras, sons e sinais para transmitir uma mensagem, a comunicação em meio digital depende de dados esses que devem ser claros e concisos, por sua vez publicizados como forma de transmissão de uma mensagem, seja para tomada de decisão da gestão ou para prestação de contas para a sociedade.

Considerando isso, o objeto de pesquisa deste trabalho se dá no Instituto Federal de Santa Catarina (IFSC) que tem por gestora das ações de extensão a

pró-reitoria de extensão e relações externas que articula as diretorias de ensino, pesquisa e extensão espalhadas pelos 22 câmpus de realidades distintas ao longo das regiões do estado catarinense. É perceptível a presença da extensão no IFSC nos planos pedagógico institucional, projetos pedagógicos de curso, os planos plurianuais da instituição (PPAs), planos de desenvolvimento institucional (PDIs), normativas, relatórios de gestão, etc. Embora haja uma diversidade de documentos que orientam as atividades de extensão, o instituto utiliza-se do sistema integrado de gestão de atividades acadêmica (SIGAA) como ferramenta facilitadora no fluxo de realização das ações de extensão e no registro de dados sobre a materialização dessas nos câmpus.

No entanto, ainda que os dados cadastrados no SIGAA sirvam para embasar os relatórios de gestão, esses não são de fácil extração para a construção do documento, nem mesmo tem uma interface de comunicação visual que transmita à sociedade e aos gestores uma mensagem de fácil compreensão sobre as ações de extensão no IFSC. Portanto, define-se a pergunta para esta pesquisa: como aprimorar a publicização e transparência dos dados da extensão no IFSC, sua comunicação com a sociedade civil e a tomada de decisão dos gestores?

1.2 OBJETIVOS

1.2.1 Objetivo Geral

Apresentar uma solução de publicização de dados por meio de um *software/aplicativo* observatório para exposição do panorama das atividades de extensão do Instituto Federal de Santa Catarina de interesse da sociedade civil e da gestão acadêmica, balizada por leis, decretos, normativas e boas práticas de governança de dados em respeito aos princípios da transparência e privacidade.

1.2.2 Objetivos Específicos

- Identificar as informações da extensão de interesse da gestão do IFSC e da sociedade civil;
- Recuperar os dados da extensão e traduzi-los para uma estrutura de dados comunicável;

- Projetar os painéis de observação em *software* de publicação dos dados e informações;
- Divulgar os painéis de observação;
- Avaliar o produto educacional.

1.3 JUSTIFICATIVA

Nos trabalhos encontrados no repositório do programa ProfEPT não há pesquisas voltadas para conservação histórica do eixo extensão em um Instituto Federal promovida em uma pró-reitoria em um escopo informatizado para publicização dos dados. Outrossim, observam-se pesquisas com propostas de normativas e boas práticas na construção de ações de extensão como em Santos (2020), Frutuoso (2021), Peres (2020), Ramos (A. 2020), Smith (2021), Ferreira (2022) e Coelho (2022).

O Instituto Federal de Santa Catarina (IFSC) comunica a necessidade de melhorias na gestão de dados pelos sistemas informatizados dentro do seu relatório de auditoria interna nº 03/2021 (IFSC, 2021). Ainda que exista uma gestão sobre os dados das ações de extensão - que também demanda melhorias -, o documento cita que há interesse e procura pela sociedade sobre assuntos voltados para a extensão na plataforma de protocolos do governo federal, falaBrasil.

Além disso, em uma análise preliminar observou-se que a forma como as plataformas de dados federais Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (INEP), a Plataforma Nilo Peçanha (PNP) e a plataforma de dados abertos da Controladoria Geral da União (CGU) publicam as informação não revelam com especificidade os dados de ações de extensão na rede federal e no IFSC; O que se apresenta são dados de fomento fundidos com o eixo da pesquisa e indicadores extensionistas de pouca clareza ou incompletude que comuniquem com sociedade civil e o corpo gestor. No sistema interno do IFSC, o SIGAA, há limitações para a extração de mais informações de dados, o sistema atende aos registros das ações de extensão, no entanto, a consulta de informações para a construção e consolidação de relatórios de informações das ações de extensão é feita parcialmente de forma manual e repetitiva, o que torna a atividade meticulosa um labor demorado e passível de erros, por conseguinte, justifica-se que facilitar a extração dessas informações e exibi-las em uma aplicação de observatório

pode oferecer ao gestor maior tempo de análise dos dados para tomada de decisão e à sociedade melhor entendimento do panorama da extensão na instituição.

1.3.1 Relevância dos resultados da pesquisa

Sem a retomada do contexto histórico da construção da extensão no Brasil e no IFSC não é possível ter uma visão crítica voltada para as melhorias na gestão, fomento de novas normativas e fortalecimento da extensão como eixo basilar da formação omnilateral e integrada do ser humano. Segundo Saviani (2007), não é possível dissociar a educação profissional e tecnológica do mundo do trabalho, pois há uma dualidade de ideias na estrutura social vigente que norteiam os passos da sociedade. Não obstante, isso reflete no campo educacional uma herança de escolas burguesas para os intelectuais e a herança da escola de oficinas para os comuns e desvalidos da sorte.

Porquanto há que se fortalecer a extensão enquanto incentivadora da formação do discente emancipado enquanto cidadão contribuinte para o desenvolvimento e empatia social, produtor de força tecnológica e de conhecimento científico no ambiente em que está inserido. Pois para que haja esse movimento, o cidadão deve tomar conhecimento do que ocorre em seu meio, assim a publicização de dados pode servir como ferramenta para a amannualidade¹ do agente transformador (Pinto, 2005).

Espera-se que como resultado da pesquisa os gestores extensionistas e a sociedade civil tenham uma visão mais clara das ações de extensão realizadas no Instituto Federal de Santa Catarina por meio da publicização de dados registrados no sistema informatizado de gestão acadêmico com o desvelamento da realidade histórica, logo também espera-se que isso fortaleça as relações de comunicação do IFSC com o público interessado nas informações, além de aprimorar as tomadas de decisões na organização do espaço pedagógico da extensão em todos os níveis de ensino.

¹ Amanualidade é uma concepção da relação ontológica do ser humano com a realidade utilizando-se de ferramentas materiais ou abstratas que possui “à mão”, ou ao seu alcance (Pinto, 2005).

1.4 ADERÊNCIA AO PROGRAMA PROFEPT

A Área de Ensino tem por direcionamento a pesquisa translacional, que conecta a ciência básica à aplicação prática do conhecimento gerado, práxis. Essa abordagem abrange todos os níveis e modalidades do ensino formal no país, desde a educação infantil à pós-graduação *stricto sensu*, abordando também modalidades em espaços não formais, como ambientes de trabalho, divulgações científicas e artísticas. Busca estabelecer conexões entre os conhecimentos acadêmicos provenientes da pesquisa em educação e ensino, aplicando-os em produtos e práticas educativas que atendam às demandas da sociedade, a nível local, regional e nacional (CAPES, 2019).

A pesquisa em Ensino é considerada estratégica para enfrentar desafios sociais, sendo essencial para promover mudanças na educação propedêutica, e na profissional e tecnológica. Nessa área, especificamente na Educação Profissional e Tecnológica, há a contribuição para o desenvolvimento do país a partir da formação de profissionais preparados para o mundo do trabalho a fim de promoverem melhorias nas condições de vida pessoal e da população em seu entorno (CAPES, 2019).

As ações na área envolvem pesquisa, ensino e extensão, incluindo a criação e avaliação de materiais didáticos, divulgação científica, assessorias a órgãos públicos, programas e políticas educacionais. Dentre estas ressalta-se a Lei nº 11.892, de 29 de dezembro de 2008, que dispõe sobre a criação da Rede Federal de Educação Profissional, Científica e Tecnológica no Brasil, nela destaca-se o propósito de expandir a EPT para as diversas regiões brasileiras de forma integrada, para tanto a coordenação da capilaridade de câmpus requer uma visão acurada de dados e informações acerca da realidade institucional, logo a publicização de dados se faz interessante para identificação de fraquezas a serem sanadas ou oportunidades de fortalecimento pelas tomadas de decisão dos gestores, ademais é possível que por meio da observação de dados os docentes e discentes tenham ciência das atividades extensionistas que ocorrem em seu câmpus e nos demais, possibilitando que essas ensejem parcerias intra e inter institucionais. Portanto, essa pesquisa alinhada ao produto educacional adere à área de Ensino, sobretudo à EPT ao fortalecer o eixo de extensão no IFSC (Brasil, 2008).

A Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), na portaria nº 60, de 20 de março de 2019 no seu art.12 destaca quais características devem ter um trabalho de pesquisa, em suma: relevância, inovação e aplicabilidade na sociedade. Essas características direcionam a debruçar sobre as linhas de pesquisa e os macroprojetos citados no regulamento do programa ProfEpt, e em como alcançar um resultado eficaz e satisfatório.

Dentre esses, para este estudo, destaca-se a linha de pesquisa 2, Organização e Memórias de Espaços Pedagógicos na Educação Profissional e Tecnológica (EPT), em que traz como objetivo a formação integral e significativa do estudante em espaços não formais, além de visar a configuração de processos de organização de espaços pedagógicos. O meio em que isso pode se tornar concreto ou que é posto em prática é no espaço de gestão e governança, em que metodologias de pesquisa são aplicadas para a construção de soluções para o fortalecimento das práticas pedagógicas de contato direto com o estudante.

Dentro dessa linha de pesquisa o que se aplica a esse trabalho é o macroprojeto 6 Organização de Espaços Pedagógicos da EPT, com o intuito de abrigar projetos que trabalham questões relacionadas à organização e planejamento do ensino, pesquisa e extensão. Quanto a organização e publicização de dados, os *softwares/aplicativos* ganham destaque, esses estão amplamente difundidos nas instituições de ensino da EPT.

No entanto, assim como um organograma institucional distribui, define e organiza as responsabilidades de cada setor, um *software/aplicativo* é construído sobre uma estrutura de dados organizada, distribuída e definida sob as necessidades institucionais, portanto, os dados gerados sobre ela tendem a refletir a estrutura organizacional de uma instituição. Por conseguinte, publicar os dados é um ato de cidadania por dar transparência às informações de interesse da sociedade que por consequência tem papel fundamental de engajamento na construção da escola.

Como discriminado no documento da área de ensino da CAPES, o quesito impacto social é um dos requisitos relevantes para a construção de um produto educacional, e este produto informatizado educacional (*software/aplicativo*) pode aprimorar a comunicação e transparência das atividades institucionais no âmbito da extensão com a sociedade interessada, a exemplo os discentes matriculados,

docentes mantenedores de atividades, gestores da extensão, comunidade envolta da instituição, órgãos de controle e análise, assim também visar melhor avaliação e internacionalização do programa de mestrado profissional EPT.

1.5 TRAJETÓRIA DO AUTOR E O TEMA

Neto de caboclos ribeirinhos baré que há gerações levam o trabalho ontológico no seu sentido mais primitivo de transformação da natureza para não somente sua transformação predatória, mas também para sua sustentabilidade, perpetuaram o conhecimento e sintonia com o ambiente natural para seus descendentes. Eu sendo um deles, na posição dialética de neto mais jovem por parte de mãe e mais velho por parte de pai, desde cedo comecei minha formação em contato a floresta amazônica e rios do estado do Amazonas, aos campos e lagos do estado de Roraima, além de cachoeiras e montes da *gran sabana* venezuelana.

Embora amazonense *manao'wara*, minha formação propedêutica se iniciou no estado de Roraima no Centro Educacional Objetivo Macunaíma, uma escola privada, que tinha uma das preocupações a formação cultural e a conscientização social de seus alunos. Além das competências nas matérias ditas científicas, a escola se preocupava em transmitir aos alunos as artes visuais, literárias e musicais. Recordo que antes de conhecer Macunaíma de Mário de Andrade, conheci o herói dos Macuxi, antes de conhecer ritmos populares fui introduzido a Chiquinha Gonzaga. Nesta época tive o primeiro contato com a realidade dos estudantes de escolas públicas, das diferenças existentes entre a dinâmica das escolas. Permaneci até a metade do ensino fundamental.

Em seguida, retornando à Paris dos trópicos, por preocupação de meu pai com minha educação física, um esportista nato, fui matriculado na escola La Salle por ser referência na formação de atletas. No entanto, pela preocupação de uma assistente social e servidora pública, minha mãe, logo fui rematriculado no Centro Educacional Lato Sensu, uma escola privada, conhecida pela rigidez e formação de cidadãos para ocuparem posições intelectuais no mundo do trabalho. Além da preparação para a vida acadêmica e civil, essa escola também não foi negligente na construção da nossa identidade cultural, das aulas de história da região norte, da composição dos seus povos indígenas, do sacrifício de Ajuricaba até a ousadia de

Eduardo Ribeiro, além claro da brasilidade do teatro Amazonas. Nesse mesmo ambiente pude vivenciar o reflexo do sistema hegemônico dentro das escolas, no que tange ao convívio dos discentes observei as segregações impostas aos filhos dos trabalhadores pelos filhos da “burguesia” local.

Em honra ao meu pai, administrador e empreendedor, ao longo desse trajeto propedêutico, tive contato com o sistema ‘S’ onde por quase duas décadas fui treinado nos esportes natação, vôlei e basquete, não obstante também participava com regularidade nos programas de empreendedorismo.

Em um próximo momento, contrariando todas as expectativas familiares e estimulado por uma curiosidade inquietante para saber como os aparelhos digitais funcionam escolhi seguir os estudos no bacharelado em Ciências da Computação em uma faculdade localizada no polo industrial de Manaus: Fundação e Centro de Análise, Pesquisa e Inovação Tecnológica (FUCAPI). Uma formação tortuosa, meu primeiro contato com a formação profissional, com matérias densas e explicadas por quem presumia que o discente já tivesse uma educação tecnológica e profissional, exigia um deslocamento diário de quatro horas até a sala de aula no transporte público. Mas com persistência e teimosia, após seis anos, restei como o último matriculado da turma inicial de quarenta alunos a conseguir o diploma.

Ao longo do curso da faculdade pude conciliar com o trabalho em instituições públicas e privadas, e ao final desse percurso, na fase de conclusão de curso, tive a primeira experiência na docência ao ministrar aulas sobre tecnologia para discentes ingressantes, o que me abriu a oportunidade de ser selecionado a me tornar docente temporário pela FUCAPI. Entretanto, no mesmo momento fui convocado pelo Instituto Federal do Amazonas (IFAM) para ocupar o cargo de Analista de Tecnologia da Informação (TI) no *Campus* Avançado de Manacapuru como gestor de TI.

Apesar de ter estudado sobre empreendedorismo e por Manaus ser um polo industrial, minha inclinação sempre foi para o serviço público e às pessoas. Portanto esse trabalho foi assumido com muita expectativa. No entanto, ser pioneiro em um *campus* avançado exige determinação, iniciar o trabalho sem uma cadeira para sentar nem sede para conduzir aulas, passar algumas noites em claro para que os discentes tivessem um mínimo de estrutura digna para uma formação de qualidade, fiscalizar obras e em quatro anos acompanhar a entrega de um *campus* para a comunidade é cansativo e gratificante, pois essa labuta me deixou mais capaz para

o mundo do trabalho.

Em paralelo a essa jornada de gestor pude experienciar mais uma vez a docência em programas profissionalizantes no ensino de jovens e adultos (EJA): Mediotec² e Pronatec³. Ambos deram bons frutos, das turmas do Pronatec surgiu o projeto de extensão de pesquisadores adjuntos de tecnologias java em que presenciei a extensão como princípio formativo de maneira mais latente, e dessa observação surgiu minha monografia de especialização em engenharia de software. Pelo Pronatec vivenciei, parcialmente, a aprendizagem flexibilizada onde parte da carga horária das disciplinas eu exercia o papel de docente mediador, precisei de criatividade para elaborar atividades que estimulam a autodisciplina e proatividade em um público jovem. Não menos importante, também conduzi um projeto extensionista independente de treinamento de rúgbi para a comunidade em parceria com a secretaria de esportes de Manacapuru e times de Manaus, esporte o qual vinha praticando por seis anos. Naturalmente o maior fruto foi contribuir para a formação dos discentes.

Após o período no *campus* avançado Manacapuru, por conta de uma demanda familiar, e por decorrência da minha formação acadêmica e experiência profissional com ferramentas de desenvolvimento de software em java, e de ambiente virtual de aprendizagem com moodle, fui redistribuído para o Instituto Federal de Santa Catarina (IFSC) para atender a demanda institucional de integração do sistema de gestão acadêmico com o de ensino a distância. Ao longo dos anos em que estou na rede federal de educação profissional, científica e tecnológica, em conversa com colegas e ao também vivenciar o ensino a distância, notei quão transformador pode ser esta modalidade de ensino se seguida de bons projetos, a exemplo o projeto dos barcos escola nos rios amazônicos, os containers escola, os laboratórios de informática em *campus*, todos com infraestrutura que permita um discente ter um ensino bem estruturado por um corpo docente

² “O MedioTec é uma ação para ofertar cursos de educação profissional técnica de nível médio na forma concomitante para o aluno das redes públicas estaduais e distrital de educação, matriculado no ensino médio regular. As vagas são gratuitas custeadas pela Setec/MEC por meio da Bolsa Formação. Entre seus objetivos, tem a missão de garantir que o estudante do ensino médio, após concluir essa etapa de ensino, esteja apto a se inserir no mundo do trabalho e renda.”

³ “O Programa Nacional de Acesso ao Ensino Técnico e Emprego (Pronatec) foi criado pelo Governo Federal em 2011, por meio da [Lei nº 12.513](#), com a finalidade de ampliar a oferta de cursos de Educação Profissional e Tecnológica (EPT), por meio de programas, projetos e ações de assistência técnica e financeira.”

especializado, e mediado pelas tecnologias informatizadas.

Portanto, essa trajetória pessoal em busca de uma formação integrada, a inclinação para a docência e um interesse no ensino a distância me despertou o interesse em cursar o mestrado profissional em educação profissional e tecnológica. E mesmo que eu não venha a exercer formalmente a docência, tenho a expectativa que o programa promova em mim a construção de uma consciência crítica para o mundo do trabalho e que me permita ser um profissional capaz de capacitar outros em espaços não formais.

1.6 ESTRUTURA DO TRABALHO

Esta pesquisa divide o primeiro capítulo nos elementos introdutórios, o segundo quanto à fundamentação teórica, sendo no primeiro subtópico sobre a relação do ser humano com a tecnologia, o trabalho ontológico e de produção, e a dualidade na educação que provoca a construção de modelos educacionais emancipadores. Por conseguinte, expõe a realidade assistencialista da educação profissional no Brasil e do caráter hegemônico até a criação da rede federal de educação profissional e tecnológica, dando enfoque à extensão enquanto articuladora da escola com a sociedade na vanguarda da organização dos processos de estruturação da gestão e governança escolar e a presença dela no IFSC. O último subtópico trata da relação da escola e as influências das tecnologias de informação e comunicação sob uma sociedade acumuladora flexível; Da relação democrática do estado com a escola na construção de políticas públicas, como a governança de dados é substancial para as discussões; E o papel do modelo difusionista de informação no meio acadêmico e a possibilidade de uso dele para a publicização de dados extensionistas. Ainda o terceiro capítulo apresenta os procedimentos metodológicos aplicados na pesquisa, o quarto aborda a identificação dos indicadores de extensão perpassando pela revisão de documentos infraconstitucionais, de fóruns da extensão e do IFSC, também a exploração de solicitações de acesso à informação de extensão pela sociedade e as entrevistas com gestores extensionistas do IFSC. O quinto capítulo traça a construção do produto educacional para publicização dos indicadores de extensão considerando uma metodologia de desenvolvimento obedecendo às etapas de processamento de

dados de extração, tratamento e carregamento, a construção de painéis para exibição dos dados no Observatório da Extensão e em como esse produto educacional foi avaliado e validado. Por fim, o sexto capítulo discorre sobre as considerações a respeito deste trabalho e das percepções a partir da aplicação do produto.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 RELAÇÃO EDUCAÇÃO, TRABALHO E TECNOLOGIA

Segundo Saviani (2007), não há como citar a relação trabalho-educação sem observar a história em que ela foi construída. Primeiro o autor dialoga com as definições de homem que o capacitam trabalhar e educar, perpassa o argumento aristotélico que o que define a capacidade do homem é a racionalidade, no entanto, essa definição reflete um caráter accidental da relação trabalho-educação e não substancial para a existência desta relação, pois não basta que o homem seja apenas racional para que esteja inclinado a trabalhar e educar, é preciso ser intencional no labor. Em seguida o autor argumenta sobre a definição de Bergson (1979) que a princípio faz uma distinção entre plantas e animais, e consequentemente pela inferência que um homem é um animal, o que o diferencia dos outros de sua classe não é apenas a inteligência, pois sim a fabricação de objetos artificiais.

Saviani (2007, apud Bergson, 1979) reflete que a necessidade de fabricação é instintiva nos primórdios da produção e a posteriori é reflexiva e consciente, capaz de refletir a existência do homem no objeto fabricado. No entanto, para Saviani (2007), essa reflexão não é suficiente para definir a essência humana. Em um terceiro momento o autor, por meio do materialismo histórico-dialético, retorna à base ontológica do homem, com o argumento de que o homem para produzir sua existência necessita transformar a natureza para que sobreviva.

Ao contrário dos animais que se adaptam à natureza, o homem adapta a natureza a si ajustando-a às suas necessidades. Esse ato de transformação é o que caracteriza o trabalho, ademais o desenvolvimento aprofundado e complexo ao longo deste é um processo histórico que resulta em artes e técnicas próprias do indivíduo. O homem por viver coletivamente e por vezes necessitar da colaboração de outros perpetua essas técnicas para os membros de sua comunidade, nesse processo os homens educavam-se e educavam as novas gerações que se apropriavam do conhecimento e o aperfeiçoavam para os seres humanos das gerações seguintes, assim produziam tecnologia. Nessas condições a educação se identifica com a subsistência humana, portanto é veraz a afirmação que a “educação

é vida”, e dessa forma a relação trabalho-educação reflete a essência do homem (Saviani, 2007).

Tecnologia é uma palavra de comum uso no mundo do trabalho, no entanto seu entendimento pode variar de acordo com o contexto. No sentido mais puro, etimológico, a tecnologia enquanto logos da técnica significa o estudo da técnica, que constroi discussões, habilidades, ciência e artes a partir da técnica. Porém, há contextos populares em que a tecnologia se confunde com a técnica quando há uma aplicação de um saber fazer, assim acabam por convergir em sinônimos. A tecnologia assume um contexto mais amplo quando entendida como um conjunto de técnicas que identificam uma sociedade em qualquer momento histórico, seja na observância do desenvolvimento de uma civilização à de um grupo social local ou global, a exemplo os conselhos de classe de trabalho. Por último, a tecnologia implica na ideologia da técnica, a que é apropriada nas forças de produção material que unilateralmente fragmenta a relação homem-objeto, nesse sentido o ser humano deixa de refletir sobre a técnica ou sequer é desprovido do contexto histórico em que esta surgiu (Pinto, 2005).

De forma ilustrativa é saber a razão por que um cozinheiro aplica óleo em uma panela antiaderente, pode ser que a técnica exija aplicar um filete de óleo para criar uma crosta em determinado alimento e atribuir uma textura que estimula as papilas gustativas, porém a razão de fazê-lo talvez seja inconsciente, pois foi lhe ensinado que esta é a maneira “correta” a se fazer para que o alimento não grude na ferramenta, mas isso vale para uma panela que não seja antiaderente, e esse não é o caso dele, logo esse depara-se em um cenário de replicação da técnica de comum aceitação social, de quando esta foi idealizada em um determinado momento histórico em que isso era apropriado para a ferramenta disponível e que não seja mais necessário para ele, no entanto a falta de reflexão crítica e compreensão histórica o exime de questionar e aprimorar sua técnica. Segundo Pinto (2005):

O homem, que por essência está destinado a procurar a natureza, para, sobre ela, se constituir a si mesmo, encontra em lugar dela cada vez mais a obra de outros homens. A perniciosidade desta situação reside não no fato em si mesmo, mas em não se saber interpretá-lo dialeticamente, no curso de um processo objetivo em que a realidade do ser humano se constitui em função da mobilidade dos suportes históricos (Pinto, 2005, pág. 225).

No que tange a relação histórica da humanidade na separação entre trabalho e educação, dado o momento em que surgiu a ideia de propriedade privada,

aparecem as divisões de classe onde os que não tem terra passam a trabalhar para os senhores e donos. Por sua vez, o agir sobre a natureza não reflete necessariamente a subsistência daquele que age sobre ela, mas implica na produção de bens materiais para o dono da terra. Assim o mesmo modelo de venda de força de trabalho para os senhores de propriedades privadas se perdura pelo tempo. Os que não necessitam direcionar sua energia para o trabalho passam a viver do ócio, e desse ócio há a abertura para o exercício do pensamento, consequentemente o surgimento da escola (Saviani, 2007).

Neste cenário também introduz-se assim uma cisão na unidade da educação, na qual a estrutura escolar passa a ser distinta para os que precisam aprender técnicas de trabalho para vender sua mão de obra e para os que vivem do ócio que aprenderão as técnicas de oratória e política para comandar os demais. Apesar das mudanças nas estruturas sociais ao longo do tempo, o modo de produção capitalista se mantém hegemônico, no entanto com os movimentos sociais se articulando o Estado passa a ter protagonismo em provocar mudanças na ideia de escola pública, universal, gratuita, leiga e obrigatória para todos os humanos, uma escola unitária, ainda assim esse novo perfil de escola não deixa de sofrer a influência hegemônica do capital e por sofrer com o sistema vigente sofre uma dualidade. Na realidade, a partir da revolução industrial e com o advento da maquinaria, a escola passa a ter essa dupla identidade, onde há a separação por cursos pré-selecionados para qual função o homem desempenhará na sociedade, seja em um trabalho dito manual, seja um trabalho dito intelectual. Todavia, sendo necessária para que capacite manualmente o homem para o mundo do trabalho e tenha sua subsistência, porém forjado intelectualmente para sua emancipação. No entanto essa dualidade corrobora com a cisão de classes (Saviani, 2007).

Saviani (2007) inspirado em Gramsci (1968) sobre o trabalho como princípio educativo em uma escola unitária, visa traçar uma conformidade deste tipo de ensino com a realidade brasileira. Primeira etapa essencial é a educação básica que transmite ao ser humano o acervo mínimo de conhecimentos sistemáticos que o permitam ser cidadão, ou seja, um participante ativo na sociedade. A segunda etapa é o ensino fundamental com foco no princípio educativo do trabalho, revela assim os conhecimentos de equilíbrio entre ordem social e ordem natural para que entenda seu papel na transformação da natureza. Por fim, a terceira etapa finda com o

ensino médio, tratando de explicitar como o conhecimento científico se converte em potência material no processo produtivo, logo o domínio não deve ser apenas teórico, mas também prático por meio de oficinas nas quais o aluno manipula processos básicos de produção. Este aluno não se limita a receber um mero adestramento em técnicas produtivas, e sim aprender o fundamento de técnicas utilizadas na produção moderna, portanto recebe uma formação politécnica. Isso se opõe ao ensino médio profissionalizante, pois este diferente daquele se propõe a adestrar o aluno nas técnicas sem que tenha o pensamento crítico, reflexivo e esclarecido. Uma vez concluído, cabe ao aluno decidir aplicar sua politécnica no mundo do trabalho sem que caia na passividade intelectual ou se especializar em uma universidade no mundo acadêmico. Independente da escolha, a formação politécnica implica em que o egresso seja um trabalhador mais engajado para a vida cidadã ativa.

2.2 A EXTENSÃO NA EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA

No início do período imperial há a necessidade de desenvolvimento da colônia para a profissionalização como meio de atender um problema social: possibilitar autonomia para os desvalidos, que ali se enquadravam os mais pobres sem condições de sustento, os órfãos, os mendigos e desocupados que não estavam alienados ao interesse do estado. Em contrapartida os filhos dos mais afortunados eram educados no exterior, principalmente na França iluminista, ou tinham o atendimento em casa de professores particulares, e para esses o império proveu a colônia de faculdades para a formação da elite do funcionalismo público da época (Nascimento, J. 2020; UNIVESP, 2010).

Da chegada da família real portuguesa no início do século XIX até 1888 (império brasileiro) o desenvolvimento dos primórdios de uma educação profissional andava paralela a um sistema escravista que teve seu enfraquecimento a partir de 1866 quando último navio de tráfico aportou, em 1871 com a promulgação da lei do ventre livre e em 1888 com a abolição do sistema de escravidão. Nesse meio tempo a relação dos senhores de engenho e os nascidos livres e alforriados muda, pois o então filho de escravo passa a vender a sua mão de obra em troca de salário, e o liberto sem amparo fica a mercê das instituições assistencialistas, muitas vezes religiosas, que davam abrigo e educação profissional. No entanto, sob influência do

racismo científico e as políticas sanitaristas os problemas sociais de segregação se agravaram, somado ao contexto histórico houve uma forte associação cultural da formação em trabalhos manuais à escravidão e a formação nas faculdades às elites (Nascimento, J. 2020).

Para fortalecer a nação, concomitante às atividades das instituições religiosas, as forças militares, sobretudo a marinha, criaram os centros de aprendizagens de ofícios para formação de operários especializados em técnicas de navegação, construção e manutenção do patrimônio militar. Com incentivo financeiro do império esses centros expandiram o alcance e a formação de artífices competentes em ferraria, serralheria, tanoaria e funilaria. Além das instituições assistencialistas e militares, Nascimento (J., 2020) cita instituições voltadas para as demandas do comércio como a Real Impressão (1808), o colégio de Fábrica (1809), a Fábrica de Lapidação de Diamantes (1812) e a Fábrica de Tecidos (1814).

Inspiradas nos colégios de fábricas (1809 a 1812), as casas de educandos artífices (1840 a 1865) foram as primeiras a ensaiar um modelo de educação profissionalizante nacional. Embora descentralizadas, serviram de modelo para instituições futuras, nelas os educandos tinham aulas de nível primário: letras, escultura, desenho, aritmética, noções de álgebra, geometria e mecânica aplicada às artes; Somadas ao ensino de um ofício, sob uma estrutura de rígida de hierarquia e disciplina na lógica militar. Em seguida, com difusão nacional semelhante, são criados os liceus de artes e ofícios (1858) mantidos por sociedades beneméritas para o ensino das artes manufatureiras, com oferta inovadora para público de ambos os sexos, jovens aprendiam ofícios mecânicos de utilidade prática para labor fabril, porém com segregação de cursos de acordo com o sexo do(a) ingressante (Nascimento, J. 2020).

Somente em 1879 com a reforma educacional proposta por Carlos Leôncio de Carvalho que o ensino profissionalizante no Brasil amplia o teor assistencialista das instituições, a reforma propõe a criação de escolas profissionais mantidas pelo estado em cada província, sem distinção de condição social, para o aprendizado das primeiras ciências e para a instrução técnica voltada para as fábricas ou ao ensino prático de proveito da população e o Estado de acordo com a localidade. Por meio dessa reforma, e da visita do imperador D. Pedro II às regiões carentes do nordeste,

criam-se as escolas agrícolas técnicas como alternativa para solucionar problemas de atrasos tecnológicos na produção agrícola (Nascimento, J. 2020).

No primeiro período da república (1889-1930) a oligarquia cafeeira do sudeste, sobretudo a paulista, detinha o poder e influência nos rumos do país em prol dos seus interesses. Havia um interesse de industrialização da região para então expansão para as demais. Todavia, por conta do modelo econômico agrícola de exportação e importação de produtos o Brasil se alia às indústrias estrangeiras para a instalação de fábricas em território nacional, assim esse cenário emergente impele em 1909 a abertura da escola de aprendizes e artífices para suprir a demanda quanto assistir aos desvalidos - dentre eles migrantes de outras regiões e alforriados - que aos poucos se instalavam nas margens das cidades formando favelas por falta de oportunidades de trabalho como meio de subsistência (Castro, 2020; Souza, F. 2022).

No mesmo interstício a república, por influência de ideias positivistas e pelo racismo científico, se vale da condição de pobreza de nações europeias e traz para o Brasil, sob promessas de incentivos, estrangeiros para ocupação do território no sudeste e sul, e para a ocupação nas fábricas nacionais para difundir o conhecimento adquirido nos seus países de origem. Por consequência as escolas de artífices deixaram de atender as necessidades das fábricas da oligarquia, e pararam de receber incentivo para que funcionassem com instalações adequadas, mestres capacitados e instrumentos para o ensino (Castro, 2020).

Em meados da década de 1920 o capitalismo tendia a implodir com seu pico na crise mundial de 1929, o que fomentou as primeiras manifestações populares pelos mesmos estrangeiros das fábricas que tinham viés anarquista, assim surgem os primeiros sindicatos no Brasil. No ano seguinte acontece a revolução de 1930 quando Getúlio Vargas sobe ao poder com um governo provisório e logo após o toma com uma proposta de formar um estado novo (1937-1945, segunda e terceira fase da república), dentre seus apoiadores Anísio Teixeira lidera o manifesto dos pioneiros e o movimento da escola nova, também assistencialista, inserindo a educação como uma pauta no debate público culminando na criação do Ministério da Educação e Saúde Pública e do Trabalho, e do Conselho Nacional de Educação, pois propusera que a escola fosse pública, de acesso a todas as classes sociais, laica e única sob responsabilidade do estado. Com essa visão acreditavam que

assim fosse possível formar cidadãos para uma nação moral e economicamente sólida (Castro, 2020 apud Saviani, 2013).

O movimento não teve a adesão de todos, a oligarquia e a igreja (então detentora da educação) ainda mantinham seus interesses de ordem corporativista e/ou cultural, portanto a segregação entre educação para os filhos dos mais abastados e para o povo se mantém presente. Nesse contexto, a reforma Capanema (1942-1943) propôs a adequação dos interesses econômicos, sociais e políticos da época à educação nacional, resultando na criação do sistema S, na mesma linha o estado estabelece o conceito de menor aprendiz sob os efeitos das leis trabalhistas, e pelo Decreto-Lei nº 4.127, de 25 de fevereiro de 1942 constitui a rede federal de estabelecimentos de ensino industrial, resgatando as escolas de aprendizes e artífices transformando-as em Escolas Técnicas ou Escolas Industriais (Castro, 2020 apud Schenkel, 2012; Souza F. 2022).

Nos anos que se seguiram, de 1945 a 1964, o Brasil entra na quarta fase da república, uma pós segunda guerra mundial em que os opositores políticos se unem para alavancar a economia nacional, o governo de Juscelino Kubitschek expande para além do sudeste o parque industrial nacional e inicia a interligação entre as regiões brasileiras, a capital é transferida para o centro-oeste, há um grande êxodo da área rural para as cidades, e o cidadão se interessa pelo ensino profissional superior com expectativa de crescimento financeiro para uma melhor qualidade de vida (Castro, 2020).

Nas fases da ditadura militar (1964-1985) há o surgimento da lei nº 5.692, de 11 de agosto de 1971 que estipula a obrigatoriedade do 2º grau profissional, hoje chamado de ensino médio. Apesar de se destacar como uma boa política, essa lei é fruto da reforma da lei nº 5.540 de 1968 que retirou das escolas técnicas federais a obrigatoriedade do ensino profissional e repassou para os sistemas estaduais a ofertá-la, o que fez com que esses sistemas já fragilizados pela precariedade em atender o ensino de primeiro grau tivessem mais desafios, tal decisão reflete por décadas a fragilidade do ensino público nos municípios e estados. Ainda o estado em 1982 promulga a lei nº 7.044 que torna facultativa a profissionalização no 2º grau (ensino médio) nas escolas públicas, e reverter os currículos para o ensino acadêmico, fragilizando o ensino profissional e técnico no Brasil (Castro, 2020).

Porém, graças a lei nº 6.545, de 30 de junho de 1978 as escolas técnicas federais passam a ter autonomia de autarquia e as escolas de Minas Gerais, Rio de Janeiro e Paraná se tornam Centros Federais de Educação Tecnológica (Cefets) (Brasil, 1978), com isso a educação profissional tem uma sobrevida até a expansão da rede federal durante a república nova e a transformação deles em institutos federais de educação ciência e tecnologia pela lei nº 11.892, de 29 de dezembro de 2008 (Brasil, 2008).

Embora a educação profissional fora incipiente até a redemocratização do Brasil, ela toma uma forma humanística com o promulgar da lei nº 11.892 que institui a rede federal de EPT ao caracterizar as finalidades e objetivos dos institutos federais por desenvolver programas de extensão e divulgação científica e tecnológica, de acordo com os princípios e finalidades da EPT, em articulação com o mundo do trabalho e os segmentos sociais (Brasil, 2008).

No entanto, a origem da extensão no Brasil não surge desse marco, ela provém do Estatuto da Universidade Brasileira, no Decreto-Lei nº 19.851 de 1931 com o entendimento herdado de ação assistencialista sob as modalidades de curso, conferência e assistência rural, com um aspecto de comunicação unilateral para com a sociedade, tornando-se distante dessa, e do progresso da pesquisa e transmissão do ensino. Esse entendimento da extensão permanece por décadas à frente, a exemplo na década de 60, no período do governo militar, é iniciado o projeto Rondon, com o objetivo de colocar os estudantes universitários a serviço dos interesses desenvolvimentista e tecnicista do Estado. A universidade materializava ações pré-definidas propostas pelo governo, ausentes das discussões problemáticas dos contextos sociais. (FORPROEXT, 2006).

O projeto Rondon foi retomado em 2005 coordenado pelo ministério da defesa com outra roupagem, com o objetivo de contribuir na formação cidadã, integrar o discente à sociedade por ações participativas, estimular no formando a produção, em parceria com a comunidade, de projetos coletivos locais, para construção do sentido de responsabilidade social e coletiva (Rodrigues, 2016).

No Brasil, a extensão tem influência de duas correntes, a estadunidense voltada aos interesses comerciais e empresariais na forma de prestações de serviços que tragam retorno ao mercado de trabalho; E a corrente europeia com

uma herança mais assistencialista, com viés de formação continuada de adultos, por conta da pobreza gerada a partir da Revolução Industrial. Sob essas, o resultado dos debates dos fóruns em volta da extensão tendeu a desempenhar um papel de superação dessas linhas extensionistas para promover a interação entre saberes científicos e populares para atender às demandas sociais, econômicas e ambientais buscando fortalecer a cidadania e estimular o protagonismo social no mundo do trabalho (Filho, 2022).

A Extensão Universitária é guiada por cinco princípios fundamentais, conhecidos como os "5 Is": interação dialógica, indissociabilidade entre ensino, pesquisa e extensão, interdisciplinaridade e interprofissionalidade, impacto na formação do estudante, e impacto e transformação social. Sua prática exige um diálogo constante entre universidade e sociedade, baseado na troca de saberes e na reciprocidade. Essa abordagem interdisciplinar enriquece as ações, ao reunir diferentes áreas e formações, promovendo a integração entre ensino, pesquisa e extensão. Além disso, as ações de extensão organizam-se em três eixos principais: territórios, grupos populacionais e áreas temáticas, contribuindo para soluções inovadoras em diversos campos. A formação estudantil é um pilar essencial da extensão, sendo indispensável para justificar sua execução na forma de programas, projetos, cursos, eventos e prestações de serviço, pois garante aprendizado com base na práxis e forma cidadãos conscientes e profissionais capacitados (CNE/CES, 2018; Filho, 2022)

Segundo Filho (2022) a extensão é um esforço contínuo conectado às demandas sociais que dialoga com os setores produtivos desde o campo às indústrias. Nessa transformação dupla, a escola enquanto agente extensionista se transforma enquanto propicia o desenvolvimento regional e social, além de estimular o aprimoramento de políticas públicas. Sob essa perspectiva motriz, no período da transição democrática foi criado o fórum de pró-reitores de extensão das instituições públicas de educação superior brasileiras em 1987 - FORPROEXT -, organizado em coordenações envoltas nas áreas temáticas de cultura, comunicação, direitos humanos e justiça, educação, meio ambiente, tecnologia, e produção, trabalho e saúde. A partir de então iniciaram-se as discussões sobre a extensão na educação brasileira que estreitaram o diálogo com o ministério da educação para a implementação de novos conceitos na comunidade acadêmica. No seu regimento o

fórum estipulou como objetivo a proposição de políticas e diretrizes básicas para a institucionalização, articulação e fortalecimento de ações comuns às instituições de extensão, desenvolvimento de práticas integradoras às atividades acadêmicas, diálogo permanente com instituições da sociedade civil, do setor produtivo e dos poderes constituídos com vista à ampliação da inserção social das instituições públicas, incentivo ao desenvolvimento da informação, avaliação, gestão e divulgação das ações da rede nacional de extensão - RENEX (Rodrigues, 2016).

Nesse contexto, as ideias basilares dos institutos federais são forjadas e/ou fortalecidas: democratização do conhecimento acadêmico, instrumentalização da práxis, promoção da interdisciplinaridade, participação efetiva da comunidade, visão integral do social, relação transformadora entre meio acadêmico e sociedade. A extensão busca romper o assistencialismo estrutural através das ideias freireanas de diálogo com a comunidade, a comunicação deixa de ser unilateral, há uma via de mão dupla da academia aprendendo com a comunidade, estabelecendo uma interlocução do conhecimento comum transladando-o para o conhecimento científico e esse retornando à sociedade com atores transformadores do seu meio. Dessa relação é endossada a indissociabilidade ensino, pesquisa e extensão, através de um currículo flexível coerente com o projeto pedagógico de curso, aquele que transversalmente, por meio de atividades as quais o discente tem afinidade e exprime o desejo de participação, compõem as suas competências, convergindo a formação do mesmo para uma atuação preparada para a atuação profissional e cidadã, conferindo uma visão transformadora e comprometida. Sob esses argumentos o FORPROEXT consegue galgar a institucionalização da extensão como elemento chave nas instituições federais de ensino (Frutuoso, 2020; Rodrigues, 2016; FORPROEXT, 2006).

Como ferramentas de consolidação da extensão e enalço para a rede federal, o FORPROEXT, a partir de 2003, apoiado pelo MEC, em parceria com a Universidade Federal de Minas Gerais - UFMG, implementa e mantém o *software* Sistema de Informação de Extensão - SIEX para a disseminação da utilização deste para as instituições que compõem o fórum, tornando-se também um guia padronizado na gestão das ações de extensão pelo Brasil, nota-se nele a segmentação de modalidades de atividades: curso, evento, produto, programa, prestação de serviços e projeto. Ademais é possível observar a posição de

vanguarda na promoção da gestão de atividades e transparência na sistematização das ações extensionistas, o que impulsionou a pesquisa e o ensino a também informatizarem suas atividades no sucessor do SIEX: o Sistema de Informação e Gestão de Projetos (SIGProj), este projeto se sustentou como um *software* livre agregando a participação de uma comunidade de técnicos administrativos, pesquisadores e discentes das universidades brasileiras no seu desenvolvimento, contribuindo para a divulgação e democratização de dados e informações para a comunidade externa (Rodrigues, 2016; FORPROEXT, 2006).

No Instituto Federal de Santa Catarina a influência do trabalho do FORPROEXT é visível na regulamentação das atividades de extensão na resolução consup nº 61 de 2016, define-a como um processo educativo, cultural, político, social, científico e tecnológico, indissociável do ensino e da pesquisa, para a promoção da interação entre o IFSC e a sociedade. Estabelece que suas atividades tenham por alvo a transformação social, o desenvolvimento regional sustentável e integral dos discentes como executores ou colaboradores de programas, projetos, cursos, eventos, produtos e serviços. Ainda regula a proposição, aprovação, registro, execução, relato e certificação das atividades, e determina diretrizes para o fomento financeiro para essas. Não obstante essa estrutura é gerida pela pró-reitoria de extensão e relações externas (PROEX) que orchestra as atividades junto às coordenações de extensão dos 22 câmpus no estado. Por fim, o documento prevê que os trâmites para a materialização das atividades de extensão devem ser registradas no Sistema Integrado de Gestão e Atividades Acadêmicas do IFSC (SIGAA) (IFSC, 2016).

2.3 PUBLICIZAÇÃO DE DADOS

2.3.1 Escola e as tecnologias da informação e comunicação

Diacrônico (2022) citando Harvey (2010), um geógrafo e sociólogo estadunidense, o regime de acumulação flexível, também conhecido como Toyotismo, é uma dinâmica aprimorada do Fordismo- Keynesianismo de produção, circulação e acumulação no sistema hegemônico capitalista do período pós-moderno. O Fordismo caracteriza-se por ser um sistema de produção rígido,

com linhas de produção específicas, com papéis definidos e um foco na durabilidade do produto, já o Keynesianismo é uma teoria econômica que defende a intervenção do estado na regulação do mercado por meio de políticas para alcançar o equilíbrio econômico e pleno emprego. Naturalmente esse regime influencia os modos de consumo, as relações sociais e as regulamentações políticas do estado fazem com que esse tenha uma postura neoliberal.

Em razão da implosão do regime de produção rígido da época vigente, o regime de acumulação flexível adotou como estratégias o avanço tecnológico célere, automação das esteiras de produção, criação de linhas de produto obsoletas, nichos de público consumidores específicos, fusões empresariais, dispersão geográfica de filiais a fim de baratear a mão de obra onde não havia uma regulamentação forte do trabalho, e aceleração de giro do capital. Isso impacta em novas dinâmicas de trabalho, uma vez que os trabalhadores capacitados em uma atividade agora se adequam em exercerem múltiplas funções ou seriam dispensados causando desemprego estrutural e crescimento do exército de reserva principalmente em regiões periféricas do globo, consequentemente o trabalhador sem uma identidade específica ou desempregado implica numa dispersão de classes sindicais e uma competitividade exacerbada por treinamentos de curta duração e formação rasa do ser humano (Harvey, 2010).

Por conta desse movimento o exército de reserva acaba fortalecendo o setor de serviços onde se terceirizam etapas da produção fabril, da extração da matéria, da manutenção de parque tecnológico até a mão de obra para as esteiras que passa a ser temporária de acordo com a demanda do mercado, livre de contratos bem estruturados e sem organização que defenda os interesses do trabalhador. Tratando-se de infraestrutura fabril a acumulação flexível impulsiona o avanço tecnológico criando-se pólos de desenvolvimento pelo mundo, sobretudo o vale do silício nos Estados Unidos da América, graças a isso ocorre o fenômeno de compressão do tempo-espço devido ao surgimento de tecnologias de telecomunicações capazes de diminuir o tempo de troca de informações entre espaços geograficamente distantes uns dos outros em um intervalo cada vez menor trazendo uma dinâmica de mudanças céleres das lógicas no mundo do trabalho e nas relações superficiais estético-culturais interpessoais (Harvey, 2010).

A respeito deste cenário Filho (J. 2020) expõe a entrevista da globo news com Zygmunt Bauman, um sociólogo polonês então professor em Londres, nela o professor explana sobre sua vivência ao longo desse período de transformações do século XX. Para ele o início do século XIX foi um momento de grandes expectativas e esperanças, o avanço tecnológico trazia em si a visão que a humanidade poderia atingir um estado de plenitude e satisfação na vida, e que o terreno dos embates políticos alcançaria um patamar menos belicoso, mais democrático em prol da evolução social. No entanto, o sociólogo define o início do século XXI como um momento interregno, pois a sociedade não é mais sólida, não caminha em conjunto, está mais diluída, como um líquido que se desloca conforme as formas em que se encontra com interesses diversos, com ideias imediatistas e rasas. A sociedade atual está mais acostumada em lidar com crises mais recentes, porém as crises também são diferentes, evoluem mais rápido e de forma mais imprevisível, pois a humanidade está em desordem.

A respeito da evolução das telecomunicações ele comenta que “estamos nos afogando em informações e famintos por sabedoria”, o homem pós-moderno não tem tempo de raciocinar sobre as informações que lhe são entregues, ele as usa de imediato e logo lhe são entregues mais informações. A sabedoria apresenta como podemos prosseguir, no entanto, para que se a adquira requer tempo de qualidade, para aprender, pesquisar e refletir até que se esgote o assunto e exercitá-lo com consistência para que seja apreendido e usado para o debate de ideias no mundo do trabalho. Ainda sobre o tema, Zygmunt comenta que essa situação criada pelo neoliberalismo e a acumulação flexível construiu uma cultura consumista aceitável na sociedade, sempre ávida pelo novo, individualista, imediatista, fragmentada e esquecida da história humana, logo o papel da escola no mundo pós-moderno é crucial para desconstruir essa cultura desigual, destrutiva e predatória (Filho, J. 2020).

Comparado por Neves (2008), o Brasil se assemelha a um ornitorrinco, um animal sem características bem definidas dentro das famílias da fauna global, da mesma maneira as características que moldam o Brasil são atípicas comparados a outros países, e movimentos que constroem a identidade nacional causam distorções sociais cada vez mais abruptas. Para evitar um colapso, a máquina pública se ajusta para incentivar cada vez mais capital estrangeiro ao privatizar

estatais que usam os serviços públicos básicos como mercadoria para o cidadão pagar, com uma esperança de melhorias na infraestrutura e sociedade do país. Esse modelo não afeta apenas serviços de subsistência como o consumo de energia elétrica e água, mas também dá liberdade de acesso à privatização de oferta de serviços na saúde e educação.

Dessa maneira o estado numa tentativa de melhoria para a população se exime parcialmente da responsabilidade sobre os serviços básicos, o que finda em uma realidade na qual os com melhores condições financeiras conseguem se beneficiar de serviços mais qualificados, e os sem ficam sob os cuidados precarizados do estado, enquanto ambos permanecem com a obrigação de pagar por impostos caros. Como consequência, os trabalhos complexos ganham destaque em detrimento dos trabalhos simples. Resumidamente, os trabalhos simples se caracterizam por atividades de execução rápida que necessitam apenas de pouca instrução para serem feitos, enquanto os trabalhos complexos são aqueles que para serem executados necessitam de uma jornada longa de apreensão de conhecimentos científicos e de treinamento prático. O destaque é notório quando veículos de comunicação expõem que vagas com bons salários estão em aberto por certo período de tempo enquanto noticiam alto índice de desemprego (Neves, 2008).

Isso remete à educação profissional, o cenário do mercado brasileiro, influenciado por organismos internacionais financeiros, passa a demandar mão de obra qualificada enquanto a classe trabalhadora também requer estudo para a inserção no mundo do trabalho com melhor recurso salarial para subsistência, o que dá margem para que o estado sem conseguir abarcar essa demanda por capacitação, passe a fomentar programas de estudo com empréstimos financeiros a estudantes que queiram ingressar em uma faculdade privada com o compromisso destes pagarem juros após o término do curso. Enquanto os estudantes de instituições públicas têm seus currículos construídos sob a exigência de cumprimento de diretrizes estabelecidas pelo estado incentivado por organismos capitalistas internacionais, os estudantes de instituições privadas são seduzidos por propagandas de facilidade de acesso aos cursos que podem ser financiados pelo programa de incentivo estatal (Neves, 2008).

Com essa nova influência de gerência educacional o conhecimento científico profissional se caracteriza como mercadoria, a escola/faculdade se torna um

empreendimento, e sob a lei de oferta e procura as mensalidades aumentam o preço e o estudante se endivida mais. Em contrapartida por influência da indústria as ofertas de cursos tecnológicos e de engenharia crescem para atender a demanda do mercado, nessa corrida pelo preenchimento de vagas de trabalho e pela ânsia da população em preenchê-la estabelece-se uma corrida do conhecimento com grades curriculares mais enxutas, pragmáticas, com pouca formação humana, objetivando a flexibilização do trabalhador apenas como um executor de tarefas, sem que se forme um humano omnilateralmente como alguém que além de ser competente no labor também seja capaz de ser crítico em suas atividades e que possa ter acesso e usufruir de outros elementos que compõem a humanidade. Portanto, essa relação mercado para o conhecimento e conhecimento para o mercado forma um homem unilateral e converge para uma escola mais enxuta em mediadores e mais vasta em investidores de conhecimento (Neves, 2008).

Porém, mesmo a escola desinteressada das práticas mercantilistas sofre influência das transformações do mundo do trabalho e das exigências do capital, portanto é de se esperar que haja integração de tecnologias da informação e comunicação, de uso natural para os ingressos, nas ações pedagógicas. Por exemplo, os ambientes virtuais de aprendizagem (AVA) e os objetos digitais de aprendizagem (ODAs) que dão suporte à abordagens interativas. Essa utilização de tecnologias prepara os discentes para os arranjos flexíveis de trabalho, porém há que se ressaltar a importância da relação de mediação direta de docente e discente na construção de um pensamento crítico e consciente sobre as atividades sociais, laborais, ambientais e os fenômenos econômicos que afetam a realidade em que estão inseridos. É necessário prover fluência tecnológica, um letramento digital, como habilidade essencial para a inclusão digital visando uma formação humana integral e politécnica a fim que compreendam, utilizem e criem as tecnologias de informação e comunicação também de maneira reflexiva, significativa e ética na produção e disseminação de informações (Souza, L. 2022).

2.3.2 Democracia e governança de dados

Após a derrota da democracia na Grécia antiga, durante a guerra do Peloponeso entre 431 e 404 A.C., a democracia praticamente desapareceu,

cedendo espaço ao ideário absolutista na Idade Média posteriormente. O ressurgimento da democracia ocorreu com o constitucionalismo moderno, destacado pelos federalistas, que contrapôs a democracia direta dos antigos à democracia representativa como a forma mais viável nos estados modernos. Esse pensamento foi expresso na obra "O Federalista" de Madison, Hamilton e Jay, defendendo a participação do povo por meio da escolha de representantes para tomar decisões em seu nome. O desenvolvimento do pensamento democrático aconteceu nas revoluções burguesas dos séculos XVII e XVIII na Europa, fundamentado no ideário liberal que, ao lado da democracia moderna, surge como produto do capitalismo. A democracia moderna, baseada no liberalismo, inicialmente vinculava a condição de cidadão à propriedade privada, estabelecendo uma perspectiva restrita que associava cidadania à propriedade. Apesar das históricas contraposições, o liberalismo e a democracia convergiram para a formação da democracia liberal (Neto, 1997).

O pensamento liberal, baseado nas formulações de Locke, destaca a construção de um Estado com autoridade legislativa fundamentada na igualdade natural. Para Locke, o Estado deve garantir propriedade e segurança, considerando o trabalho como a expressão da igualdade natural (submissão do homem às leis naturais que o regiam antes da organização legislativa estatal passível de repressão). Historicamente, o liberalismo, apesar de reconhecer a igualdade formal dos indivíduos perante a lei, vincula a cidadania à propriedade privada. O liberalismo, inicialmente não democrático, evoluiu para a democracia liberal ao se adaptar às reivindicações democráticas, resultando em uma complexa articulação entre liberalismo e democracia. A democracia liberal, surgida no século XIX, defende a soberania popular por meio da representação política, mas reconhece a necessidade de ampliação dos direitos políticos para uma participação mais efetiva dos cidadãos. O sistema representativo é criticado por sua limitação na tomada de decisões socioeconômicas, sugerindo a necessidade de mecanismos que permitam uma maior interferência dos cidadãos nesse processo. O liberalismo e a democracia, apesar de suas controvérsias, convergem para formar a democracia liberal (Neto, 1997).

Rousseau, situado no contexto do século XVIII, destaca a participação direta como uma necessidade histórica, propondo um Estado democrático baseado na

igualdade e liberdade. Defende o contratualismo para legitimar as forças sociais, alegando que a civilização perturba as relações humanas. Para Rousseau, os homens nascem livres, mas são escravizados pela modernidade. A solução seria um contrato para fundar uma sociedade igualitária e livre. Ele considera o Estado como expressão da vontade geral, não limitado ao governo, com a soberania popular promulgando leis fundamentais. Liberdade e igualdade são inseparáveis em sua visão, com a igualdade fundamentada na identidade entre os homens. Rousseau critica a democracia representativa, defendendo a participação direta como o eixo básico da teoria política. No entanto, reconhece a dificuldade prática da democracia direta em sociedades complexas. A articulação entre democracia representativa e direta, combinada com níveis adequados de democracia econômica e cultural, é vista como uma necessidade histórica (Neto, 1997).

No contexto dos séculos XIX e XX, as transformações políticas e econômicas geraram a necessidade de aprimorar os mecanismos democráticos para conciliar avanços na democracia política com a criação de condições mínimas de bem-estar para a população. Neto (1997) citando Bobbio (1987) argumenta que a expansão democrática não deve se limitar à integração da democracia representativa e direta, mas deve se estender a esferas sociais além das políticas. A ampliação da democracia política deve estar ligada à redução das desigualdades sociais, pois extremas disparidades prejudicam tanto a democracia política quanto a econômica. Discute-se o desenvolvimento do estado de bem-estar social nos países democráticos, relacionando-o a demandas sociais e mudanças estruturais, no entanto, observa-se que o modelo enfrenta desafios na contemporaneidade, com a crise econômica e a ascensão do neoliberalismo, resultando em uma possível mudança nos padrões das políticas sociais.

Markoff (2013) aponta que embora tenha havido um aumento significativo de países democráticos nas últimas décadas do século XX, surgiram preocupações crescentes sobre o descontentamento com as práticas democráticas em diversas partes do mundo. Cientistas políticos observaram uma erosão na fé e confiança do público em políticos, partidos e instituições políticas, tanto em países recentemente democráticos quanto em países com tradição democrática de longa data. O autor menciona que, historicamente, períodos de avanço democrático foram seguidos por retrocessos, levando a uma análise mais aprofundada sobre a qualidade da

democracia. Destaca que as expectativas em relação à democracia nem sempre foram atendidas, gerando debates sobre seu significado. Ao longo dos séculos, houveram mudanças nas concepções de democracia e nas exclusões originalmente estabelecidas, implicando questões importantes para o século XXI. As concepções de democracia tendem a ser mais inclusivas hoje, mas enfatiza a possibilidade de transformações futuras no entendimento desse conceito fundamental. Enseja-se o questionamento se a ideia tradicional de democracia, centrada no governo autônomo de um povo em um território delimitado, permanecerá válida no futuro.

Diante de desafios contemporâneos, a evolução do conceito de democracia encara questões transnacionais a exemplo as eleições presidenciais com discursos polarizados, alguns questionados sobre o uso de *fake news* na manipulação da massa popular. A imigração crescente de trabalhadores para países com relação mais estreita entre democracia e capitalismo em busca de melhores condições de vida, implicando em argumentos do papel do imigrante enquanto cidadão e do enfrentamento às ações xenofóbicas. Além da preocupação crescente com questões de segurança relacionadas a terrorismo, guerras, criminalidade e emprego, indicando como essas preocupações impactam esses estados ricos. Ademais, há ameaças à democracia decorrentes de problemas globais, como epidemias e mudanças climáticas, que podem exigir organizações transnacionais em rede de tomada de decisão. O autor pondera sobre o significado futuro da democracia diante da influência crescente dessas instituições e das desigualdades entre os estados. Também destaca a necessidade de repensar a democracia em um contexto global e transnacional, considerando que as instituições nacionais podem não ser suficientes para lidar com desafios críticos (Markoff, 2013).

Markoff (2013) apresenta diferentes perspectivas a serem debatidas sobre a transnacionalidade democrática: (1) existe o argumento que a escala é grande demais para permitir um governo transnacional efetivamente responsável perante os cidadãos. Max Weber expressava ceticismo em relação à supervisão popular efetiva do governo. (2) Uma visão mais otimista destaca o crescimento de formas de monitoramento público do poder, denominado por John Keane como "democracia monitorada". Erik Olin Wright destaca a capacidade das pessoas de formar associações para promover objetivos coletivos. (3) Outras perspectivas negam a possibilidade de uma democracia além do estado nacional devido a diferenças

culturais, identidades nacionais fortes e resistência de democracias existentes à diluição de poder. (4) Há a preocupação de que regimes autoritários resistam à incorporação em estruturas democráticas e que cidadãos de democracias existentes rejeitem a ideia de compartilhar poder com estados não democráticos. Portanto a necessidade de repensar a democracia em uma escala mais ampla se faz presente, explorando diferentes modelos e instituições para estimular uma renovação da imaginação política democrática.

A vida na sociedade contemporânea é complexa, há a presença frequente de conflitos de opiniões, interesses e valores. Destaca-se a necessidade de manter o conflito dentro de limites administráveis para a sobrevivência e progresso da sociedade, sendo a política uma alternativa à coerção. A política é definida como o conjunto de procedimentos formais e informais que expressam relações de poder, visando a resolver pacificamente conflitos relacionados a bens públicos. As políticas públicas são discutidas como decisões e ações do Estado sistematicamente avaliadas quanto à relevância e eficiência. O processo de formulação de políticas envolve a identificação de atores, preferências em torno de questões específicas e o reconhecimento da ambiguidade no processo decisório. A implementação de políticas é um processo complexo que requer condições específicas, e seu sucesso depende de fatores como recursos, teorias adequadas e coordenação eficaz. No entanto, a implementação muitas vezes se confunde com o processo de formulação, levando a desafios na definição de objetivos e problemas (Maldaner, 2016).

Uma maneira de exercitar o repensar a democracia é por meio do 'ciclo de política' que se caracteriza por ser uma abordagem de estudo e construção de políticas públicas sob uma orientação pós-moderna, a fim de romper com a rigidez hierárquica com que essas eram elaboradas, das influências de organizações mundiais, organizadas em redes de interesse muitas vezes alinhado ao sistema capitalista, ao texto do corpo das leis, decretos, diretrizes e normas escritos pela cúpula do estado, até a materialização delas no labor dos educadores. Baseada no trabalho dos autores ingleses Stephen Ball e Richard Bowe, essa abordagem evidencia, por exemplo, a complexidade das políticas educacionais ao tentar traçar uma articulação entre as macro e micro políticas, e o impacto delas na rotina dos profissionais que lidam com elas nos seus locais (Mainardes, 2006).

A princípio Ball e Bowe caracterizam o processo constituído de três facetas: (1) a “política de proposta” referindo-se à política oficial e às intenções do governo de estado identificados nos discursos; (2) a “política de fato” constituindo-se das etapas legislativas até a construção dos textos das leis para dar base a prática; e (3) a “política em uso” referindo-se às práticas educacionais nos ambientes institucionais. No entanto, descartaram essa classificação pois se assemelhava a estrutura rígida de outrora, pois para a construção de políticas educacionais deveria haver o envolvimento dos profissionais de educação. Para isso os autores convidam o educador a participar na produção de discursos interpretativos nas lacunas deixadas pelas políticas e nas manifestações por mudanças de políticas com impacto negativo na localidade. Esses discursos estão diretamente ligados aos contextos dos autores de discursos, sobretudo o contexto da prática pelos profissionais da educação, e neles é possível identificar movimentos complexos e contraditórios de conflito, acomodação e convergência de ideias (Mainardes, 2006).

Propuseram um ciclo contínuo se caracterizaria em três contextos principais sem uma estrutura rígida temporal, sequencial ou linear, em ambientes heterogêneos compostos por grupos de interesse envolvidos em disputas e embates: (1) “contexto de influência” onde políticas emergem a partir de discursos, e os grupos de interesse entram em disputa para influenciar os textos normativos e condutivos; O (2) “contexto da produção” está fortemente relacionado ao contexto da influência, nele ocorre a produção de fato do texto formal disputado pela influência dos grupos por meio de acordos, aqui os textos representam a vontade política das ideologias, esses se manifestam não somente em leis, mas também em discursos informais, comentários, pronunciamentos oficiais, mídias sociais, meios de comunicação massivos, etc; No entanto, o contexto da produção tem limitações materiais e de possibilidades do real, o (3) “contexto da prática” se caracteriza pela aplicação das políticas, o que nem sempre ocorre de acordo com a ideologia do grupo que a promoveu, portanto há margem para interpretação e reinterpretação desta pelos atuentes para promoção da recriação das políticas educacionais, o que permite que as políticas originais não sejam implementadas, mas não por serem ignoradas, e sim por resultarem em impactos e sofrerem resistência (Mainardes, 2006).

Ainda assim, Mainardes (2006) reconhece a falta de outros contextos que são desconsiderados por irem em um movimento contra-hegemônico pouco presentes nas organizações com maior poder de influência nos contextos de produção. O autor defende que o cenário da educação é amplo, que os demais contextos devem ser considerados e estudados para a recriação e formulação de políticas públicas em educação. De modo geral, Mainardes (2018) expõe que os estudos de política em educação abordam as seguintes questões: (a) estudos de natureza teóricas relacionados às políticas educacionais (influências ideológicas, relação entre setor público e privado, epistemologias e fundamentos teórico-metodológicos das políticas educacionais); (b) análise de políticas (implementação e avaliação); (c) políticas educacionais e gestão; (d) legislação educacional; (e) financiamento da educação; (f) políticas curriculares; (g) políticas voltadas ao trabalho docente; (h) questões relacionadas às demandas, ofertas, qualidade e acesso educacional.

Percebe-se que não existe um único campo de estudo das políticas educacionais, essas são pluralistas, sofrem as influências dos grupos hegemônicos em todas as esferas do estado (municipal, estadual e federal) seja verticalmente por determinações diretivas ou horizontalmente através de redes de influência. E por estar fragmentada, Mainardes (2018) com base nas obras de Sayer (1984) e Pawson e Tilley (1997), consegue classificar três níveis de análise inter-relacionados: (1) “políticas da educação” - estrutura - é o nível mais elevado de condicionantes mais gerais pelo estado; (2) “políticas educacionais” - mecanismos - envolve identificar os mecanismos atuantes nos contextos da educação; (3) “pesquisas sobre políticas e programas” - eventos - é o nível mais concreto em que os estudos sobre os impactos são desenvolvidos e em seguida difundidos. Considerando essa última onde o educador manifesta sua interpretação e estimula a recriação de políticas, é necessário que o mesmo se valha de argumentos muito bem estruturados e embasados, ao que, por exemplo, indicativos metodológicos concisos tem mais peso que apenas argumentos ideológicos e teorias, além do mais, o pesquisador deve observar o contexto onde está levantando as informações de pesquisa: local, cultura profissional-institucional, condição de ambiente de trabalho e de material, e as condições externas à escola que a impactam (Mainardes, 2018).

Esse cenário de construção de instrumento regulador remete à definição de Pinto (2005) sobre a cibernética, que contrariando os pensadores que defendiam que essa existia somente no mundo das ideias, se baseou no pensamento da dialética da natureza de Engels de que as forças naturais eram materiais embora invisíveis e se relacionam, da mesma forma, em um ambiente de ideias entrópicas - espalhadas - a cibernética se traduz como um movimento de organização delas em informação e conhecimento para a materialização de um propósito. Para que o homem seja capaz de aplicar tal conhecimento ele precisa ser formado, não somente como um transformador dos elementos da natureza, mas um transformador da sociedade, pela dialética das “forças de convivência” se materializando em regramentos e leis. Portanto, isso realça que a dialética vem antes da cibernética, pois a partir da reflexão e da contradição das ideias é possível experimentar e estabelecer os regramentos de estruturas de sistemas gestores e reguladores.

Da entropia à primeira aproximação ao problema da essência da informação, há a reflexão sobre a identificação e parametrização de variáveis em estudos de projetos de lei e de construção de estruturas organizacionais dentro da sociedade. A visão de entropia, que assim como a cibernética, está relacionada com o homem na sociedade e os sistemas que ele produz, pois naquela há uma energia de elementos que evoluem ou se degradam com o passar do tempo, e no que cerne um sistema produzido, sua vida útil está determinada a quanto seus parâmetros são especificados ou generalizados de maneira a atender a demanda humana. Cabe refletir sobre o que a informação significa em um contexto, e em ser crítico quanto à descoberta, porém é preciso que essa criticidade encontre um equilíbrio entre o pragmatismo e o aprofundamento excessivo. Cabe ao propositor ser coerente para que o instrumento regulador atenda uma demanda social (Pinto, 2005).

Logo, a cibernética não é neutra nas relações sociais, os sistemas se degradam constantemente para o perecimento ou evolução para atender interesses. Esse movimento reflexivo dialético entre entropia e dialética se manifesta com o exercício do trabalho da gestão e governança. Como amparo do gestor, o mundo informatizado materializou a cibernética em *softwares* de computadores, tornando-se quase sinônimos, que essencialmente são sequências encadeadas de instruções finitas, abstrações de processos, a fim de trazer uma solução tendo como fonte uma base de dados. Os primeiros a promoverem a melhoria de processos foi a indústria

japonesa no pós 2ª guerra mundial em busca de interessadas em aprimorar o nível de qualidade de produtos tomando como base conceitos de qualidade e controle estatístico, que ensinaram o ciclo PDCA: (1) planejar identificando problemas, (2) fazer (*do*), produzir, testar soluções, (3) checar resultados, (4) agir para melhorar o processo/produto. Posteriormente, outros modelos de qualidade de processos de engenharia de *software* surgiram nas décadas de 70 e 80 em solo estadunidense, como as normas ISO e o modelo CMM/CMMI que deu base ao modelo brasileiro MPS-BR décadas depois (Barbieri, 2019).

Porém, o foco em qualidade de dados se deu tardiamente, apesar da evolução de equipamentos e sistemas de armazenamento e controle de dados ter se dado paralela aos processos de governança empresariais, os primeiros modelos de governança de dados surgiram no início da década de 2000 quando as tecnologias de comunicação possibilitaram que grandes empresas tivessem acesso a uma massa de dados grande o suficiente para gerarem gargalos ou possibilitarem mudanças estratégicas, o mais célebre, DAMA-DMBoK, em sua versão de 2017, aponta que a governança de dados é uma transformação cultural em uma organização, sendo o seu ativo mais importante as pessoas que geram esses dados - *peopleware* -, para isso a organização deve atender conceitos específicos, como a arquitetura e modelagem de metadados, a infraestrutura de ambientes informatizados, a qualidade dos dados, a segurança de acesso e armazenamento, a interoperabilidade de dados entre sistemas, e análise desses (Barbieri, 2019).

No contexto social, a evolução da governança de dados dentro das grandes empresas de tecnologia da informação se deu pela escalada global de vendas engenhosas de fetichização de seus produtos de tecnologias móveis e portáteis agregados à *softwares* com algoritmos elaborados por eles próprios, que coletam dados de seus usuários para a mercantilização de produtos, mídias e serviços personalizados. Além disso, esse conglomerado empresarial veste a roupagem do que vem a ser o colonialismo digital ao explorar matéria prima no hemisfério sul para a fabricação de seus produtos, utilização de dados para manipular as massas, sobretudo na influência de eleições, eventos globais, promoção de notícias de seus interesses, e de meios para a precarização do trabalho em países subdesenvolvidos, a exemplo a terceirização de serviços de moderação de conteúdo, por vezes sujos a ponto de traumatizar trabalhadores, e fazenda de

cliques que chegam a ser análogas à escravidão, a exemplo de prisioneiros chineses, após trabalhos físicos durante o dia, forçados à noite a coletarem moeda virtual de um jogo para ser vendida por dinheiro real em plataformas digitais (Lippold, 2022; G1, 2011).

Portanto, para o capital, o dado, sobretudo o pessoal sob uma perspectiva de vigilância digital, passa a ter um valor agregado quando transformado em informação, e quando ocorre uma exploração exacerbada e que mine sistemas de poder, o estado democrático interfere com regulações a fim de balizar a prática. No Brasil há alguns instrumentos legais importantes para esse controle: a lei nº 12.527, de 18 de novembro de 2011, mais conhecida como lei de acesso à informação (LAI) que sob o princípio da publicidade e transparência estabelece procedimentos de divulgação de ações de órgãos e autarquias dos entes federativos; a lei nº 12.965, de 23 de abril de 2014, o marco civil da internet, que estabelece garantias, direitos, deveres e princípios no uso da internet no Brasil, também estabelece diretrizes de governança institucional e pessoal de dados, além de tipificar a soberania dos dados nacionais; a lei nº 13.709, de 14 de agosto de 2018, ou lei geral de proteção de dados pessoais (LGPD), que dispõe sobre o tratamento de dados pessoais em qualquer meio, em especial no meio digital, para a proteção do direito de liberdade e privacidade, nela enseja-se a criação da autoridade nacional de proteção de dados (ANPD); O decreto nº 10.046, de 9 de outubro de 2019 que estabelece critérios de governança de dados entre órgãos e entidades do poder público e cria o comitê central de governança de dados (CCGD); também o decreto nº 11.856, de 26 de dezembro de 2023, que amplia o poder de polícia do estado no espaço digital, instituindo a política e comitê nacional de cibersegurança (CCN). Isso cria um terreno de debates do uso de dados, possibilidades e riscos, com espaços para debate de cultura de dados no governo federal (GOV, 2023), debate no uso estratégico de dados para construção de políticas públicas (GOV, 2024), incentivo a integridade e segurança de dados (MGI, 2024; ANPD, 2024), e discussões sobre a regulação do uso de inteligências artificiais na análise de dados (Senado, 2023).

Para esse controle de governança estatal o Brasil tem como entidade de controle externo o Tribunal de Contas da União (TCU) , que tem por alvo aprimorar a administração pública em benefício da sociedade. Dentre seus atos o que compeliu a educação a publicização de seus dados foi o acórdão nº 2.267 de 2005 que deu

base para o ministério da educação criar a plataforma Nilo Peçanha (PNP) via portaria nº 01, de 03 de janeiro de 2018, a PNP como canal de indicadores apontados pelo TCU apresenta dados estatísticos de eficiência e eficácia das ações da rede federal EPT (PNP, 2018), porém apesar dos institutos federais terem *softwares* de gerenciamento de suas atividades, a extração de dados das suas bases de dados, por vezes é realizada manualmente na consulta de telas, então transmitidos via planilhas quando possível, por exemplo.

No IFSC percebe-se no relatório de controle interno de extensão de 2021 algumas fragilidades que compõem essa extração, como ausência de aderência do objetivos e planos estratégicos da extensão presentes no plano de desenvolvimento institucional 2020-2024, fragilidade no ambiente de controle de atividade (SIGAA) e no acompanhamento das ações, por conseguinte a extensão no IFSC expõe em seu relatório de gestão de 2023 que ações com base nos objetivos estratégicos de articulação com as coordenações do câmpus e registro das atividades registradas no SIGAA em diálogo com a comunidade - *peopleware* - têm sido feitas.

Em paralelo a esses eventos, o TCU em 2022 no seu acórdão nº 461, destaca indicadores a serem apresentados quanto à educação superior, onde a extensão também se faz presente. Além da PNP, o estado possui uma plataforma de controle interno de dados abertos da união, em que ambas apresentam lacuna de dados e/ou apresentação difusa em que a extensão e pesquisa, embora indissociáveis, são expostas como uma só, dificultando um pouco a observabilidade da extensão na EPT (CGU, 2023). Ainda o MEC por meio da SETEC, publica em 2022 um manual de indicadores de extensão, de pesquisa e de inovação para aprimoramento da publicização de dados, além do FORPROEXT, enquanto entidade em rede de extensão, em seu 52º encontro em dezembro de 2023, reafirma na sua carta o incentivo a melhoria da coleta de dados da extensão (RENEX, 2023).

2.3.3 Modelo difusionista de informação

A comunicação é base das interações sociais em todos os tempos e contextos, seja por meio de uma linguagem verbal ou corporal, pela grafia ou cunha de símbolos, na utilização de elementos naturais como a sapopema, raiz acústica da samaúma, ou de ferramentas criadas, como os cabos de fibra ótica que emitem

sinais manipulados e encriptografados. Ela se manifesta quando se emite (pessoa, roteadores, antenas) mensagem (palavras ditas e escritas, símbolos, dados, frequências) por um canal (voz, ar, cabos) até um receptor (pessoa, rádio, computador, televisão) que traduz o significado da mensagem (sinapse, decodificador, algoritmo) e por vezes retorna uma resposta (Bordenave, 1997).

Encarada pelo capital como instrumento de perpetuação de dominação da classe trabalhadora, a comunicação sofre um processo de industrialização, ao produzir não somente ferramentas de transmissão massiva, mas também no aperfeiçoamento da técnica de elaboração da mensagem por profissionais especializados. Consequentemente, o efeito se expande para o terreno político, *“visto que ela pode estabilizar ou desestabilizar governos”* (Bordenave, 1997, p.32-34, o que é comunicação), meios de comunicação independentes, privados e estatais digladiam pela difusão de interesses. O Brasil historicamente instituiu duas ferramentas de divulgação de seus atos, o programa de rádio “voz do Brasil” que deve passar no mesmo horário em canais registrados (Câmara dos Deputados, 2006), e o “Diário Oficial da União”, que desde à época do império, pós revolução industrial, os divulga por mídia impressa e a partir do ano 2000 na versão digital (Imprensa Nacional, 2018).

O campo educacional também sofreu influência da indústria da comunicação, Ramos (M. 1994) cita quatro modelos no meio científico: (1) difusionista, no qual o pesquisador a partir de seus estudos publica a sua pesquisa em periódicos especializados para a disseminação, como que epidêmica, do artigo para demais acadêmicos, dela se derivam os buscadores para publicações científicas na *internet*; (2) paradigmático que constitui um fluxo de debates informais cognitivos em que os participantes experientes convergem em ideias, práticas, paradigmas de como realizam a ciência, a exemplo a estruturação de teorias e fóruns de debate; (3) dialético em que há a contraposição de deduções hipotéticas capazes de serem testadas e do prognóstico serem refutadas ou aceitas na comunidade; (4) culturalista, que se dá o dicionário de vocábulos de um grupo de cientistas que os identificam e separam de outros no próprio meio acadêmico e na sociedade.

Todavia, o modelo difusionista foi o pioneiro na extensão, que deu por base as ações ruralistas, como o projeto Rondon supracitado. A extensão rural no Brasil se deu pela crise global de alimentos nos meados do século passado, o mundo

passou a ter a preocupação de como os governos alimentariam as pessoas para a manutenção da ordem hegemônica vigente, por isso recorrem às universidades, em busca do conhecimento científico para que esse fosse posto em prática no campo. Porém, o extensionista emergente do meio acadêmico, formado sob um modelo de comunicação culturalista, tenta por meio da difusão, transmitir o conhecimento e técnicas de manejo da terra, de forma persuasiva, para uma comunidade de trabalhadores rurais, moldados sob outra cultura, desse encontro provia ruído de comunicação e ineficácia da abordagem. Como resultado, o homem do campo fica desconfiado e incomunicável, para contornar isso, sob uma concepção dialógica de Paulo Freire, os extensionistas rurais, antes de proverem soluções acadêmicas, conversam com a comunidade para entendimento do contexto dela, e dela aprender, ressignificar, e educar. Assim é possível estabelecer um vínculo de confiança e eficácia na transformação da relação do homem com a tecnologia (Pestana, 2020; Estêvão, 2013).

A comunicação constrói a pessoa, logo, para que uma mensagem difundida possa surtir efeito, é preciso que o emissor a redija ou edite de maneira que o público receptor a entenda e seja atraente. A persuasão é carregada de intenção, mas não é necessariamente negativa, o objetivo desprovido de manipulação é comunicar para que estimule o receptor a se interessar em assuntos que complementam ou desvelam a relação dele com o espaço social em que se encontra (Bordenave, 1984; Estêvão, 2013). Ainda que em um primeiro momento o modelo difusionista da informação tivesse sua resistência no meio rural, é possível, também em meio urbano, que um receptor letrado digitalmente e com acesso às tecnologias da informação e comunicação, possa se beneficiar da disseminação de dados e informações de interesse através de *softwares* observatórios, tendo como canal a internet, para que sirva de insumo para a comunicação dialógica na construção ou fortalecimento do que se quer divulgar.

3 PROCEDIMENTOS METODOLÓGICOS

3.1 CARACTERIZAÇÃO DA PESQUISA

O método científico consolida sua caracterização através de métodos, ou maneiras de realizar algo, a partir de uma sistematização de abordagens e procedimentos controlados e observáveis. Diferente do conhecimento empírico proveniente da experiência tácita do observador, do conhecimento popular baseado no senso comum ou do religioso com base na fé das expectativas futuras fundadas em experiências metafísicas, o científico se baseia na materialidade a fim de estabelecer princípios e conceitos com o objetivo de elucidar as razões do objeto estudado.

Como método científico utilizado considerou-se uma classificação indutiva uma vez que para a construção do conhecimento legal e normativo de uma área de conhecimento existir é preciso uma análise e classificação dos fenômenos sociais materializados em documentos, e destes extrair informações. Também foi considerada a classificação dedutiva na construção do produto educacional a partir da premissa que há um espalhamento de dados em diversos artefatos institucionais que precisam ser organizados (Otani, 2023).

Esta pesquisa apresenta uma abordagem qualitativa e quantitativa, em um primeiro momento se discute e reflete o surgimento qualitativo de indicadores a partir da construção da identidade da extensão no país e no IFSC, para trazer à tona um quantitativo de dados da materialização das ações de extensão e então retoma uma análise qualitativa dos resultados com base nos documentos de planejamento institucional e entrevistas (Otani, 2023).

O objetivo tende a nortear o caminho percorrido para a construção da pesquisa, neste sentido essa se pauta como uma pesquisa exploratória na busca de compreender o cenário da extensão dentro do escopo estudado, também se torna descritiva quando encontram-se pontos convergentes no material levantado na identificação de indicadores, ao mesmo tempo tem um objetivo pragmático ao refletir a realidade do objeto estudado em comparação a um escopo amplo, por estabelecer as regras de negócio para a construção do produto educacional em um *software* (Otani, 2023; Wazlawick, 2014).

No que tange os procedimentos técnicos da pesquisa, nesta é feita uma pesquisa bibliográfica para o entendimento do cenário da extensão com base em autores de dissertações, teses e artigos científicos. Assim como nesta, há um procedimento de revisão documental na análise de artefatos legais e normativos nacionais, em como eles influenciam a construção de documentos diretivos e de planejamento, de como esses geram dados dentro da instituição e como são sistematizados para o desvelamento da realidade (Wazlawick, 2014).

A pesquisa-ação é defendida por Tripp (2005) como uma forma de investigação-ação usando técnicas de pesquisa para informar ações destinadas a melhorar a prática. Embora a pesquisa-ação tenha características pragmáticas, ela se distingue da prática rotineira e da pesquisa científica tradicional, pois pode alterar a ótica sob o objeto de pesquisa e é limitada pelo contexto, pelo meio em que o fenômeno ocorre e pela ética da prática que direciona a neutralidade do pesquisador na finalidade da investigação. Uma das técnicas de pesquisa consonantes com a pesquisa-ação é o *Design Science Research* (DSR) que se concentra na criação de artefatos, soluções ou *designs* para resolver problemas complexos. Ao contrário das tradicionais pesquisas científicas, que buscam entender fenômenos naturais, a DSR está mais preocupada em desenvolver intervenções práticas e aplicáveis (Lacerda, 2013).

Por sua vez, um artefato criado pode não gerar impacto se não houver envolvimento das pessoas pertencentes ao contexto, para tanto é necessário divulgá-lo para que haja aplicabilidade. A comunicação é fundamental no processo de inserção social do ser humano, portanto é uma necessidade básica. Através dela a interação humana se materializa, produz conhecimento, códigos, condutas, leis e relações. Para este trabalho, é destacado o modelo difusionista de informação, que caracteriza-se por estabelecer contato unilateral através de um veículo de comunicação por agente(s) transmissor(es) para uma abrangência de agentes receptores, comumente eivada de conteúdo persuasivo que seduza o receptor a assimilar a informação (Estevão, 2013; Ramos, M. 1994).

Portanto nesta pesquisa, após a revisão documental, realizaram-se entrevistas com o corpo gestor extensionista para entendimento do contexto no IFSC enquanto propositor de formação integral, além de investigar o fenômeno de interesse na publicização de dados das ações de extensão, pautado sob a ética

epistemológica da prática extensionista e dos artefatos diretivos legais e como através das características iterativas orientadas a solução da DSR contribuiu para a modelagem de um artefato em *software* que atenda os anseios dos interessados.

3.2 PARTICIPANTES

Como participantes do estudo a pró-reitoria de extensão e relações externas (PROEX), assessoria e suas diretorias de extensão e comunicação institucional se fizeram presente, além de gestores selecionados por região de departamentos e coordenações de extensão dos câmpus, também a documentação produzida por outros setores do IFSC são relevantes para a exploração de dados. A PROEX tem por missão externalizar à sociedade a articulação com o ensino e a pesquisa, promovendo ações que envolvem participações ativas de docentes, discentes e técnicos-administrativos. Dentre as ações de extensão identificam-se programas, projetos, iniciativas sociais, cursos, eventos, empresas sem fins lucrativos e processos de curricularização da extensão, articulados com a publicização de editais de fomento e participação em fóruns de discussão para o seu fortalecimento (IFSC, 2023). A pró-reitoria citada tem papel importante na articulação dos câmpus e da defesa da extensão em âmbito regional e nacional com outras instituições, portanto é coerente inferir que esta centralize dados coletados na comunidade acadêmica para fundamentar debates e fomentar editais.

3.3 DETALHAMENTO DAS ETAPAS DA PESQUISA

Nesta seção há uma breve descrição das etapas da pesquisa, seus procedimentos metodológicos de acordo com os objetivos específicos, onde cada resultado esperado encadeia com a etapa seguinte a fim de alcançar o objetivo geral conforme o quadro 1.

Quadro 1 - Etapas da pesquisa.

ETAPA	OBJETIVOS ESPECÍFICOS	PROCEDIMENTOS METODOLÓGICOS	RESULTADOS ESPERADOS
1	Identificar as informações da extensão de interesse da gestão do IFSC e da sociedade civil.	Revisão documental e realização de entrevistas semiestruturadas	Os indicadores e parâmetros que comuniquem sobre as atividades de extensão de interesse no IFSC
2	Recuperar os dados da extensão e traduzi-los para uma estrutura de dados comunicável.	Modelagem de artefatos de comunicação de dados pelo método exploratório descritivo (pesquisa-ação)	As informações das atividades de extensão organizadas de acordo com os indicadores e parâmetros identificados na etapa anterior
3	Projetar os painéis de observação em <i>software</i> de publicação dos dados e informações.		1ª versão do Produto Educacional
4	Divulgar os painéis de observação.	Divulgação do produto educacional pelo modelo difusionista de informação.	Produto educacional divulgado.
5	Avaliar o produto educacional.	Avaliação do produto educacional com grupo focal.	Produto educacional avaliado.

Fonte: autoria própria.

3.3.1 Etapa 1 - Identificação das informações da extensão de interesse da gestão do IFSC e da sociedade civil

Em suma, a extensão propõe expandir a relação do meio acadêmico com a sociedade através de suas ações embasadas no conhecimento produzido na escola. A princípio no Brasil há regulamentação da extensão universitária na curricularização dessa nos cursos de nível superior (Frutuoso, 2021), ainda assim os institutos federais expandem o conceito para a formação integral de nível médio, o que permite que as características das ações de extensão técnica possam diferir de uma autarquia para outra. Por conseguinte, fez-se uma revisão documental das normativas do IFSC para levantamento de indicadores que caracterizam a extensão na instituição. Ainda assim, se fez oportuno realizar entrevistas semiestruturadas com gestores extensionistas que conhecem a realidade nos câmpus, que apontaram indicadores de interesse coletivo, bem como extrair dados públicos de acesso à

informação ao IFSC, de portais públicos, em que se possam identificar informações de interesse que não estão previstos em documentos e observados por gestores.

3.3.2 Etapa 2 - Recuperação dos dados da extensão e tradução para uma estrutura de dados comunicável

Para a promoção das ações de extensão o IFSC fomenta editais para que essas sejam materializadas, para isso o fluxo desse processo é organizado dentro do sistema integrado de gestão de atividades acadêmicas (SIGAA), consequentemente da base de dados desse ambiente informatizado se concentram os registros históricos das ações extensionistas. Portanto, após a catalogação dos identificadores de interesse, o passo seguinte foi extrair esses dados por meio da elaboração de comandos de consulta, técnicas de extração e transformação de dados, isso foi possível por meio da linguagem de programação Python e da linguagem estrutural de consulta (SQL) baseada na teoria matemática dos conjuntos.

Os dados limpos e quantificados para serem apresentados é preciso que sejam transmitidos por outras aplicações de software separadamente do SIGAA para melhor uso dos recursos computacionais, esses artefatos de orquestração de fluxo de dados são construídos ou configurados conforme a necessidade de cada contexto, normalmente obedecem uma organização por etapas, cada uma responsável por algum tratamento dos dados, podendo por último apresentar uma visualização dos dados comunicável ou enviá-los para que uma terceira ferramenta faça essa apresentação.

3.3.3 Etapa 3 - Construção dos painéis de observação em *software* de publicação dos dados e informações

Uma vez estabelecidos os pacotes de dados comunicáveis, foi preciso modelar os arquétipos de apresentação: painéis para recebê-los. Na elaboração desses foi considerada a ferramenta Looker Studio por conta do uso dela dentro do IFSC visando a continuidade do produto educacional. Ademais, essa utiliza, ainda que abstraídos em componentes visuais, uma apresentação amigável, comunicável

e legível coerente com as 10 heurísticas de Jakob Nielsen que embasam a usabilidade de interface visual.

3.3.4 Etapa 4 - Divulgação dos painéis de observação em sítio institucional

Um artefato pronto para que seja utilizado é primordial que o público de interesse tenha acesso, uma forma presente de veículo de comunicação é a *world wide web* popularmente conhecida como internet. O IFSC se faz presente nessa rede, por meio dela é possível acessar o principal ponto de comunicação digital difusora: o sítio institucional, para tanto teve o direcionamento das diretorias de tecnologia da informação e de comunicação para hospedagem do produto educacional nos domínios do IFSC.

3.3.5 Etapa 5 - Avaliação do produto educacional

Ainda foi observada a relevância e utilidade do artefato pela validação do produto educacional por meio de um grupo focal. Essa é uma técnica de pesquisa qualitativa oriunda de conversas em grupos estimulantes à participação e interação entre os membros, proporcionando diálogo e reflexões acerca da pauta a fim de obter as impressões do grupo a respeito dos objetivos do objeto de estudo. A escolha dos participantes foi deliberada quanto ao papel que esses desempenham a respeito do objeto investigado. Portanto a aplicação dessa técnica tendo como objeto a apresentação do artefato de publicização dos indicadores intitulado Observatório da Extensão, com participação de atores na gestão extensionistas e na sociedade, teve por objetivo a coleta de impressões sobre a validação do produto se este atende o fenômeno (Trad, 2009; Dias, 2000).

3.4 LOCAL E PERÍODO

A pesquisa teve realização no Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina - IFSC, durante o período de outubro a dezembro de 2024, considera documentos institucionais elaborados no último quadriênio - 2020 a

2024, e utilizou-se de ferramenta de videoconferência para no mesmo período, com a aprovação no CEP SH (Comitê de Ética em Pesquisa com Seres Humanos), a realização de entrevistas individuais e reunião de grupo focal.

3.5 CRITÉRIOS DE INCLUSÃO

Para critério de inclusão de documentos institucionais para análise são considerados os normativos em vigor, sendo que os artefatos expositivos, como de relatório e planejamento, são considerados os do quadriênio 2020 a 2024. Quanto aos participantes foram selecionados servidores da gestão extensionista que estejam bem ambientados com as demandas da unidade em que estão lotados e que tivessem 12 meses ou mais de experiência nessa área de atuação.

3.6 CRITÉRIOS DE EXCLUSÃO

Como critério de exclusão documental são desconsiderados documentos em fase de estudo, também os documentos recentemente homologados fora do escopo do cronograma da etapa de revisão documental, mesmo que estivessem dentro do período do quadriênio 2020 a 2024. Com relação aos servidores participantes da pesquisa, ainda que atendessem critérios de inclusão, foram excluídos os que foram movimentados para a unidade há menos de 6 meses.

3.7 PROCEDIMENTOS E INSTRUMENTOS PARA A COLETA E GERAÇÃO DE DADOS

Os procedimentos iniciaram-se com uma investigação histórica bibliográfica das obras acadêmicas no cenário nacional, levando em consideração os movimentos sociais e ideológicos que constroem a identidade da extensão, bem como desvelam a importância dele. Em seguida, buscou-se realizar uma análise comparativa entre o estado da arte acadêmico e o desenvolvimento legal documental que promovem ações de extensão no Brasil e no IFSC. Em um terceiro momento foi levado em consideração procedimentos funcionalistas e estruturalistas,

que esboçam, respectivamente, como a relação entre sociedade enquanto grupos e indivíduos (comunidade) e instituição (IFSC) estabelecem regramentos para o funcionamento de sistemas construídos sob estruturas organizacionais (câmpus e pró-reitoria) e processos concretos (normativas) e abstratos (procedimentos e práticas) que retratam a estrutura social vigente (Marques, 2009).

Para a coleta de dados foram utilizados fichamentos provenientes das obras documentais, através das ferramentas *google sheets* e *notion* que condensam resumos, categorias e palavras-chave permitindo uma análise de conteúdo para a descoberta de processos concretos estabelecidos ou não na PROEX e no IFSC (Bardin, 2011). Paralelamente aos fichamentos, buscou-se realizar uma extração de dados automatizada com *scripts* em linguagem Python das plataformas FalaBR e BuscaLAI para mapeamento das solicitações de acesso à informação da extensão vindas da sociedade civil entre os anos 2020 a 2024.

Para processos abstratos foram programadas entrevistas semiestruturadas individuais em meio virtual usando a ferramenta de videoconferência *zoom* tendo como público alvo servidores participantes da gestão da extensão no IFSC, esta teve como propósito compreender as ações que a gestão tem realizado para atendimento e construção de processos concretos. A entrevista compôs questionamentos com vista ao entendimento do cumprimento de normativas e demandas da sociedade, e seção aberta para conhecimento do trabalho diário da gestão no atendimento das demandas dos câmpus. A participação em entrevistas foi respaldada por um termo de consentimento à identidade do participante (apêndice B) e foram gravadas mediante a autorização deste(s) para análise posterior. Caso a gravação fosse negada seria produzida uma ata resguardando o nome do(s) envolvido(s).

De forma semelhante houve intenção de convidar membros da gestão do IFSC presentes na reitoria e nos câmpus, bem como servidor representante da sociedade civil para a composição de grupo focal. Ao grupo foi exposto o produto educacional Observatório da Extensão produzido pela pesquisa para a coleta de impressões qualitativas acerca deste. A participação dos membros no grupo focal foi respaldada por um termo de consentimento esclarecido conforme o apêndice C.

3.8 PROCEDIMENTOS PARA A ANÁLISE DOS DADOS

A partir dos fichamentos foi realizada uma seleção de conteúdo textual para análise, interpretação e categorização dos dados para construção de indicadores que refletem a extensão no IFSC. Coube na análise e interpretação identificar as necessidades a serem manejadas na extensão e na categorização a segmentação dessas necessidades em macroindicadores ou categorias a serem publicizadas. Da mesma maneira, as entrevistas tenderam a apontar necessidades e costumes não definidos em documentos que são práticas adotadas pela gestão, assim, após assinatura do termo de consentimento livre e esclarecido individual do apêndice B, foi possível levantar quais indicadores são importantes para a gestão e sociedade civil envolta do IFSC.

Uma vez identificadas as necessidades, pôde-se convertê-las em requisitos, a modelagem de requisitos divide-se em fases de concepção, análise, construção e validação, e podem ser classificados em requisitos funcionais e não-funcionais. No que cerne a concepção e análise de requisitos, essas são abarcadas pela pesquisa exploratória bibliográfica e documental, a fase de construção e validação são atendidas pelas ferramentas de entrevistas. Por sua vez, Pressman (2011) sugere como modelagem formal de dados uma estratégia de sala limpa que tende em um processo entrópico abstrair as características divergentes das necessidades e as colocar em classificações bem definidas, macroindicadores, para então em um movimento cibernético realizar o regramento de uma sistematização de gestão. Essa técnica tem como uma das fases a estrutura de caixa onde são encadeados os requisitos em cada escopo de caixa para o ordenamento, inspeção de acurácia e verificação de impacto para garantir a conformidade da informação. Não menos importante, o autor cita a fase de verificação de correção para ajuste de entendimento e escopo de caixa, além da elaboração de um projeto formal composto pelo conjunto de caixas, que nesta pesquisa findou por desvelar a materialidade de ações de extensão por meio de indicadores quali-quantitativos.

Por fim, após o levantamento de requisitos convertidos em indicadores e então publicizados pelo produto educacional, analisou-se as impressões e contribuições qualitativas dos participantes de grupo focal sobre os painéis de observação por meio de uma breve apresentação das características,

funcionalidades e usabilidade do *software*, estimulando a interação entre os mesmos buscando a externalização de opiniões coerentes com os objetivos desta pesquisa.

3.9 ASPECTOS ÉTICOS

Alcântara (2018) cita que a ética surge na Grécia antiga quando filósofos refletindo sobre uma sociedade antropocêntrica racionaliza a construção de valores, princípios e condutas inerentes ao desenvolvimento humano com valorização do princípio da vida.

No resguardo da ética o IFSC dispõe do comitê de ética em pesquisa com seres humanos - CEPESH. O comitê recebe através da plataforma Brasil os projetos submetidos para iniciação da pesquisa científica. Todas pesquisas envolvendo seres humanos têm riscos à privacidade, segurança e saúde, por isso o comitê desfere esforços para a mitigação desses riscos e a proteção dos envolvidos na pesquisa. Para isso o comitê dispõe à comunidade pesquisadora termos de anuência que devem acompanhar o projeto.

No apêndice A segue a carta de anuência para a realização de pesquisa, nela o autor da pesquisa se compromete com a área pesquisada a obedecer as resoluções sobre ética e recebe anuência para conduzir as atividades.

No apêndice B consta o termo de consentimento livre e esclarecido (TCLE) que dá ciência sobre os riscos e benefícios para os participantes perfilados que se comprometem em envolver-se na pesquisa.

No apêndice C consta o termo de consentimento livre e esclarecido (TCLE) de grupo focal que dá ciência sobre os riscos e benefícios para os membros perfilados que o comporão ao se comprometerem com a pesquisa.

No apêndice D contém o termo de confidencialidade e sigilo em que o pesquisador se compromete com o resguardo de informações sigilosas e sensíveis à entidade e pessoas envolvidas.

No apêndice E apresenta-se o roteiro de entrevista semiestruturado a ser aplicado com os gestores dos câmpus, nele há uma breve apresentação introdutória da entrevista bem como as perguntas base.

Todos esses documentos foram aplicados de acordo com as normas e orientações do CEP SH, todos os entrevistados e os voluntários do grupo focal assinaram os termos de consentimento e permitiram a gravação do vídeo para análise posterior.

3.9.1 Riscos

No que diz respeito a pesquisa com seres humanos há fatores de riscos a serem considerados no levantamento de dados através de instrumentos como entrevistas, questionários e observações. Portanto, os documentos nos apêndices A, B e C indicam tratativas de prevenção do risco em caso de manifestação. Ainda assim o quadro 2 traz a tona com mais detalhes as precauções:

Quadro 2 - Exemplos de riscos e como preveni-los.

Riscos	Precaução/prevenção
Possibilidade de constrangimento ou desconforto ao responder à entrevista.	- Os indivíduos receberam esclarecimento prévio sobre a pesquisa através da leitura do TCLE; - A entrevista pôde ser interrompida a qualquer momento; - Foi garantida a privacidade para responder o questionário; - Participação foi voluntária; - O pesquisador esteve à disposição dos participantes para esclarecimento de dúvidas.
Quebra de sigilo/anonimato/confidencialidade	- As respostas são confidenciais e são resguardadas pelo sigilo dos pesquisadores durante a pesquisa e divulgação dos resultados, assegurado também o anonimato.
Estresse ou dano	- Assistência psicológica se necessária que fora direcionada à equipe qualificada do IFSC para encaminhamento de providências.

Cansaço ao responder às perguntas	- Questionários foram validados na Plataforma Brasil em sua versão resumida, mas ainda assim, podem se estender ao longo da entrevista; para isso foi respeitado o tempo aproximado de 1 hora, com possibilidade de realizar pausas na entrevista caso o participante apresentasse sinais de cansaço.
-----------------------------------	---

Fonte: CEP SH - IFSC (2023).

Destaca-se que na tratativa de dados virtualizados e nos descritos em documentos fica assegurada a confidencialidade de dados sensíveis de acordo com a lei geral de proteção de dados 13.709/2018. Não menos importante, os dados coletados e os termos de consentimento livre e esclarecido tem validade de armazenamento de 5 anos. Também todo custeio de pesquisa foi de responsabilidade do pesquisador.

3.9.2 Benefícios

Os benefícios diretos da pesquisa foram o fortalecimento da imagem da extensão para a sociedade diretamente envolvida com o Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina, além da promoção de uma visibilidade acurada dos dados da extensão extraídos do SIGAA para melhor tomada de decisão da gestão na busca de fomento e fortalecimento da extensão junto à comunidade interna e em fóruns e entidades externas, assim como melhorar a visibilidade da extensão para com a sociedade civil.

Como benefício indireto observou-se o fortalecimento da extensão para com os cidadãos interessados favorecidos com a visibilidade de programas, eventos, cursos, projetos e ações sociais que podem participar junto ao IFSC. Além da maior promoção da formação integrada para discentes para se tornarem cidadãos mais preparados para o mundo do trabalho. Também, em especial no grupo focal, serviu de referência para as discussões institucionais em prol da educação profissional e tecnológica e amplitude de atividades de extensão em outros níveis de educação além do ensino superior, bem como estimulou o pensamento crítico quanto à cultura de gestão de dados.

4 INDICADORES DE EXTENSÃO

4.1 REVISÃO DOCUMENTAL

Para a revisão documental foram analisados 7 (sete) documentos de abrangência nacional infraconstitucionais: Decreto Nº 7.416, de 20 de dezembro de 2010 que trata da concessão de bolsas para atividades de ensino e extensão; Lei Nº 13.005, de 25 de junho de 2014, o Plano Nacional de Educação (PNE) para 2014 a 2024; Resolução do Conselho Nacional de Educação Nº 7, de 18 de dezembro de 2018 que estabelece as diretrizes para extensão universitária conforme a meta 12.7 do PNE; Portaria do Ministério da Educação Nº 983, de 18 de novembro de 2020 que regulamenta as atividades docentes da Rede Federal de Educação Profissional, Científica e Tecnológica (EPCT); Acórdão do Tribunal de Contas da União Nº 461, de 9 de março de 2022 que propõe um modelo de indicadores de gestão e desempenho dos membros da rede federal supracitada; Portaria da Secretaria de Educação Profissional e Tecnológica do Ministério da Educação Nº 299, de 6 de maio de 2022 que dispõe sobre indicadores de pesquisa e extensão a serem utilizados pelas instituições da rede federal de EPCT; E a Lei Nº 14.695, de 10 de outubro de 2023 que abrange o acesso à bolsas de pesquisa e extensão para servidores ocupantes do cargo de técnico-administrativo em educação (TAE).

Também foram considerados 2 (dois) documentos provenientes de fóruns da extensão, a política nacional de extensão universitária de 2012 do FORPROEXT com o Conselho Nacional de Instituições da Rede Federal de Educação Profissional, Científica e Tecnológica (CONIF) trazendo uma visão ampla da extensão no Brasil com princípios e objetivos a serem alcançados com o envolvimento da sociedade em busca da transformação social, a curricularização da extensão e formação do discente, o orçamento público para educação, o estímulo no uso de tecnologias para ampliar a oferta de oportunidades e comunicação com a sociedade, o fortalecimento de parcerias interinstitucionais, o aprimoramento de políticas públicas, etc (FORPROEXT, 2012).

Por meio de estudos ao longo de três anos, com contribuição das universidades, tendo como objetivo de análise as práticas extensionistas no ensino superior e o plano de educação nacional vigente, nesse documento se amplia a

possibilidade de uma regulamentação para que TAEs se tornem bolsistas e a terem uma participação equânime ao de docentes nas atividades de extensão, sendo regulamentada somente em 2023 pela lei Nº 14.695, de 10 de outubro de 2023 que também amplia a possibilidade de TAEs exercerem o papel de coordenadores (Brasil, 2023). Além disso, a política nacional de extensão ainda destaca a importância da participação da comunidade externa à instituição de educação e do público alvo atingido por suas ações. Também discrimina que essas devem se encaixar nas áreas temáticas: comunicação, cultura, direitos humanos e justiça, tecnologia e produção, e trabalho; além que aponta para um percentual mínimo de 10% de créditos da extensão nos currículos de cursos superiores (FORPROEX, 2012).

Como complemento o outro documento considerado foi o manual de indicadores de extensão universitária elaborado por pesquisa estatística junto a comunidade acadêmica espalhada pelo país, utilizando a técnica Delphi, selecionou-se 37 (trinta e sete) indicadores quantitativos de extensão passíveis de serem computados em sistemas da informação, onde revelam-se mais características pertinentes ao fortalecimento da extensão, como: estrutura física para atividades de gestão e acadêmicas, orçamento para a extensão, participação dos servidores que atuam na extensão em fóruns, eventos e editais externos, quantidade de parcerias, além da confecção de produtos, de produções artísticas e de publicações científicas (FORPROEX, 2017).

No decreto Nº 7.416 de 2010 é dado uma visão ampla de critérios para a seleção de bolsistas discentes matriculados nas instituições, embora o documento sirva para o ensino e pesquisa, em seu art. 7º considera que atividades de extensão são cursos, eventos, programas e produtos, além disso o documento também traz a participação de técnicos-administrativos em educação (TAEs) e de docentes com papel de coordenadores (Brasil, 2010).

O Plano Nacional de Educação de 2014 a 2024 da lei Nº 13.005, com metas herdadas do plano anterior, ressalta metas voltadas para a extensão, dentre essas há o compromisso de atender ao menos 10% de “carga horária” para cursos de graduação e atividades do tipo programa e projeto, abrindo margem para a inclusão opcional de atividades de extensão na pós-graduação. Não menos importante,

destaca o envolvimento dos públicos alvo a população de jovens e adultos (EJA) e de pessoas com deficiência (PCD) (Brasil, 2014).

Os documentos subsequentes, resolução Nº 7 de 2018 do Conselho Nacional de Educação, a portaria Nº 983 de 2020 do Ministério da Educação (MEC, 2018), o acórdão Nº 461 de 2022 do Tribunal de Contas da União (TCU, 2022) que traz proposta de indicadores de extensão de interesse do controle externo (figura 1), e a portaria Nº 299 de 2022 do Ministério da Educação (MEC, 2022) que impulsiona a criação do manual de indicadores de extensão e pesquisa e inovação de 2023 (MEC, 2023) reforçam indicadores anteriormente citados, salvo a adição da prestação de serviços como uma modalidade de ação de extensão.

Figura 1 - Indicadores de extensão propostos pelo TCU.

Quadro 5 - Indicadores propostos (Extensão)

<i>Extensão</i>	<i>Indicador proposto/existente</i>	<i>O que pretende medir</i>	<i>Previsão Legal</i>
1	Quantidade anual de atividades de extensão, por modalidade	Acompanhar os tipos de atividades de extensão desenvolvidas pela Ifes	Art. 8º, da Resolução CNE/CES 7/2018
2	Público (diretamente) beneficiado por atividades de extensão, por modalidade, desenvolvidas no ano pela Ifes	Avaliar efetividade, a partir de quantitativo de pessoas diretamente atendidas em atividades de extensão, por modalidade	Art. 15, parágrafo único, da Resolução CNE/CES 7/2018
3	Número de Professores da Rede Pública atendidos por Programas e Projetos de Formação Continuada	Avaliar esforço das Ifes em ações de extensão voltadas à qualificação de docentes da educação básica	Art. 15, parágrafo único, da Resolução CNE/CES 7/2018
4	Quantitativo de pessoas atendidas com atividades de extensão no ano em relação ao total de matrículas de graduação da Ifes	Avaliar esforço de atividades de extensão em relação ao total de alunos da Ifes	Art. 15 e art. 16 da Resolução CNE/CES 7/2018
5	Número (percentual) de estudantes envolvidos em atividades de extensão	Avaliar envolvimento de estudantes em atividades de extensão	Art. 4º e art. 12, da Resolução CNE/CES 7/2018
6	Percentual (taxa) de docentes envolvidos em atividades extensão	Avaliar envolvimento de docentes em atividades de extensão	Art. 12, inciso III, e art. 18, da Resolução CNE/CES 7/2018
7	Total de técnicos envolvidos em atividades de extensão	Avaliar quantitativo de profissionais/técnicos envolvidos em atividades de extensão	Art. 18, da Resolução CNE/CES 7/2018
8	Percentual de recursos do orçamento anual destinado às atividades de extensão	Avaliar montante de recursos do orçamento anual investido em atividades de extensão	Art. 13, inciso VI, da Resolução CNE/CES 7/2018; Art. 207 da CF/1988 (autonomia de gestão financeira das Ifes) .

Fonte: Elaboração da equipe, a partir da legislação e de reuniões com stakeholders

Fonte: Acórdão do TCU Nº 461, de 9 de março de 2022.

Sob essa análise foi possível identificar temas que são abordados nos documentos e organizá-los em categorias de indicadores de extensão ou macroindicadores. Assim, por meio desses documentos nacionais infraconstitucionais e de fóruns da extensão, sob uma visão mais detalhista, extraíram-se 76 (setenta e seis) indicadores quantitativos de extensão categorizados em 13 (treze) macroindicadores conforme o quadro 3. Ao classificar um indicador de extensão buscou-se inseri-lo em macroindicador de maior coerência, um exemplo foi a aglutinação do macroindicador bolsista, pois como docentes, TAEs, discentes, e potencialmente a participantes externos podem se tornar bolsistas, fez-se mais coerente designar os indicadores quantitativos de bolsistas para o respectivo contexto tratado em outro macroindicador.

Quadro 3 - Estratificação de indicadores de extensão provenientes de documentos infraconstitucionais e de fóruns da extensão.

MACROINDICADORES	INDICADORES
1. Docentes em atividades de extensão	- quantitativo por papel de participante voluntário; - quantitativo por papel de bolsista; - quantitativo por papel de coordenador; - quantitativo por papel de orientador; - percentual de docentes envolvidos em atividades de extensão.
2. TAEs em atividades de extensão	- quantitativo por papel de participante voluntário; - quantitativo por papel de bolsista; - quantitativo por papel de coordenador; - percentual de TAEs envolvidos em atividades de extensão.
3. Discentes em atividades de extensão	- quantitativo por papel de participante voluntário; - quantitativo por papel de bolsista;

	- percentual de discentes envolvidos em ações de extensão.
4. Participantes	- quantitativo total de servidores envolvidos em atividades de extensão; - quantitativo de participantes externos envolvidos em atividades de extensão; - proporção de participantes externos por total de atividade de extensão; - proporção de participantes externos por total de participantes; - percentual de servidores envolvidos em atividades de extensão.
5. Modalidades de atividades de extensão	- quantitativo de curso; - quantitativo de evento; - quantitativo de programa; - quantitativo de produto; - quantitativo de projeto; - quantitativo de prestação de serviço; - proporção discentes com matrícula ativa por programas e projetos; - proporção discentes com matrícula ativa por programas; - proporção discentes com matrícula ativa por projetos; - proporção discentes com matrícula ativa por produtos; - proporção discentes com matrícula ativa por prestação de serviço; - proporção de atividades de extensão por modalidade projeto; - proporção de atividades de extensão por modalidade curso;

	- quantitativo de empreendimentos graduados em incubadoras resultados da extensão; - quantitativo de cooperativas populares graduadas em incubadoras resultadas da extensão.
6. Curricularização da extensão	- percentual de créditos ou carga horária destinados à extensão no currículo de graduação; - percentual de créditos ou carga horária destinados à extensão no currículo de pós-graduação; - percentual de créditos ou carga horária de ações de extensão no currículo de cursos; - quantitativo de atividades de extensão por modalidade de curso (técnico, superior, especialização, mestrado, doutorado).
7. Orçamento	- percentual de recursos financeiros do orçamento anual público aplicados em extensão; - quantitativo de recursos para extensão captados via prestação de serviços acadêmicos especializados; - quantitativo de recursos para extensão captados via editais externos.

8. Público alvo	- quantitativo de pessoas envolvidas por ações de extensão; - quantitativo de pessoas envolvidas por atividades de modalidade prestação de serviço; - quantitativo de pessoas envolvidas por atividades de modalidade programas e projetos; - quantitativo de pessoas envolvidas por atividades de modalidade cursos e eventos; - quantitativo de ações de extensão para capacitação de população EJA; - quantitativo de ações de extensão para capacitação de população PCD; - quantitativo de pessoas envolvidas por discentes de graduação por meio de ações de extensão; - quantitativo de docentes da rede pública capacitados por programas e projetos de formação continuada; - percentual de ações de extensão destinadas à inclusão de população vulnerável; - percentual de ações de extensão dirigidas às escolas públicas; - percentual de municípios atendidos pelas ações extensionistas em relação ao total de municípios que são abrangidos pela respectiva instituição de ensino/câmpus.
9. Parcerias	- percentual de ações de extensão com parcerias institucionais; - quantitativo de ações de extensão com parcerias institucionais; - percentual de representação oficial da instituição junto à sociedade civil;

	- taxa de aprovação de propostas de extensão em editais externos.
10. Produção extensionista	- quantitativo de produção de livros ou capítulos com base em resultados da extensão; - quantitativo de publicação de artigos em periódicos com base em resultados da extensão; - quantitativo de mídias de comunicações em eventos com base em resultados da extensão; - quantitativo de produções audiovisuais resultadas da extensão; - quantitativo de produções artísticas (exposições, espetáculos, outros) resultantes da extensão.
11. Áreas temáticas de extensão	- quantitativo de atividades de extensão por área da comunicação; - quantitativo de atividades de extensão por área da cultura; - quantitativo de atividades de extensão por área dos direitos humanos e justiça; - quantitativo de atividades de extensão por área da tecnologia e produção; - quantitativo de atividades de extensão por área do trabalho.
12. Infraestrutura e recursos humanos	- proporção de horas de treinamento por servidor da gestão extensionista; - proporção de participação em eventos por servidor da gestão extensionista; - proporção de participação em eventos promovidos ou apoiados pela pró-reitoria de extensão por servidor da gestão extensionista; - proporção espaço físico (área) por servidor;

	- percentual de capacidade de atendimento de demandas de transporte logístico para ações de extensão; - percentual de servidores em representação oficial da instituição junto à sociedade civil.
13. Atividades de extensão	- quantitativo total de atividades de extensão; - proporção de ações de extensão com status aprovada por servidor; - quantitativo de atividades de extensão com status favorável; - quantitativo de atividades de extensão com status desfavorável; - percentual de atividade de extensão que articularam com a pesquisa; - percentual de atividade de extensão que articularam com o ensino.

Fonte: autoria própria.

Prosseguindo com a revisão documental, sob o escopo do IFSC, foram selecionados 8 (oito) documentos normativos: Deliberação do Colegiado de Ensino, Pesquisa e Extensão (CEPE) Nº 18, de 12 de abril de 2010 que trata do voluntariado em pesquisa e extensão; Resolução do Conselho Superior (CONSUP) Nº 23, de 3 de agosto de 2012 que regulamenta parcerias de crédito a projetos de ensino, pesquisa e extensão; GB Instrução Normativa Nº 03/2016 que dispõe sobre processo de parcerias interinstitucionais; Resolução do CONSUP Nº 40, de 29 de agosto de 2016 que aprova as diretrizes de curricularização da extensão; Resolução do CONSUP Nº 48, de 24 de outubro de 2016 que dispõe sobre a modalidade de prestação de serviços à comunidade externa por meio de projetos; Resolução do CONSUP Nº 61, de 12 de dezembro de 2016 que regulamenta as atividades de extensão; Resolução do CONSUP Nº 31, de 18 de setembro de 2017 que trata da captação de apoio à realização de eventos; E Resolução do CONSUP Nº 20, de 25 de junho de 2018 que aprova o regulamento didático-pedagógico.

Desses, nota-se o quão de acordo estão com os documentos de abrangência nacional, sobretudo com os que se originam dos fóruns de extensão. Embora a Deliberação do CEPE Nº 18 de 2010 seja anterior a esses documentos, percebe-se que o IFSC já aplica princípios de acordo com a EPT que vieram a estar previstos na política nacional da extensão. No seu teor há o estímulo do serviço voluntário pelos discentes, docentes, TAEs e da comunidade externa promovendo a formação docente, a interação dialógica com a sociedade e a indissociabilidade entre ensino, pesquisa e extensão.

Da mesma forma, considerando a participação da sociedade, as Resoluções do CONSUP Nº 23 de 2012 e Nº 31 de 2017 e a Instrução Normativa Nº 03 de 2016 dão direcionamento à comunidade interna do IFSC e oportunizam a participação de outras instituições que estimulam a produção técnico científica por meio de fomentos e de partilha de ambientes e ferramentas para que ocorram as atividades de pesquisa e extensão. Em contrapartida, da contribuição da escola com a sociedade, atendendo a interação dialógica, há a Resolução do CONSUP Nº 48 de 2016 estimulando a participação da comunidade interna na prestação de serviços por meio de projetos à comunidade externa.

Não obstante, quanto à formação discente e indissociabilidade entre ensino, pesquisa e extensão, notam-se esses princípios nas Resoluções do CONSUP Nº 40 de 2016, Nº 61 de 2016 e Nº 20 de 2018. A primeira e a última, de acordo com o Plano Nacional de Educação, focam na curricularização da extensão e em como se dá nos planos pedagógicos de curso. A segunda, senão a mais importante, não só trata da formação e indissociabilidade, mas abrange as características da extensão no IFSC a fim de preparar o formando para o mundo do trabalho respeitando o desenvolvimento sustentável, a participação cidadã ativa, o convívio e desenvolvimento social. Nesse são discriminados o que são as modalidades de extensão, as áreas temáticas, o processo de submissão e aprovação de propostas por meio de editais, dos recursos e envolvimento dos participantes. Ressalta-se nesse documento o art. 10 em que determina que esses processos devem tramitar exclusivamente via SIGAA.

Ainda sob o escopo do IFSC foram selecionados 6 (seis) documentos diretivos: Relatório de avaliação dos exercícios 2019 e 2020 da Auditoria Interna, de 31 de março de 2023 que focou nos processos de gestão das atividades de

extensão; Plano de Desenvolvimento Institucional (PDI) de 2020-2024; Planejamento Estratégico Institucional (PEI) de 2020-2024; Plano de Dados Abertos (PDA) do IFSC 2021-2023; Manual de Gestão da Extensão da PROEX de 2022; Relatório de Gestão de 2023.

Seguindo uma análise cronológica, segundo o relatório da auditoria interna de 2019 e 2020, com base nos dados do SIGAA as atividades de extensão cresceram gradativamente, inferindo que cada vez mais há participação da comunidade interna na submissão de propostas com participação da comunidade externa. Em contrapartida, o mesmo documento aponta para melhorias na construção de processos e registro de dados no SIGAA, além de requerer a transparência de indicadores sobre as atividades de extensão.

Todavia, nos planos de desenvolvimento e estratégico institucional percebem-se objetivos, planos de ação e impactos que atendem os apontamentos do relatório da auditoria, como o registro mais acurado dos dados das atividades de extensão no SIGAA, maior treinamento e articulação entre os gestores de extensão dos câmpus e envolvimento dos setores no desenvolvimento da plataforma institucional. Também percebe-se o reflexo dessas ações no manual de gestão de extensão, especificamente no mapeamento dos trâmites processuais para a comunidade interna na condução de atividades de extensão no SIGAA.

O resultado dessas campanhas de gestão se observa no relatório de gestão de 2023, nesse é apresentada a estrutura do IFSC bem como são apresentados dados de desempenho do ensino, pesquisa e extensão, e discorre com afinco a respeito das estratégias adotadas e da prestação de contas. Nota-se o relatório de gestão como uma ferramenta de publicização de indicadores construído a partir da consulta manual das fontes de dados do IFSC, sobretudo o SIGAA. Esse relatório atende a 15 (quinze) indicadores qualitativos trazidos pelo manual de indicadores de extensão, a exemplo a presença da extensão na missão institucional, a estrutura organizacional, a valorização da prática extensionista na formação cidadã, sistemas informatizados de apoio à extensão, existência de critérios de curricularização, acesso e transparência das ações de extensão, etc. Apesar dos relatórios anuais de gestão atenderem esses indicadores, ainda assim o plano de dados aberto de 2021 a 2023 aponta 375 (trezentos e setenta e cinco) solicitações de acesso à informação vindos da sociedade sobre projetos, inovações, bolsas e patentes de pesquisa e

extensão. Essas solicitações, somente as de extensão, são discutidas no tópico seguinte.

Assim como nos documentos infraconstitucionais e dos fóruns de extensão, os documentos do IFSC abordam no seu texto elementos que remetem a indicadores e suas categorias, sendo assim o quadro de estratificação de indicadores passa a agregar 4 (quatro) indicadores adicionais não observáveis nos documentos externos ao IFSC: (1) quantitativo de participantes externos voluntários; (2) créditos ou carga horária em PPCs de nível médio técnico; (3) percentual de PPCs com curricularização da extensão; e (4) quantitativo de instituições ativas nos fóruns de extensão nos câmpus, totalizando 80 (oitenta) indicadores de extensão segundo o quadro 4.

Quadro 4 - Estratificação de indicadores de extensão provenientes de documentos infraconstitucionais, de fóruns da extensão, e de documentos do IFSC.

MACROINDICADORES	INDICADORES
1. Docentes em atividades de extensão	- quantitativo por papel de participante voluntário; - quantitativo por papel de bolsista; - quantitativo por papel de coordenador; - quantitativo por papel de orientador; - percentual de docentes envolvidos em atividades de extensão.
2. TAEs em atividades de extensão	- quantitativo por papel de participante voluntário; - quantitativo por papel de bolsista; - quantitativo por papel de coordenador; - percentual de TAEs envolvidos em atividades de extensão.
3. Discentes em atividades de extensão	- quantitativo por papel de participante voluntário; - quantitativo por papel de bolsista;

	- percentual de discentes envolvidos em ações de extensão.
4. Participantes	- quantitativo total de servidores envolvidos em atividades de extensão; - quantitativo de participantes externos envolvidos em atividades de extensão; - quantitativo de participantes externos envolvidos em atividades de extensão com papel de voluntário; (AGREGADO) - proporção de participantes externos por total de atividade de extensão; - proporção de participantes externos por total de participantes; - percentual de servidores envolvidos em atividades de extensão.
5. Modalidades de atividades de extensão	- quantitativo de curso; - quantitativo de evento; - quantitativo de programa; - quantitativo de produto; - quantitativo de projeto; - quantitativo de prestação de serviço; - proporção discentes com matrícula ativa por programas e projetos; - proporção discentes com matrícula ativa por programas; - proporção discentes com matrícula ativa por projetos; - proporção discentes com matrícula ativa por produtos; - proporção discentes com matrícula ativa por prestação de serviço;

	- proporção de atividades de extensão por modalidade projeto; - proporção de atividades de extensão por modalidade curso; - quantitativo de empreendimentos graduados em incubadoras resultados da extensão; - quantitativo de cooperativas populares graduadas em incubadoras resultadas da extensão.
6. Curricularização da extensão	- percentual de créditos ou carga horária destinados à extensão no currículo de nível técnico; (AGREGADO) - percentual de créditos ou carga horária destinados à extensão no currículo de graduação; - percentual de créditos ou carga horária destinados à extensão no currículo de pós-graduação; - percentual de créditos ou carga horária de ações de extensão no currículo de cursos; - quantitativo de atividades de extensão por modalidade de curso (técnico, superior, especialização, mestrado, doutorado); - Percentual de PPC com curricularização da extensão e atividades complementares. (AGREGADO)

7. Orçamento	- percentual de recursos financeiros do orçamento anual público aplicados em extensão; - quantitativo de recursos para extensão captados via prestação de serviços acadêmicos especializados; - quantitativo de recursos para extensão captados via editais externos.
8. Público alvo	- quantitativo de pessoas envolvidas por ações de extensão; - quantitativo de pessoas envolvidas por atividades de modalidade prestação de serviço; - quantitativo de pessoas envolvidas por atividades de modalidade programas e projetos; - quantitativo de pessoas envolvidas por atividades de modalidade cursos e eventos; - quantitativo de ações de extensão para capacitação de população EJA; - quantitativo de ações de extensão para capacitação de população PCD; - quantitativo de pessoas envolvidas por discentes de graduação por meio de ações de extensão; - quantitativo de docentes da rede pública capacitados por programas e projetos de formação continuada; - percentual de ações de extensão destinadas à inclusão de população vulnerável; - percentual de ações de extensão dirigidas às escolas públicas; - percentual de municípios atendidos pelas ações extensionistas em relação ao total de

	municípios que são abrangidos pela respectiva instituição de ensino/câmpus.
9. Parcerias	- percentual de ações de extensão com parcerias institucionais; - quantitativo de ações de extensão com parcerias institucionais; - percentual de representação oficial da instituição junto à sociedade civil; - taxa de aprovação de propostas de extensão em editais externos; - quantitativo de instituições ativas nos fóruns de extensão e relações externas no câmpus. (AGREGADO)
10. Produção extensionista	- quantitativo de produção de livros ou capítulos com base em resultados da extensão; - quantitativo de publicação de artigos em periódicos com base em resultados da extensão; - quantitativo de mídias de comunicações em eventos com base em resultados da extensão; - quantitativo de produções audiovisuais resultadas da extensão; - quantitativo de produções artísticas (exposições, espetáculos, outros) resultantes da extensão.
11. Áreas temáticas de extensão	- quantitativo de atividades de extensão por área da comunicação; - quantitativo de atividades de extensão por área da cultura; - quantitativo de atividades de extensão por área dos direitos humanos e justiça;

	- quantitativo de atividades de extensão por área da tecnologia e produção; - quantitativo de atividades de extensão por área do trabalho.
12. Infraestrutura e recursos humanos	- proporção de horas de treinamento por servidor da gestão extensionista; - proporção de participação em eventos por servidor da gestão extensionista; - proporção de participação em eventos promovidos ou apoiados pela pró-reitoria de extensão por servidor da gestão extensionista; - proporção espaço físico (área) por servidor; - percentual de capacidade de atendimento de demandas de transporte logístico para ações de extensão; - percentual de servidores em representação oficial da instituição junto à sociedade civil.
13. Atividades de extensão	- quantitativo total de atividades de extensão; - proporção de ações de extensão com status aprovada por servidor; - quantitativo de atividades de extensão com status favorável; - quantitativo de atividades de extensão com status desfavorável; - percentual de atividade de extensão que articularam com a pesquisa; - percentual de atividade de extensão que articularam com o ensino.

Fonte: autoria própria.

4.2 SOLICITAÇÕES DE ACESSO À INFORMAÇÃO DA SOCIEDADE CIVIL

Como abordado pelo plano de dados abertos de 2021 a 2023, a sociedade tem buscado informações sobre as atividades do IFSC e demandado transparência por meio do sistema e-SIC - renomeado para FalaBR e mantido pelo TCU - de acesso da ouvidoria, com visibilidade restrita às demandas do IFSC. Considerando isso, esta pesquisa requisitou à ouvidoria do IFSC, respaldada pelo processo Nº 23292.017805/2024-35, o acesso às solicitações para análise. O mesmo foi concedido, ressaltando o respeito à Lei Geral de Proteção de Dados Pessoais, por tempo limitado com o papel de observador, para as solicitações de acesso à informação. Todavia, ao aplicar os filtros de busca de acesso à informação para o escopo de tempo de 2020 (dois mil e vinte) a 2024 (dois mil e vinte e quatro) não foi retornada nenhuma manifestação.

Em consulta à Coordenadoria de Gestão de Dados do IFSC (CGD), foi levantada a hipótese de que o histórico de dados fora migrado para a plataforma da Lei de Acesso à Informação, a BuscaLAI, mantida pela CGU, durante a alteração da plataforma e-SIC em FalaBR. Ao buscar as solicitações na plataforma BuscaLAI, no mesmo escopo de anos, houve o retorno de 5 (cinco) arquivos compactados contendo em cada 6 (seis) outros arquivos em formato .csv com os nomes [data da busca]_Pedidos_csv_[ano do escopo].csv, [data da busca]_PedidosLinkArquivo_csv_[ano do escopo].csv, [data da busca]_Recursos_csv_[ano do escopo].csv, [data da busca]_RecursosLinkArquivo_csv_[ano do escopo].csv, [data da busca]_SolicitantesPedidos_csv_[ano do escopo].csv e [data da busca]_SolicitantesRecursos_csv_[ano do escopo].csv.

Ao inspecioná-los notou-se que as solicitações de acesso à informação com mapeamento das entidades públicas destinatárias são identificadas nos arquivos [data da busca]_Pedidos_csv_[ano do escopo].csv. No entanto, a soma das solicitações de 2020 a 2024 resultaram em 403.293 (quatrocentos e três mil, duzentos e noventa e três) registros. Logo, por ser inviável a análise de cada manifestação, recorreu-se à confecção de *scripts* em linguagem Python para a extração de solicitações de acesso à informação direcionadas ao IFSC de 2020 a 2024.

O primeiro passo foi elaborar um *script* (código 1) que agregasse todos os arquivos [data da busca]_Pedidos_csv_[ano do escopo].csv em apenas um com as solicitações destinadas ao IFSC. No exemplo posterior, os arquivos de pedidos são listados na variável `arquivos`, em seguida são percorridos todos os registros de arquivos e salvando os que tem a coluna 'orgaoDestinatario' correspondente a 'IFSC - Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina' em uma estrutura de matriz chamada de *dataframe* para então convertê-la no arquivo Pedidos_IFSC_2020_2024.csv.

Código 1 - Script Python de seleção de solicitações de acesso à informação direcionadas ao IFSC entre 2020 e 2024.

```
import pandas as pd
import numpy as np

# Lista de arquivos CSV
arquivos = [
    '20240916_Pedidos_csv_2020.csv',
    '20240916_Pedidos_csv_2021.csv',
    '20240916_Pedidos_csv_2022.csv',
    '20240916_Pedidos_csv_2023.csv',
    '20240916_Pedidos_csv_2024.csv',
]

# Lista para armazenar os DataFrames filtrados
dataframes = []

# Itera sobre os arquivos
for arquivo in arquivos:
    try:
        # Lê o arquivo CSV com encoding 'utf-16' e delimitador ';'
        df = pd.read_csv(arquivo, encoding='utf-16', sep=';')

        # Filtra as linhas onde a coluna 'orgaoDestinatario' contém 'IFSC'
        # Substitui valores NaN na coluna por uma string vazia para que a
        # comparação seja possível
        df['OrgaoDestinatario'] = df['OrgaoDestinatario'].fillna("")
        df_filtrado = df[df['OrgaoDestinatario'].str.contains('IFSC - Instituto
Federal de Educação, Ciência e Tecnologia de Santa Catarina')]

        # Adiciona o DataFrame filtrado à lista
        dataframes.append(df_filtrado)
    except FileNotFoundError:
        print(f'Arquivo {arquivo} não encontrado.')
```

```

except pd.errors.ParserError as e:
    print(f'Erro ao processar {arquivo}: {e}')

# Concatena os DataFrames filtrados
df_final = pd.concat(dataframes)
# Salva o DataFrame final como um arquivo CSV
df_final.to_csv('Pedidos_IFSC_2020_2024.csv', index=False, encoding='utf-8',
sep=';')
print('Arquivo "Pedidos_IFSC_2020_2024.csv" criado com sucesso!')
Fonte: autoria própria.

```

Ao analisar o arquivo Pedidos_IFSC_2020_2024.csv observou-se que os registros foram reduzidos para 664 (seiscentas e sessenta e quatro) solicitações com conteúdo direcionado a várias frentes da EPT. Para tal, outro *script* (código 2) foi elaborado para filtrar as que são destinadas à extensão. Nele o arquivo Pedidos_IFSC_2020_2024.csv é inserido diretamente em um *dataframe*, sobre esse é aplicada uma máscara de filtragem em que se verificam as linhas de registros onde em suas colunas 'DetalhamentoSolicitacao', 'FormaResposta', 'AssuntoPedido', 'SubAssuntoPedido', 'Tag', 'Resposta', 'Decisao' e 'EspecificacaoDecisao' citam a extensão para então transformar o resultado no arquivo Pedidos_Extensao_IFSC_2020_2024.csv.

Código 2 - Script Python de seleção de solicitações de acesso à informação direcionadas à extensão do IFSC entre 2020 e 2024.

```

import pandas as pd

# Carrega o arquivo CSV
df = pd.read_csv("Pedidos_IFSC_2020_2024.csv", encoding='utf-8', sep=';')
# Lista de colunas a serem verificadas
colunas = ['DetalhamentoSolicitacao', 'FormaResposta', 'AssuntoPedido',
'SubAssuntoPedido', 'Tag', 'Resposta', 'Decisao', 'EspecificacaoDecisao']
# Cria uma máscara booleana para identificar linhas que contêm "extensão" ou
"extensao"
mascara = df[colunas].apply(lambda x:
x.str.lower().str.contains('extensão|extensao')).any(axis=1)
# Filtra o DataFrame usando a máscara
df_filtrado = df[mascara]
# Salva o DataFrame filtrado em um novo arquivo CSV
df_filtrado.to_csv('Pedidos_Extensao_IFSC_2020_2024.csv', index=False,
encoding='utf-8', sep=';')
print('Arquivo "Pedidos_Extensao_IFSC_2020_2024.csv" criado com sucesso!')
Fonte: autoria própria.

```

Com uma quantidade de registros reduzida para 35 (trinta e cinco) ainda se fez necessário criar um *script* conforme o código 3 para converter o arquivo Pedidos_Extensao_IFSC_2020_2024.csv para o formato .xlsx com o objetivo de manipulá-lo em ferramentas de tabulação de dados para análise de conteúdo.

Código 3 - Script Python de conversão de formato de arquivo de .csv para .xlsx.

```
import pandas as pd

# Nome do arquivo CSV de entrada
arquivo_csv = 'Pedidos_Extensao_IFSC_2020_2024.csv'
# Nome do arquivo Excel de saída
arquivo_xlsx = 'Pedidos_Analisados_Extensao_IFSC_2020_2024.xlsx'
# Lê o arquivo CSV com separador ';'
df = pd.read_csv(arquivo_csv, sep=';')
# Salva o DataFrame como um arquivo Excel
df.to_excel(arquivo_xlsx, index=False)
print(f'Arquivo {arquivo_csv} convertido para {arquivo_xlsx} com sucesso!')
```

Fonte: autoria própria.

Dentre essas 35 (trinta e cinco) solicitações 5 (cinco) foram descartadas por serem requisições de dados qualitativos que são atendidos por documentos institucionais, ou que são divulgados no sítio institucional, ou que trazem especificidades respondidas por correio eletrônico, como informação sobre um evento ocorrido, atividade de extensão de um currículo de curso e resultado de seleção por edital. Das 30 (trinta) solicitações restantes 15 (quinze) foram descartadas, pois apesar de citarem a extensão, o teor do texto pedia para demandas técnicas de ensino, pesquisa ou sobre decisões de gestão e governança do IFSC.

Portanto, restaram 15 (quinze) solicitações de acesso à informação sobre a extensão direcionadas ao IFSC entre os anos de 2020 a 2024, nas quais foi possível extrair indicadores de extensão, o quadro 5 apresenta os indicadores extraídos. Nessas foram levantados 11 (onze) indicadores já identificados pela revisão documental, mas 9 (nove) outros não identificados na etapa anterior foram agregados na listagem completa conforme o quadro 6.

Quadro 5 - Estratificação de indicadores de extensão provenientes de solicitações da sociedade de dados extensionistas.

MACROINDICADORES	INDICADORES	ID SOLICITAÇÃO(ÕES)
1. Docentes em atividades de extensão	- quantitativo por papel de bolsista;	238009
	- quantitativo de docentes envolvidos em atividades de extensão. (AGREGADO)	
2. TAEs em atividades de extensão	- quantitativo por papel de bolsista;	238009, 3294411
	- quantitativo de TAEs envolvidos em atividades de extensão. (AGREGADO)	238009
3. Discentes em atividades de extensão	- quantitativo por papel de bolsista;	3229626, 6407192
	- quantitativo de discentes envolvidos em atividades de extensão. (AGREGADO)	238009
5. Modalidades de atividades de extensão	- quantitativo de projeto;	3229626, 2383009, 4756658
	- quantitativo de discentes com matrícula ativa em projetos; (AGREGADO)	3229626
	- quantitativo de incubadoras; (AGREGADO)	6159185, 6159187

	- quantitativo de cooperativas populares graduadas em incubadoras resultadas da extensão.	7512020
7. Orçamento	- percentual de recursos financeiros do orçamento anual público aplicados em extensão;	5880549, 6159185, 6159187, 6407192
	- quantitativo de recursos para extensão captados via editais externos;	6159185, 6159187
	- quantitativo de recurso aplicado por edital interno de extensão; (AGREGADO)	7394845
	- quantitativo de recurso repassado aos coordenadores de atividade de extensão; (AGREGADO)	6407192
	- quantitativo de recurso repassado aos bolsistas discentes. (AGREGADO)	
8. Público alvo	- quantitativo de ações de extensão para capacitação de população EJA;	5495713
	- percentual de municípios atendidos pelas ações extensionistas em relação ao total de municípios que são abrangidos	5873972, 6107362

	pela respectiva instituição de ensino/câmpus.	
10. Produção extensionista	- quantitativo de revistas e periódicos institucionais; (AGREGADO)	2086627, 2525317
11. Áreas temáticas de extensão	- quantitativo de atividades de extensão por área da cultura;	2383009
13. Atividades de extensão	- quantitativo total de atividades de extensão.	2383009, 6159185, 6159187

Fonte: autoria própria.

Quadro 6 - Estratificação de indicadores de extensão provenientes de documentos infraconstitucionais, de fóruns da extensão, de documentos do IFSC, e de solicitações da sociedade de dados extensionistas.

MACROINDICADORES	INDICADORES
1. Docentes em atividades de extensão	- quantitativo por papel de participante voluntário; - quantitativo por papel de bolsista; - quantitativo por papel de coordenador; - quantitativo por papel de orientador; - quantitativo de docentes envolvidos em atividades de extensão; (AGREGADO) - percentual de docentes envolvidos em atividades de extensão.
2. TAEs em atividades de extensão	- quantitativo por papel de participante voluntário; - quantitativo por papel de bolsista; - quantitativo por papel de coordenador;

	- quantitativo de TAES envolvidos em atividades de extensão; (AGREGADO) - percentual de TAEs envolvidos em atividades de extensão.
3. Discentes em atividades de extensão	- quantitativo por papel de participante voluntário; - quantitativo por papel de bolsista; - quantitativo de discentes envolvidos em atividades de extensão; (AGREGADO) - percentual de discentes envolvidos em ações de extensão.
4. Participantes	- quantitativo total de servidores envolvidos em atividades de extensão; - quantitativo de participantes externos envolvidos em atividades de extensão; - quantitativo de participantes externos envolvidos em atividades de extensão com papel de voluntário; - proporção de participantes externos por total de atividade de extensão; - proporção de participantes externos por total de participantes; - percentual de servidores envolvidos em atividades de extensão.

5. Modalidades de atividades de extensão	- quantitativo de curso; - quantitativo de evento; - quantitativo de programa; - quantitativo de produto; - quantitativo de projeto; - quantitativo de prestação de serviço; - proporção discentes com matrícula ativa por programas e projetos; - proporção discentes com matrícula ativa por programas; - proporção discentes com matrícula ativa por projetos; - quantitativo de discentes com matrícula ativa em projetos; (AGREGADO) - proporção discentes com matrícula ativa por produtos; - proporção discentes com matrícula ativa por prestação de serviço; - proporção de atividades de extensão por modalidade projeto; - proporção de atividades de extensão por modalidade curso; - quantitativo de incubadoras; (AGREGADO) - quantitativo de empreendimentos graduados em incubadoras resultados da extensão; - quantitativo de cooperativas populares graduadas em incubadoras resultadas da extensão.
---	--

6. Curricularização da extensão	- percentual de créditos ou carga horária destinados à extensão no currículo de nível técnico; - percentual de créditos ou carga horária destinados à extensão no currículo de graduação; - percentual de créditos ou carga horária destinados à extensão no currículo de pós-graduação; - percentual de créditos ou carga horária de ações de extensão no currículo de cursos; - quantitativo de atividades de extensão por modalidade de curso (técnico, superior, especialização, mestrado, doutorado); - Percentual de PPC com curricularização da extensão e atividades complementares.
7. Orçamento	- percentual de recursos financeiros do orçamento anual público aplicados em extensão; - quantitativo de recursos para extensão captados via prestação de serviços acadêmicos especializados; - quantitativo de recursos para extensão captados via editais externos; - quantitativo de recurso aplicado por edital interno de extensão; (AGREGADO) - quantitativo de recurso repassado aos coordenadores de atividade de extensão; (AGREGADO)

	- quantitativo de recurso repassado aos bolsistas discentes. (AGREGADO)
8. Público alvo	- quantitativo de pessoas envolvidas por ações de extensão; - quantitativo de pessoas envolvidas por atividades de modalidade prestação de serviço; - quantitativo de pessoas envolvidas por atividades de modalidade programas e projetos; - quantitativo de pessoas envolvidas por atividades de modalidade cursos e eventos; - quantitativo de ações de extensão para capacitação de população EJA; - quantitativo de ações de extensão para capacitação de população PCD; - quantitativo de pessoas envolvidas por discentes de graduação por meio de ações de extensão; - quantitativo de docentes da rede pública capacitados por programas e projetos de formação continuada; - percentual de ações de extensão destinadas à inclusão de população vulnerável; - percentual de ações de extensão dirigidas às escolas públicas; - percentual de municípios atendidos pelas ações extensionistas em relação ao total de municípios que são abrangidos pela respectiva instituição de ensino/câmpus.

9. Parcerias	- percentual de ações de extensão com parcerias institucionais; - quantitativo de ações de extensão com parcerias institucionais; - percentual de representação oficial da instituição junto à sociedade civil; - taxa de aprovação de propostas de extensão em editais externos; - quantitativo de instituições ativas nos fóruns de extensão e relações externas no câmpus.
10. Produção extensionista	- quantitativo de revistas e periódicos institucionais; (AGREGADO) - quantitativo de produção de livros ou capítulos com base em resultados da extensão; - quantitativo de publicação de artigos em periódicos com base em resultados da extensão; - quantitativo de mídias de comunicações em eventos com base em resultados da extensão; - quantitativo de produções audiovisuais resultadas da extensão; - quantitativo de produções artísticas (exposições, espetáculos, outros) resultantes da extensão.

11. Áreas temáticas de extensão	- quantitativo de atividades de extensão por área da comunicação; - quantitativo de atividades de extensão por área da cultura; - quantitativo de atividades de extensão por área dos direitos humanos e justiça; - quantitativo de atividades de extensão por área da tecnologia e produção; - quantitativo de atividades de extensão por área do trabalho.
12. Infraestrutura e recursos humanos	- proporção de horas de treinamento por servidor da gestão extensionista; - proporção de participação em eventos por servidor da gestão extensionista; - proporção de participação em eventos promovidos ou apoiados pela pró-reitoria de extensão por servidor da gestão extensionista; - proporção espaço físico (área) por servidor; - percentual de capacidade de atendimento de demandas de transporte logístico para ações de extensão; - percentual de servidores em representação oficial da instituição junto à sociedade civil.
13. Atividades de extensão	- quantitativo total de atividades de extensão; - proporção de ações de extensão com status aprovada por servidor; - quantitativo de atividades de extensão com status favorável;

	- quantitativo de atividades de extensão com status desfavorável; - percentual de atividade de extensão que articularam com a pesquisa; - percentual de atividade de extensão que articularam com o ensino.
--	---

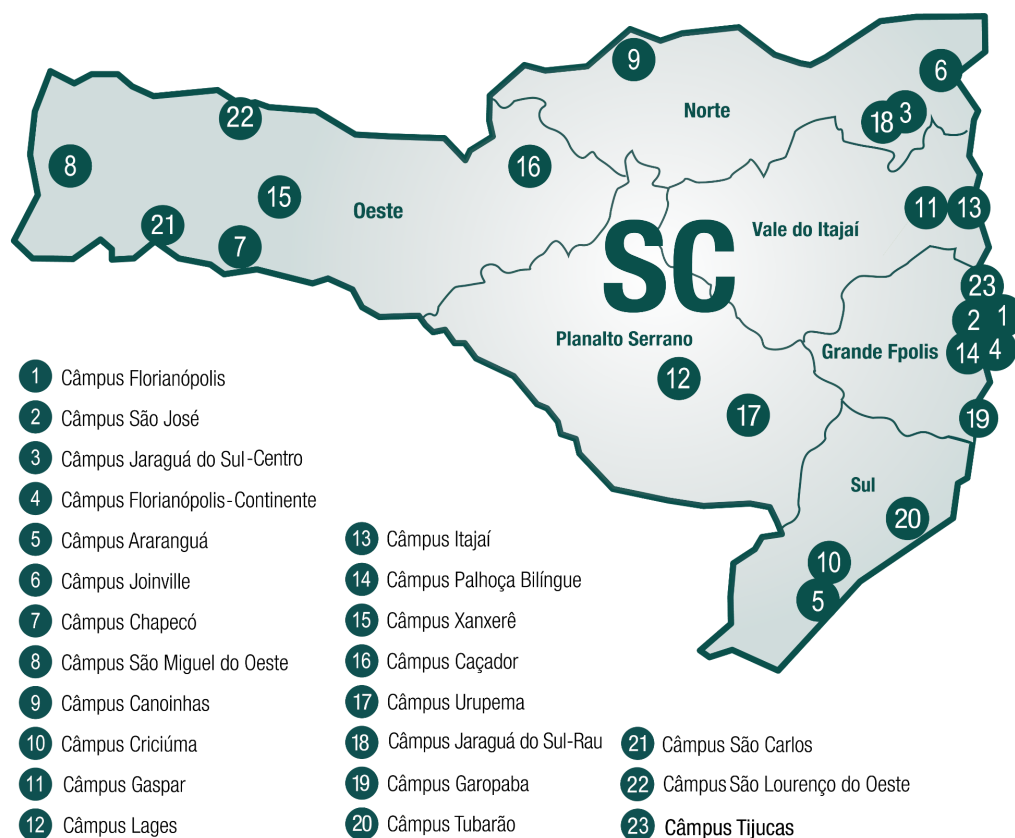
Fonte: autoria própria.

4.3 ENTREVISTAS COM A GESTÃO DA EXTENSÃO

A última etapa de identificação dos identificadores de extensão contou com as entrevistas semiestruturadas conforme o roteiro previsto no apêndice E, com participação voluntária de gestores da extensão do IFSC mediante a assinatura do termo de consentimento livre e esclarecido individual constante no apêndice B.

O IFSC está presente no estado por meio de 22 (vinte e dois) câmpus consolidados, tendo o câmpus São Lourenço do Oeste com características de avançado até 2024 (dois mil e vinte e quatro), e em processo de consolidação iniciada em 2024 (dois mil e vinte e quatro) há o câmpus Tijucas, totalizando 23 (vinte e três) câmpus e a reitoria sob a gestão do IFSC, a figura 2 exibe a lista de câmpus do IFSC por região e ordem de criação. Dentre os consolidados optou-se por retirar uma amostra de 7 (sete) voluntários da gestão de extensão, sendo 1 (um) representante de cada região mais 1 (um) da reitoria, para manter o anonimato desses esta pesquisa se dirige aos voluntários conforme a região.

Figura 2 - Lista de câmpus do IFSC por região em ordem de criação.



Fonte: Sítio institucional do IFSC.

O primeiro contato com os câmpus se deu pelo número de telefone cadastrado no sítio institucional, para os que atenderam fez-se uma breve apresentação da pesquisa para que entendessem o papel do voluntário nela. Em alguns casos o ouvinte não fazia parte de alguma unidade gestora da extensão, mas através deste foi possível ter indicação de servidor que pudesse contribuir. Uma vez a comunicação estabelecida entre todos os selecionados, foi encaminhado um e-mail formalizando o convite dando ciência aos chefes imediatos.

Ao todo foram contatados 11 (onze) câmpus mais a reitoria para a seleção da amostra, isso se deu pois existiram situações de impedimento da condução da pesquisa, a saber, na região Norte o indicado do primeiro câmpus contatado caiu nos critérios exclusão. Na região do Vale do Itajaí, seja pela proximidade do evento da Semana Nacional de Ciência e Tecnologia em que muitos estão em atividades externas ao câmpus ou por outro motivo, não foi conseguido contato por telefone ou

e-mail com o primeiro câmpus selecionado. O mesmo se deu na região Oeste, não obteve-se retorno do primeiro câmpus selecionado, e o segundo indicado da região não atendeu aos critérios de inclusão. Na região do Planalto Serrano que houve uma situação diferente, o indicado pela gestão do primeiro contatado estava de licença. Porém, dentre os que foram possíveis conversar, notou-se uma boa disposição em contribuir para esta pesquisa.

As entrevistas ocorreram por videoconferência, entre as datas de 24 (vinte e quatro) de outubro a 6 (seis) de novembro de 2024 (dois mil e vinte e quatro) no turno matutino ou vespertino, dentro do horário de trabalho, respeitando os dias e horários de descanso do entrevistado. Houve anuência de gravação para análise posterior, assim também, por ser uma entrevista semiestruturada, foi possível ter conversas fluidas e confortáveis sobre a atuação da extensão em cada região com cenários e características distintas.

A primeira pessoa entrevistada, docente da Grande Florianópolis, trabalha na região há mais ou menos 15 (quinze) anos, envolvida nas ciências exatas, já experienciou coordenar atividades de extensão e há 2 (dois) anos atua como gestora de extensão. A coordenação de extensão tem incentivado a participação dos colegas, e o câmpus, focado em cursos das áreas exatas, tem se destacado na submissão de atividades em editais, frequentemente superando as expectativas de projetos contemplados. A relação com a comunidade externa é ativa, com o câmpus sendo procurado por escolas e outras instituições interessadas em seus projetos. O papel da extensão nessa região é tão presente que antes os coordenadores das atividades de extensão careciam de parcerias para implementá-las, e atualmente a demanda de parceria parte da própria comunidade. Os discentes do câmpus, por serem membros das atividades, acabam por divulgá-las e servem como ponte entre o câmpus e a comunidade externa. Os servidores, especialmente docentes, demonstram interesse na extensão, buscando orientação sobre demandas da comunidade e parcerias, porém há expectativa de maior envolvimento dos TAEs.

Outra pessoa entrevistada é TAE, atua na EPT há pelo menos 10 (dez) anos, sendo 7 (sete) na região Norte após ser redistribuída. Apesar de ser formada nas ciências da biologia, por natureza do seu cargo, trabalhou com foco em atividades administrativas, na secretaria de registro acadêmico e, há pouco mais de um ano, assumiu a gestão da extensão em seu câmpus, porém nunca participou como

membro de alguma atividade de extensão. No câmpus, percebe-se um crescente interesse na submissão de projetos de extensão por docentes e TAEs, com aumento da participação e maior demonstração de interesse destes últimos. No entanto, há dúvidas por parte da comunidade interna sobre como estruturar essas ações na curricularização da extensão. A relação com a comunidade externa ainda precisa ser fortalecida, pois a percepção desses sobre o IFSC é superficial, sendo visto apenas como uma escola para jovens. A entrevistada destaca a necessidade de ampliar essa visão e reforçar o papel da instituição na transformação social, sem cair no assistencialismo. Os discentes do câmpus, em sua maioria, ainda não compreendem bem o conceito de extensão, muitas vezes participando de atividades extensionistas sem perceber. Além disso, há o apontamento na superação de desafios no acesso da comunidade e dos discentes aos canais de comunicação do IFSC, especialmente ao site institucional, no uso do sistema acadêmico SIGAA, e na infraestrutura logística para eventos extensionistas.

Já na região Sul, a pessoa voluntária atua há 11 (onze) anos como TAE, sendo gestora de extensão há aproximadamente 5(cinco) anos. Graduada em administração, com especialização em gestão pública e mestrado em tecnologias da informação e comunicação, seu estudo, por meio da extensão, envolveu a inserção de tecnologias para documentar a tradição oral de comunidades indígenas. Sua trajetória na extensão inclui a implantação do programa Mulheres Mil promovido por editais do IFSC, além da coordenação e participação como membro em atividades extensionistas em seu câmpus. Ela percebe um aumento no envolvimento dos docentes devido à curricularização da extensão, mas acredita que a participação dos TAEs ainda é baixa e poderia ser incentivada. Apesar do interesse dos servidores, esses ainda requisitam por capacitação sobre os processos da extensão, e buscam a gestão por esclarecimentos sobre editais e na submissão de propostas, também destaca que a documentação de uso do SIGAA ainda requer aprimoramentos. Em relação ao envolvimento dos discentes, esses poucos interagem com a coordenação, normalmente levam as demandas para o(a) coordenador(a) da ação de extensão. O câmpus mantém parcerias com empresas para oferta de cursos de extensão dentro dessas instituições, além de projetos voltados a bairros locais e ao centro de convivência de idosos.

A participação da região do Planalto Serrano contou com a entrevista de uma pessoa também TAE que atua há 10 (dez) anos em seu câmpus. Com formação em música, já foi docente em outra instituição e pôde observar a formação discente sob outra perspectiva. Tem experiência na coordenação e participação de projetos de extensão, e há 6 (seis) anos está na gestão da extensão. Ela destaca que a extensão no câmpus tem a participação ativa de servidores com perfil extensionista, porém alguns docentes acabam se envolvendo por conta da obrigatoriedade da curricularização da extensão nos cursos de graduação. Os TAEs mantêm uma presença significativa na extensão, embora ainda haja dificuldades na compreensão do processo de submissão de ações. A relação com a comunidade externa é considerada positiva, com uma boa quantidade de extensionistas atuando como intermediários entre o IFSC e os parceiros. No entanto, a maioria das parcerias ocorrem informalmente, sem grandes compromissos financeiros ou institucionais. O câmpus é precursor no IFSC na transparência de dados da pesquisa e extensão através de painéis de observação, no entanto ela destaca que existem dificuldades em extrair informações necessárias do SIGAA para a alimentação manualmente desses painéis que não são extensos.

Do Vale do Itajaí houve a participação de uma pessoa com formação linguística. Atuante há 8 (oito) anos como docente no IFSC, há 6 (seis) meses assumiu a lotação e gestão da coordenação de extensão. No entanto, sua experiência com a extensão começou enquanto docente, participando de projetos e eventos. Embora há pouco tempo como gestora, percebe que no câmpus os servidores enfrentam desafios na submissão de propostas de atividades extensionistas, o que leva a recusas ocasionais dessas, em contrapartida há servidores experientes que conseguem submeter e desenvolver ações bem-sucedidas, envolver a comunidade, e que podem ser agentes de transformação na capacitação da comunidade interna nos trâmites da extensão. Da mesma forma que em outras regiões, os servidores revelam à gestão a dificuldade no uso do SIGAA, em especial os com pouco tempo de atuação na EPT. Também aponta que há ruídos com relação à imagem do IFSC pela comunidade externa, muitos desconhecem a atuação da instituição na região, e vê que os eventos promovidos pela extensão são essenciais para atender essa demanda reprimida, mas reconhece que os próprios eventos podem ser aprimorados para maior exposição das

atividades de interesse da sociedade, e que a manutenção orçamentária se faz necessária nesse cenário.

Na região Oeste, por ter grande atividade do setor primário, é esperado que os câmpus do IFSC ali tenham essa expertise, a pessoa entrevistada dessa região, docente zootecnista, atuou como extensionista rural pouco antes de receber nomeação para trabalhar no IFSC onde está há 4 (quatro) anos. Tem experiência em coordenar projetos de extensão no incentivo à cultura sustentável de meliponários na comunidade do entorno do seu câmpus, atualmente ocupa a função de chefia na coordenação de extensão, e ainda se mantém envolvida com atividades de extensão curricularizadas. O câmpus tem um forte envolvimento em ações extensionistas na produção rural, agroindústria e conscientização ambiental, mas também nas iniciativas culturais, como exposição de obras, dança regional, banda e teatro. Apesar dos projetos exitosos, a entrevistada percebe baixa visibilidade do IFSC na região e gostaria de fortalecer sua imagem, destacando a diferença com instituições de outros estados que têm maior reconhecimento local enquanto o IFSC tem sua marca confundida com a de uma escola de idiomas. No entanto, a relação com o público externo já envolvido com o IFSC é sólida, especialmente por meio dos próprios discentes do meio rural, que levam conhecimento para suas comunidades e ambiente familiar. Outro exemplo exitoso é a parceria do câmpus com escolas da região no incentivo ao cultivo de hortas comunitárias e consumo de alimentos. Entre os servidores, surgem várias dúvidas burocráticas sobre a submissão de atividades de extensão. Já os discentes raramente procuram a coordenação, exceto para informações sobre bolsas ou para buscar oportunidades de participação em atividades de extensão.

Por fim, a pessoa entrevistada da Reitoria, atua no IFSC há 14 (quatorze) anos, removida de um dos câmpus, além da docência, ocupou funções de gestão na pesquisa e extensão, que permitiram que passasse a atuar na gestão da extensão na Reitoria, onde está há 3 (três) anos. Sua atuação esteve mais focada na gestão da extensão do que no envolvimento direto nas ações de extensão, ainda assim, por ter um perfil extensionista ainda se faz presente como membro de projetos. Além disso, tem a oportunidade de representar o IFSC em fóruns de abrangência nacional e regional. A Reitoria articula a extensão entre os câmpus, lança editais com e sem financiamento, e promove o diálogo com a sociedade. Mesmo com essa

centralização, sendo comum que a comunidade externa busque os câmpus, eventualmente surge a procura por parcerias e consultas à informação. Essas que são manifestadas por e-mail, pela ouvidoria ou plataforma FalaBr. A gestão da extensão na reitoria recebe as requisições de acesso a dados, de melhorias nos processos e na sistematização desses pelos câmpus. Em paralelo, há sempre interação com os setores de tecnologia da informação que customizam o SIGAA a fim de diminuir a burocracia e melhorar a cultura de gestão de dados.

Ao final das conversas, os entrevistados foram apresentados ao conceito de *softwares* com painéis de observação, ou observatórios. Isso teve como finalidade ambientá-los da expectativa de se dar transparência às informações da extensão contidas no SIGAA por meio de um produto educacional do tipo observatório. Uma vez superada a apresentação, esses foram questionados, com respeito às suas considerações, sobre quais indicadores de extensão eles percebem serem relevantes para suas atividades de gestão e para envolvimento da sociedade civil. O resultado é observado no quadro 7, nele consta uma listagem de 77 (setenta e sete) indicadores, sendo 24 (vinte e quatro) ainda não identificados e agregados à estratificação geral conforme o quadro 8 com o total de 113 (cento e treze) indicadores.

Quadro 7 - Estratificação de indicadores de extensão provenientes das entrevistas semiestruturadas com a gestão da extensão do IFSC.

MACROINDICADORES	INDICADORES	REGIÃO(ÕES) ENTREVISTADA(S)
1. Docentes em atividades de extensão	- quantitativo por papel de participante voluntário;	Oeste
	- quantitativo por papel de bolsista;	
	- quantitativo de docentes que submetem ações de extensão; (AGREGADO)	Planalto Serrano

	- quantitativo de docentes envolvidos em atividades de extensão;	Reitoria
	- percentual de docentes envolvidos em atividades de extensão.	Grande Florianópolis
2. TAEs em atividades de extensão	- quantitativo por papel de participante voluntário;	Oeste
	- quantitativo por papel de bolsista;	
	- quantitativo de TAEs envolvidos em atividades de extensão;	Reitoria, Sul
	- percentual de TAEs envolvidos em atividades de extensão.	Grande Florianópolis
3. Discentes em atividades de extensão	- quantitativo por papel de participante voluntário;	Reitoria, Grande Florianópolis, Oeste
	- quantitativo por papel de bolsista;	Reitoria, Planalto Serrano, Vale do Itajaí, Oeste
	- quantitativo de discentes envolvidos em atividades de extensão;	Reitoria, Sul
	- quantitativo de discentes envolvidos em atividades de extensão além dos créditos obrigatórios. (AGREGADO)	Grande Florianópolis
4. Participantes	- quantitativo de participantes externos envolvidos em atividades de extensão;	Reitoria, Norte, Sul

	- quantitativo de participantes externos envolvidos em atividades de extensão com papel de voluntário;	Reitoria, Norte
	- quantitativo de participantes de atividades de extensão por papel; (AGREGADO)	Reitoria
	- quantitativo de participantes de atividades de extensão por status; (AGREGADO)	
	- proporção de participantes externos por total de atividade de extensão;	Norte
	- proporção de participantes externos por total de participantes;	Norte, Oeste
5. Modalidades de atividades de extensão	- quantitativo de curso;	Reitoria, Grande Florianópolis, Planalto Serrano, Vale do Itajaí, Oeste
	- quantitativo de evento;	
	- quantitativo de programa;	
	- quantitativo de produto;	
	- quantitativo de projeto;	
	- quantitativo de prestação de serviço;	
	- quantitativo de discentes com matrícula ativa por programas; (AGREGADO)	Oeste
	- quantitativo de discentes com matrícula ativa em projetos;	

	- quantitativo de discentes com matrícula ativa por produtos; (AGREGADO)	
	- quantitativo de discentes com matrícula ativa por prestação de serviço; (AGREGADO)	
	- quantitativo de discentes com matrícula ativa por curso; (AGREGADO)	
	- quantitativo de discentes com matrícula ativa por evento. (AGREGADO)	
6. Curricularização da extensão	- quantitativo de discentes em atividade curricular; (AGREGADO)	Reitoria, Vale do Itajaí, Norte, Oeste
	- percentual de créditos ou carga horária destinados à extensão no currículo de nível médio técnico;	Vale do Itajaí, Norte, Oeste
	- percentual de créditos ou carga horária destinados à extensão no currículo de graduação;	
	- percentual de créditos ou carga horária destinados à extensão no currículo de especialização; (AGREGADO)	
	- percentual de créditos ou carga horária destinados à extensão no currículo de mestrado; (AGREGADO)	

	- percentual de créditos ou carga horária destinados à extensão no currículo de doutorado; (AGREGADO)	
	- percentual de créditos ou carga horária destinados à extensão no currículo de pós-graduação (mestrado e doutorado);	
	- percentual de créditos ou carga horária de ações de extensão no currículo de cursos;	
	- quantitativo de atividades de extensão por tipo de curso;	
	- Percentual de PPC com curricularização da extensão e atividades complementares.	
7. Orçamento	- quantitativo de recurso financeiro orçamentário total por câmpus; (AGREGADO)	Planalto Serrano
	- quantitativo de recursos financeiros do orçamento anual público obtido para extensão; (AGREGADO)	Planalto Serrano, Sul
	- quantitativo de recursos financeiros do orçamento anual público aplicados em extensão; (AGREGADO)	Reitoria, Sul

8. Público alvo	- quantitativo de ações de extensão demandadas pelo público externo; (AGREGADO)	Planalto Serrano, Norte
	- quantitativo de pessoas envolvidas por ações de extensão;	Planalto Serrano, Norte, Sul
	- quantitativo de pessoas alcançadas por atividades de modalidade prestação de serviço;	Norte
	- quantitativo de pessoas alcançadas por atividades de modalidade programas e projetos;	
	- quantitativo de pessoas alcançadas por atividades de modalidade cursos e eventos;	
	- quantitativo de ações de extensão para capacitação de população EJA;	
	- quantitativo de ações de extensão para capacitação de população PCD;	
	- quantitativo de pessoas atendidas por discentes de	

	graduação por meio de ações de extensão;	
	- quantitativo de docentes da rede pública capacitados por programas e projetos de formação continuada;	
	- percentual de ações de extensão destinadas à inclusão de população vulnerável;	
	- percentual de ações de extensão dirigidas às escolas públicas;	
	- percentual de municípios atendidos pelas ações extensionistas em relação ao total de municípios que são abrangidos pela respectiva instituição de ensino/câmpus;	
9. Parcerias	- quantitativo de ações de extensão com parcerias institucionais;	Planalto Serrano, Norte
	- listagem nominal das parcerias institucionais com ações de extensão. (AGREGADO)	Vale do Itajaí, Sul
10. Produção extensionista	- quantitativo de revistas e periódicos institucionais;	Vale do Itajaí, Sul
	- quantitativo de produção de livros ou capítulos com base em resultados da extensão;	

	- quantitativo de publicação de artigos em periódicos com base em resultados da extensão;	
	- quantitativo de mídias de comunicações em eventos com base em resultados da extensão;	
	- quantitativo de produções audiovisuais resultadas da extensão;	
	- quantitativo de produções artísticas (exposições, espetáculos, outros) resultantes da extensão.	
11. Áreas temáticas de extensão	- quantitativo de atividades de extensão por área da comunicação;	Reitoria, Grande Florianópolis, Planalto Serrano, Vale do Itajaí, Oeste
	- quantitativo de atividades de extensão por área da cultura;	
	- quantitativo de atividades de extensão por área dos direitos humanos e justiça;	
	- quantitativo de atividades de extensão por área da tecnologia e produção;	
	- quantitativo de atividades de extensão por área do trabalho;	
12. Infraestrutura e recursos humanos	- quantitativo de unidades organizacionais proponentes de ação de extensão; (AGREGADO)	Planalto Serrano

	- listagem nominal das unidades organizacionais proponentes de ação de extensão. (AGREGADO)	
13. Atividades de extensão	- quantitativo de atividades de extensão contínuas; (AGREGADO)	Reitoria, Grande Florianópolis, Norte, Sul
	- proporção de ações de extensão submetidas por aprovadas; (AGREGADO)	Sul
	- quantitativo de atividades de extensão com status favorável;	Reitoria, Sul
	- quantitativo de atividades de extensão com status desfavorável;	
	- quantitativo de atividade de extensão por status; (AGREGADO)	
	- quantitativo de atividades de extensão por objetivo sustentável da ONU. (AGREGADO)	Grande Florianópolis

Fonte: autoria própria.

Quadro 8 - Estratificação de indicadores de extensão provenientes de documentos infraconstitucionais, de fóruns da extensão, de documentos do IFSC, de solicitações da sociedade de dados extensionistas, e de entrevistas semiestruturadas com a gestão da extensão do IFSC.

MACROINDICADORES	INDICADORES
1. Docentes em atividades de extensão	- quantitativo por papel de participante voluntário; - quantitativo por papel de bolsista; - quantitativo por papel de coordenador; - quantitativo por papel de orientador; - quantitativo de docentes que submetem ações de extensão; (AGREGADO) - quantitativo de docentes envolvidos em atividades de extensão; - percentual de docentes envolvidos em atividades de extensão.
2. TAEs em atividades de extensão	- quantitativo por papel de participante voluntário; - quantitativo por papel de bolsista; - quantitativo por papel de coordenador; - quantitativo de TAEs envolvidos em atividades de extensão; - percentual de TAEs envolvidos em atividades de extensão.

3. Discentes em atividades de extensão	- quantitativo por papel de participante voluntário; - quantitativo por papel de bolsista; - quantitativo de discentes envolvidos em atividades de extensão; - quantitativo de discentes envolvidos em atividades de extensão além dos créditos obrigatórios; (AGREGADO) - percentual de discentes envolvidos em ações de extensão.
4. Participantes	- quantitativo total de servidores envolvidos em atividades de extensão; - quantitativo de participantes externos envolvidos em atividades de extensão; - quantitativo de participantes externos envolvidos em atividades de extensão com papel de voluntário; - quantitativo de participantes de atividades de extensão por papel; (AGREGADO) - quantitativo de participantes de atividades de extensão por status; (AGREGADO) - proporção de participantes externos por total de atividade de extensão; - proporção de participantes externos por total de participantes; - percentual de servidores envolvidos em atividades de extensão;

5. Modalidades de atividades de extensão	- quantitativo de curso; - quantitativo de evento; - quantitativo de programa; - quantitativo de produto; - quantitativo de projeto; - quantitativo de prestação de serviço; - proporção discentes com matrícula ativa por programas e projetos; - proporção discentes com matrícula ativa por programas; - quantitativo de discentes com matrícula ativa por programas; (AGREGADO) - proporção discentes com matrícula ativa por projetos; - quantitativo de discentes com matrícula ativa em projetos; - proporção discentes com matrícula ativa por produtos; - quantitativo de discentes com matrícula ativa por produtos; (AGREGADO) - proporção discentes com matrícula ativa por prestação de serviço; - quantitativo de discentes com matrícula ativa por prestação de serviço; (AGREGADO) - quantitativo de discentes com matrícula ativa por curso; (AGREGADO) - quantitativo de discentes com matrícula ativa por evento; (AGREGADO) - proporção de atividades de extensão no ano por modalidade projeto; - proporção de atividades de extensão no ano por modalidade curso;
---	--

	- quantitativo de incubadoras; - quantitativo de empreendimentos graduados em incubadoras resultados da extensão; - quantitativo de cooperativas populares graduadas em incubadoras resultantes da extensão;
6. Curricularização da extensão	- quantitativo de discentes em atividade curricular; (AGREGADO) - percentual de créditos ou carga horária destinados à extensão no currículo de nível médio técnico; - percentual de créditos ou carga horária destinados à extensão no currículo de graduação; - percentual de créditos ou carga horária destinados à extensão no currículo de especialização; (AGREGADO) - percentual de créditos ou carga horária destinados à extensão no currículo de mestrado; (AGREGADO) - percentual de créditos ou carga horária destinados à extensão no currículo de doutorado; (AGREGADO) - percentual de créditos ou carga horária destinados à extensão no currículo de pós-graduação (mestrado e doutorado); - percentual de créditos ou carga horária de ações de extensão no currículo de cursos; - quantitativo de atividades de extensão por tipo de curso; - Percentual de PPC com curricularização da extensão e atividades complementares.

7. Orçamento	- percentual de recursos financeiros do orçamento anual público aplicados em extensão; - quantitativo de recurso financeiro orçamentário total por câmpus; (AGREGADO) - quantitativo de recursos financeiros do orçamento anual público obtido para extensão; (AGREGADO) - quantitativo de recursos financeiros do orçamento anual público aplicados em extensão; (AGREGADO) - quantitativo de recursos para extensão captados via prestação de serviços acadêmicos especializados; - quantitativo de recursos para extensão captados via editais externos; - quantitativo de recurso aplicado por edital interno de extensão; - quantitativo de recurso repassado aos coordenadores de atividade de extensão; - quantitativo de recurso repassado aos bolsistas discentes;
--------------	--

8. Público alvo	- quantitativo de ações de extensão demandadas pelo público externo; (AGREGADO) - quantitativo de pessoas envolvidas por ações de extensão; - quantitativo de pessoas envolvidas por atividades de modalidade prestação de serviço; - quantitativo de pessoas envolvidas por atividades de modalidade programas e projetos; - quantitativo de pessoas envolvidas por atividades de modalidade cursos e eventos; - quantitativo de ações de extensão para capacitação de população EJA; - quantitativo de ações de extensão para capacitação de população PCD; - quantitativo de pessoas envolvidas por discentes de graduação por meio de ações de extensão; - quantitativo de docentes da rede pública capacitados por programas e projetos de formação continuada; - percentual de ações de extensão destinadas à inclusão de população vulnerável; - percentual de ações de extensão dirigidas às escolas públicas; - percentual de municípios atendidos pelas ações extensionistas em relação ao total de municípios que são abrangidos pela respectiva instituição de ensino/câmpus.
9. Parcerias	- percentual de ações de extensão com parcerias institucionais;

	- quantitativo de ações de extensão com parcerias institucionais; - percentual de representação oficial da instituição junto à sociedade civil; - taxa de aprovação de propostas de extensão em editais externos; - listagem nominal das parcerias institucionais com ações de extensão; (AGREGADO) - quantitativo de instituições ativas nos fóruns de extensão e relações externas no câmpus.
10. Produção extensionista	- quantitativo de revistas e periódicos institucionais; - quantitativo de produção de livros ou capítulos com base em resultados da extensão; - quantitativo de publicação de artigos em periódicos com base em resultados da extensão; - quantitativo de mídias de comunicações em eventos com base em resultados da extensão; - quantitativo de produções audiovisuais resultadas da extensão; - quantitativo de produções artísticas (exposições, espetáculos, outros) resultantes da extensão.

11. Áreas temáticas de extensão	- quantitativo de atividades de extensão por área da comunicação; - quantitativo de atividades de extensão por área da cultura; - quantitativo de atividades de extensão por área dos direitos humanos e justiça; - quantitativo de atividades de extensão por área da tecnologia e produção; - quantitativo de atividades de extensão por área do trabalho;
12. Infraestrutura e recursos humanos	- proporção de horas de treinamento por servidor da gestão extensionista; - proporção de participação em eventos por servidor da gestão extensionista; - proporção de participação em eventos promovidos ou apoiados pela pró-reitoria de extensão por servidor da gestão extensionista; - proporção espaço físico (área) por servidor; - percentual de capacidade de atendimento de demandas de transporte logístico para ações de extensão; - percentual de servidores em representação oficial da instituição junto à sociedade civil; - quantitativo de unidades organizacionais proponentes de ação de extensão; (AGREGADO) - listagem nominal das unidades organizacionais proponentes de ação de extensão. (AGREGADO)

13. Atividades de extensão	- quantitativo total de atividades de extensão; - quantitativo de atividades de extensão contínuas; (AGREGADO) - proporção de ações de extensão com status aprovada por servidor; - proporção de ações de extensão submetidas por aprovadas; (AGREGADO) - quantitativo de atividades de extensão com status favorável; - quantitativo de atividades de extensão com status desfavorável; - quantitativo de atividade de extensão por status; (AGREGADO) - percentual de atividade de extensão que articularam com a pesquisa; - percentual de atividade de extensão que articularam com o ensino; - quantitativo de atividades de extensão por objetivo sustentável da ONU. (AGREGADO)
----------------------------	--

Fonte: autoria própria.

5 PRODUTO EDUCACIONAL

5.1 CARACTERIZAÇÃO E FINALIDADE DO PRODUTO EDUCACIONAL

O produto educacional proposto tem por finalidade dar visibilidade à informações e suporte a análise quali-quantitativa com base nos indicadores identificados, através da mineração do banco de dados utilizados pela pró-reitoria de extensão e relações exteriores do IFSC via SIGAA.

De acordo com o documento orientador de avaliação de propostas de cursos novos (APCN) da Capes - MEC - de 2022, o produto se enquadra como um “i) Software/Aplicativo (aplicativos de modelagem, aplicativos de aquisição e análise de dados [...])”. Por meio de uma visualização acurada dos dados é possível observar a abrangência da extensão e as lacunas a serem preenchidas por atividades de gestão para o seu fortalecimento, bem como suprir a sociedade de informações relevantes para exercício da cidadania e interação com a instituição. Desta maneira o produto também se enquadra na linha 2 de pesquisa - organização e memórias de espaços pedagógicos na educação profissional e tecnológica - , de aderência ao macroprojeto 6: organização de espaços pedagógicos na EPT.

Este produto atende os critérios de complexidade apontados no documento supracitado, sendo este o (1) impacto que abrange toda uma comunidade institucional distribuída em 23 (vinte e três) câmpus mais a reitoria no estado de Santa Catarina, (2) aplicabilidade por objetivar a publicização em domínio institucional por meio de painéis de fácil acesso e leitura, podendo ser replicável em outras instituições com arquiteturas de banco de dados e indicadores semelhantes ao IFSC, (3) tendo aderência à área de ensino para a contribuição da área de conhecimento, e percebe-se o critério de (4) inovação uma vez que não foram identificados produtos e estudos voltados para essa problemática.

5.2 ELABORAÇÃO DO PRODUTO EDUCACIONAL

O produto por ser um *software* dispõe de um leque de técnicas de elaboração e de modelagem de arquitetura. Dentre as escolhidas foi construído um produto educacional com base em um modelo iterativo incremental em espiral (figura 3) por

meio das fases clássicas do modelo cascata (figura 4) respectivamente: requisito, projeto, implementação, verificação e manutenção (Pressman, 2011). No qual cada iteração houve a construção e incremento de uma categoria de indicadores (macroindicadores) de extensão identificados na pesquisa dividindo a iteração em quatro etapas de desenvolvimento de fluxo de dados: extração, tratamento, carregamento e exibição.

O produto foi elaborado utilizando o sistema operacional Linux Ubuntu versão 22.04.05 LTS em um mini computador com a seguinte configuração: 8 GB de memória principal; 250 GB de memória secundária; Processador Intel Core i7-8700T com clock de 2.40 GHz; E placa de vídeo Intel UHD Graphics 630. Sob essas configurações as ferramentas utilizadas individualmente conseguiam executar seus processos. Porém, ao executar 2 (duas) ou mais ferramentas ou iterações simultaneamente o computador apresentava lentidão e em alguns casos travamentos necessitando de reinicialização.

A fase de requisito foi suprida pelos indicadores de extensão identificados na pesquisa descritos no tópico 4 (quatro). A fase de projeto contou com pesquisa adicional para modelagem de arquitetura da solução em *software* e levantamento de ferramentas considerando a expectativa de adoção do produto pelo IFSC, e escalabilidade para outras instituições e demais frentes da EPT.

Ainda na fase de projeto houve a identificação da base de dados PostgreSQL versão 9.0.13 do tipo relacional utilizada pelo SIGAA. Como ferramenta de acesso e consulta dessa, foi escolhido o gerenciador de banco de dados DBeaver 24.3.4 que oferece flexibilidade de acesso a diversos tipos de banco de dados, interface visual para rastreamento na navegação entre agrupamentos (*schemas*) e tabelas, além de exibir diagramas de relacionamentos entre esses e permitir consultas utilizando a sintaxe da linguagem de consulta estruturada (*Structured Query Language - SQL*).

Também optou-se por utilizar a linguagem de programação Python 3.10.12 e seu indexador de pacotes PyPi 22.0.2 para elaboração dos scripts das etapas de extração, tratamento e carregamento por conta de seu desempenho de processamento, baixa verbosidade, alta curva de aprendizado e bibliotecas que abstraem funções de tratamento de dados.

Como orquestrador de fluxo de dados das etapas de iteração foi escolhido o Apache Airflow 2.10.3 por ser uma ferramenta livre, leve, de código aberto, com interface visual de fácil navegabilidade, permite criar, agendar e monitorar fluxo de dados utilizando o Python, ainda tem por característica alta confiabilidade no controle a falhas, registro de histórico de execução de tarefas e conexão a várias fontes de dados. Considerando a portabilidade dessa ferramenta, essa foi configurada em um container Docker 27.5.1 que abstrai as dependências dos sistemas operacionais, tendo suas configurações mapeadas em arquivos de extensão .yaml e orquestradas com o Docker Compose 1.29.2. Ao todo foram despendidos 2 (dois) meses de aprendizado no Python e Apache Airflow com Docker, de julho a agosto de 2024 (dois mil e vinte e quatro).

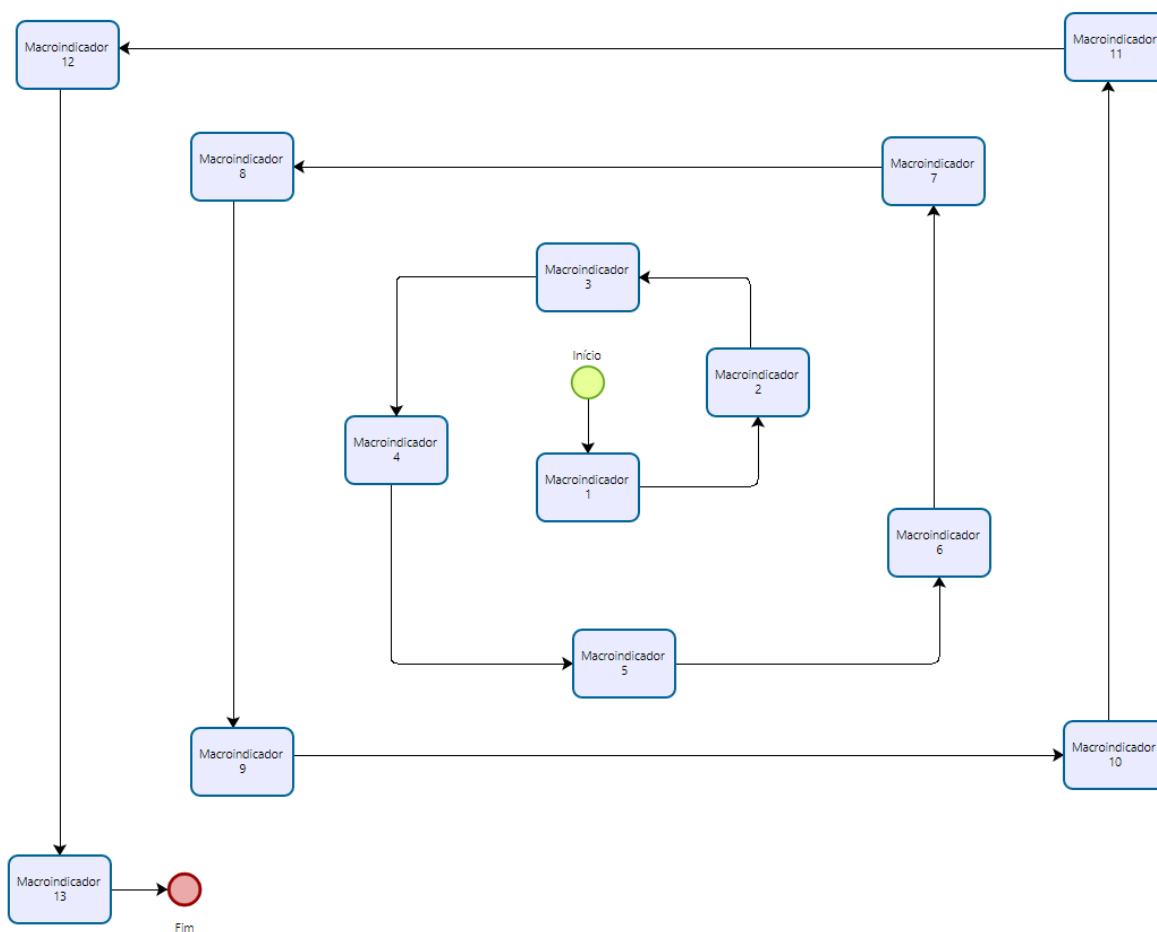
Como última ferramenta escolhida no projeto para exibição dos dados, foi levado em consideração a expertise da gestão do IFSC, pois ao consultar os documentos institucionais observou-se que há referências à painéis de dados no Looker Studio de domínio institucional, acessível somente por navegador, escolheu-se usar o Google Chrome 133.0.6943.53 por serem do mesmo mantenedor. Da mesma forma, observou-se que as funcionalidades dessa ferramenta atende as 10 (dez) heurísticas de Nielsen para construção de interfaces gráficas. Assim, os estudos para modelagem de projeto para as iterações somaram 3 (três) meses e meio, de julho a metade de outubro de 2024 (dois mil e vinte e quatro).

A fase de implementação contou com o DBeaver na construção dos *scripts* de consulta aos dados da extensão, teve o Apache Airflow para mapeamentos dos *scripts* SQL e executor dos *scripts* Python, sendo esses escritos via Visual Studio Code 1.96.4, e os componentes do tipo *what you see is what you get* (WYSIWYG) do Looker Studio para exibição dos dados.

A verificação foi executada no console de visualização de dados do DBeaver, no histórico de tarefas do Apache Airflow e na exibição de dados do tipo rascunho do Looker Studio. Por fim, a manutenção se fez presente ao observar e ajustar inconsistências de dados nas consultas SQL, na presença de erros no apontamento de falhas do Apache Airflow, no tratamento de erros nos *scripts* Python e na ocorrência de falhas de exibição de dados no Looker Studio.

Uma vez definida a fase de requisito e projeto para cada iteração, o desenvolvimento das fases seguintes do produto educacional presentes nas figuras 3 e 4 ocorreram ao longo dos meses de novembro a meados de dezembro de 2024 (dois mil e vinte e quatro).

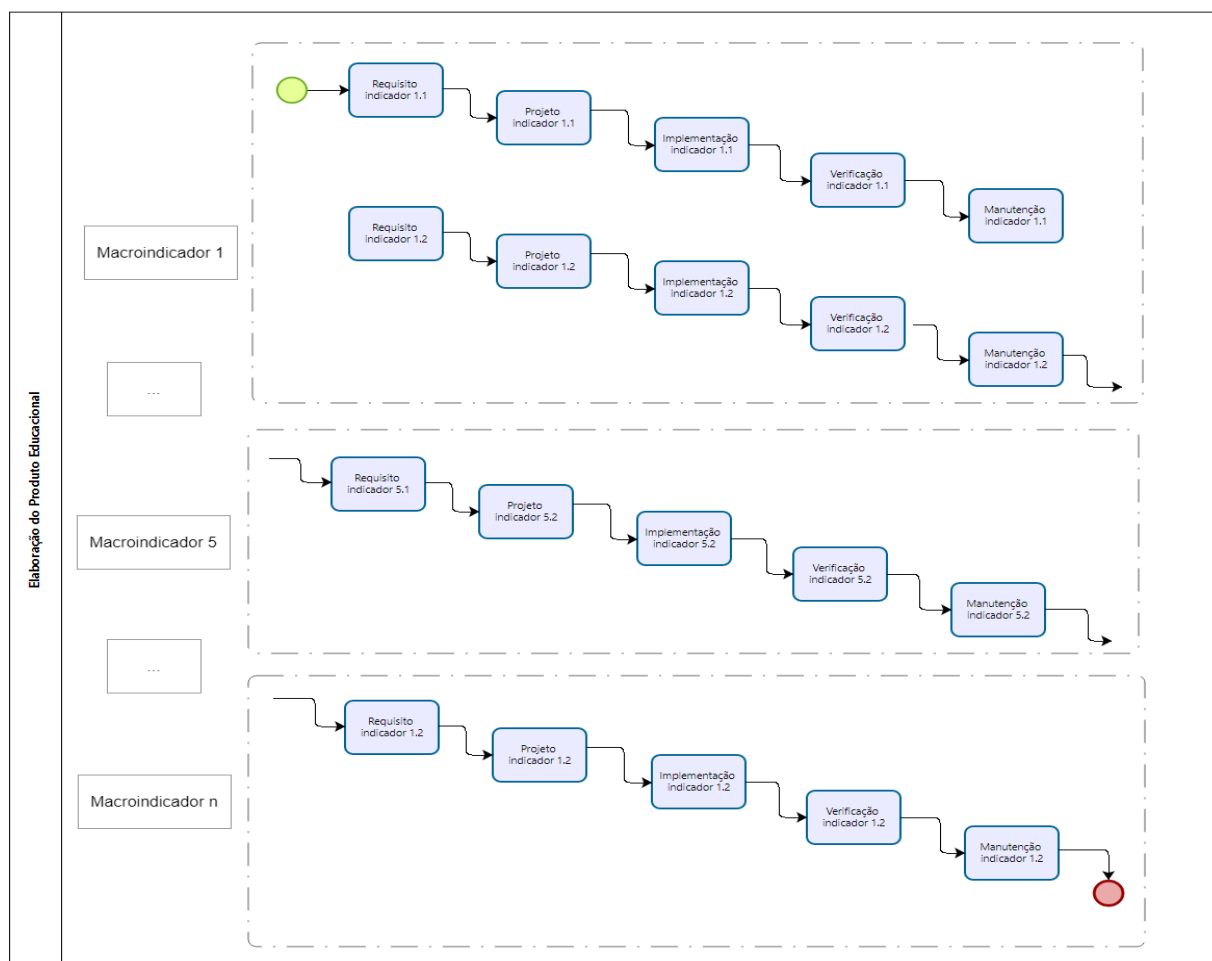
Figura 3 - Modelo de desenvolvimento incremental em espiral.



Fonte: autoria própria.

Nota: o modelo foi elaborado com o programa Bizagi Modeler.

Figura 4 - Modelo de desenvolvimento iterativo incremental em cascata.



Fonte: autoria própria.

Nota: o modelo foi elaborado com o programa Bizagi Modeler.

5.2.1 Extração dos indicadores de extensão

Desconsiderando a extração dos dados de acesso à informação pela sociedade, a etapa de extração de dados pelo produto educacional teve dois momentos. O primeiro com foco na exploração dos conjuntos e tabelas da base de dados do SIGAA, e o segundo no mapeamento dos *scripts* SQL no Apache Airflow em variáveis que são invocadas nos *scripts* Python.

O banco de dados do SIGAA, do tipo relacional, é um sistema de gerenciamento de dados baseado na teoria matemática dos conjuntos que organiza informações em tabelas compostas por linhas e colunas. Na estrutura do banco de

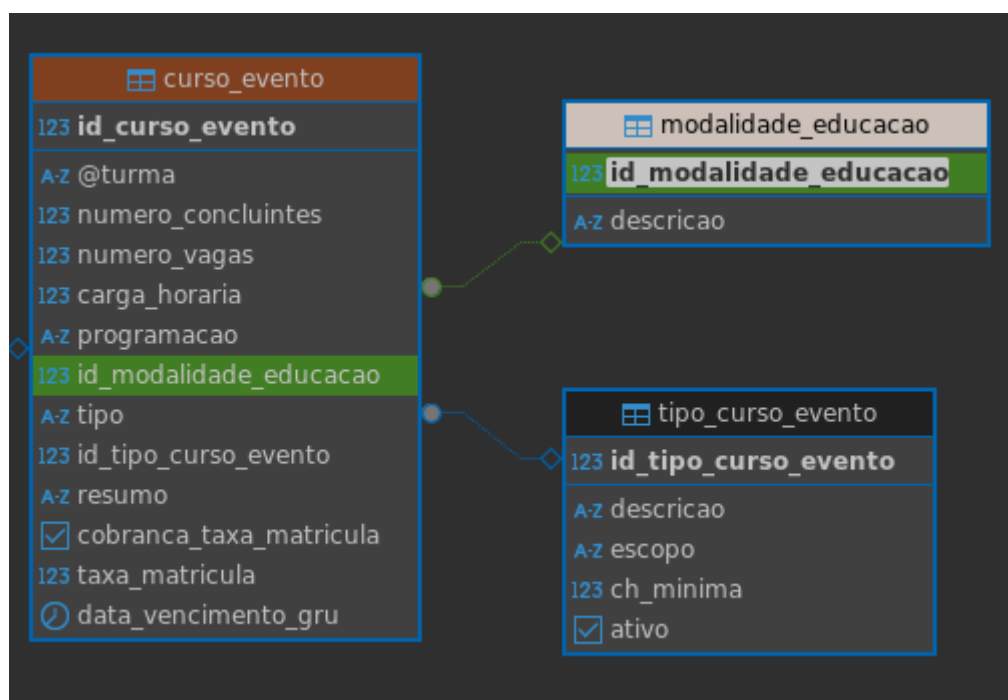
dados relacional é possível ter um agrupamento de tabelas chamado de esquema, onde cada tabela abstrai um conceito do mundo real, em que suas colunas representam os atributos dessas entidades, enquanto as linhas representam os registros de dados. Os atributos dessas são fortemente tipificados, em outras palavras, são definidos os tipos de dados antes que se submeta um registro, a exemplo, um atributo do tipo inteiro não aceitará registro de um número com casas decimais (Navathe, 2005).

Os atributos também podem ser designados como chaves para identificação única de registros e estabelecimento de relacionamentos entre tabelas. Essas podem ser classificadas em chave primária que identifica o registro através de um valor único, evitando duplicações do mesmo registro, sendo exclusiva e não podendo ser nula; Chave estrangeira que faz referência à chave primária de outra tabela, estabelecendo um relacionamento e permitindo a recuperação de dados de outra tabela (Navathe, 2005).

A respeito de relacionamentos, esses podem ocorrer de forma redundante ou entre tabelas obedecendo a integridade referencial entre chaves e a normalização de atributos para evitar redundâncias e inconsistências. O fenômeno intrínseco aos relacionamentos é a cardinalidade, que podem ser de 1 (um) para 1 (um), 1 (um) para N, N para 1 (um) e N para N registros, por exemplo, uma tabela imóvel tem como chave primária o atributo matrícula, essa tem uma cardinalidade de 1 (um) para N com a tabela residente que tem entre os seus atributos a chave estrangeira matrícula fazendo referência ao imóvel, portanto diz-se que nessa relação a casa pode ter 1 (um) ou mais residentes.

Na figura 5 o DBEaver apresenta o diagrama de relacionamento da tabela curso_evento onde há a cardinalidade de N modalidade_educacao para 1 (um) curso_evento, em que curso_evento tem a chave estrangeira id_modalidade_educacao fazendo referência à chave primária de modalidade_educacao. Observa-se a mesma cardinalidade de relacionamento entre curso_evento e tipo_curso_evento. Além disso, nota-se que os demais atributos estão fortemente tipados, por exemplo, o numero_vagas de curso_evento é um tipo numérico inteiro, cobranca_taxa_matricula da mesma tabela é do tipo booleano, descricao de modalidade_educacao é uma cadeia de caracteres conhecida como *string*, e assim por diante.

Figura 5 - Relacionamento entre tabelas no SIGAA.



Fonte: esquema de extensão da base de dados do SIGAA.

Nota: o diagrama foi retirado da interface do DBeaver.

Segundo Navathe (2005), um banco de dados normalizado tem abstrações bem definidas em suas tabelas e como consequência a consulta dos seus registros (tuplas) se torna mais eficiente. Antes de citar SQL, o autor define notações com base na teoria dos conjuntos para definir as operações entre tabelas que são: seleção, projeção, junção theta, equijunção, junção natural, união, interseção, diferença, produto cartesiano e divisão.

De forma resumida, a seleção (σ) é uma operação que extrai um subconjunto de registros de uma condição específica de um ou mais atributos da tabela. Já a projeção (π) lista quais atributos se quer apresentar em detrimento de outros. Essas duas notações podem ser identificadas na linguagem SQL através dos comandos *select from* e *where* para projeção e seleção respectivamente (Navathe, 2005). Exemplo de projeção em SQL de matrícula e endereço de imóvel que seleciona logradouros correspondente a terrenos é observável no código 4:

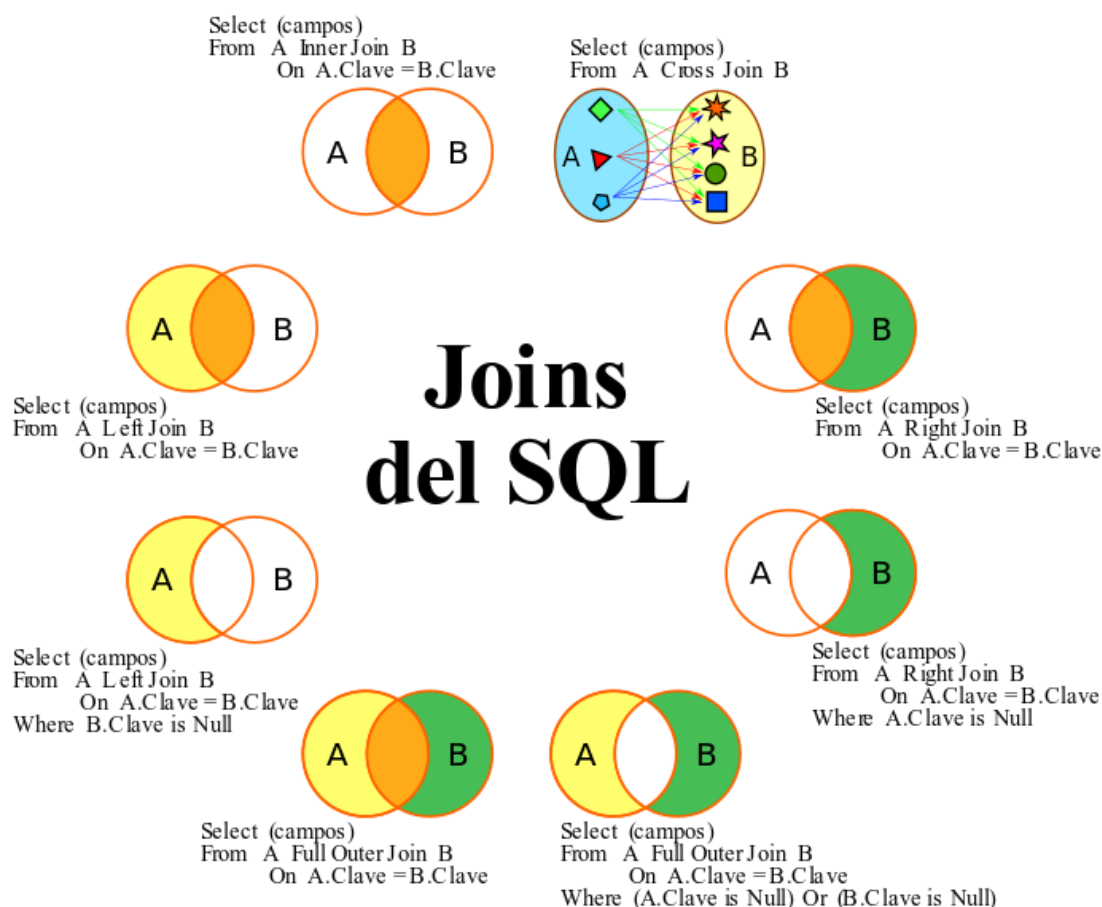
Código 4 - Script SQL com projeção e seleção.

```
select matricula, endereco from imovel where logradouro = "terreno"
```

Fonte: autoria própria.

No campo das relações, Navathe (2005) expõe a junção theta ($\bowtie$) que combina todas os registros de duas tabelas que satisfazem a condição lógica de agregação; A equijunção, semelhante a junção theta, exceto que a condição lógica de comparação corresponde ao sinal de igualdade; A junção natural semelhante a equijunção excetuando que a condição é a combinação automática dos atributos em comum mantendo a nomenclatura da tabela à esquerda; A união ($\cup$) é a combinação de todas os registros removendo os atributos duplicados automaticamente; A interseção ($\cap$) retorna apenas registros duplicados entre duas tabelas; A diferença ($-$) é uma operação que retorna registros que pertencem a uma tabela à esquerda e não pertencem a tabela à direita; O produto cartesiano ($\times$) combina todos os registros entre duas tabelas desconsiderando qualquer condição. A tradução dessas notações em linguagem SQL, sob os comandos *join*, é observável na figura 6.

Figura 6 - Comandos de agregação em linguagem SQL.



Fonte: Navathe, 2005.

Nota: a imagem foi extraída da plataforma wikiversity.

Por fim, a divisão ($\div$) é a operação que agrupa os registros de interesse da tabela à esquerda - de maior peso - com os registros existentes em uma relação com a tabela à direita (Navathe, 2005). Na linguagem SQL essa notação se implementa pelos comandos *group by* para a tabela à esquerda e *having* para a tabela à direita. Como no exemplo do código 5 da relação entre as tabelas imóvel e residente, nas quais a chave primária de imóvel é matrícula e a chave primária de residente é cpf, realiza-se a operação de divisão para retornar quais imóveis tem mais de um residente e quantos:

Código 5 - Script SQL de operação de divisão com *group by* e *having*.

```
select i.matricula, i.endereco, count(r.cpf) as TotalResidentes
from imovel i
join residente r on i.matricula = r.matricula
group by i.matricula, i.endereco
having count(r.cpf) > 1
```

Fonte: autoria própria.

Ao analisar a base de dados do SIGAA percebeu-se a existência do esquema da extensão o qual armazena boa parte dos dados de interesse desta pesquisa, no entanto, ao observar os relacionamentos das tabelas notou-se que essas não os mantêm exclusivamente com tabelas do mesmo conjunto, para tanto houve que se realizar operações de agregação através do relacionamento com tabelas de outros conjuntos para obtenção do resultado desejado.

Além dos critérios de quantitativo, proporção e percentual descritos nos indicadores de extensão identificados, foi agregado nos requisitos do produto educacional a extração de 3 (três) atributos que servem para filtragem da visão em gráficos. Essa decisão foi tomada com base na característica do IFSC pela capilaridade de suas unidades, portanto o levantamento por câmpus e CEP se fez necessário, assim também a exibição da temporalidade anual dos dados conforme os documentos do quadro 9.

Quadro 9 - Documentos que apontam a visão temporal dos dados.

Documento	Temporalidade
Deliberação CEPE/IFSC Nº 18, de 12 de abril de 2010	Anual
Portaria SETEC/MEC Nº 299, de 6 de maio de 2022	Semestral ou Anual
Manual de Indicadores de Extensão Universitária (IBEU) FORPROEXT/2017	Anual
Resolução CNE Nº 7, de 18 de dezembro de 2018	Anual
Acórdão TCU Nº 461, 9 de março de 2022	Anual
Relatório de Gestão do IFSC de 2023	Anual
Planejamento Estratégico Institucional do IFSC (PEI) 2020-2024	Anual

Fonte: autoria própria.

Para tanto, como boa parte dos indicadores da extensão identificados têm como elemento comum as atividades de extensão e os seus participantes, a exploração dos dados partiu das tabelas atividade e participante_acao_extensao, observou-se que ambas têm 46 (quarenta e seis) e 5 (cinco) relacionamentos respectivamente, que desses fora possível extrair 99 (noventa e nove) indicadores de extensão, e 14 (quatorze) carecem de customização no sistema para gestão,

sendo 9 (nove) sobre o macroindicador de orçamento e as demais de infraestrutura e recursos humanos.

Para cada 1 (um) dos 99 (noventa e nove) indicadores de extensão foram elaborados *scripts* em SQL de consulta de dados apresentados no anexo A. Em alguns casos 1 (um) *script* pôde atender mais de 1 (um) indicador. No exemplo seguinte, no código 6, o *script* atendeu todos os indicadores do macroindicador área temática de extensão. Nele ainda nota-se a projeção dos dados campus, CEP, ano, área temática e quantitativo de atividades, sendo sua seleção as áreas temáticas que estão ativas, também há a presença de 7 (sete) equijunções, onde 5 (cinco) correspondem a tabelas de outros esquemas de dados, por fim há a presença da operação de divisão das agregações pelos agrupamentos da projeção e ordenação dos mesmos.

Código 6 - Script SQL de seleção de quantitativo do macroindicador de áreas temáticas da extensão por câmpus, ano, CEP e área temática.

```
select ci.nome as campus, endereco.cep as cep, proj.ano as ano_atividade,
area.descricao as area_extensao, count(a.id_atividade) as
quantidade_atividade_extensao
from extensao.atividade a
inner join extensao.area_tematica area on area.id_area_tematica =
a.id_area_tematica_principal
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco =
ci.id_endereco
where area.ativo = true
group by ci.nome, proj.ano, endereco.cep, area.descricao
order by ci.nome, proj.ano, area.descricao asc;
```

Fonte: autoria própria.

Ainda na extração observou-se que a base de dados do SIGAA tem uma boa normalização, no entanto foram encontrados dados redundantes, como por exemplo ao buscar pelo tipo de participante, em um dado momento encontrou-se uma tabela agregável em que fora possível identificar se o participante era servidor docente ou TAE, ou se era discente, no entanto pela falta de registro de participantes externos,

houve que buscar em outro relacionamento em uma tabela de outro esquema a completude desses registros completos.

Em outro caso, considerando que os comandos *left join* têm maior eficiência em sua execução em relação ao *inner join*, optou-se por usá-lo para aumentar o desempenho das buscas, no entanto essa apresentou-se ineficaz para com a base de dados do SIGAA, pois se há inconsistência no registro de dados ou na deleção forçada de atributos que mantêm relacionamentos, há uma quebra no encadeamento de agregações, por isso retomou-se a adoção do *inner join* como garantia de completude dos dados para atendimento dos indicadores.

Uma vez com os *scripts* SQL elaborados, o segundo passo foi designá-los para variáveis no Apache Airflow utilizando a interface em Admin -> Variables. Adotando essa estratégia é possível ter uma melhor manutenção dos códigos de consulta em pares, onde alguém lida com a exploração dos dados e outro trabalha com os *scripts* Python que farão a extração. Ainda, uma etapa fundamental para que haja conectividade entre o banco de dados do SIGAA e o Apache Airflow é a configuração dessa em Admin -> Connections.

Assim, o script de extração Python, através das bibliotecas do Apache Airflow, pode invocar uma tarefa (*task*) para conectar à base de dados do SIGAA e executar o *script* de consulta SQL. No exemplo seguinte (código 7) é criada uma tarefa de identificação `extraí_quantitativo_areas_tematicas_extensao` que utiliza o `PostgresOperator` provido pelo Apache Airflow em que invoca a conexão `IFSC_SIGAA` configurada previamente chamando o *script* mapeado na variável `script_SIGAA_IFSC_consulta_quantitativo_areas_tematicas_extensao`. Adotar essa nomenclatura de variáveis, apesar de verbosa, oferece desde o projeto do produto educacional uma organização em que se prevê a inclusão de mapeamento de variáveis por outras instituições e banco de dados.

Código 7 - Script Python com função PostgresOperator do Apache Airflow de conexão com o banco de dados e chamada de consulta SQL.

```
extrai_quantitativo_areas_tematicas_extensao = PostgresOperator(
    task_id='extrai_quantitativo_areas_tematicas_extensao',
    postgres_conn_id='IFSC_SIGAA',
    sql =
Variable.get("script_SIGAA_IFSC_consulta_quantitativo_areas_tematicas_extensa
o"),
)
```

Fonte: autoria própria.

5.2.2 Tratamento de dados

A etapa de tratamento de dados é um conceito que deve ser aplicado da criação à exibição do dado. Portanto esse deve ser amplamente apoiado pela governança de dados de uma instituição. Dados mal governados refletem distorções na realidade, são o alicerce para a construção de planejamentos estratégicos que podem ter um mau direcionamento se esses não tiverem acurácia (Damabok, 2025; Barbieri, 2019).

Esta pesquisa não envolveu criação de dados, logo o tratamento não atendeu essa etapa. No entanto, procedimentos de transformação, varredura, limpeza e filtragem foram aplicados nas etapas de desenvolvimento do produto educacional, como exemplo, a filtragem de dados temporais inconsistentes nos painéis de exibição, e a limpeza dos dados de CEP para obtenção somente de valores numéricos no *script* de consulta e extração no código 8:

Código 8 - Script SQL com limpeza dos dados de CEP para obtenção apenas do valor numérico.

```
select ci.nome as campus, regexp_replace(endereco.cep,'[^0-9]', '', 'g') as
cep, proj.ano as ano_atividade, area.descricao as area_extensao,
count(a.id_atividade) as quantidade_atividade_extensao
from extensao.atividade a
inner join extensao.area_tematica area on area.id_area_tematica =
a.id_area_tematica_principal
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
```

```

inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco =
ci.id_endereco
where area.ativo = true
group by ci.nome, proj.ano, endereco.cep, area.descricao;
Fonte: autoria própria.

```

Os procedimentos de transformação e varredura foram os mais utilizados para obtenção de dados quantitativos, percentuais e proporções requeridos pelos indicadores de extensão, esses são identificáveis nos *scripts* em Python com Apache Airflow. Para elaborá-los é necessário compreender o papel de cada uma dessas ferramentas complementares.

O Python surgiu com uma proposta de ter uma sintaxe simples com encadeamento lógico de boa assimilação humana, para esta pesquisa foi fundamental entender algumas palavras reservadas e estruturas de processamento. No código 9 a seguir é apresentada uma função que abstrai um comportamento do código identificada pela palavra *def* seguida do nome da função indicando que para usá-la deve-se informar um texto e palavras-chave podendo retornar opcionalmente uma cadeia de caracteres. Nas linhas seguintes existe a estrutura condicional *if* que verifica se o texto está vazio ou nulo, se estiver retorna-se nada para quem chamou a função, dentro das estruturas condicionais também existem o *elif* e o *else* correspondentes a proposições lógicas a serem verificadas caso a primeira condição for falsa. Na sequência observa-se a declaração de uma variável identificável pelo sinal de igual, entendido pela linguagem como um sinal de atribuição, nela o texto todo é transcrito para caixa baixa. Por último existe um laço de repetição designado pelas palavras *for* e *in* onde se faz uma varredura de todas as palavras contidas em palavras-chave, na qual se existir uma palavra dentro do texto, então será retornada a chave ou categoria em que essa palavra se encaixa. Caso nenhuma categoria seja identificada, nada se retorna. Outra característica relevante no Python é que cada estrutura pertence a outra estrutura superior, o que vai designar essa hierarquia é a presença da tabulação para as que estão abaixo, e como indicativo do fim de uma função para outra há a presença de um espaço em branco.

Código 9 - Script Python com estrutura básica da sintaxe da linguagem.

```

def categorizar_produto(texto, palavras_chave) -> Optional[str]:
    if pd.isnull(texto):
        return None

```

```

if pd.isna(texto):
    return None
texto = texto.lower()
for categoria, palavras in palavras_chave.items():
    if any(palavra in texto for palavra in palavras):
        return categoria
return None

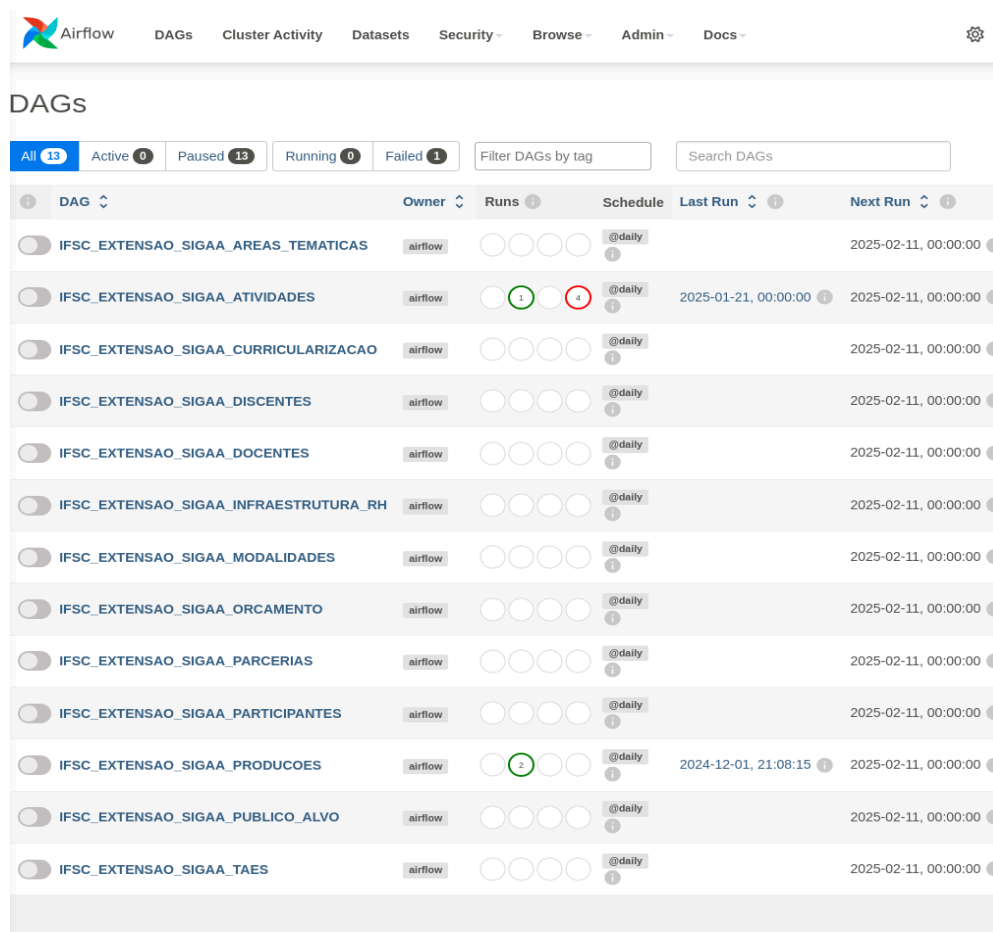
```

Fonte: autoria própria.

Essas funções são os blocos de código onde são escritos os tratamentos de dados, porém para que processem é preciso orquestrá-las quanto a sua ordem e rotina de execução. Para isso utiliza-se o Apache Airflow, nesse o fluxo é chamado de *pipeline*, pois assimila o fluxo de dados como o de canos hidráulicos por onde o líquido percorre uma direção. O *pipeline* então é organizado em DAG, acrônimo para *Directed Acyclic Graph*, traduzido por Grafo Acíclico Direcionado, conceito baseado na teoria dos grafos onde há diversos nós e arestas que se ligam. Assim funcionam as DAGs, de forma acíclica, com processamento agendado e independente uma das outras, com *pipeline* de dados direcionado, sob o formato de nós que são chamados de tarefas ou *tasks*.

Para o produto educacional foram construídas 13 (treze) DAGs (figura 7), uma para cada categoria, que estruturam as tarefas e funções em etapas para seus respectivos indicadores, o código completo das DAGs são encontrados no anexo B.

Figura 7 - Página de DAGs do Apache Airflow.



DAG	Owner	Runs	Schedule	Last Run	Next Run
IFSC_EXTENSAO_SIGAA_AREAS_TEMATICAS	airflow	0	@daily		2025-02-11, 00:00:00
IFSC_EXTENSAO_SIGAA_ATIVIDADES	airflow	1	@daily	2025-01-21, 00:00:00	2025-02-11, 00:00:00
IFSC_EXTENSAO_SIGAA_CURRICULARIZACAO	airflow	0	@daily		2025-02-11, 00:00:00
IFSC_EXTENSAO_SIGAA_DISCENTES	airflow	0	@daily		2025-02-11, 00:00:00
IFSC_EXTENSAO_SIGAA_DOCENTES	airflow	0	@daily		2025-02-11, 00:00:00
IFSC_EXTENSAO_SIGAA_INFRAESTRUTURA_RH	airflow	0	@daily		2025-02-11, 00:00:00
IFSC_EXTENSAO_SIGAA_MODALIDADES	airflow	0	@daily		2025-02-11, 00:00:00
IFSC_EXTENSAO_SIGAA_ORCAMENTO	airflow	0	@daily		2025-02-11, 00:00:00
IFSC_EXTENSAO_SIGAA_PARCERIAS	airflow	0	@daily		2025-02-11, 00:00:00
IFSC_EXTENSAO_SIGAA_PARTICIPANTES	airflow	0	@daily		2025-02-11, 00:00:00
IFSC_EXTENSAO_SIGAA_PRODUCOES	airflow	2	@daily	2024-12-01, 21:08:15	2025-02-11, 00:00:00
IFSC_EXTENSAO_SIGAA_PUBLICO_ALVO	airflow	0	@daily		2025-02-11, 00:00:00
IFSC_EXTENSAO_SIGAA_TAES	airflow	0	@daily		2025-02-11, 00:00:00

Fonte: autoria própria.

No exemplo do código 10 observa-se a estrutura da DAG de orçamento visto que é mais enxuta e não há tabelas para extração dessa categoria no SIGAA. No primeiro bloco de código são importadas as bibliotecas com funções do Apache Airflow e do Python, logo abaixo segue a declaração de variáveis que serão manipuladas no *script*. Em seguida há a função central *with DAG() as dag* em que em seu interior são preenchidos parâmetros para que o motor da ferramenta as execute, o *dag_id* determina o nome da DAG, o *start_date* recebe a informação de quando será a primeira execução, o *schedule_interval* recebe a frequência em que a execução se repetirá, e o *catchup* estabelece se irá ou não executar as tarefas seguintes ao nó em que ocorreu erro.

Dentro da função DAG é o espaço para criação de outras funções, de tarefas que as invocam, e da sequência das tarefas. Ainda na sequência do exemplo há a função *def cria_diretorio()* que manipula as variáveis antes declaradas para criação

de um lago de dados caso ainda não exista, essa função é executada pela tarefa `cria_datalake_orcamento_extensao` através de um operador de *script* Python que em seus parâmetros `task_id` recebe o nome da tarefa e `python_callable` recebe o nome da função Python a ser executada. Por último é listada a ordem de execução das tarefas seguidas do símbolo `>>` e tarefas entre colchetes para designar um conjunto a ser executado em paralelo, como no exemplo há somente uma tarefa então apenas essa é listada.

Código 10 - Script Python com mapeamento de DAG do Apache Airflow.

```
from airflow import DAG
from airflow.operators.python import PythonOperator
from airflow.models import Variable
from airflow.utils.dates import days_ago
from datetime import datetime
import pandas as pd
import os

datalake_extensao_dir = Variable.get("datalake")+ 'extensao/'
datalake_orcamento_dir = datalake_extensao_dir+ 'orcamento/'

with DAG(
    dag_id='IFSC_EXTENSAO_SIGAA_ORCAMENTO',
    start_date=days_ago(1),
    schedule_interval='@daily',
    catchup=False,
) as dag:

    def cria_diretorio():
        if not os.path.exists(datalake_extensao_dir):
            os.makedirs(datalake_extensao_dir)
        elif not os.path.exists(datalake_orcamento_dir):
            os.makedirs(datalake_orcamento_dir)

    cria_datalake_orcamento_extensao = PythonOperator(
        task_id='cria_datalake_orcamento_extensao',
        python_callable=cria_diretorio,
    )

    cria_datalake_orcamento_extensao
```

Fonte: autoria própria.

Uma vez considerados os aspectos das ferramentas, um exemplo de tratamento de dados com varredura se deu no levantamento quantitativo de produções extensionistas. No exemplo a seguir no código 11 a função Python `classifica_producoes_de_relatorios_extensao`, invocada pela tarefa do Apache Airflow `transforma_quantitativo_producoes_extensao`, recebe o resultado da extração dos relatórios de extensão e o transforma em uma estrutura tabulada, em forma de matriz, chamada de *dataframe*. Em seguida é feito um dicionário de palavras-chave que são listadas palavras correspondentes a categorias ou chaves. Nas linhas a seguir percebe-se a utilização da função `categorizar_produto` para varredura do texto contido em cada registro tabulado das colunas de resultados qualitativos e quantitativos. Por sua vez, esse *dataframe* é transformado criando-se uma nova coluna `tipo_produto` com a categoria de produção extensionista identificada na varredura. Por fim, é realizado um agrupamento por câmpus, CEP, ano, tipo de produção e a quantidade dessa para obtenção dos indicadores.

Código 11 - Script Python com Apache Airflow de tratamento de dados com dicionário de palavras e *dataframes*.

```
def classifica_producoes_de_relatorios_extensao(ti) -> pd.DataFrame:
    #Obtém o resultado da query
    result_resultados_relatorio_extensao =
ti.xcom_pull(task_ids='extrai_resultados_relatorio_extensao')
    #Cria um dataframe pandas
    df_relatorio = pd.DataFrame(result_resultados_relatorio_extensao,
columns=['campus','cep','ano','resultados_qualitativos','resultados_quantitativos'])
    #dicionário de categoria: palavras-chave
    palavras_chave = {
        "Revistas e Periódicos": ["revista", "periódico", "journal"],
        "Livros ou Capítulos": ["livro", "capítulo", "book", "chapter"],
        "Artigos": ["artigo", "paper", "article"],
        "Mídias de Comunicações em Eventos":
["evento","comunicação","comunicacao"],
        "Produções Audiovisuais": ["audiovisual"],
        "Produções Artísticas": ["artística","arte"]
    }
    #categoriza conforme a coluna de resultados_qualitativos
    df_relatorio['tipo_produto_qualitativo'] = df_relatorio.apply(lambda x:
categorizar_produto(x['resultados_qualitativos'], palavras_chave), axis=1)
    #categoriza conforme a coluna de resultados_quantitativos
```

```

df_relatorio['tipo_produto_quantitativo'] = df_relatorio.apply(lambda x:
categorizar_produto(x['resultados_quantitativos'], palavras_chave), axis=1)
#faz a junção das colunas tipo_produto_qualitativo e
tipo_produto_quantitativo conforme a categoria
df_relatorio["tipo_produto"] =
df_relatorio["tipo_produto_qualitativo"].combine_first(df_relatorio["tipo_produto_quant
itativo"])

# Agrupar e contar em um novo dataframe
df_quantitativo_producoes = df_relatorio.groupby(['campus', 'cep', 'ano',
'tipo_produto']).size().reset_index(name='quantidade')
return df_quantitativo_producoes

transforma_quantitativo_producoes_extensao = PythonOperator(
    task_id='transforma_quantitativo_producoes_extensao',
    python_callable=classifica_producoes_de_relatorios_extensao
)

```

Fonte: autoria própria.

Embora o uso de dicionário de palavras-chave seja uma abordagem coerente na busca de produtos gerados a partir das ações de extensão, essa se limita pela quantidade de palavras catalogadas, tal qual a identificação de uma palavra não garante que o contexto em que se encontra não confere necessariamente a relação com um produto extensionista gerado. Outra forma identificada de aperfeiçoar essa abordagem é pela comunicação do Apache Airflow com alguma inteligência artificial generativa já treinada por técnica de *deep learning* a fazer análise semântica dos relatórios. No entanto, para uso dessa tecnologia baseada em modelos matemáticos requer uma seleção de fornecedores bem como firma de contrato, o que inviabilizou a adoção dela na pesquisa. Além do mais, nem sempre as inteligências artificiais garantem acurácia nas suas análises. Portanto, a possibilidade de customização do SIGAA e sua base para computar os produtos através de metadados do tipo numérico se faz mais apropriada para obtenção dos indicadores.

No caso de existência de metadados do tipo numérico nos registros foi possível realizar tratamentos de transformação a fim de se obter não somente a quantidade, também a proporção e porcentagem. No exemplo do código 12 é apresentado o cálculo percentual de discentes na extensão por câmpus e nível de curso. Primeiro obtêm-se os resultados das extrações de total de discentes e o total de discentes envolvidos, ambos por câmpus, por conseguinte são transformados em

dataframes e mesclados para que em seguida se faça a criação da coluna *percentual_discentes_na_extensao* que recebe o resultado do cálculo percentual (que pode ser um cálculo proporcional retirando-se o * 100) entre os registros das colunas *discentes_extensao* * 100 / *discentes_campus*. Por último o resultado percentual recebe uma máscara de arredondamento das casas decimais para apenas 2 (duas), outra de arredondamento de resultados infinitesimais e retorna os dados tratados em um *dataframe* composto por câmpus, cep, ano, tipo ou nível de curso e o percentual de discentes na extensão. Dessa forma, todas as lógicas de tratamento de dados usadas nesta pesquisa foram apresentadas.

Código 12 - Script Python com Apache Airflow de transformação e tratamento de dados percentuais.

```
def calcula_percentual_discentes_na_extensao(ti) -> pd.DataFrame:
    #Obtém o resultado das queries
    result_total_discentes_por_campus =
ti.xcom_pull(task_ids='extrai_total_discentes_por_campus')
    result_discentes_envolvidos_extensao =
ti.xcom_pull(task_ids='extrai_discentes_envolvidos_extensao')
    #Cria dataframes pandas a partir das queries
    df_result_total_discentes_por_campus =
pd.DataFrame(result_total_discentes_por_campus,
columns=['campus','cep','ano','tipo_curso','discentes_campus'])
    df_discentes_envolvidos_extensao =
pd.DataFrame(result_discentes_envolvidos_extensao,
columns=['campus','cep','ano','tipo_curso','discentes_extensao'])
    #Mescla os dois dataframes com base e cep preservando
df_discentes_envolvidos_extensao
    df_merged = pd.merge(
        df_discentes_envolvidos_extensao,
        df_result_total_discentes_por_campus,
        on=['campus','cep','ano','tipo_curso'],
        how='left'
    ).fillna(0)
    #Calcula o percentual de discentes na extensão por campus
    df_merged['percentual_discentes_na_extensao'] = (
        (df_merged['discentes_extensao'] * 100) /
df_merged['discentes_campus']
    ).fillna(0)
    # formata o percentual para até 2 casas decimais
    df_merged['percentual_discentes_na_extensao'] =
df_merged['percentual_discentes_na_extensao'].round(2)
```

```

# Garante que os anos que não tiveram discentes na extensão tenha
como resultado 0%
df_merged.loc[df_merged['discentes_extensao'] == 0,
'percentual_discentes_na_extensao'] = 0
# substitui valores infinitos por <0.01
df_merged.replace([np.inf, -np.inf], 0, inplace=True)
# Cria um novo dataframe apenas com as colunas interessadas
df_resultado_percentual_discentes_na_extensao =
df_merged[['campus','cep','ano','tipo_curso','percentual_discentes_na_extensao']]
return df_resultado_percentual_discentes_na_extensao

```

Fonte: autoria própria.

5.2.3 Carregamento de dados

O carregamento de dados é a fase final no *pipeline* do Apache Airflow. As estratégias se dividem entre carga completa em que há uma sobrescrita completa dos dados já existentes no destino, a carga incremental em que os novos dados são adicionados aos anteriormente processados, e a carga por partição em que se faz um carregamento com base no escopo unidade de tempo, a exemplo a carga por data de processamento gerando arquivos tabulados de dias diferentes para o mesmo conjunto de dados, sendo essa estratégia adotada para análise preditiva e regressiva por meio de inteligências artificiais e algoritmos de análise de dados (Inmon, 2005; Kimball, 2013).

Essas estratégias de carregamento de dados não definem os seus ambientes de armazenamento, esses são definidos como um *data lake*, ou lago de dados, um diretório que guarda dados brutos, podendo ser em nuvem ou em ambiente local. Outra definição é a de *data warehouse*, ou armazém de dados, onde existe uma base de dados com arquitetura bem definida para construção de relatórios, sendo que sua estrutura de relacionamentos entre tabelas se dá pela arquitetura estrela ou floco de neve. Na arquitetura estrela uma ou várias tabelas fato tem relacionamento com uma tabela dimensão, e na arquitetura floco de neve, uma extensão do modelo estrela, as tabelas dimensão sofrem o processo de normalização se dividindo em outras tabelas dimensão ligadas a uma tabela fato central (Inmon, 2005; Kimball, 2013).

Em suma as tabelas fato armazenam dados sobre eventos, e as tabelas dimensão os dados sobre características do objeto ou ser relacionado ao evento, por exemplo a venda de um automóvel, a venda é um evento que tem a data registrada, local, negociação, etc. Essa venda está relacionada a um objeto automóvel que tem por característica seu chassi, motor, etc. Também está relacionada a dois seres: o vendedor e comprador que possuem características físicas, cpf, idade, etc. Que por sua vez estão relacionados a eventos que os levaram ao evento compra, e essas influências têm conexão com outros objetos e/ou seres (Inmon, 2005; Kimball, 2013).

Esses tipos de cargas em *datalake* ou *datawarehouse* podem ocorrer de diversas formas, seja por requisições via protocolo usado na internet por meio de interfaces de programação de aplicações (API), seja por transações de banco de dados relacionais, armazenamento em dicionários chave-valor, ou mesmo na criação de arquivos tabulados com dados brutos.

Visto que o Looker Studio tem integração com o banco de dados Big Query, ambos do mesmo ecossistema, foi considerado na fase de projeto da elaboração do produto educacional a carga desse *datawarehouse* através de requisições escritas no *script* Python direcionadas à API desse banco de dados. Porém, para que a comunicação seja estabelecida é necessário gerar uma chave de segurança com a área da plataforma de domínio do IFSC, e como essa ferramenta não está inclusa no rol do contrato com a instituição, a sua utilização não estava disponível gratuitamente, portanto descartada.

Como alternativa optou-se por criar um *datalake* para cada macroindicador no mesmo ambiente do Apache Airflow para armazenamento de dados brutos em arquivos tabulados do tipo .csv, sendo que a cada processamento diário de *pipeline* há uma sobrescrita do arquivo com os dados atualizados do SIGAA, dessa maneira, sempre há indicadores com dados atualizados e corrigidos e o *datalake* não fica poluído com arquivos em excesso. Em contrapartida, essa abordagem depende que um operador de dados faça a transferência dos arquivos para o ecossistema do Looker Studio em detrimento da automação.

No código 13, nota-se a função `cria_planilha_quantitativo_atividades` escrita em Python e chamada pela tarefa `carrega_quantitativo_atividades`, que espera um parâmetro de comunicação entre tarefas de onde se recebe o resultado da extração,

neste caso não houve necessidade de tratamento desses pelo Apache Airflow. Em seguida o resultado da extração é transformado em um *dataframe* e o cabeçalho das colunas mapeado. Por fim, as estruturas condicionais criam o arquivo com nomenclatura previamente definida sobrescrevendo qualquer arquivo de mesmo nome dentro do *datalake*.

Código 13 - Script Python com Apache Airflow de carregamento de dados em arquivo .csv.

```
# Caminho do arquivo CSV de saída
datalake_extensao_dir = Variable.get("datalake")+ 'extensao/'
datalake_atividades_dir = datalake_extensao_dir+'atividades/'
csv_quantitativo_atividades_extensao =
datalake_atividades_dir+'IFSC_EXTENSAO_SIGAA_QUANTITATIVO_ATIVIDADES
_EXTENSAO.csv'

def cria_planilha_quantitativo_atividades(ti):
    #Obtém o resultado da transformação
    result_quantitativo_atividades =
ti.xcom_pull(task_ids='extrai_quantitativo_atividades')
    #Cria um dataframe pandas
    df_quantitativo_atividades =
pd.DataFrame(result_quantitativo_atividades)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_quantitativo_atividades = ['Câmpus','CEP','Ano','Ações de
Extensão']

    #Exporta para CSV
    if df_quantitativo_atividades.empty:
        df_quantitativo_atividades =
pd.DataFrame(columns=cabecalho_quantitativo_atividades)
        df_quantitativo_atividades.to_csv(csv_quantitativo_atividades_extensao,
index=False, encoding="utf-8", sep=",")
    else:
        df_quantitativo_atividades.to_csv(csv_quantitativo_atividades_extensao,
index=False, header=cabecalho_quantitativo_atividades, encoding="utf-8", sep=",")

    carrega_quantitativo_atividades = PythonOperator(
        task_id='carrega_quantitativo_atividades',
        python_callable=cria_planilha_quantitativo_atividades
    )
```

Fonte: autoria própria.

Em conclusão à estrutura dos *scripts* das DAGs a tarefa é mapeada ao final no fluxo do *pipeline* conforme o código 14. Uma vez elaboradas as tarefas o Apache Airflow as identifica como um grafo conforme a figura 8.

Código 14 - Script Apache Airflow de mapeamento de fluxo de processamento de dados.

```
# Definir as dependências entre as tasks
[extrai_quantitativo_atividades,
 extrai_quantitativo_atividades_continuas,
 extrai_quantitativo_atividades_aprovadas,
 extrai_quantitativo_atividades_submetidas,
 extrai_quantitativo_atividades_favoraveis,
 extrai_quantitativo_atividades_desfavoraveis,
 extrai_quantitativo_atividades_por_status,
 extrai_quantitativo_atividades_articuladas_pesquisa,
 extrai_quantitativo_atividades_articuladas_ensino,
 extrai_quantitativo_atividades_por_objetivo_sustentavel,
 extrai_participantes_servidores_envolvidos_extensao,
 extrai_listagem_editais_internos]
>> inicia_transformacao >>

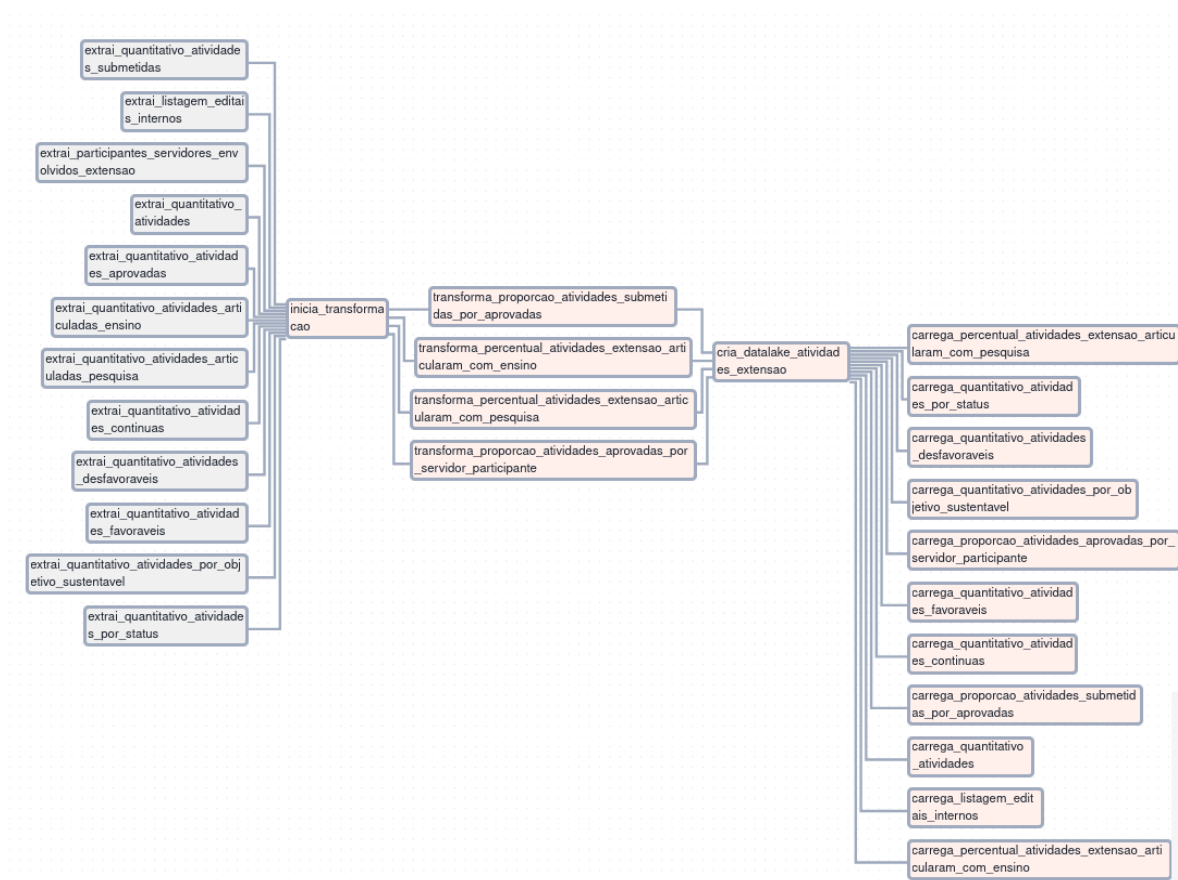
[transforma_proporcao_atividades_aprovadas_por_servidor_participante,
 transforma_proporcao_atividades_submetidas_por_aprovadas,
 transforma_percentual_atividades_extensao_articularam_com_pesquisa,
 transforma_percentual_atividades_extensao_articularam_com_ensino]
>> cria_datalake_atividades_extensao >>

[carrega_quantitativo_atividades,
 carrega_quantitativo_atividades_continuas,
 carrega_quantitativo_atividades_favoraveis,
 carrega_quantitativo_atividades_desfavoraveis,
 carrega_quantitativo_atividades_por_status,
 carrega_quantitativo_atividades_por_objetivo_sustentavel,
 carrega_proporcao_atividades_aprovadas_por_servidor_participante,
 carrega_proporcao_atividades_submetidas_por_aprovadas,
 carrega_percentual_atividades_extensao_articularam_com_pesquisa,
 carrega_percentual_atividades_extensao_articularam_com_ensino,
 carrega_listagem_editais_internos]
Fonte: autoria própria.
```

Com o grafo apresentável é possível observar a execução das tarefas a cada iteração conforme a figura 9. A leitura do gráfico ocorre da seguinte maneira: as colunas representam a execução do grafo todo e as linhas correspondem às

execuções de uma tarefa. Cada cor de quadrado está ligado a um significado, o verde corresponde a sucesso, o vermelho a falha, e o amarelo a uma suspensão normalmente ocorrida após um ou mais erros na execução de uma tarefa anterior. Também é possível notar em alguns casos a inexistência de quadrados, isso significa que naquele espaço de tempo a tarefa não existia ainda.

Figura 8 - Grafo do fluxo de dados de atividades de extensão.



Fonte: autoria própria.

Figura 9 - Execução do grafo de curricularização da extensão.



Fonte: autoria própria.

5.2.4 Exibição de dados e aplicação do produto educacional

No início da popularização da *world wide web*, comumente chamada de internet, as páginas de sítios e sistemas informatizados eram estáticos, compostos por textos com pouco ou nenhum componente visual agradável aos usuários, o que em seguida, após os primeiros mecanismos de animação, gerou um crescimento de páginas com muitos elementos visuais que causam desconforto. Sob esse cenário, Nielsen (1993) por meio do refinamento de experimentos e testes associados a necessidade de manter eficiência na navegação de páginas, estabeleceu 10 (dez) heurísticas que servem de padrão e adotada em fábricas de softwares no desenvolvimento de interfaces gráficas dos seus produtos buscando a melhor experiência dos usuários.

Heurísticas são princípios práticos para resolverem tarefas de maneira eficiente de fácil cognição e baixa análise exaustiva. Tendo por base este conceito as heurísticas sobre interface visual de *softwares* de Nielsen (1993) são: (1) a visibilidade do estado do sistema deve manter o usuário informado sobre o que está acontecendo em tempo razoável; (2) Os símbolos visuais e linguagem devem ser familiares ao usuário, corresponder entre o sistema e o mundo real; (3) O usuário deve ter controle e liberdade para corrigir ou desfazer suas ações; (4) O sistema

deve ter consistência em seus padrões visuais e funcionais para que o usuário saiba como interagir; (5) O sistema deve fazer prevenção de erros por meio de comportamentos dos componentes visuais, como exemplo desabilitar botões de submissão de formulários incompletos; (6) Reconhecimento em vez de memorização, o sistema deve reconhecer ao máximo a intenção do usuário ao navegar em vez de ter que forçar sua cognição e memória, por exemplo ao editar um formulário os campos devem vir preenchidos com as informações antes submetidas; (7) Flexibilidade e eficiência no uso, o sistema deve prover atalhos e personalizações para que usuários iniciantes e avançados tenham experiências semelhantes, como exemplo o fechamento de telas podem ocorrer pelo comando alt+f4 ou usar o componente visual de fechamento; (8) O *design* estético e minimalista, deve-se evitar poluição visual e excesso de informações para que o usuário não se distraia, as interfaces devem ser limpas e ao mesmo tempo agradáveis de observar; (9) O usuário deve ser capaz de reconhecer, diagnosticar e corrigir erros, para isso as mensagens devem ser claras explicando o problema e sugerindo soluções; (10) Ajuda e documentação, mesmo que o sistema seja intuitivo deve ser provida documentação de uso do sistema, tutoriais, área de perguntas frequentes, e canais de comunicação para ajuda.

Considerando esses aspectos, esta pesquisa, em um primeiro momento, considerou construir painéis de observação utilizando linguagens de programação, no entanto a revisão documental apresentou a possibilidade de utilização de ferramentas voltadas para exibição de dados fornecendo celeridade no processo de desenvolvimento do produto educacional. Das ferramentas levantadas, Power BI, Tableau e Looker Studio, todas atenderam às heurísticas. Porém, o Looker Studio teve como peso na tomada de decisão as heurísticas de familiaridade na linguagem, símbolos, padrões e *design* pelo corpo gestor do IFSC.

No entanto, por conta da ferramenta já possuir documentação de uso e não se enquadrar no desenvolvimento de uma API não foi necessário produzir documentação técnica em swagger. Todavia, foi elaborada documentação de implementação das ferramentas no repositório de códigos institucional do IFSC, bem como a produção de um encarte de aplicação e reprodução do produto educacional para a comunidade acadêmica.

O Looker Studio é uma ferramenta de inteligência de negócio (*business intelligence*) pertencente a uma plataforma como serviço (PaaS) em nuvem da qual o IFSC é contratante que possibilita a criação de relatórios e painéis de observação interativos de fácil manipulação, de uso colaborativo, publicizáveis, com conectividade a várias fontes de dados.

Quanto à conectividade de dados, além da possibilidade de integração do Apache Airflow com BigQuery, percebeu-se que o Looker Studio oferece a possibilidade de integração direta com o SIGAA. Porém, considerando os aspectos de segurança da informação ao expor as credenciais do banco de dados na internet, e a escalabilidade que o Apache Airflow oferece, optou-se por não seguir com essa abordagem.

O primeiro passo tomado para confecção dos painéis de indicadores foi a transferência dos arquivos .csv do *data lake* do Apache Airflow para a ferramenta Drive da plataforma do Looker Studio. Uma vez feita a integração do Drive com o Looker Studio, cada arquivo tabulado foi adicionado como uma fonte de dados, sendo necessário tratar os metadados das fontes de dados, a exemplo na figura 10, que apresenta os metadados da fonte de dados de produções extensionistas, nesta a coluna Ano foi alterada de tipo Texto para Número, a coluna CEP de tipo Número para tipo Geolocalização, ainda nota-se a criação de uma coluna personalizada chamada CEP-Câmpus em que se concatena dois registros de coluna para aprimorar a precisão de exibição de dados em gráficos em mapas.

Figura 10 - Metadados de fonte de dados no Looker Studio.

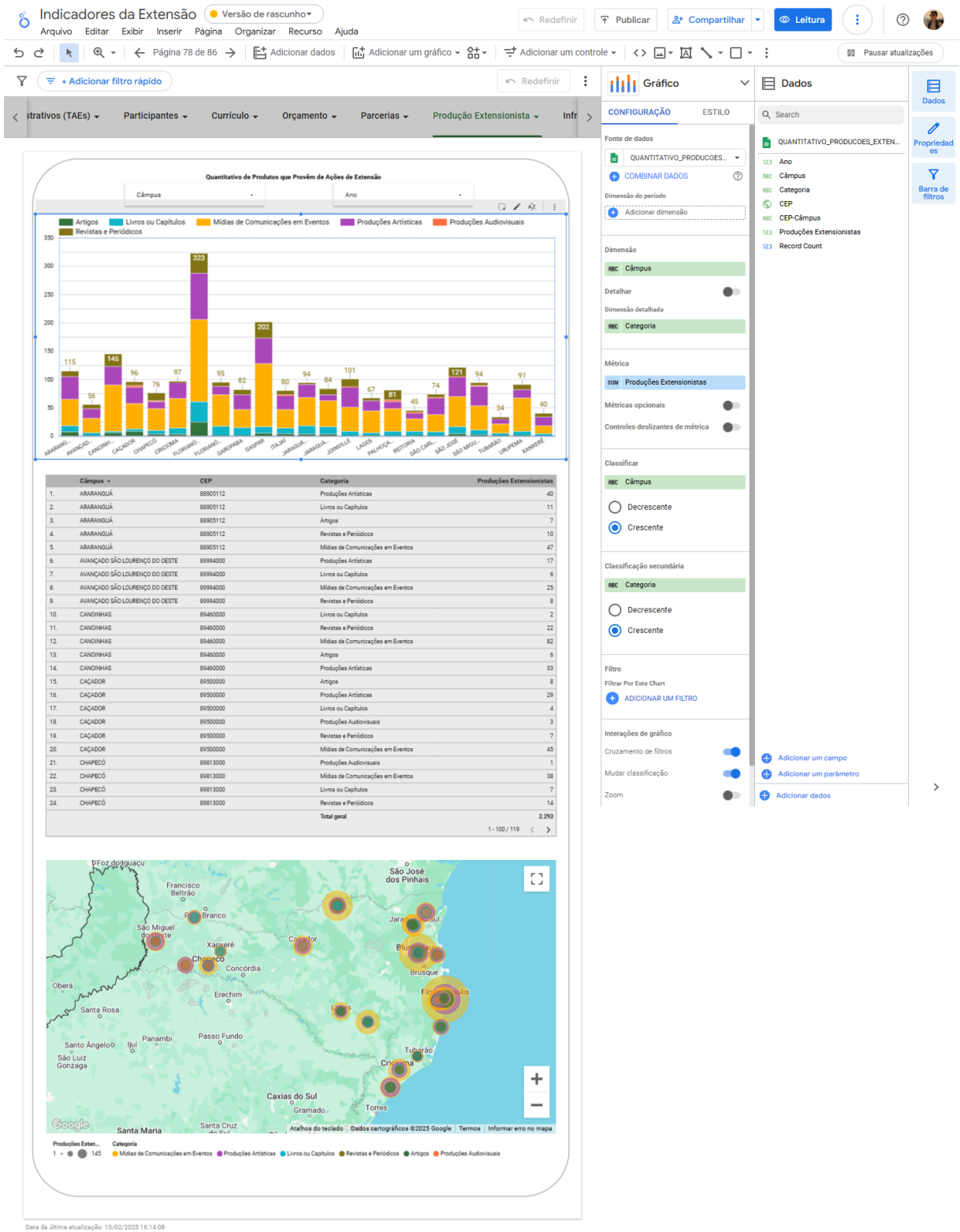
Campo ↓		Tipo ↓		Agregação padrão ↓	Descrição ↓
DIMENSÕES (6)					
Ano	⋮	123	Número	▼	Nenhum
Câmpus	⋮	ABC	Texto	▼	Nenhum
Categoria	⋮	ABC	Texto	▼	Nenhum
CEP	⋮		CEP	▼	Nenhum
CEP-Câmpus	fx ⋮	ABC	Texto	▼	Nenhum
Produções Extensionistas	⋮	123	Número	▼	Soma
MÉTRICAS (1)					
Record Count	⋮	123	Número	▼	Automático

Fonte: autoria própria.

Ao todo foram configuradas e tratadas 71 (setenta e uma) fontes de dados que em sua estrutura supriram a relação fato-dimensão. Dessa forma foi possível construir os painéis de dados. Em cada painel optou-se por trazer um gráfico de colunas simples ou empilhadas para uma visão mais abrangente dos dados, também deu-se destaque aos dados tabulados para melhor detalhamento, um gráfico mapa balão para observação dos dados por território, também a opção de filtros de câmpus e ano para maior detalhamento dos gráficos. A figura 11 apresenta a página de produções extensionistas, sendo seu gráfico de colunas do tipo empilhado por haver no total de produções a segmentação por categoria. Na mesma figura é possível observar nos menus a presença das opções de configuração da fonte de dados, a dimensão e métricas apresentadas nos gráficos.

Vale ressaltar que na construção dos filtros observou-se distorções nos dados da coluna ano, para contornar isso, criou-se um sub filtro para excluir a exibição desses conforme a figura 12.

Figura 11 - Página de edição do painel de dados de produções da extensão.



Fonte: autoria própria.

Figura 12 - Sub filtro do filtro de ano.

Nome: Filtro de Ano

QUANTITATIVO_PRODUCOES_EXTENSI...

☒ Mostrar valores sugeridos ao digitar

Excluir 123 Ano Maior que (>) 2030

AND

Excluir 123 Ano Menor que (<) 1900

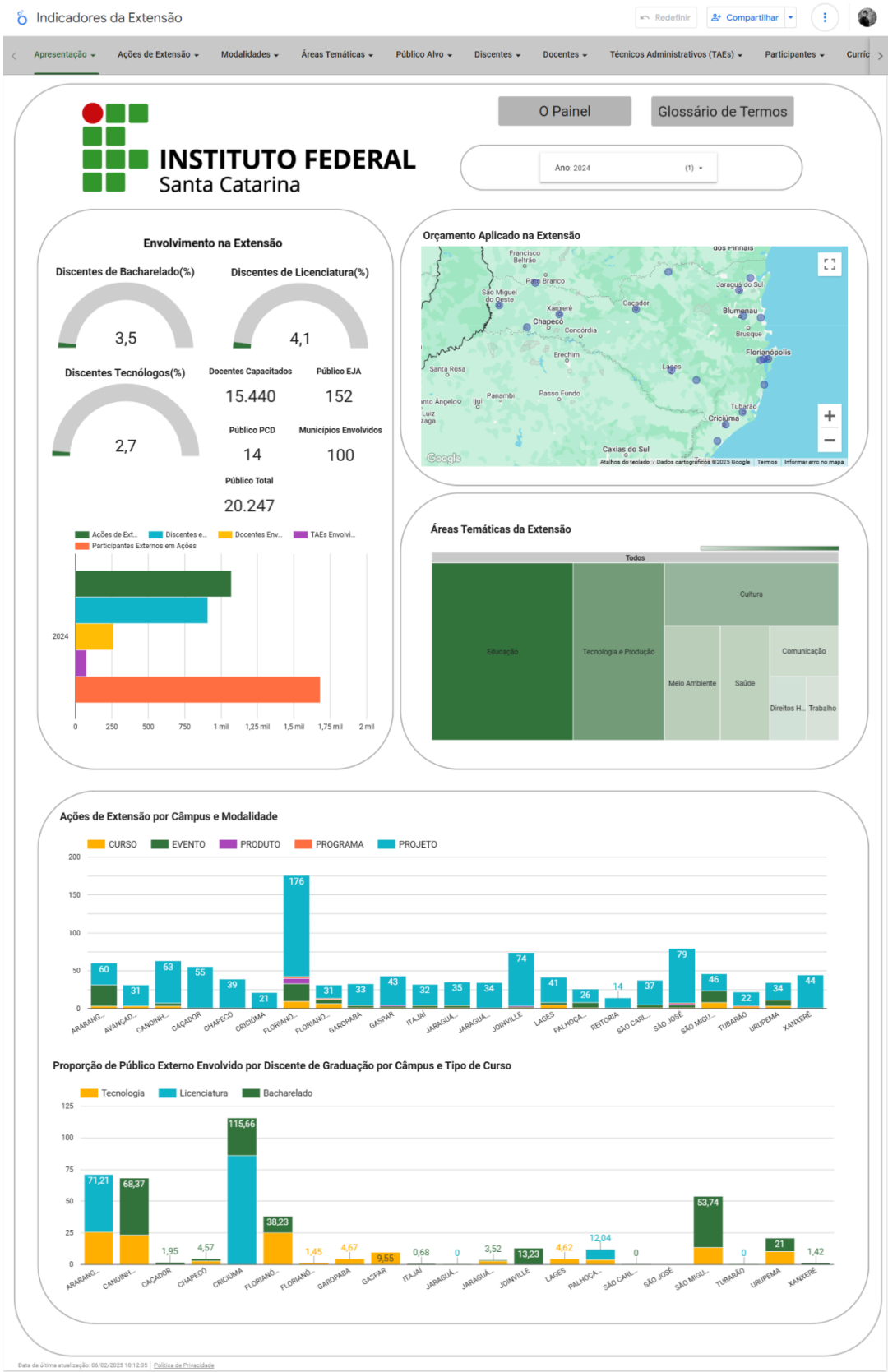
AND

Fonte: autoria própria.

Um *software* observatório é um sistema organizado para gestão e análise de informações contínuas sobre um determinado tema de observação (Schmidt, 2018). Com base nisso o produto educacional, batizado de Observatório da Extensão, também contou com uma página de apresentação do produto educacional, da pesquisa e sua navegabilidade. Ademais, criou-se uma página de glossário de termos para os usuários que não tem familiaridade com a EPT e a extensão. E como painel principal, foi construída uma página com os indicadores que dão um panorama da extensão segundo os estudos do Acórdão do TCU Nº 461, de 9 de março de 2022 fornecendo uma filtragem apenas por ano observável na figura 13.

O desenvolvimento do Observatório da Extensão produziu números relevantes que demonstram o esforço empregado em um escopo de tempo pré-definido para a pesquisa. Além dos 99 (noventa e nove) indicadores de extensão extraídos, tratados e carregados e das 71 (setenta e uma) fontes de dados mapeadas, ao todo foram customizadas 84 (oitenta e quatro) páginas de publicização de dados e escritas em torno de 5.565 (cinco mil, quinhentas e sessenta e cinco) linhas de *scripts* código, além da edição e configuração de arquivos de preparação de ambiente de desenvolvimento da solução.

Figura 13 - Tela de dados gerais do Observatório da Extensão.



Fonte: autoria própria.

5.3 AVALIAÇÃO E VALIDAÇÃO DO PRODUTO EDUCACIONAL

Desde a invenção do computador eletrônico a humanidade vem realizando tentativas de se domar os dados, buscando o equilíbrio entre o estabelecimento de processos e significado dos dados gerados. O esforço dessas tentativas não foi em vão, as instituições começaram a compreender que a tecnologia da informação deve ser ferramenta para a gestão de dados e que seu domínio vai além. É preciso estabelecer por meio de políticas e normas os papéis de responsabilidade sobre esses, como donos, gestores, curadores, cientistas e engenheiros de dados. Considerando que a organização tem processos bem definidos de gestão e governança, o passo seguinte é estabelecer estratégias de segurança e tratamento de dados e metadados desde a sua criação à publicização para então haver intervenção das tecnologias da informação (Barbieri, 2019).

No entanto, esse cenário só será possível se toda a instituição fizer um esforço de mudança de cultura. Se as pessoas mudarem sua forma de pensar sobre os dados, estes mudarão a forma de representar valor para a instituição. São as pessoas que mudam cultura e que podem construir convicções e valores pela crença obtida a partir do que é observável pelos dados (Barbieri, 2019; Inmon, 2005).

Considerando isso, para a avaliação e validação do Observatório da Extensão foram selecionados voluntários com papéis de potencial mudança de cultura na gestão de dados da extensão no IFSC para compor um grupo focal que pudessem contribuir com observações qualitativas sobre o produto. Segundo Dias (2000), um grupo focal não deve ser grande, recomenda-se que o pesquisador no papel de moderador do grupo consiga conduzir as discussões em um período de tempo que seja possível manter o engajamento dos participantes e que tanto os introvertidos e os extrovertidos se sintam impelidos a dar voz a suas impressões.

Considerando isso, foram convidados 2 (dois) membros da pró-reitoria de extensão e relações externas com o ponto de vista da unidade central de gestão da extensão; 2 (dois) membros de câmpus lotados no departamento de ensino, pesquisa e extensão e na coordenadoria de extensão e relações externas ou unidade com atribuições semelhantes, 1 (um) membro da diretoria de gestão do conhecimento da pró-reitoria de desenvolvimento institucional enquanto enquanto

especialista da instrumentalização de processos e avaliação institucional; 1 (um) membro da coordenadoria de gestão de dados; e 1 (um) membro da ouvidoria do IFSC como representante da sociedade civil uma vez que essa unidade recebe as demandas de interesse desses.

Foi dado aos membros voluntários ciência e assinatura do termo de consentimento livre e esclarecido de grupo focal conforme o apêndice C, foram reunidos no dia 17 (dezessete) de dezembro de 2024 (dois mil e vinte e quatro) em uma sala de videoconferência, na qual teve duas ausências. Houve uma apresentação ao produto educacional Observatório da Extensão quanto a sua construção, funcionalidade e usabilidade a fim de induzir discussões de como os painéis de observação dos indicadores da extensão contribuem para a tomada de decisão da gestão e para a publicização de dados extensionistas de interesse da sociedade civil e comunidade acadêmica, bem como obter sugestões de melhorias do produto e contribuir com discussões de melhoria de processos.

O grupo focal reconheceu que o Observatório da Extensão trouxe avanços significativos para a extração e análise de informações. Segundo o representante da coordenadoria de gestão de dados, os painéis conferiram maior plasticidade na obtenção de dados do sistema, proporcionando melhorias na elaboração de relatórios e facilitando a visualização de informações que antes eram limitadas pelo SIGAA. Destacou-se, ainda, a possibilidade de filtragem e segmentação de informações por câmpus, algo que a arquitetura inicial do SIGAA não contemplava de maneira satisfatória.

Entre os cinco participantes, quatro consideraram a visualização e apresentação dos dados confortável e intuitiva. A exposição das informações contribuiu para tornar perceptíveis algumas tendências antes tácitas, como a maior participação de docentes em atividades de extensão em comparação aos Técnicos Administrativos em Educação (TAEs) e o impacto da curricularização da extensão nos câmpus.

Duas voluntárias de gestão da extensão, em uma primeira análise dos painéis, identificaram oportunidades de melhoria na tomada de decisão em seu câmpus, reforçando o papel do Observatório da Extensão como um instrumento estratégico para a gestão. Essas mesmas participantes enfatizaram que esse produto educacional é uma ferramenta de transparência, permitindo que o IFSC

demonstre à sociedade as iniciativas desenvolvidas na extensão e aproxime a comunidade externa à instituição.

Durante a análise, alguns participantes notaram inconsistências nos gráficos apresentados, levando à reflexão sobre possíveis fragilidades no registro de informações no sistema. Essa percepção resultou no entendimento de que o Observatório da Extensão pode contribuir para fortalecer a cultura institucional de gestão de dados, promovendo a revisão de processos e da pluralidade de sistemas utilizados para o registro de dados.

Isso ensejou a discussão sobre a dificuldade de registro de informações pela comunidade interna. Muitas vezes, determinados dados não aparecem devido a diferentes interpretações de nomenclaturas e processos no sistema. Além disso, foi levantada a hipótese de que a carga horária docente pode influenciar nesse contexto, uma vez que os professores podem não dedicar tempo suficiente ao preenchimento das informações devido à ausência de previsão normativa interna ou ao excesso de atividades acadêmicas. Essa reflexão indica que o Observatório da Extensão pode fomentar revisões de políticas institucionais, levando a debates sobre normativas e boas práticas para aprimorar o registro de dados.

Uma outra análise relevante foi a do representante da Ouvidoria, este avaliou positivamente o painel no que se refere à transparência de dados, especialmente em relação à Lei de Acesso à Informação (LAI). A possibilidade de acesso rápido às informações contribui para agilizar respostas e minimizar requisições de acesso à informação, além de fortalecer a cultura da transparência institucional.

Entre os apontamentos negativos, uma participante que recentemente ingressou na gestão da extensão observou que, à primeira vista, o painel pode representar um desafio para novos usuários que não estão familiarizados com *softwares* observatórios. A grande quantidade de indicadores e termos técnicos pode dificultar a leitura e interpretação das informações, especialmente para aqueles que ainda estão se adaptando ao sistema. No entanto, ela reconheceu que a estruturação das páginas é intuitiva, facilitando a navegação entre os diferentes macroindicadores.

Apesar dessa navegabilidade fluida, foi identificada uma dificuldade na usabilidade dos filtros, tornando a segmentação dos dados menos intuitiva para usuários inexperientes. Ainda assim, a participante enfatizou seu encantamento com

o painel e reconheceu seu potencial como ferramenta facilitadora na rotina de trabalho. Além disso, sugeriu que a iniciativa pode motivar a abrangência de painéis específicos para ensino e pesquisa, reforçando a importância da ferramenta na gestão acadêmica de forma ampla.

Um ponto de atenção levantado pelo grupo focal é a necessidade de incrementar a automação na integração do processo de carregamento com a exibição dos dados. Isso permitiria reduzir a intervenção manual e minimizar possíveis erros na publicização de informações, sobretudo quando há transição da função de operador de dados entre servidores, o que está assumindo a tarefa de carregamento precisa assimilar a execução da atividade e disso podem surgir falhas.

Outra sugestão de melhoria foi a inclusão de mais um indicador de extensão, a listagem nominal para editais com e sem fomento, com fonte orçamentária e valor por câmpus e ano. Sendo assim, o total de indicadores de extensão nesta pesquisa passou para 114 (cento e quatorze) conforme o quadro 10, com atendimento a 100 (cem) pelo Observatório da Extensão, na confecção de 85 (oitenta e cinco) páginas com painéis composto por gráficos, sendo o de listagem dos editais observáveis na figura 14.

De maneira geral, o grupo focal avaliou positivamente a implementação do Observatório da Extensão, destacando que a publicização dos indicadores de extensão atinge seu objetivo de apoiar a tomada de decisão pela gestão proporcionando maior clareza na análise de dados e dá maior transparência das atividades que podem aproximar a sociedade da instituição. A iniciativa demonstrou seu potencial não apenas para aprimorar a gestão e transparência da extensão, mas também para influenciar políticas institucionais. Dessa forma, o Observatório da Extensão pode se consolidar como um instrumento fundamental para o IFSC, promovendo maior integração, análise estratégica e aproximação com a comunidade.

Considerando a avaliação positiva pelo grupo focal, foi encaminhada à PROEX o resultado da análise e validação do Observatório da Extensão para confecção de matéria institucional junto à diretoria de comunicação para divulgação do produto educacional dentro do domínio do IFSC. Porém, foi esclarecido que há procedimento institucional para isso após a aprovação da pesquisa pelo programa

de mestrado e período de estudo do uso da ferramenta pela diretoria de extensão do IFSC, portanto postergado para momento futuro.

Quadro 10 - Estratificação de indicadores de extensão provenientes de documentos infraconstitucionais, de fóruns da extensão, de documentos do IFSC, de solicitações da sociedade de dados extensionistas, de entrevistas semiestruturadas com a gestão da extensão do IFSC, com apontamentos do grupo focal.

MACROINDICADORES	INDICADORES
1. Docentes em atividades de extensão	- quantitativo por papel de participante voluntário; - quantitativo por papel de bolsista; - quantitativo por papel de coordenador; - quantitativo por papel de orientador; - quantitativo de docentes que submetem ações de extensão; - quantitativo de docentes envolvidos em atividades de extensão; - percentual de docentes envolvidos em atividades de extensão.
2. TAEs em atividades de extensão	- quantitativo por papel de participante voluntário; - quantitativo por papel de bolsista; - quantitativo por papel de coordenador; - quantitativo de TAEs envolvidos em atividades de extensão; - percentual de TAEs envolvidos em atividades de extensão.

3. Discentes em atividades de extensão	- quantitativo por papel de participante voluntário; - quantitativo por papel de bolsista; - quantitativo de discentes envolvidos em atividades de extensão; - quantitativo de discentes envolvidos em atividades de extensão além dos créditos obrigatórios; - percentual de discentes envolvidos em ações de extensão.
4. Participantes	- quantitativo total de servidores envolvidos em atividades de extensão; - quantitativo de participantes externos envolvidos em atividades de extensão; - quantitativo de participantes externos envolvidos em atividades de extensão com papel de voluntário; - quantitativo de participantes de atividades de extensão por papel; - quantitativo de participantes de atividades de extensão por status; - proporção de participantes externos por total de atividade de extensão; - proporção de participantes externos por total de participantes; - percentual de servidores envolvidos em atividades de extensão;

5. Modalidades de atividades de extensão	- quantitativo de curso; - quantitativo de evento; - quantitativo de programa; - quantitativo de produto; - quantitativo de projeto; - quantitativo de prestação de serviço; - proporção discentes com matrícula ativa por programas e projetos; - proporção discentes com matrícula ativa por programas; - quantitativo de discentes com matrícula ativa por programas; - proporção discentes com matrícula ativa por projetos; - quantitativo de discentes com matrícula ativa em projetos; - proporção discentes com matrícula ativa por produtos; - quantitativo de discentes com matrícula ativa por produtos; - proporção discentes com matrícula ativa por prestação de serviço; - quantitativo de discentes com matrícula ativa por prestação de serviço; - quantitativo de discentes com matrícula ativa por curso; - quantitativo de discentes com matrícula ativa por evento; - proporção de atividades de extensão no ano por modalidade projeto; - proporção de atividades de extensão no ano por modalidade curso;
---	--

	- quantitativo de incubadoras; - quantitativo de empreendimentos graduados em incubadoras resultados da extensão; - quantitativo de cooperativas populares graduadas em incubadoras resultantes da extensão;
6. Curricularização da extensão	- quantitativo de discentes em atividade curricular; - percentual de créditos ou carga horária destinados à extensão no currículo de nível médio técnico; - percentual de créditos ou carga horária destinados à extensão no currículo de graduação; - percentual de créditos ou carga horária destinados à extensão no currículo de especialização; - percentual de créditos ou carga horária destinados à extensão no currículo de mestrado; - percentual de créditos ou carga horária destinados à extensão no currículo de doutorado; - percentual de créditos ou carga horária destinados à extensão no currículo de pós-graduação (mestrado e doutorado); - percentual de créditos ou carga horária de ações de extensão no currículo de cursos; - quantitativo de atividades de extensão por tipo de curso; - Percentual de PPC com curricularização da extensão e atividades complementares.

7. Orçamento	- percentual de recursos financeiros do orçamento anual público aplicados em extensão; - quantitativo de recurso financeiro orçamentário total por câmpus; - quantitativo de recursos financeiros do orçamento anual público obtido para extensão; - quantitativo de recursos financeiros do orçamento anual público aplicados em extensão; - quantitativo de recursos para extensão captados via prestação de serviços acadêmicos especializados; - quantitativo de recursos para extensão captados via editais externos; - quantitativo de recurso aplicado por edital interno de extensão; - quantitativo de recurso repassado aos coordenadores de atividade de extensão; - quantitativo de recurso repassado aos bolsistas discentes;
8. Público alvo	- quantitativo de ações de extensão demandadas pelo público externo; - quantitativo de pessoas envolvidas por ações de extensão; - quantitativo de pessoas envolvidas por atividades de modalidade prestação de serviço; - quantitativo de pessoas envolvidas por atividades de modalidade programas e projetos; - quantitativo de pessoas envolvidas por atividades de modalidade cursos e eventos;

	- quantitativo de ações de extensão para capacitação de população EJA; - quantitativo de ações de extensão para capacitação de população PCD; - quantitativo de pessoas envolvidas por discentes de graduação por meio de ações de extensão; - quantitativo de docentes da rede pública capacitados por programas e projetos de formação continuada; - percentual de ações de extensão destinadas à inclusão de população vulnerável; - percentual de ações de extensão dirigidas às escolas públicas; - percentual de municípios atendidos pelas ações extensionistas em relação ao total de municípios que são abrangidos pela respectiva instituição de ensino/câmpus.
9. Parcerias	- percentual de ações de extensão com parcerias institucionais; - quantitativo de ações de extensão com parcerias institucionais; - percentual de representação oficial da instituição junto à sociedade civil; - taxa de aprovação de propostas de extensão em editais externos; - listagem nominal das parcerias institucionais com ações de extensão; - quantitativo de instituições ativas nos fóruns de extensão e relações externas no câmpus.

10. Produção extensionista	- quantitativo de revistas e periódicos institucionais; - quantitativo de produção de livros ou capítulos com base em resultados da extensão; - quantitativo de publicação de artigos em periódicos com base em resultados da extensão; - quantitativo de mídias de comunicações em eventos com base em resultados da extensão; - quantitativo de produções audiovisuais resultadas da extensão; - quantitativo de produções artísticas (exposições, espetáculos, outros) resultantes da extensão.
11. Áreas temáticas de extensão	- quantitativo de atividades de extensão por área da comunicação; - quantitativo de atividades de extensão por área da cultura; - quantitativo de atividades de extensão por área dos direitos humanos e justiça; - quantitativo de atividades de extensão por área da tecnologia e produção; - quantitativo de atividades de extensão por área do trabalho;

12. Infraestrutura e recursos humanos	- proporção de horas de treinamento por servidor da gestão extensionista; - proporção de participação em eventos por servidor da gestão extensionista; - proporção de participação em eventos promovidos ou apoiados pela pró-reitoria de extensão por servidor da gestão extensionista; - proporção espaço físico (área) por servidor; - percentual de capacidade de atendimento de demandas de transporte logístico para ações de extensão; - percentual de servidores em representação oficial da instituição junto à sociedade civil; - quantitativo de unidades organizacionais proponentes de ação de extensão; - listagem nominal das unidades organizacionais proponentes de ação de extensão.
13. Atividades de extensão	- quantitativo total de atividades de extensão; - quantitativo de atividades de extensão contínuas; - proporção de ações de extensão com status aprovada por servidor; - proporção de ações de extensão submetidas por aprovadas; - quantitativo de atividades de extensão com status favorável; - quantitativo de atividades de extensão com status desfavorável; - quantitativo de atividade de extensão por status; - percentual de atividade de extensão que articularam com a pesquisa;

	- percentual de atividade de extensão que articularam com o ensino; - quantitativo de atividades de extensão por objetivo sustentável da ONU; - listagem de editais internos com e sem fomento. (AGREGADO)
--	---

Fonte: autoria própria.

Figura 14 - Tela listagem de editais com e sem fomento do Observatório da Extensão.

<

Apresentação ▾

Ações de Extensão ▾

Modalidades ▾

Áreas Temáticas ▾

Público Alvo ▾

Discentes ▾

Docentes ▾

Técnicos Administrativos (TAEs) ▾

Participantes ▾

Currículo ▾

Listagem Nominal dos Editais Internos com e sem Fomento

Câmpus ▾

Ano ▾

	Câmpus ⓘ ▾	Ano ⓘ ▾	Descrição	Fonte Orçamentária	Financiamento
1.	ARARANGUÁ	2024	2024_Edital EPE PROPP/PROEX nº 02/2024 - Fomento às atividades indissociáveis entre ensino, pesquisa e extensão do Câmpus Araranguá	Câmpus	R\$ 24.600,00
2.	ARARANGUÁ	2023	2023_Edital EPE PROPP/PROEX 02 - Fomento às atividades EPE do Câmpus Araranguá	Câmpus	R\$ 19.750,00
3.	ARARANGUÁ	2022	2022_Edital PROPP/PROEX 02/2022 Fomento às atividades EPE do IFSC Câmpus Araranguá	Câmpus	R\$ 18.500,00
4.	ARARANGUÁ	2021	2021_PROPP/PROEX 50 - Edital de Câmpus_ARU - Fomento às atividades EPE	Câmpus	R\$ 16.500,00
5.	ARARANGUÁ	2020	2019_PROEX 36 - Edital de Câmpus_ARU	Câmpus	R\$ 62.700,00
6.	AVANÇADO SÃO LOURENÇO DO OESTE	2024	2024_PROEX 04 - Edital de Câmpus_SLO	Câmpus	R\$ 30.000,00
7.	AVANÇADO SÃO LOURENÇO DO OESTE	2023	2023_PROEX 03 - Edital de Câmpus_SLO	Câmpus	R\$ 15.000,00
8.	AVANÇADO SÃO LOURENÇO DO OESTE	2021	2021_PROEX 20 - Edital de Câmpus_SLO	Câmpus	R\$ 6.800,00
9.	AVANÇADO SÃO LOURENÇO DO OESTE	2019	2019_PROEX 23 - Edital de Câmpus_SLO	Câmpus	R\$ 5.000,00
10.	CANOINHAS	2024	Edital PROEX nº25/2024 Fomento às atividades de extensão do Câmpus Canoinhas	Câmpus	R\$ 70.000,00
11.	CANOINHAS	2023	2023_PROEX 18 - Fomento às atividades de extensão do Câmpus Canoinhas	Câmpus	R\$ 60.000,00
12.	CANOINHAS	2022	2022_PROEX 07 - Edital de Câmpus_Canoinhas	Câmpus	R\$ 40.003,00
13.	CANOINHAS	2021	2021_PROEX 25 - Edital de Câmpus_CAN	Câmpus	R\$ 10.000,00
14.	CANOINHAS	2021	2021_PROEX 30 - Edital de Câmpus_CAN_2o_Lançamento	Câmpus	R\$ 12.000,00
15.	CANOINHAS	2020	2020_PROEX 14 - Edital de Câmpus_CAN - Fomento às atividades de	Câmpus	R\$ 10.000,00

1 - 100 / 297 < >

Fonte: autoria própria.

6 CONSIDERAÇÕES FINAIS

A presente pesquisa teve como objetivo central a identificação e publicização de indicadores de extensão no IFSC que se revelaram por meio dos documentos, solicitações de acesso à informação e entrevistas com gestores, o meio para alcançar isso é devido a implementação e avaliação do Observatório da Extensão, um produto educacional voltado à gestão e transparência de dados extensionistas no Instituto Federal de Santa Catarina. Com base nos objetivos gerais e específicos, foi possível demonstrar como a ferramenta desenvolvida contribui para a sistematização, análise e divulgação dos dados da extensão, promovendo uma maior compreensão sobre o tema e possibilitando a melhoria de processos institucionais.

A aplicação dos procedimentos metodológicos adotados, que incluíram revisão documental e bibliográfica, exploração de dados, entrevistas com gestores extensionistas e a formação de um grupo focal, permitiu a construção de um artefato funcional e alinhado às necessidades institucionais. A abordagem baseada na pesquisa-ação e no *Design Science Research* possibilitou não apenas o desenvolvimento do produto educacional, mas também a reflexão sobre seu impacto e aperfeiçoamento com base na opinião dos usuários por meio do grupo focal. Logo, para que o produto educacional seja aprimorado, é desejável que esses procedimentos sejam reaplicados em uma escala maior a nível institucional e aplicados a nível nacional.

O Observatório da Extensão aprimora o cenário relacionado à gestão de informações sobre a extensão acadêmica, que antes eram fragmentadas, de difícil acesso e exigiam extração manual. A ferramenta automatiza processos, aprimora a exibição de dados e possibilita maior transparência, facilitando a tomada de decisão dos gestores e o acompanhamento da comunidade externa sobre as ações desenvolvidas pelo IFSC. Além disso, a pesquisa revelou fragilidades na cultura institucional de registro de dados, indicando a necessidade de revisão de customizações de sistemas, processos e capacitação contínua dos envolvidos. Essa lacuna na gestão dos dados reforça a importância de uma ferramenta que não apenas publique informações, mas também sirva como um instrumento pedagógico para fomentar a cultura da transparência, aprimoramento de normativas e

discussões progressistas. Todavia, essa cultura não deve ser direcionada para estimular um sistema de vigilância, e sim um escudo para a educação profissional, científica e tecnológica.

Apesar do produto educacional oferecer uma usabilidade intuitiva, a navegabilidade entre páginas se mostra desafiadora para novos usuários inexperiente, especialmente no que diz respeito à compreensão de termos e à usabilidade dos filtros, o que pode ser atribuído à fase de adaptação ao sistema. Porém, ainda há margem para melhorias de *design* (UI) e experiência do usuário (UX) na construção ou adoção de outras ferramentas de visualização de dados.

Não obstante, a automatização entre o processo de carregamento e exibição dos dados se mostra crucial para a continuidade da solução, para tanto há que rever o contrato da plataforma de ferramentas ou considerar outra de uso livre que resolva esse gargalo. Sanar isso é relevante sobretudo em contextos de transição de funções de gestão, minimizando riscos de inconsistências na carga dos dados, essas que devem ser constantemente revisadas para desvelamento das informações condizentes com a realidade.

A transparência dos dados, corrobora a importância do produto educacional no atendimento às exigências da LAI e na aproximação do IFSC com a comunidade externa, evidenciando seu potencial como ferramenta estratégica de comunicação e melhoria contínua das práticas institucionais e dos mecanismos democráticos. Além da solução ensejar a construção de painéis do ensino e da pesquisa, possibilita a replicabilidade da solução em outras instituições independente do sistema de registro de dados que utilizam, também traz o potencial institucionalização da ferramenta em escala nacional ser mantido pelo Ministério da Educação, com o objetivo de fortalecer a educação e o preparo da população para o mundo do trabalho.

Ainda assim, alternativamente, percebe-se que outras técnicas de gestão de dados possam ser exploradas na evolução do Observatório da Extensão, como o uso de aprendizado de máquina para a identificação de padrões nos dados extensionistas ou a incorporação de técnicas de ciência de dados mais especializadas. Outras abordagens metodológicas, como o uso de métodos quantitativos para mensuração do impacto das políticas extensionistas, também poderiam enriquecer futuras pesquisas sobre o tema. Ademais, a integração do

Observatório com outras plataformas de gestão educacional pode ampliar sua aplicabilidade, tornando-o um recurso fundamental para diversas frentes institucionais. Além disso, técnicas como a modelagem preditiva de dados e a análise de redes sociais podem ser incorporadas para mapear interações e parcerias extensionistas.

Outra possibilidade de abordagem metodológica seria a adoção da mineração de dados educacionais, permitindo uma análise mais aprofundada dos padrões de participação na extensão e suas implicações para a formulação de políticas. Também se recomenda a exploração de metodologias de modelagem de interface visual centrado no usuário para aprimorar a usabilidade do Observatório da Extensão, tornando-o ainda mais acessível a diferentes perfis de usuários, sendo esses jovens, adultos e idosos, com ou sem deficiências físicas.

Apesar dos avanços proporcionados pelo Observatório da Extensão, desafios estruturais e culturais ainda persistem. A gestão de dados na extensão depende diretamente do comprometimento dos servidores em registrar e atualizar informações com precisão. No entanto, há desafios em envolvê-los na melhoria dos processos sistêmicos, reflexo de fatores como alta rotatividade de pessoal, sobrecarga de atividades, precariedade da força de trabalho na rede EPCT e uma cultura de gestão e governança de dados incipiente.

Associado a isso, o gestor da extensão se depara com a complexa tarefa de equilibrar a necessidade de aprimorar processos, atender às demandas da sociedade, desempenhar seu papel nos fóruns de discussão, executar as atividades intrínsecas ao seu cargo, prover capacitação à equipe de trabalho, tudo isso com recursos orçamentários limitados, o que compromete a continuidade e evolução de políticas institucionais.

Entretanto, o produto educacional aqui desenvolvido se apresenta como um instrumento facilitador da gestão, estimulando não apenas a transparência, mas também o engajamento da comunidade interna e externa. A sistematização dos dados proporciona uma visão histórica e crítica da extensão, o que pode fomentar debates mais aprofundados sobre a formulação de políticas públicas. A transparência dos dados permite que a sociedade compreenda o papel da extensão na formação acadêmica e no desenvolvimento social, fortalecendo a relação entre a instituição e a comunidade.

Sendo assim, o Observatório da Extensão também se põe como meio de prover amannualidade para os interessados, reduzindo o distanciamento entre a entropia e cibernética, que se reflete na melhoria da organização e controle de processos, reduzindo a incerteza na gestão de dados extensionistas. Ao estruturar informações de forma sistemática, contribui-se para a previsibilidade e confiabilidade dos registros, minimizando falhas operacionais.

Por fim, a evolução da ferramenta se dará mediante iniciativas de formação e sensibilização dos servidores sobre a importância da cultura de gestão de dados, garantindo sua continuidade e efetividade como um instrumento de gestão e transparência, pois, são as pessoas, a cultura, que sistematiza processos, logo uma solução de tecnologia da informação e comunicação é o reflexo da organização de uma instituição. Em tempos de popularização de análises feitas com inteligências artificiais, de registro enorme de informações em ambiente eletrônico, do combate a desinformação em massa nas democracias, e da digitalização de governos, sendo o dado o combustível para esse mecanismo, cabe a reflexão individual sobre que tipo de dado se está registrando, um que vai revelar com fidedignidade as atividades de trabalho que fortalecem a educação profissional, científica e tecnológica ou que deixará uma visão histórica distorcida para gerações seguintes.

REFERÊNCIAS

ALCÂNTARA, Dalmi; MOZART, Júnior; GRAÇA, Luziray; ALE, Roberta. **Ética e qualidade no atendimento**. Goiânia, GO: Kelps Editora, 2018.

ALVES, Dayane. **Plataforma Interativa de Controle e Monitoramento Orçamentário em uma Instituição Pública de Educação Profissional e Tecnológica**. Instituto Federal de Educação, Ciência e Tecnológica de Santa Catarina, 2020. Disponível em: https://sucupira.capes.gov.br/sucupira/public/consultas/coleta/trabalhoConclusao/viewTrabalhoConclusao.xhtml?popup=true&id_trabalho=9284366

ANPD. **Programa de integridade e comissão gestora**. 2024. Disponível em <https://www.gov.br/anpd/pt-br/assuntos/noticias/anpd-institui-programa-de-integridade-e-cria-comissao-gestora>.

BARBIERI, Carlos. **Governança de Dados: práticas, conceitos e novos caminhos**. Rio de Janeiro, RJ: Alta Books, 2019.

BERGSON, Henri. **A evolução criadora**. In: . Cartas, conferências e outros escritos. São Paulo: Abril Cultural, 1979. p. 153-205. (coleção os Pensadores.)

BOBBIO, N. **Estado governo sociedade: para uma teoria geral da política**. 2ª ed. Paz e Terra, Rio de Janeiro, 1987.

BORDENAVE, Juan E. Díaz. **O que é comunicação**. Tatuapé, SP: editora brasiliense, 1997.

BORDENAVE, Juan E. Díaz. **Além dos meios e mensagens**: Introdução à comunicação como processo, tecnologia, sistema e ciência. Petrópolis, RJ: editora vozes, 1998.

BRASIL. **Lei nº 6.545, de 30 de junho de 1978**. Criação dos centros federais de educação tecnológica. Disponível em: https://www.planalto.gov.br/ccivil_03/leis/l6545.htm

BRASIL. **Constituição da República Federativa do Brasil de 1988, capítulo VII, art. 227.** Direitos da criança, adolescentes e jovens. Disponível em: http://www.planalto.gov.br/ccivil_03/constituicao/constituicao.htm.

BRASIL. **Lei nº 9.394, de 20 de dezembro de 1996.** Diretrizes e bases da educação nacional. Disponível em: http://www.planalto.gov.br/Ccivil_03/leis/L9394.htm

BRASIL. **Lei nº 11.892, de 29 de dezembro de 2008.** Fundação da Rede Federal de Educação. Disponível em: https://www.planalto.gov.br/ccivil_03/_ato2007-2010/2008/lei/l11892.htm.

BRASIL. **Decreto nº 7.416, de 30 de dezembro de 2010.** Concessão de bolsas para desenvolvimento de atividades de ensino e extensão universitária. Disponível em http://www.planalto.gov.br/ccivil_03/_ato2007-2010/2010/decreto/d7416.htm

BRASIL. **Lei nº 12.527, de 18 de novembro de 2011.** Lei de Acesso à Informação. Disponível em https://www.planalto.gov.br/ccivil_03/_ato2011-2014/2011/lei/l12527.htm.

BRASIL. **Lei nº 12.965, de 23 de abril de 2014.** Marco Civil da Internet. Disponível em https://www.planalto.gov.br/ccivil_03/_ato2011-2014/2014/lei/l12965.htm.

BRASIL. **Lei nº 13.005, de 25 de junho de 2014.** Plano Nacional de Educação. Disponível em: http://www.planalto.gov.br/ccivil_03/_ato2011-2014/2014/lei/l13005.htm

BRASIL. **Lei nº 13.709, de 14 de agosto de 2018.** Lei Geral de Proteção de Dados Pessoais. Disponível em https://www.planalto.gov.br/ccivil_03/_ato2015-2018/2018/lei/l13709.htm.

BRASIL. **Decreto nº 10.046, de 9 de outubro de 2019.** Governança de dados entre entes públicos e comitê central de governança de dados. Disponível em https://www.planalto.gov.br/ccivil_03/_Ato2019-2022/2019/Decreto/D10046.htm.

BRASIL. **Decreto nº 11.856, de 26 de dezembro de 2023.** Política e comitê nacional de cibersegurança. Disponível em <https://www.in.gov.br/en/web/dou/-/decreto-n-11.856-de-26-de-dezembro-de-2023-533845289>.

CÂMARA DOS DEPUTADOS. **A Voz do Brasil: O mais tradicional noticiário de rádio do país**. 2006. Disponível em <https://www.camara.leg.br/radio/programas/270473-a-voz-do-brasil-o-mais-tradicional-noticiario-de-radio-do-pais-03-53/>. Acesso em 16 abril 2024.

CAPES. **Portaria nº 60, de 20 de março de 2019**. Mestrados e doutorados profissionais. Fundação Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, 2019. Disponível em: <http://cad.capes.gov.br/ato-administrativo-detallar?idAtoAdmElastic=884#anchor>

CAPES. **Documento de Área**. 2019. Disponível em: <https://www.gov.br/capes/pt-br/aceso-a-informacao/acoes-e-programas/avaliacao/sobre-a-avaliacao/areas-avaliacao/sobre-as-areas-de-avaliacao/colégio-de-ciencias-exatas-tecnologicas-e-multidisciplinar/multidisciplinar/ensino>. Acesso em 18 nov. 2023.

CASTRO, Cloves Alexandre de; Plácido, Reginaldo Leandro; SCHENKEL, Cladecir Alberto. **História Socioespacial Do Trabalho No Brasil, Educação Profissional E Tecnológica e a Questão Regional**. Revista labor, Fortaleza, V 1, N. 24 2020. Disponível em: https://moodle.ifsc.edu.br/pluginfile.php/234147/mod_resource/content/1/44200-Texto%20do%20artigo-218646-1-10-20201019.pdf. Acesso em 14 ago. 2023.

CGU. **Painel de Monitoramento de Dados Abertos do Governo Federal**. 2023. Controladoria Geral da União. Disponível em: <https://centralpaineis.cgu.gov.br/visualizar/dadosabertos>

CNE/CES. **Diretrizes para Políticas de Extensão da Educação Superior Brasileira**. 2018. Ministério da Educação. Conselho Nacional de Educação. Câmara de Educação Superior. Disponível em: https://www.in.gov.br/materia/-/asset_publisher/Kujrw0TZC2Mb/content/id/55877808

COELHO, Sandra Aparecida Pinheiro. **A Extensão no Ensino Médio Integrado: Desafios da sua Consolidação no Instituto Federal do Sudeste de Minas Gerais - Campus Rio Pomba**. Instituto Federal de Educação, Ciência e Tecnologia do Sudeste de Minas Gerais, 2022. Disponível

em:https://sucupira.capes.gov.br/sucupira/public/consultas/coleta/trabalhoConclusao/viewTrabalhoConclusao.jsf?popup=true&id_trabalho=11866297

DAMABOK. **Manifesto sobre governança de dados em tempos de IA**. 2025. Data Management Body of Knowledge. Disponível em: <https://www.damabrasil.net/manifesto>. Acesso em: 27 fev. 2025.

DIACRÔNICO | HISTÓRIA, SOCIOLOGIA E FILOSOFIA. **Regime de Acumulação Flexível | 'Condição Pós-Moderna' de David Harvey**. YouTube, 16 ago. 2022. Disponível em: <https://www.youtube.com/watch?v=ErcgcenUW-o>. Acesso em: 23 set. 2023.

DIAS, Cláudia Augusto. **Grupo focal: técnica de coleta de dados em pesquisas qualitativas**. Informação & Sociedade, v. 10, n. 2, 2000. Disponível em: https://d1wqtxts1xzle7.cloudfront.net/31815169/330-269-1-PB_%281%29.pdf20130901-24809-1qssaph-libre-libre.pdf?1378030762=&response-content-disposition=inline%3B+filename%3D330_269_1_PB_1.pdf&Expires=1720651274&Signature=f7cP7h2inhnNFhbiINijqtKKu72PP5aFzHpDNPa79ONdtM2Y7q8Esk209KLKvwgPlkfBx4c5-QvXqKi3SeDHka4W4vqISelwSFpJX-PKtBqYHedx9zDnA0n4Uqb5r8tbl8LLQCZ2Mw6rNwXYr4wzrTkXq50IS6ia7leOuzcqyImK02k9N5Y-Pj2AYK5Kfkfrxytz4hDJZYPve5qeGBcli9WTg8f61L5GS4ahXMMY8xhHd1ZlcW-D9sveABbi7b-OM1Z48JbcsFoQm8MK8yE1uW0gH1kkhbJi~iwnE~HMCORx3oqaalPvuYJquSMI2HOYXEPNtMkhaq-MwtnJYeO8A__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA. Acesso em: 22 set. 2024.

ESTEVIÃO, Priscila; Coelho, France Maria Gontijo; Sousa, Diego Neves de. **Sedução, persuasão e manipulação no contexto da comunicação rural**. Revista de C. Humanas, Viçosa, v. 13, n. 1, p. 153-167, 2023. Disponível em: <https://periodicos.ufv.br/RCH/article/view/3925/Sedu%C3%A7%C3%A3o,%20Persuas%C3%A3o%20e%20Manipula%C3%A7%C3%A3o%20no%20Contexto%20da%20Comunica%C3%A7%C3%A3o%20Rura>. Acesso em: 11 mar. 2024.

FILHO, Jorge Martins. **Entrevista com Zygmunt Bauman**. YouTube, 2 jun. 2020. Disponível em: <https://www.youtube.com/watch?v=u77PSYsCGiM>. Acesso em: 23 set. 2023.

FILHO, Flavi Ferreira Lisboa. **Extensão Universitária: Gestão, Comunicação e Desenvolvimento Regional**. FACOS-UFSM, Santa Maria, 2022.

FORPROEXT. **Indissociabilidade entre ensino-pesquisa-extensão e flexibilização curricular: uma visão da extensão**. Porto Alegre: UFRGS; Brasília: MEC/SESu, 2006. Disponível em:
http://www.uemg.br/downloads/indissociabilidade_ensino_pesquisa_extensao.pdf. Acesso em 5 de abril de 2024.

FORPROEXT. **Política nacional de extensão universitária**. Manaus, 2012. Disponível em
<https://proex.ufsc.br/files/2016/04/Pol%C3%ADtica-Nacional-de-Extens%C3%A3o-Universit%C3%A1ria-e-book.pdf>. Acesso em 27 de junho de 2024.

FORPROEXT. **O fórum de pró-reitores de extensão e sua contribuição no debate sobre a extensão universitária**. Revista Educação e Políticas em Debate – v. 4, n.2 – ago./dez. 2015. Disponível em:
<http://www.seer.ufu.br/index.php/revistaeducaopoliticas/article/viewFile/34562/18318>. Acesso em 5 de abril de 2024.

FORPROEXT. **Indicadores Brasileiros de Extensão Universitária (IBEU)**. Campina Grande, 2017. Disponível em
https://www.ufmg.br/proex/renex/images/documentos/Relat%C3%B3rio_de_Pesquisa_Forproex_EBOOK.pdf. Acesso em 26 de junho de 2024.

FERREIRA, Rosangela. **Curricularização da Extensão nos Institutos Federais de Educação, Ciência e Tecnologia: Contexto e Possibilidades**. Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul, 2022. Disponível em:
https://sucupira.capes.gov.br/sucupira/public/consultas/coleta/trabalhoConclusao/viewTrabalhoConclusao.jsf?popup=true&id_trabalho=11693810.

FRUTUOSO, Tomé de Pádua. **O processo de curricularização da extensão nos cursos de graduação do Instituto Federal de Santa Catarina — IFSC**. Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina, 2020. Disponível em:
https://sucupira.capes.gov.br/sucupira/coleta_online/manutencaoTrabalhoConclusao/viewTrabalho.jsf?popup=true&id_trabalho_conclusao=9375771.

G1. **Prisioneiros chineses fazem trabalho escravo em jogo on-line.** 2011.

Disponível em

<https://g1.globo.com/tecnologia/noticia/2011/05/prisioneiros-chineses-fazem-trabalho-escravo-em-jogo-line.html>. Acesso em 12 abril 2024.

GOV. **Fórum debate políticas de governança de dados no governo federal.**

2023. Disponível em

<https://www.gov.br/gestao/pt-br/assuntos/noticias/2023/marco/forum-debate-politicas-de-governanca-de-dados-no-governo-federal>. Acesso em 12 de abril de 2024.

GOV. **Gestão debate o uso estratégico de dados nas políticas públicas.** 2024.

Disponível em

<https://www.gov.br/gestao/pt-br/assuntos/noticias/2024/abril/gestao-debate-o-uso-estrategico-de-dados-nas-politicas-publicas>. Acesso em 12 de abril de 2024.

GRAMSCI, Antonio. **Os intelectuais e a organização da cultura.** Civilização Brasileira, Rio de Janeiro, 1968.

HARVEY, David. **Condição Pós-moderna: Uma Pesquisa sobre as Origens da Mudança Cultural.** 20ª ed. São Paulo: Edições Loyola, 2010.

IFSC. **Resolução Consup nº 61, de 12 de Dezembro de 2016.** 2016. Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina. Disponível em http://cs.ifsc.edu.br/portal/files/consup_resolucao61_2016_extensao.pdf.

IFSC. **Plano de Desenvolvimento Institucional 2020-2024.** 2020. Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina. Disponível em: <https://www.ifsc.edu.br/pdi-2020-2024>.

IFSC. **Relatório de Auditoria Interna nº 03/2021 - Gestão da Extensão.** Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina. 2021. Disponível em: <https://www.ifsc.edu.br/relatorios-de-auditoria-interna-2021>.

IFSC. **Plano de Dados Abertos 2021-2023.** 2021. Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina. Disponível em: <https://www.ifsc.edu.br/12-dados-abertos>.

IFSC. **Relatório de Gestão 2022**. Instituto Federal de Educação ,Ciência e Tecnologia de Santa Catarina. 2023. Disponível em:
<https://www.ifsc.edu.br/relatorio-de-gestao>.

IMPrensa Nacional. **Diário Oficial da União completa 152 anos**. 2018.
Disponível em
<https://www.gov.br/impresnanacional/pt-br/assuntos/noticias/diario-oficial-da-uniao-completa-152-anos>. Acesso em 16 abril 2024.

INMON, William H. ***Building the Data Warehouse***. 4. ed. Indianapolis: Wiley, 2005.

KIMBALL, Ralph; ROSS, Margy. ***The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling***. 3. ed. Indianapolis: Wiley, 2013.

LACERDA, Daniel Pacheco; DRESCH, Aline; PROENÇA, Adriano; JÚNIOR, José Antonio Valle Antunes. **Design Science Research: método de pesquisa para engenharia de produção**. Scielo, Rio de Janeiro, 2013. Disponível em:
<https://www.scielo.br/j/gp/a/3CZmL4JJxLmxCv6b3pnQ8pq>.

LIPPOLD, W., & Faustino, D.. **Colonialismo digital, racismo e acumulação primitiva de dados**. Germinal: Marxismo E educação Em Debate, 14(2), 56–78. 2022. Disponível em
<https://periodicos.ufba.br/index.php/revistagerminal/article/view/49760>. Acesso em 9 abril 2024.

MAINARDES, Jefferson. **Abordagem do ciclo de políticas: uma contribuição para a análise de políticas educacionais**. Educ. Soc., vol. 27, n. 94, p. 47-69, Campinas, Unicamp, 2006. Disponível em:
https://ava.cefor.ifes.edu.br/pluginfile.php/3228625/mod_resource/content/4/MAINARDES%20-%20Ciclo%20de%20pol%C3%ADticas%20de%20Ball%20e%20Bowe.pdf. Acesso em: 03 out. 2023.

MAINARDES, Jefferson. **Reflexões sobre o objeto de estudo da política educacional**. Laplage, vol. 4, n. 1, p. 186-201, Ponta Grossa, UEPG, 2018.
Disponível em:
https://ava.cefor.ifes.edu.br/pluginfile.php/3167623/mod_resource/content/5/Reflex%20sobre%20o%20objeto%20de%20estudo%20da%20pol%C3%ADtica%20educacional.pdf

C3%B5es%20sobre%20o%20objeto%20de%20estudo%20da%20pol%C3%ADtica%20educacional.pdf. Acesso em: 03 out. 2023.

MARKOFF, John. **Democracia: transformações passadas, desafios presentes e perspectivas futuras**. Sociologias, Porto Alegre, ano 15, n. 32, p.18-50, 2013.

Disponível em:

https://moodle.ifsc.edu.br/pluginfile.php/223294/mod_resource/content/1/baixados%20%281%29.pdf. Acesso em 20 nov. 2023.

MALDANER, Jair José. **O papel da formação docente na efetividade das políticas públicas de EPT no Brasil - período 2003-2015**: implicações políticas e pedagógicas na atuação de professores. 2016. 207 f., il. Tese (Doutorado em Educação)—Universidade de Brasília, Brasília, 2016. Disponível em:

https://ava.cefor.ifes.edu.br/pluginfile.php/3167625/mod_resource/content/3/CAP.%202%20Tese%20MALDANER.pdf. Acesso em: 14 nov. 2023.

MARQUES, Dorli João Carlos. **Trabalhos acadêmicos: normas e fundamentos**. Manaus, AM: 2ª ed, Faculdade Salesiana Dom Bosco e Universidade Federal do Amazonas, 2009.

MEC. **Resolução Nº 7, de 18 de dezembro de 2018**. Diretrizes para a extensão na educação superior brasileira. Ministério da Educação. 2018. Disponível em https://normativasconselhos.mec.gov.br/normativa/pdf/CNE_RES_CNECESN72018.pdf

MEC. **Portaria Nº 983, de 18 de novembro de 2020**. Complementa a regulamentação das atividades docentes na rede federal de educação profissional, científica e tecnológica. Ministério da Educação. 2020. Disponível em <https://www.in.gov.br/en/web/dou/-/portaria-n-983-de-18-de-novembro-de-2020-289277573>.

MEC. **Programa EduCA+: Fomento a projetos de Iniciação Tecnológica**.

Ministério da Educação. 2022. Disponível

em:<https://www.gov.br/mec/pt-br/aceso-a-informacao/institucional/secretarias/secretaria-de-educacao-profissional/educacao-no-mundo-4-0-edmu4.0>.

MEC. **Portaria Nº 299, de 18 de novembro de 2022.** Indicadores de pesquisa e extensão na educação profissional, científica e tecnológica. Ministério da Educação. 2022. Disponível em

<https://www.in.gov.br/en/web/dou/-/portaria-n-299-de-6-de-maio-de-2022-399680297>

MEC. **Manual dos indicadores de extensão e pesquisa e inovação das instituições da rede federal de educação profissional, científica e tecnológica.**

Ministério da Educação. 2023. Disponível em

<https://monitorsetec.gitlab.io/docs/assets/files/ManualIndicadoresPortaria2992022-567e46e1ccfd907cd908c39fa61fb85.pdf>

MEC. **Relatório de Gestão de Contas da Secretaria Executiva.** Ministério da Educação. 2023. Disponível em:

<https://www.gov.br/mec/pt-br/processo-de-contas-anuais/processo-de-contas-anuais-secretaria-executiva-se>

MEC. **Dados Abertos.** Ministério da Educação. 2023. Disponível em:

<https://www.gov.br/mec/pt-br/acesso-a-informacao/dados-abertos>

MEC. **Consulta Pública ao Plano de Dados Abertos do MEC.** Ministério da Educação. 2023. Disponível em:

<https://www.gov.br/mec/pt-br/assuntos-1/paineis-de-monitoramento-e-indicadores/consulta-publica-pda>

MEC. **Plano Estratégico 2020-2023.** Ministério da Educação. 2020. Disponível em:

<http://portal.mec.gov.br/institucional/plano-estrategico>

MEC. **Competências.** Ministério da Educação. 2023. Disponível em:

<https://www.gov.br/mec/pt-br/acesso-a-informacao/institucional/competencias>

MEC. **Gestão de Resultados e Painel de Fomento.** Ministério da Educação. 2023. Disponível em:

<https://www.gov.br/mec/pt-br/assuntos/eventos/educacaoprofissionaltecnologica/documentos/gestaoeresultados>.

MGI. **Centro de Excelência em Privacidade e Segurança da Informação.** 2024.

Disponível em

<https://www.gov.br/gestao/pt-br/assuntos/noticias/2024/abril/gestao-lanca-centro-de-e-xcelencia-em-privacidade-e-seguranca-da-informacao-ceps-gov-br>.

NASCIMENTO, Leila Celin; AZEVEDO, Elizabeth Premoli; ALLEDI, Carla Therezinha Dalvi Borjaille; BAPTISTA, Gercyr Junior. **A Formação Humana e Cidadã em Atividades de Extensão para Cursos Técnicos**. Ensino, Pesquisa e Extensão no Brasil: Uma Abordagem Pluralista, volume III, 3ª edição, páginas 38-47. Editora Conhecimento Livre, 2020. Disponível em: <https://conhecimentolivre.org/wp-content/uploads/2020/08/L.28.3-2020.pdf>

NASCIMENTO, José Mateus do. **Ensino profissional brasileiro no século XIX: Ações assistencialistas e de reeducação pela aprendizagem de ofícios**. Hist. R., Goiânia, v. 25, n. 2, p. 76-98, mai./ago. 2020. Disponível em: https://moodle.ifsc.edu.br/pluginfile.php/222911/mod_resource/content/5/Ensino%20profissional%20brasileiro%20no%20s%C3%A9culo%20XIX%20a%C3%A7%C3%B5es%20assistencialistas%20e%20de%20reeduca%C3%A7%C3%A3o%20pela%20aprendizagem%20de%20of%C3%ADcios.pdf. Acesso em 31 jul. 2023.

NAVATHE, Shamkant B; ELMASRI, Ramez E. **Sistemas de banco de dados**. São Paulo, SP. 6ª ed. Pearson Education, 2005.

NETO, Antônio Cabral. **Democracia: velhas e novas controvérsias**. Estudos de Psicologia 1997, p.287-312. Disponível em: https://moodle.ifsc.edu.br/pluginfile.php/223287/mod_resource/content/1/baixados.pdf. Acesso em 06 nov. 2023.

NEVES, Lúcia M. W.; PRONKO, Marcela. A. **O mercado do conhecimento e o conhecimento do mercado: da formação para o trabalho complexo no Brasil contemporâneo**. Rio de Janeiro: EPSJV/Fiocruz, 2008. Disponível em : <https://www.epsjv.fiocruz.br/sites/default/files/l40.pdf>. Acesso em: 23 jun. 2023.

NIELSEN, Jakob. **Usability Engineering**. San Francisco: Morgan Kaufmann, 1993.

OTANI, Nilo; Juliani, Paulesky Douglas. **Metodologia de Pesquisa**. 2.ed. ampl. e rev. 2023. Disponível em: <https://moodle.ifsc.edu.br/mod/book/view.php?id=759217>. Acesso em: 25 jun. 2023.

PAWSON, R.; Tilley, N. **Realistic evaluation**. Londres: Sage, 1997.

PERES, Aline Neves. **Extensão na Educação Profissional e Técnica de Nível Médio no IFAM: Formação Humana Integral e Reflexo Social**. Instituto Federal de Educação, Ciência e Tecnologia do Amazonas, 2020. Disponível em: https://sucupira.capes.gov.br/sucupira/coleita_online/manutencaoTrabalhoConclusao/viewTrabalho.jsf?popup=true&id_trabalho_conclusao=9386926

PESTANA, Marlon Borges. **Métodos e meios de comunicação em Extensão Rural**. YouTube, 06 out. 2020. Disponível em <https://www.youtube.com/watch?v=wPeZuJ6EARk>. Acesso em 15 abril 2024.

PINTO, Álvaro Vieira. **O conceito de tecnologia v2**. Rio de Janeiro, RJ: Contraponto Editora Ltda, 2005.

PNP. **Plataforma Nilo Peçanha**. Publicização de dados oficiais da rede federal EPCT. 2018. Ministério da Educação. Disponível em: <http://portal.mec.gov.br/plataforma-nilo-pecanha>.

PNP. **Indicadores de Gestão na Educação**. Plataforma Nilo Peçanha. 2022. Disponível em: <https://app.powerbi.com/view?r=eyJrljoiZDhkNGNiYzgtMjQ0My00OGVILWJjNzYtZWQwYjI2OThhYWM1IiwidCI6IjllNjgyMzU5LWQxMjgtNGVkYi1iYjU4LTgyYjJhMTUzNDBmZiJ9>

PRESSMAN, Roger S. **Engenharia de Software: Uma abordagem profissional**. Porto Alegre, RS: 7ª ed, Armed Editora S.A. e AMGH Editora, 2011.

RAMOS, Andressa Freire. **A Indissociabilidade entre Ensino, Pesquisa e Extensão na EPT: Uma Proposta para o Planejamento Integrador no IFES - Campus Colatina**. Instituto Federal de Educação, Ciência e Tecnologia do Espírito Santo, 2020. Disponível em: <http://biblioteca.ifes.edu.br:8080/pergamumweb/vinculos/00001b/00001b6b.pdf>.

RAMOS, Marcos Gonçalves. **Modelos de comunicação e divulgação científicas - uma revisão de perspectivas**. Brasília, IBICT, v.23, n.3, p. 340-348, 1994.

Disponível em: <https://revista.ibict.br/ciinf/article/download/532/532/533>. Acesso em: 14 dez. 2023.

RAMOS, Marise Nogueira. **História e política da educação profissional**. 1ª ed. Formação Pedagógica. Volume V. Curitiba: Instituto Federal do Paraná, 2014.

Disponível em:

<https://ifpr.edu.br/curitiba/wp-content/uploads/sites/11/2016/05/Historia-e-politica-da-educacao-profissional.pdf>.

RENEX. **Cartas e Memória - Encontros Nacionais do FORPREX**. 2023. Rede Nacional de Extensão RENEX. Disponível em <https://www.ufmg.br/proex/renex/index.php/documentos/cartas-e-memoria>.

RODRIGUES, Valéria M. **O Fórum de pró-reitores de extensão e sua contribuição no debate sobre a extensão universitária**. Revista Educação e Políticas em Debate, [S. l.], v. 4, n. 2, 2016. DOI:

10.14393/REPOD-v4n2a2015-34562. Disponível em:

<https://seer.ufu.br/index.php/revistaeducaopoliticas/article/view/34562>. Acesso em: 29 jun. 2024.

SANTOS, Andre Luis da Silva. **A Importância da Extensão Tecnológica Desenvolvida pelas ITCPS para a Travessia Rumo à Educação Politécnica**.

Instituto Federal de Educação, Ciência e Tecnologia da Bahia, 2020. Disponível em: <https://profept.ifes.edu.br/consulta-de-egressos-dissertacoes-produtos-educacionais>

SAVIANI, Dermeval. **Trabalho e educação: fundamentos ontológicos e históricos**. Revista Brasileira de Educação. Volume 12. nº 34. Campinas:

Universidade Estadual de Campinas, Faculdade de Educação, 2007. Disponível em: <https://www.scielo.br/j/rbedu/a/wBnPGNkvstzMTLYkmXdrkWP/?format=pdf&lang=pt>

SAVIANI, Dermeval. **Histórias das ideias pedagógicas no Brasil**. 4. ed. Campinas: Autores Associados.

SAYER, A. **Method in Social Science: a realist approach**. Londres: Hutchinson Publishing Group, 1984.

SCHENKEL, C.A. **Gestão ambiental: perfil profissional e formação em cursos superiores de tecnologia e de bacharelado**. 2012. 348 f. Tese (Doutorado em Ciências Humanas) - Universidade Federal de Uberlândia, Uberlândia, 2012.

SCHMIDT, N. S., & da Silva, C. L. (2018). **Observatório como instrumento de prospectiva estratégica para as Instituições de Ciência e Tecnologia (ICTs)**. Interações (Campo Grande), 19(2), 387–400. Disponível em <https://interacoesucdb.emnuvens.com.br/interacoes/article/view/1689>

SENADO. **Projeto de Lei 2.338/2023. Marco Legal da Inteligência Artificial**. 2023. Disponível em <https://www12.senado.leg.br/radio/1/projeto-em-destaque/2023/06/13/marco-legal-da-inteligencia-artificial>.

SETEC/MEC. **Portaria nº 299, de 6 de maio de 2022**. Indicadores de Pesquisa e Extensão a serem utilizados pelas instituições da rede federal EPCT. Disponível em: <https://www.in.gov.br/en/web/dou/-/portaria-n-299-de-6-de-maio-de-2022-399680297>

SMITH, Telma Carolina. **Na Roda da Extensão: Proposta para Ações Extensionistas nos Cursos Técnicos na Modalidade A Distância**. Instituto Federal de Educação, Ciência e Tecnologia do Espírito Santo, 2021. Disponível em: https://sucupira.capes.gov.br/sucupira/public/consultas/coleta/trabalhoConclusao/viewTrabalhoConclusao.jsf?popup=true&id_trabalho=11143280.

SOUZA, Francisco das Chagas Silva; SILVA, Edilana Carlos da. **Políticas educacionais e verticalização da Educação Profissional e Tecnológica no Brasil (séculos XX e XXI)**. Essentia Editora IFFluminense, 2022, Dossiê Temático, v. 24, n. 22022, p. 236-266. Disponível em: https://moodle.ifsc.edu.br/pluginfile.php/222963/mod_resource/content/1/02.pdf. Acesso em 14 ago. 2023.

SOUZA, Lucas de. **O uso das tecnologias digitais nas práticas pedagógicas como competência para o mundo do trabalho: um estudo no Instituto Federal De Educação, Ciência e Tecnologia de Santa Catarina (IFSC) - Campus Florianópolis-Continente**. Instituto Federal de Educação, Ciência e Tecnologia de

Santa Catarina, 2022. Disponível

em:https://sucupira.capes.gov.br/sucupira/public/consultas/coleta/trabalhoConclusao/viewTrabalhoConclusao.jsf?popup=true&id_trabalho=11850407

TCU. **Acórdão 461/2022**. Indicadores de Gestão e Desempenho da Rede Federal de Educação. Tribunal de Contas da União, 2022. Disponível em

https://pesquisa.apps.tcu.gov.br/documento/acordao-completo/*/NUMACORDAO%253A461%2520ANOACORDAO%253A2022/DTRELEVANCIA%2520desc%252C%2520NUMACORDAOINT%2520desc/0/%2520. Acesso em 11 abril 2024.

TRAD, Leny A. Bomfim. **Grupos focais: conceitos, procedimentos e reflexões baseadas em experiências com o uso da técnica em pesquisas de saúde**.

Scielo, Rio de Janeiro, 2009. Disponível em:

<https://www.scielo.br/j/physis/a/gGZ7wXtGXqDHNCHv7gm3srw>.

TRIPP, David. **Pesquisa-ação: uma introdução metodológica**. Scielo, Rio de Janeiro, 2005. Disponível em:

<https://www.scielo.br/j/ep/a/3DkbXnqBQyyq5bV4TCL9NSH>.

UNIVESP. **D-06 - D. João Vi: Ensino Superior e Profissional (2/2)**. YouTube, 8 jul. 2010. Disponível em: <https://www.youtube.com/watch?v=GeWv1aDqebk>. Acesso em 31 jul. 2023.

WAZLAWICK, Raul Sidnei. **Metodologia de pesquisa para ciência da computação**. Rio de Janeiro, RJ: 2ª ed, Elsevier Editora Ltda, 2014.

APÊNDICE A – CARTA DE ANUÊNCIA PARA REALIZAÇÃO DA PESQUISA

DECLARAÇÃO

Declaro para os devidos fins e efeitos legais que tenho conhecimento da pesquisa intitulada OBSERVATÓRIO DA EXTENSÃO NO INSTITUTO FEDERAL DE SANTA CATARINA, sob a responsabilidade de PAULO VICTOR REBOUÇAS SOARES. Diante da análise da proposta de pesquisa, realizada pela Pró-Reitoria de Pesquisa, Pós-Graduação e Inovação, autorizo a sua execução. Esta autorização não exime, contudo, a responsabilidade do pesquisador em atender à Resolução CNS 466/12, de 12/12/2012, e à Resolução CNS 510/16, de 07/04/2016 e complementares.

Nome do responsável pelo local de aplicação da pesquisa.

Florianópolis, data.

APÊNDICE B – TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO (TCLE) INDIVIDUAL

Elaborado por:

Vanessa L Tuono Jardim e Luciana Senter

Adaptado por:

Comissão Permanente de Gestão de Dados Institucionais Portaria do(a) Reitor(a) N° 2756 de 14 de setembro de 2021)

Versão n.3 Maio/2022

Você está sendo convidado(a) para participar como voluntário em uma pesquisa. Leia os termos abaixo e, caso aceite fazer parte do estudo, assine este termo.

Para o caso de documento em cópia física: Rubrique todas as páginas e assine no final deste documento, que está em duas vias. Uma delas é sua e a outra é do pesquisador.

Para o caso de documento virtual: Será possível inserir ao término do documento a assinatura com certificado digital, sendo que este documento será encaminhado para o seu e-mail pelo pesquisador.

Título da pesquisa: OBSERVATÓRIO DA EXTENSÃO NO INSTITUTO FEDERAL DE SANTA CATARINA

Pesquisador responsável (Operador de dados): Paulo Victor Rebouças Soares

Telefone para contato: (48) 99807-8599

O Comitê de Ética em Pesquisa envolvendo seres humanos (CEPSH) é um colegiado interdisciplinar e independente, de relevância pública, de caráter consultivo, deliberativo e educativo, criado para defender os interesses dos participantes da pesquisa em sua integridade e dignidade e para contribuir no desenvolvimento da pesquisa dentro de padrões éticos. O CEP/IFSC está localizado dentro da própria Instituição, à Rua 14 de julho n°150, 1º andar, sala 33B, Florianópolis-SC, CEP 88075-010. Horário de funcionamento definido de segunda-feira a sexta-feira das 8h às 12h para contato dos pesquisadores e participantes das pesquisas. Telefone para contato (48) 3877-9054 e e-mail cepsh@ifsc.edu.br.

O objetivo desta pesquisa é apresentar o panorama do eixo extensão no Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina e dar visibilidade à sociedade da materialização das ações de extensão promovidas pelo instituto.

A sua participação na pesquisa consiste em contribuir para a elucidação de boas práticas e execução de atividades para a realização de ações de extensão e da construção de dados para o fortalecimento do eixo, sem qualquer prejuízo ou constrangimento para o(a) pesquisado(a). Os procedimentos aplicados por esta pesquisa eventualmente podem oferecer riscos (*Ressalta-se que toda pesquisa tem riscos, mesmo que sejam mínimos. Podendo esta pesquisa impactar na sua saúde mental ou na integridade moral*). Com a finalidade de uma posterior análise pragmática para a pesquisa, ressalta-se que a entrevista será gravada mediante seu consentimento de uso de imagem e som ao firmar este documento, ademais a mídia gravada será guardada em diretório institucional privado protegido por controle de acesso, podendo ser acessada pelo(a) pesquisado(a) mediante solicitação. Caso queira, informe ao pesquisador qualquer condição de saúde ou desconforto que possa interferir em sua participação na pesquisa. Caso ocorram efeitos indesejáveis ao(a) pesquisado(a), encaminharemos para o serviço integrado de atenção à saúde do servidor, sendo garantida assistência imediata, sem ônus de qualquer espécie a sua pessoa com todos os cuidados necessários a sua participação de acordo com seus direitos individuais e respeito ao seu bem-estar físico e psicológico. Os benefícios esperados pela sua participação na pesquisa são a identificação de boas práticas e informações relevantes para o fortalecimento do eixo de extensão no IFSC em acordo com as normativas estabelecidas e na construção de futuras normativas por meio da publicização de dados para a melhor visibilidade das atividades pela sociedade e para aprimoramento da tomada de decisão pela gestão extensionista. A atividade não prevê nenhum tipo de ressarcimento ou compensação material para os participantes.

É garantida indenização diante de eventuais danos decorrentes da pesquisa a sua pessoa.

CONSENTIMENTO DA PESSOA (TITULAR) COMO PARTICIPANTE DE PESQUISA

Eu, _____, abaixo assinado, concordo em participar do estudo. Fui devidamente informado e esclarecido pelo pesquisador sobre a pesquisa e os procedimentos nela envolvidos, bem como os riscos e benefícios decorrentes da minha participação. Foi-me garantido que posso retirar meu consentimento a qualquer momento e ter acesso ao registro do consentimento sempre que solicitado.

Local: _____

Data ____/____/____.

Assinatura do participante da pesquisa:

Assinatura do pesquisador responsável:

APÊNDICE C – TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO (TCLE) DE GRUPO FOCAL

Elaborado por:

Vanessa L Tuono Jardim e Luciana Senter

Adaptado por:

Comissão Permanente de Gestão de Dados Institucionais Portaria do(a) Reitor(a) N° 2756 de 14 de setembro de 2021)

Versão n.3 Maio/2022

Você está sendo convidado(a) para participar como voluntário de um grupo focal de uma pesquisa. Leia os termos abaixo e, caso aceite fazer parte do estudo, assine este termo.

Para o caso de documento em cópia física: Rubrique todas as páginas e assine no final deste documento, que está em duas vias. Uma delas é sua e a outra é do pesquisador.

Para o caso de documento virtual: Será possível inserir ao término do documento a assinatura com certificado digital, sendo que este documento será encaminhado para o seu e-mail pelo pesquisador.

Título da pesquisa: OBSERVATÓRIO DA EXTENSÃO NO INSTITUTO FEDERAL DE SANTA CATARINA

Pesquisador responsável (Operador de dados): Paulo Victor Rebouças Soares

Telefone para contato: (48) 99807-8599

O Comitê de Ética em Pesquisa envolvendo seres humanos (CEPSH) é um colegiado interdisciplinar e independente, de relevância pública, de caráter consultivo, deliberativo e educativo, criado para defender os interesses dos participantes da pesquisa em sua integridade e dignidade e para contribuir no desenvolvimento da pesquisa dentro de padrões éticos. O CEP/IFSC está localizado dentro da própria Instituição, à Rua 14 de julho n°150, 1º andar, sala 33B, Florianópolis-SC, CEP 88075-010. Horário de funcionamento definido de segunda-feira a sexta-feira das 8h às 12h para contato dos pesquisadores e participantes das pesquisas. Telefone para contato (48) 3877-9054 e e-mail cepsch@ifsc.edu.br.

O objetivo desta pesquisa é apresentar o panorama do eixo extensão no Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina e dar visibilidade à sociedade da materialização das ações de extensão promovidas pelo instituto.

A participação do grupo focal na pesquisa consiste em contribuir para a validação da aplicabilidade do produto educacional sob os aspectos extensionistas observados na pesquisa, sem qualquer prejuízo ou constrangimento para os(as) voluntários(as). Os procedimentos aplicados por esta pesquisa eventualmente podem oferecer riscos *(Ressalta-se que toda pesquisa tem riscos, mesmo que sejam mínimos. Podendo esta pesquisa impactar na sua saúde mental ou na integridade moral)*. Com a finalidade de uma posterior análise pragmática para a pesquisa, ressalta-se que a entrevista será gravada mediante consentimento de uso de imagem e som ao firmar este documento, ademais a mídia gravada será guardada em diretório institucional privado protegido por controle de acesso, podendo ser acessada pelos(as) voluntários(as) mediante solicitação. Caso queiram, informem ao pesquisador qualquer condição de saúde ou desconforto que possa interferir em sua participação no grupo. Caso ocorram efeitos indesejáveis aos(as) voluntários(as), encaminharemos para o serviço integrado de atenção à saúde do servidor, sendo garantida assistência imediata, sem ônus de qualquer espécie a sua pessoa com todos os cuidados necessários a sua participação de acordo com seus direitos individuais e respeito ao seu bem-estar físico e psicológico. Os benefícios esperados pela sua participação na pesquisa são a identificação de boas práticas e informações relevantes para o fortalecimento do eixo de extensão no IFSC em acordo com as normativas estabelecidas e na construção de futuras normativas por meio da publicização de dados para a melhor visibilidade das atividades pela sociedade e para aprimoramento da tomada de decisão pela gestão extensionista. A atividade não prevê nenhum tipo de ressarcimento ou compensação material para os participantes.

É garantida indenização diante de eventuais danos decorrentes da pesquisa a sua pessoa.

CONSENTIMENTO DA(S) PESSOA(S) (TITULAR(ES)) COMO VOLUNTÁRIO DE

PESQUISA

O abaixo assinado, remete à concordância em participar do grupo focal do estudo. Fomos devidamente informados e esclarecidos pelo pesquisador sobre a pesquisa e os procedimentos nela envolvidos, bem como os riscos e benefícios decorrentes da participação. Foi-nos garantido que podemos retirar o consentimento a qualquer momento e ter acesso ao registro do consentimento sempre que solicitado.

Local: _____

Data ____/____/____.

Assinatura dos voluntários da pesquisa:

Assinatura do pesquisador responsável:

APÊNDICE D – TERMO DE CONFIDENCIALIDADE E SIGILO

Eu, **Paulo Victor Rebouças Soares, brasileiro, casado, servidor público**, inscrito(a) no CPF/ MF sob o nº **929.219.062-87**, abaixo firmado, assumo o compromisso de manter confidencialidade e sigilo sobre todas as informações técnicas obtidas por meio do processo **23292.017923/2024-43/IFSC** e outras relacionadas ao projeto de pesquisa intitulado **observatório da extensão no instituto federal de Santa Catarina**, a que tiver acesso nas dependências e dados da **pró-reitoria de extensão e relações externas e da diretoria de tecnologia da informação e comunicação do instituto federal de educação, ciência e tecnologia de Santa Catarina**.

Por este termo de confidencialidade e sigilo comprometo-me a:

1. não utilizar as informações confidenciais a que tiver acesso, para gerar benefício próprio exclusivo e/ou unilateral, presente ou futuro, ou para o uso de terceiros;
2. orientar a equipe e os(as) envolvidos(as) quanto às boas práticas em sigilo e confidencialidade;
3. não efetuar nenhuma gravação ou cópia da documentação confidencial a que tiver acesso;
4. não me apropriar, copiar ou efetuar duplicação de material confidencial e/ou sigiloso que venha a ser disponível;
5. não repassar o conhecimento das informações confidenciais, responsabilizando-me por todas as pessoas que vierem a ter acesso às informações, por meu intermédio, e obrigando-me, assim, a ressarcir a ocorrência de qualquer dano e/ou prejuízo oriundo de uma eventual quebra de sigilo das informações fornecidas.

Neste Termo, as seguintes expressões serão assim definidas:

Informação Confidencial: significará toda informação revelada ou cedida pelo participante da pesquisa, a respeito da pesquisa, ou associada à avaliação de seus dados, sob a forma escrita, verbal ou por quaisquer outros meios. **Avaliação** significará todas e quaisquer discussões, conversações ou negociações entre, ou com as partes, de alguma forma relacionada ou associada com o desenvolvimento da pesquisa.

Informação Confidencial inclui, mas não se limita, aos dados pessoais, informação relativa à operações, processos, planos ou intenções, informações sobre produção, instalações, equipamentos, segredos de negócio, segredo de fábrica, dados, habilidades especializadas, projetos, métodos e metodologia, fluxogramas, especializações, componentes, fórmulas, produtos, amostras, diagramas, desenhos de esquema industrial, patentes, oportunidades de mercado e questões relativas a negócios.

Pelo não cumprimento do presente Termo de Confidencialidade e Sigilo, fica o abaixo assinado ciente de que sanções judiciais poderão advir.

Local, ____/____/____.

Ass. _____

Nome do Pesquisador(a) Responsável

APÊNDICE E – ROTEIRO DE ENTREVISTA COM GESTORES DA EXTENSÃO

Essa entrevista faz parte da pesquisa intitulada "observatório da extensão no IFSC" que tem como objetivo dar publicidade à dados por meio de indicadores apresentados em painéis *online* que caracterizem a extensão no IFSC para fortalecimento da imagem dela, aproximar a comunidade, e ajudar a gestão na tomada de decisões. Além da busca de indicadores em documentos normativos e diretivos, decretos e leis, é importante entender no que os câmpus estão trabalhando e quais indicadores são importantes para as suas atividades extensionistas.

Sua participação é voluntária, os dados serão tratados considerando o sigilo, a confidencialidade, o anonimato e privacidade do(a) entrevistado(a), conforme o termo de consentimento livre e esclarecido. Essa atividade não prevê nenhum tipo de ressarcimento ou compensação material. Caso você sinta algum desconforto que possa interferir em sua participação na pesquisa essa entrevista pode ser interrompida a qualquer momento, e se você sentir algum dano a sua saúde acionaremos o serviço integrado de atenção à saúde do servidor. Por fim, se não houver dúvidas, com seu consentimento iremos gravar essa conversa.

Perguntas:

1. Há quanto tempo você atua na Educação Profissional e Tecnológica (EPT)?
2. Quais cargos e funções você já ocupou e ocupa na EPT?
3. Como suas atividades ao longo desse tempo se relacionam com a extensão?
4. Como o(s) campus tem trabalhado as ações de extensão, você pode fazer um breve relato dessa atuação?
5. Como é a relação do(s) campus com o público externo nas ações de extensão, há envolvimento de parceiros e da comunidade?
6. Como é o envolvimento dos servidores do(s) campus com a extensão, há alguma manifestação que comumente trazem à gestão?
7. Como é o envolvimento dos discentes do(s) campus com a extensão, trazem alguma demanda ou dão retorno para a gestão?
8. Quais informações da extensão você considera importantes para a gestão?
9. Quais informações da extensão você considera importantes para a sociedade civil?

ANEXO A - LISTA DE SCRIPTS SQL PARA EXTRAÇÃO DOS INDICADORES DE EXTENSÃO DO SIGAA

-- DOCENTES EM ATIVIDADES DE EXTENSÃO

-- docentes em atividades de extensão com papel de voluntário por campus e ano

```
select ci.nome as campus, regexp_replace(endereco.cep, '^0-9', ', 'g') as cep,
proj.ano as ano_atividade, count(distinct pae.cpf) as
quantidade_docentes_voluntarios
from extensao.participante_acao_extensao pae
inner join extensao.tipo_participacao tp on tp.id_tipo_participacao =
pae.id_tipo_participacao
inner join extensao.atividade a on a.id_atividade = pae.id_acao_extensao
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.membro_projeto mp on mp.id_projeto = proj.id_projeto
inner join projetos.funcao_membro fm on fm.id_funcao_membro =
mp.id_funcao_membro
inner join projetos.categoria_membro cm on cm.id_categoria_membro =
mp.id_categoria_membro
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
inner join rh.servidor s on s.id_servidor = pae.id_servidor
inner join rh.tipo_categoria tc on tc.id_tipo_categoria = s.id_categoria
where tc.denominacao = 'Docente'
and pae.cpf != 0
and fm.descricao = 'VOLUNTARIO(A)'
or fm.descricao = 'voluntário'
and fm.extensao = true
group by ci.nome, proj.ano, endereco.cep
order by ci.nome, proj.ano asc;
```

-- docentes em atividades de extensão com papel de bolsista por campus e ano

-- não há registro de docente bolsista no SIGAA

```
select ci.nome as campus, regexp_replace(endereco.cep, '^0-9', ', 'g') as cep,
proj.ano as ano_atividade, count(distinct pae.cpf) as quantidade_docentes_bolsistas
from extensao.participante_acao_extensao pae
inner join extensao.tipo_participacao tp on tp.id_tipo_participacao =
pae.id_tipo_participacao
inner join extensao.atividade a on a.id_atividade = pae.id_acao_extensao
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.membro_projeto mp on mp.id_projeto = proj.id_projeto
```

```

inner join projetos.funcao_membro fm on fm.id_funcao_membro =
mp.id_funcao_membro
inner join projetos.categoria_membro cm on cm.id_categoria_membro =
mp.id_categoria_membro
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
inner join rh.servidor s on s.id_servidor = pae.id_servidor
inner join rh.tipo_categoria tc on tc.id_tipo_categoria = s.id_categoria
where tc.denominacao = 'Docente'
and false
group by ci.nome, proj.ano, endereco.cep
order by ci.nome, proj.ano asc;

```

```

-- docentes em atividades de extensão com papel de coordenador por campus e ano
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, count(distinct pae.cpf) as
quantidade_docentes_coordenadores
from extensao.participante_acao_extensao pae
inner join extensao.tipo_participacao tp on tp.id_tipo_participacao =
pae.id_tipo_participacao
inner join extensao.atividade a on a.id_atividade = pae.id_acao_extensao
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.membro_projeto mp on mp.id_projeto = proj.id_projeto
inner join projetos.funcao_membro fm on fm.id_funcao_membro =
mp.id_funcao_membro
inner join projetos.categoria_membro cm on cm.id_categoria_membro =
mp.id_categoria_membro
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
inner join rh.servidor s on s.id_servidor = pae.id_servidor
inner join rh.tipo_categoria tc on tc.id_tipo_categoria = s.id_categoria
where tc.denominacao = 'Docente'
and pae.cpf != 0
and fm.descricao = 'COORDENADOR(A)'
and fm.extensao = true
group by ci.nome, proj.ano, endereco.cep
order by ci.nome, proj.ano asc;

```

```

-- docentes em atividades de extensão com papel de orientador por campus e ano
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, count(distinct pae.cpf) as
quantidade_docentes_orientadores
from extensao.participante_acao_extensao pae
inner join extensao.tipo_participacao tp on tp.id_tipo_participacao =
pae.id_tipo_participacao

```

```

inner join extensao.atividade a on a.id_atividade = pae.id_acao_extensao
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
inner join rh.servidor s on s.id_servidor = pae.id_servidor
inner join rh.tipo_categoria tc on tc.id_tipo_categoria = s.id_categoria
where tc.denominacao = 'Docente'
and pae.cpf != 0
and tp.descricao = 'ORIENTADOR(A)'
group by ci.nome, proj.ano, endereco.cep
order by ci.nome, proj.ano asc;

```

```

-- quantitativo de docentes que submetem ações de extensão por campus e ano
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, count(distinct pae.cpf) as
quantidade_docentes_que_submetem_extensao
from extensao.participante_acao_extensao pae
inner join extensao.tipo_participacao tp on tp.id_tipo_participacao =
pae.id_tipo_participacao
inner join extensao.atividade a on a.id_atividade = pae.id_acao_extensao
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.membro_projeto mp on mp.id_projeto = proj.id_projeto
inner join projetos.funcao_membro fm on fm.id_funcao_membro =
mp.id_funcao_membro
inner join projetos.categoria_membro cm on cm.id_categoria_membro =
mp.id_categoria_membro
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
inner join rh.servidor s on s.id_servidor = pae.id_servidor
inner join rh.tipo_categoria tc on tc.id_tipo_categoria = s.id_categoria
where tc.denominacao = 'Docente'
and pae.cpf != 0
and fm.extensao = true
group by ci.nome, proj.ano, endereco.cep
order by ci.nome, proj.ano asc;

```

```

-- docentes envolvidos em atividades de extensão por campus e ano
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, count(distinct pae.cpf) as
quantidade_docentes_envolvidos_atividades
from extensao.participante_acao_extensao pae
inner join extensao.tipo_participacao tp on tp.id_tipo_participacao =
pae.id_tipo_participacao

```

```

inner join extensao.atividade a on a.id_atividade = pae.id_acao_extensao
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.membro_projeto mp on mp.id_projeto = proj.id_projeto
inner join projetos.funcao_membro fm on fm.id_funcao_membro =
mp.id_funcao_membro
inner join projetos.categoria_membro cm on cm.id_categoria_membro =
mp.id_categoria_membro
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
inner join rh.servidor s on s.id_servidor = pae.id_servidor
inner join rh.tipo_categoria tc on tc.id_tipo_categoria = s.id_categoria
where tc.denominacao = 'Docente'
and pae.cpf != 0
and fm.extensao = true
group by ci.nome, proj.ano, endereco.cep
order by ci.nome, proj.ano asc;

```

```

-- quantitativo total de servidores docentes por campus para percentual
-- não há registro histórico no SIGAA para agrupar com precisão por ano (está
considerando docentes temporários e visitantes)
select ci.nome as campus, regexp_replace(endereco.cep, '^[0-9]', ', 'g') as cep,
count(s.id_servidor) as quantidade_docentes
from rh.servidor s
inner join rh.tipo_categoria tc on tc.id_tipo_categoria = s.id_categoria
inner join rh.ativo a on a.id_ativo = s.id_ativo
inner join comum.unidade u on u.id_unidade = s.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where tc.denominacao = 'Docente' and s.id_ativo = 1
group by ci.nome, endereco.cep
order by ci.nome asc;

```

-- TAES EM ATIVIDADES DE EXTENSÃO

```
-- TAEs em atividades de extensão com papel de voluntário por campus e ano
select ci.nome as campus, regexp_replace(endereco.cep,'[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, count(distinct pae.cpf) as quantidade_taes_voluntario
from extensao.participante_acao_extensao pae
inner join extensao.tipo_participacao tp on tp.id_tipo_participacao =
pae.id_tipo_participacao
inner join extensao.atividade a on a.id_atividade = pae.id_acao_extensao
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.membro_projeto mp on mp.id_projeto = proj.id_projeto
inner join projetos.funcao_membro fm on fm.id_funcao_membro =
mp.id_funcao_membro
inner join projetos.categoria_membro cm on cm.id_categoria_membro =
mp.id_categoria_membro
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
inner join rh.servidor s on s.id_servidor = pae.id_servidor
inner join rh.tipo_categoria tc on tc.id_tipo_categoria = s.id_categoria
where tc.denominacao = 'Tecnico Administrativo'
and pae.cpf != 0
and fm.descricao = 'VOLUNTARIO(A)'
or fm.descricao = 'voluntário'
and fm.extensao = true
group by ci.nome, proj.ano, endereco.cep
order by ci.nome, proj.ano asc;
```

-- TAEs em atividades de extensão com papel de bolsista por campus e ano

-- não há registro de TAE bolsista no SIGAA

```
select ci.nome as campus, regexp_replace(endereco.cep,'[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, count(distinct pae.cpf) as quantidade_taes_bolsistas
from extensao.participante_acao_extensao pae
inner join extensao.tipo_participacao tp on tp.id_tipo_participacao =
pae.id_tipo_participacao
inner join extensao.atividade a on a.id_atividade = pae.id_acao_extensao
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.membro_projeto mp on mp.id_projeto = proj.id_projeto
inner join projetos.funcao_membro fm on fm.id_funcao_membro =
mp.id_funcao_membro
inner join projetos.categoria_membro cm on cm.id_categoria_membro =
mp.id_categoria_membro
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
```



```

inner join rh.servidor s on s.id_servidor = pae.id_servidor
inner join rh.tipo_categoria tc on tc.id_tipo_categoria = s.id_categoria
where tc.denominacao = 'Tecnico Administrativo'
and false
group by ci.nome, proj.ano, endereco.cep
order by ci.nome, proj.ano asc;

```

```

-- TAEs em atividades de extensão com papel de coordenador por campus e ano
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, count(distinct pae.cpf) as quantidade_taes_coordenador
from extensao.participante_acao_extensao pae
inner join extensao.tipo_participacao tp on tp.id_tipo_participacao =
pae.id_tipo_participacao
inner join extensao.atividade a on a.id_atividade = pae.id_acao_extensao
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.membro_projeto mp on mp.id_projeto = proj.id_projeto
inner join projetos.funcao_membro fm on fm.id_funcao_membro =
mp.id_funcao_membro
inner join projetos.categoria_membro cm on cm.id_categoria_membro =
mp.id_categoria_membro
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
inner join rh.servidor s on s.id_servidor = pae.id_servidor
inner join rh.tipo_categoria tc on tc.id_tipo_categoria = s.id_categoria
where tc.denominacao = 'Tecnico Administrativo'
and pae.cpf != 0
and fm.descricao = 'COORDENADOR(A)'
and fm.extensao = true
group by ci.nome, proj.ano, endereco.cep
order by ci.nome, proj.ano asc;

```

```

-- TAEs em atividades de extensão por campus e ano
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, count(distinct pae.cpf) as
quantidade_taes_envolvidos_atividades
from extensao.participante_acao_extensao pae
inner join extensao.tipo_participacao tp on tp.id_tipo_participacao =
pae.id_tipo_participacao
inner join extensao.atividade a on a.id_atividade = pae.id_acao_extensao
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.membro_projeto mp on mp.id_projeto = proj.id_projeto
inner join projetos.funcao_membro fm on fm.id_funcao_membro =
mp.id_funcao_membro

```



```

inner join projetos.categoria_membro cm on cm.id_categoria_membro =
mp.id_categoria_membro
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
inner join rh.servidor s on s.id_servidor = pae.id_servidor
inner join rh.tipo_categoria tc on tc.id_tipo_categoria = s.id_categoria
where tc.denominacao = 'Tecnico Administrativo'
and pae.cpf != 0
and fm.extensao = true
group by ci.nome, proj.ano, endereco.cep
order by ci.nome, proj.ano asc;

```

```

-- quantitativo total de servidores TAEs por campus para percentual
-- não há registro histórico no SIGAA para agrupar por ano
select ci.nome as campus, regexp_replace(endereco.cep, '^0-9', ", 'g') as cep,
count(s.id_servidor)
from rh.servidor s
inner join rh.tipo_categoria tc on tc.id_tipo_categoria = s.id_categoria
inner join rh.ativo a on a.id_ativo = s.id_ativo
inner join comum.unidade u on u.id_unidade = s.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where tc.denominacao = 'Tecnico Administrativo' and s.id_ativo = 1
group by ci.nome, endereco.cep
order by ci.nome asc;

```

-- DISCENTES EM ATIVIDADES DE EXTENSÃO

-- discentes em atividades de extensão com papel de voluntário por campus, ano e tipo de curso

```
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, tca.descricao as tipo_curso, count(distinct d.matricula) as
quantidade_discentes_voluntarios
from extensao.participante_acao_extensao pae
inner join public.discente d on d.id_discente = pae.id_discente
inner join public.curso c on c.id_curso = d.id_curso
inner join ensino.tipo_curso_academico tca on tca.id_tipo_curso_academico =
c.id_tipo_curso_academico
inner join extensao.tipo_participacao tp on tp.id_tipo_participacao =
pae.id_tipo_participacao
inner join extensao.atividade a on a.id_atividade = pae.id_acao_extensao
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.membro_projeto mp on mp.id_projeto = proj.id_projeto
inner join projetos.funcao_membro fm on fm.id_funcao_membro =
mp.id_funcao_membro
inner join projetos.categoria_membro cm on cm.id_categoria_membro =
mp.id_categoria_membro
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where cm.descricao = 'DISCENTE'
and pae.cpf != 0
and fm.descricao = 'VOLUNTARIO(A)'
or fm.descricao = 'voluntário'
and fm.extensao = true
group by ci.nome, proj.ano, endereco.cep, tca.descricao
order by ci.nome, proj.ano asc;
```

-- discentes em atividades de extensão com papel de bolsista por campus, ano e tipo de curso

```
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, tca.descricao as tipo_curso, count(distinct d.matricula) as
quantidade_discentes_bolsistas
from extensao.participante_acao_extensao pae
inner join public.discente d on d.id_discente = pae.id_discente
inner join public.curso c on c.id_curso = d.id_curso
inner join ensino.tipo_curso_academico tca on tca.id_tipo_curso_academico =
c.id_tipo_curso_academico
inner join extensao.tipo_participacao tp on tp.id_tipo_participacao =
pae.id_tipo_participacao
inner join extensao.atividade a on a.id_atividade = pae.id_acao_extensao
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
```

```

inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.membro_projeto mp on mp.id_projeto = proj.id_projeto
inner join projetos.funcao_membro fm on fm.id_funcao_membro =
mp.id_funcao_membro
inner join projetos.categoria_membro cm on cm.id_categoria_membro =
mp.id_categoria_membro
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where cm.descricao = 'DISCENTE'
and pae.cpf != 0
and fm.descricao = 'DISCENTE BOLSISTA'
and fm.extensao = true
group by ci.nome, proj.ano, tca.descricao, endereco.cep
order by ci.nome, proj.ano asc;

```

```

-- discentes em atividades de extensão por campus, ano e tipo de curso
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, tca.descricao as tipo_curso, count(distinct d.matricula) as
quantidade_discentes_envolvidos_atividades
from extensao.participante_acao_extensao pae
inner join public.discente d on d.id_discente = pae.id_discente
inner join public.curso c on c.id_curso = d.id_curso
inner join ensino.tipo_curso_academico tca on tca.id_tipo_curso_academico =
c.id_tipo_curso_academico
inner join extensao.tipo_participacao tp on tp.id_tipo_participacao =
pae.id_tipo_participacao
inner join extensao.atividade a on a.id_atividade = pae.id_acao_extensao
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.membro_projeto mp on mp.id_projeto = proj.id_projeto
inner join projetos.funcao_membro fm on fm.id_funcao_membro =
mp.id_funcao_membro
inner join projetos.categoria_membro cm on cm.id_categoria_membro =
mp.id_categoria_membro
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where cm.descricao = 'DISCENTE'
and pae.cpf != 0
and fm.extensao = true
group by ci.nome, proj.ano, endereco.cep, tca.descricao
order by ci.nome, proj.ano asc;

```

-- quantitativo de discentes envolvidos em atividades de extensão além dos créditos obrigatórios por campus, ano e tipo de curso

```
SELECT ci.nome as campus, regexp_replace(endereco.cep,['^0-9'], ", 'g') as cep,
proj.ano as ano_atividade, tca.descricao as tipo_curso, count(distinct d.matricula) as
quantidade_discente_realizando_atividade_extensao_nao_obrigatoria
FROM extensao.participante_acao_extensao pae
INNER JOIN public.discente d ON d.id_discente = pae.id_discente
INNER JOIN extensao.atividade a ON a.id_atividade = pae.id_acao_extensao
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join public.curso c on c.id_curso = d.id_curso
inner join ensino.tipo_curso_academico tca on tca.id_tipo_curso_academico =
c.id_tipo_curso_academico
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
GROUP BY ci.nome, proj.ano, endereco.cep, tca.descricao
HAVING
COUNT(CASE WHEN a.id_componente_curricular IS NOT NULL THEN 1
END) > 0
AND COUNT(CASE WHEN a.id_componente_curricular IS NULL THEN 1
END) > 0
ORDER BY ci.nome, proj.ano asc;
```

-- quantitativo total de discentes por campus, ano de ingresso e tipo de curso para percentual

```
select ci.nome as campus, regexp_replace(endereco.cep,['^0-9'], ", 'g') as cep,
d.ano_ingresso, tca.descricao as tipo_curso, count(d.matricula)
from public.discente d
inner join public.curso c on c.id_curso = d.id_curso
inner join ensino.tipo_curso_academico tca on tca.id_tipo_curso_academico =
c.id_tipo_curso_academico
inner join comum.unidade u on u.id_unidade = c.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
group by ci.nome, endereco.cep, d.ano_ingresso, tca.descricao
order by ci.nome asc, d.ano_ingresso asc;
```

-- PARTICIPANTES DA EXTENSÃO

-- quantitativo de servidores envolvidos em atividades de extensão por campus e ano

```
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, count(distinct pae.cpf) as
quantidade_servidores_envolvidos_atividades
from extensao.participante_acao_extensao pae
inner join rh.servidor s on s.id_servidor = pae.id_servidor
inner join extensao.tipo_participacao tp on tp.id_tipo_participacao =
pae.id_tipo_participacao
inner join extensao.atividade a on a.id_atividade = pae.id_acao_extensao
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where pae.cpf != 0
group by ci.nome, proj.ano, endereco.cep
order by ci.nome, proj.ano asc;
```

-- quantitativo de servidores envolvidos em atividades com instituições ativas nos fóruns de extensão por campus e ano

```
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano, count(distinct pae.cpf) as
quantidade_servidores_envolvidos_atividades_extensao_parcerias_autorizadas
from extensao.participante_acao_extensao pae
inner join rh.servidor s on s.id_servidor = pae.id_servidor
inner join extensao.atividade a on a.id_atividade = pae.id_acao_extensao
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.tipo_situacao_projeto tsp on tsp.id_tipo_situacao_projeto =
proj.id_tipo_situacao_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where tsp.contexto = 'E'
and tsp.descricao in ('PARCERIA AUTORIZADA')
and a.atividade_possui_parceria = true
group by ci.nome, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;
```

-- quantitativo de participantes externos envolvidos em atividades de extensão por campus e ano

```
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, count(distinct pae.cpf) as
quantidade_externos_envolvidos_atividades
from extensao.participante_acao_extensao pae
inner join extensao.tipo_participacao tp on tp.id_tipo_participacao =
pae.id_tipo_participacao
```

```

inner join extensao.atividade a on a.id_atividade = pae.id_acao_extensao
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.membro_projeto mp on mp.id_projeto = proj.id_projeto
inner join projetos.funcao_membro fm on fm.id_funcao_membro =
mp.id_funcao_membro
inner join projetos.categoria_membro cm on cm.id_categoria_membro =
mp.id_categoria_membro
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where cm.descricao = 'EXTERNO'
and pae.cpf != 0
and fm.extensao = true
group by ci.nome, proj.ano, endereco.cep
order by ci.nome, proj.ano asc;

```

-- quantitativo de participantes externos com papel de voluntario envolvidos em atividades de extensão por campus e ano

```

select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, count(distinct pae.cpf) as
quantidade_externos_voluntarios_envolvidos_atividades
from extensao.participante_acao_extensao pae
inner join extensao.tipo_participacao tp on tp.id_tipo_participacao =
pae.id_tipo_participacao
inner join extensao.atividade a on a.id_atividade = pae.id_acao_extensao
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.membro_projeto mp on mp.id_projeto = proj.id_projeto
inner join projetos.funcao_membro fm on fm.id_funcao_membro =
mp.id_funcao_membro
inner join projetos.categoria_membro cm on cm.id_categoria_membro =
mp.id_categoria_membro
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where cm.descricao = 'EXTERNO'
and pae.cpf != 0
and fm.extensao = true
and fm.descricao = 'VOLUNTARIO(A)'
or fm.descricao = 'voluntário'
group by ci.nome, proj.ano, endereco.cep
order by ci.nome, proj.ano asc;

```

-- quantitativo de participantes externos envolvidos em atividades de extensão por campus, ano e papel

```
SELECT
    ci.nome AS campus,
    regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
    proj.ano AS ano_atividade,
    CASE
        WHEN fm.descricao IN ('VOLUNTARIO(A)', 'voluntário') THEN
'VOLUNTARIO(A)'
        when fm.descricao = 'AVALIADOR(A)' then 'AVALIADOR(A)'
        when fm.descricao = 'Coautor (SEURS)' then 'Coautor (SEURS)'
        when fm.descricao = 'COLABORADOR(A)' then 'COLABORADOR(A)'
        when fm.descricao = 'COORDENADOR(A) DE EVENTOS MULHERES SIM'
then 'COORDENADOR(A) DE EVENTOS MULHERES SIM'
        WHEN fm.descricao IN ('discente', 'DISCENTE EXTENSIONISTA') THEN
'DISCENTE EXTENSIONISTA'
        when fm.descricao = 'Oficineiro (SEURS)' then 'Oficineiro (SEURS)'
        ELSE fm.descricao END AS papel,
    COUNT(DISTINCT pae.cpf) AS
quantidade_externos_por_papel_envolvidos_atividades
FROM
    extensao.participante_acao_extensao pae
    inner join extensao.tipo_participacao tp on tp.id_tipo_participacao =
pae.id_tipo_participacao
    inner join extensao.atividade a on a.id_atividade = pae.id_acao_extensao
    inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
    inner join projetos.edital e on e.id_edital = ee.id_edital
    inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
    inner join projetos.membro_projeto mp on mp.id_projeto = proj.id_projeto
    inner join projetos.funcao_membro fm on fm.id_funcao_membro =
mp.id_funcao_membro
    inner join projetos.categoria_membro cm on cm.id_categoria_membro =
mp.id_categoria_membro
    inner join comum.unidade u on u.id_unidade = proj.id_unidade
    inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
    inner join comum.endereco endereco on endereco.id_endereco =
ci.id_endereco
WHERE
    cm.descricao = 'EXTERNO'
    AND pae.cpf != 0
    AND fm.extensao = TRUE
GROUP BY
    ci.nome, proj.ano, endereco.cep,
    CASE
        WHEN fm.descricao IN ('VOLUNTARIO(A)', 'voluntário') THEN
'VOLUNTARIO(A)'
        when fm.descricao = 'AVALIADOR(A)' then 'AVALIADOR(A)'
        when fm.descricao = 'Coautor (SEURS)' then 'Coautor (SEURS)'
        when fm.descricao = 'COLABORADOR(A)' then 'COLABORADOR(A)'
```



```

        when fm.descricao = 'COORDENADOR(A) DE EVENTOS MULHERES SIM'
then 'COORDENADOR(A) DE EVENTOS MULHERES SIM'
        WHEN fm.descricao IN ('discente', 'DISCENTE EXTENSIONISTA') THEN
'DISCENTE EXTENSIONISTA'
        when fm.descricao = 'Oficineiro (SEURS)' then 'Oficineiro (SEURS)'
        ELSE fm.descricao END
ORDER BY
        ci.nome, proj.ano ASC;

```

-- quantitativo de participantes externos envolvidos em atividades de extensão por campus, ano e status

```

select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, cm.descricao as status, count(distinct pae.cpf) as
quantidade_externos_por_status_envolvidos_atividades
from extensao.participante_acao_extensao pae
inner join extensao.tipo_participacao tp on tp.id_tipo_participacao =
pae.id_tipo_participacao
inner join extensao.atividade a on a.id_atividade = pae.id_acao_extensao
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.membro_projeto mp on mp.id_projeto = proj.id_projeto
inner join projetos.categoria_membro cm on cm.id_categoria_membro =
mp.id_categoria_membro
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where pae.cpf != 0
group by ci.nome, proj.ano, endereco.cep, cm.descricao
order by ci.nome, proj.ano asc, cm.descricao asc;

```

-- proporção de participantes externos por total de atividade de extensão
-- (usar tabela de quantitativo total de participantes externos e tabela de quantitativo total de atividades de extensão da categoria ATIVIDADES DE EXTENSÃO)

-- proporção de participantes externos por total de participantes (retorna total de participantes de extensão por campus e ano)

```

select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, count(pae.cpf) as
quantidade_participantes_atividades_extensao
from extensao.participante_acao_extensao pae
inner join extensao.atividade a on a.id_atividade = pae.id_acao_extensao
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where cpf != 0

```



```
group by ci.nome, endereco.cep, proj.ano  
order by ci.nome, proj.ano asc;
```

-- percentual de servidores envolvidos em atividades de extensão (retorna total de servidores ativos por campus e ano, considerar o quantitativo de servidores envolvidos em extensão)

```
select ci.nome as campus, regexp_replace(endereco.cep, '^0-9', ", 'g') as cep,  
count(s.id_servidor) as servidores_ativos  
from rh.servidor s  
inner join rh.tipo_categoria tc on tc.id_tipo_categoria = s.id_categoria  
inner join rh.ativo a on a.id_ativo = s.id_ativo  
inner join comum.unidade u on u.id_unidade = s.id_unidade  
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies  
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco  
where s.id_ativo = 1  
group by ci.nome, endereco.cep  
order by ci.nome asc;
```

-- MODALIDADES DE ATIVIDADES DE EXTENSÃO

-- quantitativo de curso extensão por campus e ano
-- quantitativo de evento por campus e ano
-- quantitativo de programa por campus e ano
-- quantitativo de produto por campus e ano
-- quantitativo de projeto por campus e ano
-- quantitativo de prestação de serviço por campus e ano

```
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, tae.descricao as modalidade_extensao,
count(a.id_atividade) as quantidade_atividade_extensao
from extensao.atividade a
inner join extensao.tipo_atividade_extensao tae on tae.id_tipo_atividade_extensao =
a.id_tipo_atividade_extensao
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
group by ci.nome, proj.ano, endereco.cep, tae.descricao
order by ci.nome, proj.ano, tae.descricao asc;
```

-- proporção de discentes com matrícula ativa por programas e projetos por campus e ano (retorna quantitativo de discentes com matrícula ativa em programas e projetos)

```
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, count(a.id_atividade) as
quantidade_matricula_ativa_programa_projeto
from extensao.participante_acao_extensao pae
inner join extensao.atividade a on a.id_atividade = pae.id_acao_extensao
inner join extensao.tipo_atividade_extensao tae on tae.id_tipo_atividade_extensao =
a.id_tipo_atividade_extensao
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
inner join public.discente d on d.id_discente = pae.id_discente
inner join public.status_discente sd on sd.status = d.status
where tae.descricao in ('PROGRAMA', 'PROJETO')
and sd.descricao like '%ATIVO%'
group by ci.nome, proj.ano, endereco.cep
order by ci.nome, proj.ano asc;
```

```

-- quantitativo de programa e projeto por campus e ano
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, count(a.id_atividade) as
quantidade_programa_projeto_extensao
from extensao.atividade a
inner join extensao.tipo_atividade_extensao tae on tae.id_tipo_atividade_extensao =
a.id_tipo_atividade_extensao
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where tae.descricao in ('PROGRAMA', 'PROJETO')
group by ci.nome, proj.ano, endereco.cep
order by ci.nome, proj.ano asc;

```

```

-- quantitativo de discentes com matrícula ativa por programas por campus e ano
-- quantitativo de discentes com matrícula ativa por projetos por campus e ano
-- quantitativo de discentes com matrícula ativa por produtos por campus e ano
-- quantitativo de discentes com matrícula ativa por prestação de serviço por campus
e ano
-- quantitativo de discentes com matrícula ativa por curso extensão por campus e
ano
-- quantitativo de discentes com matrícula ativa por evento por campus e ano
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, tae.descricao as modalidade_extensao,
count(d.matricula) as quantidade_matricula_ativa_por_modalidade
from extensao.participante_acao_extensao pae
inner join public.discente d on d.id_discente = pae.id_discente
inner join public.status_discente sd on sd.status = d.status
inner join extensao.atividade a on a.id_atividade = pae.id_acao_extensao
inner join extensao.tipo_atividade_extensao tae on tae.id_tipo_atividade_extensao =
a.id_tipo_atividade_extensao
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where sd.descricao like '%ATIVO%'
group by ci.nome, proj.ano, endereco.cep, tae.descricao
order by ci.nome, proj.ano, tae.descricao asc;

```

```

-- proporção de discentes com matrícula ativa por programas por campus e ano
-- proporção de discentes com matrícula ativa por projetos por campus e ano
-- proporção de discentes com matrícula ativa por produtos por campus e ano

```

-- proporção de discentes com matrícula ativa por prestação de serviço por campus e ano
-- proporção de atividades de extensão no ano por modalidade projeto por campus e ano
-- proporção de atividades de extensão no ano por modalidade curso por campus e ano

-- retorna a quantidade de matrículas ativas por campus e ano de ingresso
`select ci.nome as campus, regexp_replace(endereco.cep,'[^0-9]', '', 'g') as cep, d.ano_ingresso, count(d.matricula) as quantidade_matriculas_ativas
from public.discente d
inner join public.status_discente sd on sd.status = d.status
inner join public.curso c on c.id_curso = d.id_curso
inner join comum.unidade u on u.id_unidade = c.id_unidade
inner join comum.campus_ies ci on u.id_campus_ies = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where sd.descricao like '%ATIVO%'
group by ci.nome, endereco.cep, d.ano_ingresso
order by ci.nome, d.ano_ingresso asc;`

-- quantitativo de incubadoras por campus e ano
`select ci.nome as campus, regexp_replace(endereco.cep,'[^0-9]', '', 'g') as cep, proj.ano, count(a.id_atividade) as incubadoras
from extensao.atividade a
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where UPPER(proj.titulo) like '%INCUBADORA%'
group by ci.nome, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;`

-- quantitativo de empreendimentos graduados em incubadoras resultados da extensão por campus e ano
`select ci.nome as campus, regexp_replace(endereco.cep,'[^0-9]', '', 'g') as cep, proj.ano, count(a.id_atividade) as incubadoras_graduadas
from extensao.atividade a
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.tipo_situacao_projeto tsp on tsp.id_tipo_situacao_projeto = proj.id_tipo_situacao_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where UPPER(proj.titulo) like '%INCUBADORA%'
and tsp.contexto = 'E'
and tsp.descricao in ('CONCLUÍDA','REGISTRO APROVADO')
group by ci.nome, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;`

-- quantitativo de cooperativas populares graduadas em incubadoras resultadas da extensão por campus e ano

```
select ci.nome as campus, regexp_replace(endereco.cep,['^0-9'], '', 'g') as cep,
proj.ano, count(a.id_atividade) as cooperativas_graduadas
from extensao.atividade a
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.tipo_situacao_projeto tsp on tsp.id_tipo_situacao_projeto =
proj.id_tipo_situacao_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where UPPER(proj.titulo) like '%COOPERATIVA%'
and tsp.contexto = 'E'
and tsp.descricao in ('CONCLUÍDA','REGISTRO APROVADO')
group by ci.nome, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;
```

-- CURRICULARIZAÇÃO DA EXTENSÃO

```
-- quantitativo de discentes em atividade curricular de extensão por campus e ano
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, count(d.matricula) as
quantidade_discentes_atividade_extensao_curricular
from extensao.participante_acao_extensao pae
inner join public.discente d on d.id_discente = pae.id_discente
inner join public.status_discente sd on sd.status = d.status
inner join extensao.atividade a on a.id_atividade = pae.id_acao_extensao
inner join extensao.atividade_cursos_componentes acc on acc.id_atividade =
a.id_atividade
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where sd.descricao like '%ATIVO%'
group by ci.nome, proj.ano, endereco.cep
order by ci.nome, proj.ano asc;
```

-- percentual de créditos ou carga horária destinados à extensão no currículo de nível médio técnico por campus e ano

-- percentual de créditos ou carga horária destinados à extensão no currículo de graduação por campus e ano

-- percentual de créditos ou carga horária destinados à extensão no currículo de especialização por campus e ano

-- percentual de créditos ou carga horária destinados à extensão no currículo de mestrado por campus e ano

-- percentual de créditos ou carga horária destinados à extensão no currículo de doutorado por campus e ano

-- percentual de créditos ou carga horária destinados à extensão no currículo de pós-graduação (mestrado e doutorado) por campus e ano

-- retorna o quantitativo de créditos e carga horária destinados à extensão

```
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, tca.descricao as tipo_curso, sum(ccd.cr_total) as
total_creditos_extensao, sum(ccd.ch_total) as total_carga_horaria_extensao
from extensao.atividade a
inner join extensao.atividade_cursos_componentes acc on acc.id_atividade =
a.id_atividade
inner join public.curso c on c.id_curso = acc.id_curso
inner join ensino.componente_curricular cc on cc.id_disciplina =
acc.id_componente_curricular
inner join ensino.componente_curricular_detalhes ccd on
ccd.id_componente_detalhes = cc.id_detalhe
inner join ensino.tipo_curso_academico tca on tca.id_tipo_curso_academico =
c.id_tipo_curso_academico
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
```

```

inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
group by ci.nome, proj.ano, endereco.cep, tca.descricao
order by ci.nome, proj.ano, tca.descricao asc;

-- retorna o quantitativo de créditos e carga horária total dos cursos
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
mc.ano as ano_atividade, tca.descricao as tipo_curso, sum(ccd.cr_total) as
total_creditos, sum(ccd.ch_total) as total_carga_horaria
from ensino.matricula_componente mc
inner join ensino.componente_curricular cc on cc.id_disciplina =
mc.id_componente_curricular
inner join ensino.componente_curricular_detalhes ccd on
ccd.id_componente_detalhes = cc.id_detalhe
inner join public.curso c on c.id_curso = cc.id_curso
inner join ensino.tipo_curso_academico tca on tca.id_tipo_curso_academico =
c.id_tipo_curso_academico
inner join comum.unidade u on u.id_unidade = c.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
group by ci.nome, mc.ano, endereco.cep, tca.descricao
order by ci.nome, mc.ano, tca.descricao asc;

```

-- percentual de créditos ou carga horária de ações de extensão no currículo de cursos por campus e ano

```

-- retorna o quantitativo destinados à extensão
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, sum(ccd.cr_total) as total_creditos_extensao,
sum(ccd.ch_total) as total_carga_horaria_extensao
from extensao.atividade a
inner join extensao.atividade_cursos_componentes acc on acc.id_atividade =
a.id_atividade
inner join public.curso c on c.id_curso = acc.id_curso
inner join ensino.componente_curricular cc on cc.id_disciplina =
acc.id_componente_curricular
inner join ensino.componente_curricular_detalhes ccd on
ccd.id_componente_detalhes = cc.id_detalhe
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
group by ci.nome, proj.ano, endereco.cep
order by ci.nome, proj.ano asc;

```

```

-- retorna o quantitativo total dos cursos
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
mc.ano as ano_atividade, sum(ccd.cr_total) as total_creditos, sum(ccd.ch_total) as
total_carga_horaria
from ensino.matricula_componente mc
inner join ensino.componente_curricular cc on cc.id_disciplina =
mc.id_componente_curricular

```

```

inner join ensino.componente_curricular_detalhes ccd on
ccd.id_componente_detalhes = cc.id_detalhe
inner join public.curso c on c.id_curso = cc.id_curso
inner join comum.unidade u on u.id_unidade = c.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
group by ci.nome, mc.ano, endereco.cep
order by ci.nome, mc.ano asc;

```

```

-- quantitativo de atividades de extensão por tipo de curso por campus e ano
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', ", 'g') as cep,
proj.ano as ano_atividade, tca.descricao as tipo_curso, count(a.id_atividade) as
quantidade_atividades_extensao_curricularizadas_por_tipo_curso
from extensao.atividade a
inner join extensao.atividade_cursos_componentes acc on acc.id_atividade =
a.id_atividade
inner join public.curso c on c.id_curso = acc.id_curso
inner join ensino.tipo_curso_academico tca on tca.id_tipo_curso_academico =
c.id_tipo_curso_academico
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
group by ci.nome, proj.ano, endereco.cep, tca.descricao
order by ci.nome, proj.ano, tca.descricao asc;

```

```

-- Percentual de PPCs com curricularização da extensão e atividades
complementares por campus, ano e tipo de curso
-- retorna o total de cursos ativos em que há curricularização da extensão
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', ", 'g') as cep,
date_part('year', c.data_cadastro) as ano_cadastro, tca.descricao as tipo_curso,
count(c.id_curso) as ppcs_com_curricularizacao_extensao
from public.curso c
inner join ensino.tipo_curso_academico tca on tca.id_tipo_curso_academico =
c.id_tipo_curso_academico
inner join comum.unidade u on u.id_unidade = c.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where c.ativo = true
and c.id_curso in (
    select acc.id_curso from extensao.atividade_cursos_componentes acc
)
group by ci.nome, ano_cadastro, endereco.cep, tca.descricao
order by ci.nome, ano_cadastro, tca.descricao asc;

```



```

-- retorna o total de cursos ativos cadastrados no sigaa
select ci.nome as campus, regexp_replace(endereco.cep,['^0-9'], ", 'g') as cep,
date_part('year', c.data_cadastro) as ano_cadastro, tca.descricao as tipo_curso,
count(c.id_curso) as quantidade_ppcs
from public.curso c
inner join ensino.tipo_curso_academico tca on tca.id_tipo_curso_academico =
c.id_tipo_curso_academico
inner join comum.unidade u on u.id_unidade = c.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where c.ativo = true
group by ci.nome, ano_cadastro, endereco.cep, tca.descricao
order by ci.nome, ano_cadastro, tca.descricao asc;

```

-- ORÇAMENTO (SEM TABELAS NO SIGAA EXTENSÃO)

- percentual de recursos financeiros do orçamento anual público aplicados em extensão
- quantitativo de recurso financeiro orçamentário total por câmpus e ano
- quantitativo de recursos financeiros do orçamento anual público obtido para extensão por câmpus e ano
- quantitativo de recursos financeiros do orçamento anual público aplicados em extensão por câmpus e ano
- quantitativo de recursos para extensão captados via prestação de serviços acadêmicos especializados por câmpus e ano
- quantitativo de recursos para extensão captados via editais externos por câmpus e ano
- quantitativo de recurso aplicado por edital interno de extensão por câmpus e ano
- quantitativo de recurso repassado aos coordenadores de atividade de extensão por câmpus e ano
- quantitativo de recurso repassado aos bolsistas discentes por câmpus e ano

-- PÚBLICO ALVO

-- quantitativo de ações de extensão demandadas pelo público externo por câmpus e ano

```
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano, count(a.id_atividade) as quantitativo_publico_externo_demandante
from extensao.atividade a
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where proj.protagonismo_externo != ""
or proj.protagonismo_externo != null
group by ci.nome, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;
```

-- quantitativo de pessoas envolvidas por atividades de extensão por câmpus e ano

```
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano, cast(sum(rae.publico_real_atingido) as INTEGER) as
quantitativo_publico_envolvido_atividades_extensao
from extensao.atividade a
inner join extensao.relatorio_acao_extensao rae on rae.id_atividade = a.id_atividade
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
group by ci.nome, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;
```

-- quantitativo de pessoas alcançadas por atividades de modalidade prestação de serviço por câmpus e ano

```
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano, cast(sum(rae.publico_real_atingido) as INTEGER) as
quantitativo_publico_envolvido_prestacao_servico
from extensao.atividade a
inner join extensao.tipo_atividade_extensao tae on tae.id_tipo_atividade_extensao =
a.id_tipo_atividade_extensao
inner join extensao.relatorio_acao_extensao rae on rae.id_atividade = a.id_atividade
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where tae.descricao = 'PRESTAÇÃO DE SERVIÇOS'
group by ci.nome, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;
```

-- quantitativo de pessoas alcançadas por atividades de modalidade programas e projetos por câmpus e ano

```
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano, cast(sum(rae.publico_real_atingido) as INTEGER) as
quantitativo_publico_envolvido_programa_projeto
from extensao.atividade a
inner join extensao.tipo_atividade_extensao tae on tae.id_tipo_atividade_extensao =
a.id_tipo_atividade_extensao
inner join extensao.relatorio_acao_extensao rae on rae.id_atividade = a.id_atividade
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where tae.descricao = 'PROGRAMA'
or tae.descricao = 'PROJETO'
group by ci.nome, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;
```

-- quantitativo de pessoas alcançadas por atividades de modalidade cursos e eventos por câmpus e ano

```
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano, cast(sum(rae.publico_real_atingido) as INTEGER) as
quantitativo_publico_envolvido_curso_evento
from extensao.atividade a
inner join extensao.tipo_atividade_extensao tae on tae.id_tipo_atividade_extensao =
a.id_tipo_atividade_extensao
inner join extensao.relatorio_acao_extensao rae on rae.id_atividade = a.id_atividade
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where tae.descricao = 'CURSO'
or tae.descricao = 'EVENTO'
group by ci.nome, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;
```

-- quantitativo de ações de extensão para capacitação de população EJA por câmpus e ano

```
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano, count(a.id_atividade) as quantitativo_publico_EJA
from extensao.atividade a
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where UPPER(proj.objetivos) like '%EJA%'
or UPPER(proj.objetivos) like '%JOVENS E ADULTOS%'
or UPPER(proj.objetivos) like '%ADULTOS%'
or UPPER(proj.objetivos) like '%EDUCAÇÃO DE JOVENS E ADULTOS%'
```

```

or UPPER(proj.objetivos) like '%IDOSO%'
or UPPER(proj.objetivos) like '%IDOSOS%'
group by ci.nome, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;

```

-- quantitativo de ações de extensão para capacitação de população PCD por câmpus e ano

```

select ci.nome as campus, regexp_replace(endereco.cep,'[^0-9]', '', 'g') as cep,
proj.ano, count(a.id_atividade) as quantitativo_publico_PCD
from extensao.atividade a
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where UPPER(proj.objetivos) like '%PCD%'
or UPPER(proj.objetivos) like '%PESSOA COM DEFICIÊNCIA%'
or UPPER(proj.objetivos) like '%PESSOA COM DEFICIENCIA%'
or UPPER(proj.objetivos) like '%PESSOA COM DEFICIÊNCIAS%'
or UPPER(proj.objetivos) like '%PESSOA COM DEFICIENCIAS%'
or UPPER(proj.objetivos) like '%PESSOAS COM DEFICIÊNCIA%'
or UPPER(proj.objetivos) like '%PESSOAS COM DEFICIENCIA%'
or UPPER(proj.objetivos) like '%PESSOAS COM DEFICIÊNCIAS%'
or UPPER(proj.objetivos) like '%PESSOAS COM DEFICIENCIAS%'
or UPPER(proj.objetivos) like '%DEFICIÊNCIA%'
or UPPER(proj.objetivos) like '%DEFICIENCIA%'
or UPPER(proj.objetivos) like '%DEFICIÊNCIAS%'
or UPPER(proj.objetivos) like '%DEFICIENCIAS%'
or UPPER(proj.objetivos) like '%DEFICIENTE%'
or UPPER(proj.objetivos) like '%DEFICIENTES%'
group by ci.nome, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;

```

-- quantitativo de pessoas envolvidas por discentes de graduação por meio de ações de extensão por câmpus e ano (considerar o quantitativo de público envolvido pela consulta abaixo de discentes de graduacao)

```

select ci.nome as campus, regexp_replace(endereco.cep,'[^0-9]', '', 'g') as cep,
d.ano_ingresso, tca.descricao as tipo_curso, count(distinct d.matricula) as
quantidade_discentes_graduacao
from public.discente d
inner join public.status_discente sd on sd.status = d.status
inner join public.curso c on c.id_curso = d.id_curso
inner join ensino.tipo_curso_academico tca on tca.id_tipo_curso_academico =
c.id_tipo_curso_academico
inner join comum.unidade u on u.id_unidade = c.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where tca.nivel = 'G'
group by ci.nome, d.ano_ingresso, endereco.cep, tca.descricao
order by ci.nome, d.ano_ingresso asc;

```

-- quantitativo de docentes da rede pública capacitados por programas e projetos de formação continuada por câmpus e ano

```
select ci.nome as campus, regexp_replace(endereco.cep,['^0-9'], ", 'g') as cep,
proj.ano, cast(sum(rae.publico_real_atingido) as INTEGER) as
quantitativo_docente_rede_publica_capacitado_por_programa_projeto
from extensao.atividade a
inner join extensao.relatorio_acao_extensao rae on rae.id_atividade = a.id_atividade
inner join extensao.tipo_atividade_extensao tae on tae.id_tipo_atividade_extensao =
a.id_tipo_atividade_extensao
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where UPPER(proj.objetivos) like '%DOCENTES REDE PÚBLICA%'
or UPPER(proj.objetivos) like '%DOCENTE REDE PÚBLICA%'
or UPPER(proj.objetivos) like '%PROFESSOR REDE PÚBLICA%'
or UPPER(proj.objetivos) like '%PROFESSORES REDE PÚBLICA%'
OR UPPER(proj.objetivos) like '%DOCENTES DA REDE PÚBLICA%'
or UPPER(proj.objetivos) like '%DOCENTE DA REDE PÚBLICA%'
or UPPER(proj.objetivos) like '%PROFESSOR DA REDE PÚBLICA%'
or UPPER(proj.objetivos) like '%PROFESSORES DA REDE PÚBLICA%'
OR UPPER(proj.objetivos) like '%DOCENTES REDE PUBLICA%'
or UPPER(proj.objetivos) like '%DOCENTE REDE PUBLICA%'
or UPPER(proj.objetivos) like '%PROFESSOR REDE PUBLICA%'
or UPPER(proj.objetivos) like '%PROFESSORES REDE PUBLICA%'
OR UPPER(proj.objetivos) like '%DOCENTES DA REDE PUBLICA%'
or UPPER(proj.objetivos) like '%DOCENTE DA REDE PUBLICA%'
or UPPER(proj.objetivos) like '%PROFESSOR DA REDE PUBLICA%'
or UPPER(proj.objetivos) like '%PROFESSORES DA REDE PUBLICA%'
OR UPPER(proj.objetivos) like '%DOCENTES ESTADUAIS%'
or UPPER(proj.objetivos) like '%DOCENTE ESTADUAIS%'
or UPPER(proj.objetivos) like '%PROFESSOR ESTADUAIS%'
or UPPER(proj.objetivos) like '%PROFESSORES ESTADUAIS%'
OR UPPER(proj.objetivos) like '%DOCENTES DO ESTADO%'
or UPPER(proj.objetivos) like '%DOCENTE DO ESTADO%'
or UPPER(proj.objetivos) like '%PROFESSOR DO ESTADO%'
or UPPER(proj.objetivos) like '%PROFESSORES DO ESTADO%'
OR UPPER(proj.objetivos) like '%DOCENTES MUNICIPAIS%'
or UPPER(proj.objetivos) like '%DOCENTE MUNICIPAIS%'
or UPPER(proj.objetivos) like '%PROFESSOR MUNICIPAIS%'
or UPPER(proj.objetivos) like '%PROFESSORES MUNICIPAIS%'
OR UPPER(proj.objetivos) like '%DOCENTES DO MUNICIPIO%'
or UPPER(proj.objetivos) like '%DOCENTE DO MUNICIPIO%'
or UPPER(proj.objetivos) like '%PROFESSOR DO MUNICIPIO%'
or UPPER(proj.objetivos) like '%PROFESSORES DO MUNICIPIO%'
OR UPPER(proj.objetivos) like '%DOCENTES DO MUNICÍPIO%'
or UPPER(proj.objetivos) like '%DOCENTE DO MUNICÍPIO%'
or UPPER(proj.objetivos) like '%PROFESSOR DO MUNICÍPIO%'
```

```

or UPPER(proj.objetivos) like '%PROFESSORES DO MUNICÍPIO%'
OR UPPER(proj.objetivos) like '%DOCENTES DA ESCOLA%'
or UPPER(proj.objetivos) like '%DOCENTE DA ESCOLA%'
or UPPER(proj.objetivos) like '%PROFESSOR DA ESCOLA%'
or UPPER(proj.objetivos) like '%PROFESSORES DA ESCOLA%'
OR UPPER(proj.objetivos) like '%DOCENTES DO CENTRO%'
or UPPER(proj.objetivos) like '%DOCENTE DO CENTRO%'
or UPPER(proj.objetivos) like '%PROFESSOR DO CENTRO%'
or UPPER(proj.objetivos) like '%PROFESSORES DO CENTRO%'
and tae.descricao = 'PROGRAMA'
or tae.descricao = 'PROJETO'
group by ci.nome, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;

```

-- percentual de ações de extensão destinadas à inclusão de população vulnerável por câmpus e ano (consultar a quantidade de ações de extensão na categoria ATIVIDADES DE EXTENSÃO)

```

select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano, count(a.id_atividade) as quantitativo_atividades_populacao_vulneravel
from extensao.atividade a
inner join extensao.relatorio_acao_extensao rae on rae.id_atividade = a.id_atividade
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where UPPER(proj.objetivos) like '%POPULAÇÃO VULNERÁVEL%'
or UPPER(proj.objetivos) like '%POPULAÇÃO VULNERAVEL%'
or UPPER(proj.objetivos) like '%VULNERABILIDADE%'
or UPPER(proj.objetivos) like '%VULNERÁVEL%'
or UPPER(proj.objetivos) like '%VULNERAVEL%'
group by ci.nome, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;

```

-- percentual de ações de extensão dirigidas às escolas públicas por câmpus e ano (consultar a quantidade de ações de extensão na categoria ATIVIDADES DE EXTENSÃO)

```

select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano, count(a.id_atividade) as quantitativo_atividades_escolas_publicas
from extensao.atividade a
inner join extensao.relatorio_acao_extensao rae on rae.id_atividade = a.id_atividade
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where UPPER(proj.objetivos) like '%ESCOLA PÚBLICA%'
or UPPER(proj.objetivos) like '%ESCOLA PUBLICA%'
or UPPER(proj.objetivos) like '%ESCOLA ESTADUAL%'
or UPPER(proj.objetivos) like '%ESCOLA DO ESTADO%'
or UPPER(proj.objetivos) like '%ESCOLA MUNICIPAL%'

```



```
or UPPER(proj.objetivos) like '%ESCOLA DO MUNICÍPIO%'
or UPPER(proj.objetivos) like '%ESCOLA DO MUNICIPIO%'
or UPPER(proj.objetivos) like '%CENTRO ESCOLAR%'
group by ci.nome, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;
```

-- percentual de municípios envolvidos pelas ações extensionistas em relação ao total de municípios que são abrangidos pela respectiva IES

-- possível resgatar somente o quantitativo de municípios

```
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano, count(distinct lr.id_municipio) as quantidade_municipios_envolvidos
from extensao.atividade a
inner join extensao.local_realizacao lr on lr.id_atividade = a.id_atividade
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.tipo_situacao_projeto tsp on tsp.id_tipo_situacao_projeto =
proj.id_tipo_situacao_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
group by ci.nome, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;
```

-- PARCERIAS

-- percentual de ações de extensão com parcerias institucionais por câmpus e ano (considerar a consulta de total de atividades de extensão por câmpus e ano na categoria ATIVIDADES DE EXTENSÃO)

-- quantitativo de ações de extensão com parcerias institucionais por câmpus e ano

```
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano, count(a.id_atividade) as quantidade_parcerias
from extensao.atividade a
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.tipo_situacao_projeto tsp on tsp.id_tipo_situacao_projeto =
proj.id_tipo_situacao_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where tsp.contexto = 'E'
and a.atividade_possui_parceria = true
group by ci.nome, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;
```

-- percentual de representação oficial da instituição junto à sociedade civil por câmpus e ano

--(considerar o quantitativo de instituições ativas nos fóruns como parcerias oficiais por quantitativo total de atividades na categoria ATIVIDADES DE EXTENSÃO)

-- taxa de aprovação de propostas de extensão em editais externos por câmpus e ano (considerar quantitativo total de atividades na categoria ATIVIDADES DE EXTENSÃO)

```
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano, count(a.id_atividade) as
quantidade_atividades_extensao_financiamento_externo
from extensao.atividade a
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.tipo_situacao_projeto tsp on tsp.id_tipo_situacao_projeto =
proj.id_tipo_situacao_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where tsp.contexto = 'E'
and proj.financiamento_externo = true
group by ci.nome, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;
```

-- listagem nominal das parcerias institucionais com ações de extensão por câmpus e ano

```
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano, proj.titulo as nome_atividade_extensao
from extensao.atividade a
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
```



```

inner join projetos.tipo_situacao_projeto tsp on tsp.id_tipo_situacao_projeto =
proj.id_tipo_situacao_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where tsp.contexto = 'E'
and a.atividade_possui_parceria = true
group by ci.nome, endereco.cep, proj.ano, proj.titulo
order by ci.nome, proj.ano asc;

```

-- quantitativo de instituições ativas nos fóruns de extensão e relações externas nos
câmpus por ano.

```

select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano, count(a.id_atividade) as
quantidade_atividades_extensao_parcerias_autorizadas
from extensao.atividade a
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.tipo_situacao_projeto tsp on tsp.id_tipo_situacao_projeto =
proj.id_tipo_situacao_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where tsp.contexto = 'E'
and tsp.descricao in ('PARCERIA AUTORIZADA')
and a.atividade_possui_parceria = true
group by ci.nome, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;

```

-- PRODUÇÃO EXTENSIONISTA (SEM TABELA ESPECÍFICA NO SIGAA, EXTRAÇÃO FEITA DOS RELATÓRIOS DE ATIVIDADES)

- quantitativo de revistas e periódicos institucionais por câmpus e ano
- quantitativo de produção de livros ou capítulos com base em resultados da extensão por câmpus e ano
- quantitativo de publicação de artigos em periódicos com base em resultados da extensão por câmpus e ano
- quantitativo de mídias de comunicações em eventos com base em resultados da extensão por câmpus e ano
- quantitativo de produções audiovisuais resultadas da extensão por câmpus e ano
- quantitativo de produções artísticas (exposições, espetáculos, outros) resultadas da extensão por câmpus e ano
- com o resultado da consulta abaixo considerar o(s) prompt(s) para uma IA generativa:

- - construa um arquivo csv (ou dataframe) com o quantitativo de producao_extensionista agrupado por campus, cep, ano e tipo_produto, onde o tipo_produto pode ser classificado como 'Revistas e Periódicos', 'Livros ou Capítulos', 'Artigos', 'Mídias de Comunicações em Eventos', 'Produções Audiovisuais', 'Produções Artísticas' que podem ser extraídos de resultados_qualitativos e resultados_quantitativos

- - construa um arquivo csv (ou dataframe) com a descrição de producao_extensionista agrupado por campus, cep, ano e tipo_produto, onde o tipo_produto pode ser classificado como 'Revistas e Periódicos', 'Livros ou Capítulos', 'Artigos', 'Mídias de Comunicações em Eventos', 'Produções Audiovisuais', 'Produções Artísticas' que podem ser extraídos de resultados_qualitativos e resultados_quantitativos

```
select ci.nome as campus, regexp_replace(endereco.cep,'[^0-9]', '', 'g') as cep,
proj.ano, rae.resultados_qualitativos, rae.resultados_quantitativos
from extensao.atividade a
```

```
inner join extensao.relatorio_acao_extensao rae on rae.id_atividade = a.id_atividade
```

```
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
```

```
inner join comum.unidade u on u.id_unidade = proj.id_unidade
```

```
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
```

```
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
```

```
where rae.resultados_qualitativos != ""
```

```
or rae.resultados_quantitativos != ""
```

```
group by ci.nome, endereco.cep, proj.ano, rae.resultados_qualitativos,
rae.resultados_quantitativos
```

```
order by ci.nome, proj.ano asc;
```

-- ÁREA TEMÁTICA DE EXTENSÃO

```
-- quantitativo de atividades de extensão por área da comunicação por câmpus e ano
-- quantitativo de atividades de extensão por área da cultura por câmpus e ano
-- quantitativo de atividades de extensão por área dos direitos humanos e justiça por câmpus e ano
-- quantitativo de atividades de extensão por área da tecnologia e produção por câmpus e ano
-- quantitativo de atividades de extensão por área do trabalho por câmpus e ano
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, area.descricao as area_extensao, count(a.id_atividade)
as quantidade_atividade_extensao
from extensao.atividade a
inner join extensao.area_tematica area on area.id_area_tematica =
a.id_area_tematica_principal
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where area.ativo = true
group by ci.nome, proj.ano, endereco.cep, area.descricao
order by ci.nome, proj.ano, area.descricao asc;
```

-- INFRAESTRUTURA E RECURSOS HUMANOS

-- proporção de horas de treinamento por servidor da gestão extensionista por
câmpus e ano
-- proporção de participação em eventos por servidor da gestão extensionista por
câmpus e ano
-- proporção de participação em eventos promovidos ou apoiados pela pró-reitoria
de extensão por servidor da gestão extensionista por câmpus e ano
-- proporção espaço físico (área) por servidor por câmpus e ano
-- percentual de capacidade de atendimento de demandas de transporte logístico
para ações de extensão por câmpus e ano
-- percentual de servidores em representação oficial da instituição junto à sociedade
civil por câmpus e ano
--(considerar o quantitativo de instituições ativas em fóruns na categoria
PARCERIAS por quantitativo total de servidores na categoria PARTICIPANTES DA
EXTENSÃO)

-- quantitativo de unidades organizacionais proponentes de ação de extensão por
câmpus e ano

```
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, count(distinct u.id_unidade) as
quantidade_unidades_proponentes_extensao
from extensao.participante_acao_extensao pae
inner join rh.servidor s on s.id_servidor = pae.id_servidor
inner join extensao.tipo_participacao tp on tp.id_tipo_participacao =
pae.id_tipo_participacao
inner join extensao.atividade a on a.id_atividade = pae.id_acao_extensao
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where pae.cpf != 0
group by ci.nome, proj.ano, endereco.cep
order by ci.nome, proj.ano asc;
```

-- listagem nominal das unidades organizacionais proponentes de ação de extensão
por câmpus e ano

```
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, u.nome as unidade_proponente_extensao
from extensao.participante_acao_extensao pae
inner join rh.servidor s on s.id_servidor = pae.id_servidor
inner join extensao.tipo_participacao tp on tp.id_tipo_participacao =
pae.id_tipo_participacao
inner join extensao.atividade a on a.id_atividade = pae.id_acao_extensao
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
```

```
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where pae.cpf != 0
group by ci.nome, proj.ano, endereco.cep, u.nome
order by ci.nome, proj.ano, u.nome asc;
```

-- ATIVIDADES DE EXTENSÃO

-- quantitativo total de atividades de extensão por campus e ano

```
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, count(a.id_atividade) as quantidade_atividades_extensao
from extensao.atividade a
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
group by ci.nome, proj.ano, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;
```

-- quantitativo de atividades de extensão contínuas por campus e ano

```
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano as ano_atividade, count(a.id_atividade) as
quantidade_atividades_continuadas_extensao
from extensao.atividade a
inner join extensao.edital_extensao ee on ee.id_edital_extensao = a.id_edital
inner join projetos.edital e on e.id_edital = ee.id_edital
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where a.projeto_formacao_continuada = true
or a.projeto_formacao_continuada_multiplos_cursos = true
group by ci.nome, proj.ano, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;
```

-- proporção de ações de extensão com status aprovada por servidor por campus e ano (considerar percentual de servidores envolvidos em atividades de extensão (retorna total de servidores por campus e ano) da categoria PARTICIPANTES)

```
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano, count(a.id_atividade) as quantidade_atividades_extensao_aprovadas
from extensao.atividade a
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.tipo_situacao_projeto tsp on tsp.id_tipo_situacao_projeto =
proj.id_tipo_situacao_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where tsp.contexto = 'E'
and tsp.descricao in ('APROVADA COM RECURSOS DO
CÂMPUS', 'CLASSIFICADA (APROVADA SEM RECURSOS)')
group by ci.nome, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;
```

-- proporção de ações de extensão submetidas por aprovadas por câmpus e ano
(considerar a consulta anterior)

```
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano, count(a.id_atividade) as quantidade_atividades_extensao_submetidas
from extensao.atividade a
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.tipo_situacao_projeto tsp on tsp.id_tipo_situacao_projeto =
proj.id_tipo_situacao_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where tsp.contexto = 'E'
and tsp.descricao in ('SUBMETIDA')
group by ci.nome, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;
```

-- quantitativo de atividades de extensão com status favorável (segundo o SIGAA)
por câmpus e ano

```
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano, count(a.id_atividade) as quantidade_atividades_extensao_favoraveis
from extensao.atividade a
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.tipo_situacao_projeto tsp on tsp.id_tipo_situacao_projeto =
proj.id_tipo_situacao_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where tsp.contexto = 'E'
and tsp.descricao in ('AGUARDANDO APROVAÇÃO DOS
DEPARTAMENTOS','AGUARDANDO AVALIAÇÃO','ANALISANDO SOLICITAÇÃO
DE RECONSIDERAÇÃO','CADASTRO EM ANDAMENTO','CONCLUÍDA',
'EM EXECUÇÃO','ENTREGA DE RELATÓRIO FINAL EXPIRADA','PENDENTE DE
RELATÓRIO FINAL','PROPOSTA DEVOLVIDA PARA COORDENADOR
REEDITAR','PROPOSTA DEVOLVIDA PARA DEPARTAMENTO REVALIDAR',
'REGISTRO APROVADO','RELATÓRIO FINAL DO PROJETO ENTREGUE
(AGUARDANDO AVALIAÇÃO)','SUBMETIDA')
group by ci.nome, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;
```

-- quantitativo de atividades de extensão com status desfavorável (segundo o
SIGAA) por câmpus e ano

```
select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano, count(a.id_atividade) as quantidade_atividades_extensao_desfavoraveis
from extensao.atividade a
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.tipo_situacao_projeto tsp on tsp.id_tipo_situacao_projeto =
proj.id_tipo_situacao_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
```

```

where tsp.contexto = 'E'
and tsp.descricao in ('NÃO APROVADA','NÃO APROVADO PELOS
DEPARTAMENTOS','PARCERIA NÃO AUTORIZADA','PARCERIA NÃO
AUTORIZADA
REGISTRO NÃO APROVADO','REMOVIDA')
group by ci.nome, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;

```

```

-- quantitativo de atividade de extensão por status por câmpus e ano
select ci.nome as campus, regexp_replace(endereco.cep,['^0-9'], ", 'g') as cep,
proj.ano, tsp.descricao as status, count(a.id_atividade) as
quantidade_atividades_extensao_por_status
from extensao.atividade a
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join projetos.tipo_situacao_projeto tsp on tsp.id_tipo_situacao_projeto =
proj.id_tipo_situacao_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where tsp.contexto = 'E'
group by ci.nome, endereco.cep, proj.ano, tsp.descricao
order by ci.nome, proj.ano asc, tsp.descricao;

```

```

-- percentual de atividade de extensao que articularam com a pesquisa por câmpus
e ano (considerar quantitativo total de atividades de extensão por câmpus e ano)
select ci.nome as campus, regexp_replace(endereco.cep,['^0-9'], ", 'g') as cep,
proj.ano, count(a.id_atividade) as
quantidade_atividades_extensao_relacionada_pesquisa
from extensao.atividade a
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where proj.relacao_ensino_pesquisa_extensao != null or
proj.relacao_ensino_pesquisa_extensao != "
and proj.ensino = false
or proj.apoio_grupo_pesquisa = true
or proj.apoio_novos_pesquisadores = true
or proj.iniciacao_cientifica = true
group by ci.nome, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;

```

```

-- percentual de atividade de extensao que articularam com o ensino por câmpus e
ano (considerar quantitativo total de atividades de extensão por câmpus e ano)
select ci.nome as campus, regexp_replace(endereco.cep,['^0-9'], ", 'g') as cep,
proj.ano, count(a.id_atividade) as
quantidade_atividades_extensao_relacionada_ensino
from extensao.atividade a
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto

```



```

inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where proj.ensino = true
or proj.melhoria_qualidade_ensino = true
group by ci.nome, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;

```

-- quantitativo de atividades de extensão por objetivo sustentável da ONU por
câmpus e ano (não há na base de dados o registro dos objetivos sustentáveis, mas
é possível retornar o quantitativo)

```

select ci.nome as campus, regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep,
proj.ano, count(a.id_atividade) as
quantidade_atividades_extensao_objetivos_sustentaveis
from extensao.atividade a
inner join projetos.projeto proj on proj.id_projeto = a.id_projeto
inner join comum.unidade u on u.id_unidade = proj.id_unidade
inner join comum.campus_ies ci on ci.id_campus = u.id_campus_ies
inner join comum.endereco endereco on endereco.id_endereco = ci.id_endereco
where a.possui_objetivo_sustentavel = true
group by ci.nome, endereco.cep, proj.ano
order by ci.nome, proj.ano asc;

```

-- listagem de editais internos com e sem financiamento

```

SELECT ci.nome as campus,
regexp_replace(endereco.cep, '[^0-9]', '', 'g') as cep, e.ano, e.descricao,
case
    WHEN ee.tipo_fonte_orcamentaria = 'CAMPUS' THEN 'Câmpus'
    WHEN ee.tipo_fonte_orcamentaria = 'PRO_REITORIA_EXTENSAO' THEN
'Pró-reitoria de Extensão'
    WHEN ee.tipo_fonte_orcamentaria = 'PRO_REITORIA_ENSINO' THEN
'Pró-reitoria de Ensino'
end as tipo_fonte_orcamentaria,
ee.valor_financiamento
FROM extensao.edital_extensao ee
INNER JOIN projetos.edital e ON e.id_edital = ee.id_edital
INNER JOIN comum.unidade u ON COALESCE(u.id_unidade, 727) =
COALESCE(ee.id_campus, 727)
INNER JOIN comum.campus_ies ci ON ci.id_campus = u.id_campus_ies
INNER JOIN comum.endereco endereco ON endereco.id_endereco = ci.id_endereco
group by ci.nome, e.ano, e.descricao, endereco.cep, ee.tipo_fonte_orcamentaria,
ee.valor_financiamento
order by ci.nome, e.ano, e.descricao, ee.tipo_fonte_orcamentaria,
ee.valor_financiamento;

```

ANEXO B - LISTA DE SCRIPTS PYTHON COM APACHE AIRFLOW PARA EXTRAÇÃO, TRATAMENTO E CARREGAMENTO DOS DADOS DA EXTENSÃO

DOCENTES EM ATIVIDADES DE EXTENSÃO

```
from airflow import DAG
from airflow.operators.python import PythonOperator
from airflow.providers.postgres.operators.postgres import PostgresOperator
from airflow.providers.smtp.operators.smtp import EmailOperator
from airflow.models import Variable
from airflow.utils.dates import days_ago
from datetime import datetime
import pandas as pd
import numpy as np
import os

# Caminho do arquivo CSV de saída
datalake_extensao_dir = Variable.get("datalake")+ 'extensao/'
datalake_docentes_dir = datalake_extensao_dir+'docentes/'
encoding = "utf-8"
separador = ","
csv_docentes_envolvidos =
datalake_docentes_dir+'IFSC_EXTENSAO_SIGAA_DOCENTES_ENVOLVIDOS.csv'

csv_docentes_voluntarios_envolvidos =
datalake_docentes_dir+'IFSC_EXTENSAO_SIGAA_DOCENTES_VOLUNTARIOS_E
NVOLVIDOS.csv'
csv_docentes_bolsistas_envolvidos =
datalake_docentes_dir+'IFSC_EXTENSAO_SIGAA_DOCENTES_BOLSISTAS_ENV
OLVIDOS.csv'
csv_docentes_coordenadores_envolvidos =
datalake_docentes_dir+'IFSC_EXTENSAO_SIGAA_DOCENTES_COORDENADOR
ES_ENVOLVIDOS.csv'
csv_docentes_orientadores_envolvidos =
datalake_docentes_dir+'IFSC_EXTENSAO_SIGAA_DOCENTES_ORIENTADORES
_ENVOLVIDOS.csv'
csv_docentes_que_submetem_extensao =
datalake_docentes_dir+'IFSC_EXTENSAO_SIGAA_DOCENTES_QUE_SUBMETEM
_EXTENSAO.csv'
#csv_docentes_por_campus =
datalake_docentes_dir+'IFSC_EXTENSAO_SIGAA_DOCENTES_POR_CAMPUS.cs
v'
csv_percentual_docentes_na_extensao =
datalake_docentes_dir+'IFSC_EXTENSAO_SIGAA_PERCENTUAL_DOCENTES.cs
v'
lista_arquivos =
[csv_docentes_envolvidos,csv_docentes_voluntarios_envolvidos,csv_docentes_bolsi
stas_envolvidos,csv_docentes_coordenadores_envolvidos,csv_docentes_orientador
```

```
es_envolvidos, csv_docentes_que_submetem_extensao, csv_percentual_docentes_n  
a_extensao]
```

```
with DAG(  
    dag_id='IFSC_EXTENSAO_SIGAA_DOCENTES',  
    start_date=days_ago(1),  
    schedule_interval='@daily',  
    catchup=False,  
) as dag:
```

```
    # docentes em atividades de extensão com papel de voluntário por campus e  
ano
```

```
    extrai_docentes_voluntarios_envolvidos_extensao = PostgresOperator(  
        task_id='extrai_docentes_voluntarios_envolvidos_extensao',  
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em  
Admin -> Connections  
        sql =  
Variable.get("script_SIGAA_IFSC_consulta_docentes_voluntarios_envolvidos_exten  
sao"), # script definido em Admin -> Variables  
        #do_xcom_push=True,  
    )
```

```
    # docentes em atividades de extensão com papel de bolsista por campus e  
ano
```

```
    extrai_docentes_bolsistas_envolvidos_extensao = PostgresOperator(  
        task_id='extrai_docentes_bolsistas_envolvidos_extensao',  
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em  
Admin -> Connections  
        sql =  
Variable.get("script_SIGAA_IFSC_consulta_quantidade_docentes_bolsistas_envolvi  
dos_extensao"), # script definido em Admin -> Variables  
        #do_xcom_push=True,  
    )
```

```
    # docentes em atividades de extensão com papel de coordenador por campus  
e ano
```

```
    extrai_docentes_coordenadores_envolvidos_extensao = PostgresOperator(  
        task_id='extrai_docentes_coordenadores_envolvidos_extensao',  
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em  
Admin -> Connections  
        sql =  
Variable.get("script_SIGAA_IFSC_consulta_quantidade_docentes_coordenadores_e  
nvolvidos_extensao"), # script definido em Admin -> Variables  
        #do_xcom_push=True,  
    )
```

```
    # docentes em atividades de extensão com papel de orientador por campus e  
ano
```

```
    extrai_docentes_orientadores_envolvidos_extensao = PostgresOperator(  

```

```

        task_id='extrai_docentes_orientadores_envolvidos_extensao',
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
        sql =
Variable.get("script_SIGAA_IFSC_consulta_quantidade_docentes_orientadores_env
olvidos_extensao"), # script definido em Admin -> Variables
        #do_xcom_push=True,
    )

```

quantitativo de docentes que submetem ações de extensão por campus e ano (para o SIGAA todos que submetem ações de extensão estão envolvidos com extensão)

```

        extrai_docentes_que_submetem_extensao = PostgresOperator(
        task_id='extrai_docentes_que_submetem_extensao',
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
        sql =
Variable.get("script_SIGAA_IFSC_consulta_docentes_que_submetem_extensao"), #
script definido em Admin -> Variables
        #do_xcom_push=True,
    )

```

docentes envolvidos em atividades de extensão por campus e ano

```

        extrai_docentes_envolvidos_extensao = PostgresOperator(
        task_id='extrai_docentes_envolvidos_extensao',
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
        sql =
Variable.get("script_SIGAA_IFSC_consulta_quantidade_docentes_envolvidos_exten
sao"), # script definido em Admin -> Variables
        #do_xcom_push=True,
    )

```

quantitativo total de servidores docentes por campus para percentual (não há registro histórico no SIGAA para agrupar com precisão por ano, está consultando o quantitativo do ano corrente - consider docentes temporários e visitantes)

```

        extrai_total_docentes_por_campus = PostgresOperator(
        task_id='extrai_total_docentes_por_campus',
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
        sql =
Variable.get("script_SIGAA_IFSC_consulta_total_docentes_por_campus"), # script
definido em Admin -> Variables
        #do_xcom_push=True,
    )

```

```

def calcula_percentual_docentes_na_extensao(ti) -> pd.DataFrame:
#Obtém o resultado das queries

```

```

        result_total_docentes_por_campus =
ti.xcom_pull(task_ids='extrai_total_docentes_por_campus')
        result_docentes_envolvidos_extensao =
ti.xcom_pull(task_ids='extrai_docentes_envolvidos_extensao')
        #Cria dataframes pandas a partir das queries
        df_result_total_docentes_por_campus =
pd.DataFrame(result_total_docentes_por_campus,
columns=['campus','cep','docentes_campus'])
        df_docentes_envolvidos_extensao =
pd.DataFrame(result_docentes_envolvidos_extensao,
columns=['campus','cep','ano','docentes_extensao'])
        #Mescla os dois dataframes com base e cep preservando
df_docentes_envolvidos_extensao
        df_merged = pd.merge(
df_docentes_envolvidos_extensao,
df_result_total_docentes_por_campus,
on=['campus', 'cep'],
how='left'
).fillna(0)
        #Calcula o percentual de docentes na extensão por campus
df_merged['percentual_docentes_na_extensao'] = (
(df_merged['docentes_extensao'] * 100) /
df_merged['docentes_campus']
).fillna(0)
        # formata o percentual para até 2 casas decimais
df_merged['percentual_docentes_na_extensao'] =
df_merged['percentual_docentes_na_extensao'].round(2)
        # Garante que os anos que não tiveram docentes na extensão tenha como
resultado 0%
df_merged.loc[df_merged['docentes_extensao'] == 0,
'percentual_docentes_na_extensao'] = 0
        # substitui valores infinitos por <0.01
df_merged.replace([np.inf, -np.inf], 0, inplace=True)
        # Cria um novo dataframe apenas com as colunas interessadas
df_resultado_percentual_docentes_na_extensao = df_merged[['campus',
'cep', 'ano', 'percentual_docentes_na_extensao']]
        return df_resultado_percentual_docentes_na_extensao

transforma_percentual_docentes_na_extensao = PythonOperator(
task_id='transforma_percentual_docentes_na_extensao',
python_callable=calcula_percentual_docentes_na_extensao
)

```

```
def cria_diretorio():
```

```
#cria os diretórios se não existirem
```

```
if not os.path.exists(datalake_extensao_dir):
```

```
os.makedirs(datalake_extensao_dir)
```

```
elif not os.path.exists(datalake_docentes_dir):
```

```
os.makedirs(datalake_docentes_dir)
```

```

cria_datalake_docentes_extensao = PythonOperator(
    task_id='cria_datalake_docentes_extensao',
    python_callable=cria_diretorio,
)

def cria_planilha_docentes_voluntarios_envolvidos_extensao(ti):
    #Obtém o resultado da query
    result_docentes_voluntarios_envolvidos_extensao =
ti.xcom_pull(task_ids='extraí_docentes_voluntarios_envolvidos_extensao')
    #Cria um dataframe pandas
    df_docentes_voluntarios =
pd.DataFrame(result_docentes_voluntarios_envolvidos_extensao)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_docentes_voluntarios_extensao = ['Câmpus',
'CEP','Ano','Docentes Voluntários em Ações']
    #Exporta para CSV
    if df_docentes_voluntarios.empty:
        df_docentes_voluntarios =
pd.DataFrame(columns=cabecalho_docentes_voluntarios_extensao)
        df_docentes_voluntarios.to_csv(csv_docentes_voluntarios_envolvidos,
index=False, encoding=encoding, sep=separador)
    else:
        df_docentes_voluntarios.to_csv(csv_docentes_voluntarios_envolvidos,
index=False, header=cabecalho_docentes_voluntarios_extensao,
encoding=encoding, sep=separador)

carrega_docentes_voluntarios_envolvidos_extensao = PythonOperator(
    task_id='carrega_docentes_voluntarios_envolvidos_extensao',
    python_callable=cria_planilha_docentes_voluntarios_envolvidos_extensao
)

```

```

def cria_planilha_docentes_bolsistas_envolvidos_extensao(ti):
    #Obtém o resultado da query
    result_docentes_bolsistas_envolvidos_extensao =
ti.xcom_pull(task_ids='extraí_docentes_bolsistas_envolvidos_extensao')
    #Cria um dataframe pandas
    df_docentes_bolsistas =
pd.DataFrame(result_docentes_bolsistas_envolvidos_extensao)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_docentes_bolsistas_extensao = ['Câmpus', 'CEP','Ano','Docentes
Bolsistas em Ações']
    #Exporta para CSV
    if df_docentes_bolsistas.empty:
        df_docentes_bolsistas =
pd.DataFrame(columns=cabecalho_docentes_bolsistas_extensao)
        df_docentes_bolsistas.to_csv(csv_docentes_bolsistas_envolvidos,
index=False, encoding=encoding, sep=separador)
    else:

```

```
df_docentes_bolsistas.to_csv(csv_docentes_bolsistas_envolvidos,
index=False, header=cabecalho_docentes_bolsistas_extensao, encoding=encoding,
sep=separador)
```

```
carrega_docentes_bolsistas_envolvidos_extensao = PythonOperator(
task_id='carrega_docentes_bolsistas_envolvidos_extensao',
python_callable=cria_planilha_docentes_bolsistas_envolvidos_extensao
)
```

```
def cria_planilha_docentes_coordenadores_envolvidos_extensao(ti):
#Obtém o resultado da query
result_docentes_coordenadores_envolvidos_extensao =
ti.xcom_pull(task_ids='extraí_docentes_coordenadores_envolvidos_extensao')
#Cria um dataframe pandas
df_docentes_coordenadores =
pd.DataFrame(result_docentes_coordenadores_envolvidos_extensao)
#Mapea o cabeçalho do arquivo CSV
cabecalho_docentes_coordenadores_extensao = ['Câmpus',
'CEP','Ano','Docentes Coordenadores em Ações']
#Exporta para CSV
if df_docentes_coordenadores.empty:
df_docentes_coordenadores =
pd.DataFrame(columns=cabecalho_docentes_coordenadores_extensao)
```

```
df_docentes_coordenadores.to_csv(csv_docentes_coordenadores_envolvidos,
index=False, encoding=encoding, sep=separador)
else:
```

```
df_docentes_coordenadores.to_csv(csv_docentes_coordenadores_envolvidos,
index=False, header=cabecalho_docentes_coordenadores_extensao,
encoding=encoding, sep=separador)
```

```
carrega_docentes_coordenadores_envolvidos_extensao = PythonOperator(
task_id='carrega_docentes_coordenadores_envolvidos_extensao',

python_callable=cria_planilha_docentes_coordenadores_envolvidos_extensao
)
```

```
def cria_planilha_docentes_orientadores_envolvidos_extensao(ti):
#Obtém o resultado da query
result_docentes_orientadores_envolvidos_extensao =
ti.xcom_pull(task_ids='extraí_docentes_orientadores_envolvidos_extensao')
#Cria um dataframe pandas
df_docentes_orientadores =
pd.DataFrame(result_docentes_orientadores_envolvidos_extensao)
#Mapea o cabeçalho do arquivo CSV
cabecalho_docentes_orientadores_extensao = ['Câmpus',
'CEP','Ano','Docentes Orientadores em Ações']
#Exporta para CSV
```



```

        if df_docentes_orientadores.empty:
            df_docentes_orientadores =
pd.DataFrame(columns=cabecalho_docentes_orientadores_extensao)
            df_docentes_orientadores.to_csv(csv_docentes_orientadores_envolvidos,
index=False, encoding=encoding, sep=separador)
        else:
            df_docentes_orientadores.to_csv(csv_docentes_orientadores_envolvidos,
index=False, header=cabecalho_docentes_orientadores_extensao,
encoding=encoding, sep=separador)

carrega_docentes_orientadores_envolvidos_extensao = PythonOperator(
    task_id='carrega_docentes_orientadores_envolvidos_extensao',
    python_callable=cria_planilha_docentes_orientadores_envolvidos_extensao
)

```

```

def cria_planilha_docentes_que_submetem_extensao(ti):
    #Obtém o resultado da query
    result_docentes_que_submetem_extensao =
ti.xcom_pull(task_ids='extrai_docentes_que_submetem_extensao')
    #Cria um dataframe pandas
    df_docentes_que_submetem =
pd.DataFrame(result_docentes_que_submetem_extensao)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_docentes_que_submetem_extensao = ['Câmpus',
'CEP','Ano','Docentes que Submetem Ações']
    #Exporta para CSV
    if df_docentes_que_submetem.empty:
        df_docentes_que_submetem =
pd.DataFrame(columns=cabecalho_docentes_que_submetem_extensao)

    df_docentes_que_submetem.to_csv(csv_docentes_que_submetem_extensao,
index=False, encoding=encoding, sep=separador)
    else:

df_docentes_que_submetem.to_csv(csv_docentes_que_submetem_extensao,
index=False, header=cabecalho_docentes_que_submetem_extensao,
encoding=encoding, sep=separador)

```

```

carrega_docentes_que_submetem_extensao = PythonOperator(
    task_id='carrega_docentes_que_submetem_extensao',
    python_callable=cria_planilha_docentes_que_submetem_extensao
)

```

```

def cria_planilha_docentes_envolvidos_extensao(ti):
    #Obtém o resultado da query
    result_docentes_envolvidos_extensao =
ti.xcom_pull(task_ids='extrai_docentes_envolvidos_extensao')
    #Cria um dataframe pandas

```



```

df_docentes_envolvidos =
pd.DataFrame(result_docentes_envolvidos_extensao)
#Mapea o cabeçalho do arquivo CSV
cabecalho_docentes_envolvidos_extensao = ['Câmpus', 'CEP', 'Ano', 'Docentes
Envolvidos em Ações']
#Exporta para CSV
if df_docentes_envolvidos.empty:
df_docentes_envolvidos =
pd.DataFrame(columns=cabecalho_docentes_envolvidos_extensao)
df_docentes_envolvidos.to_csv(csv_docentes_envolvidos, index=False,
encoding=encoding, sep=separador)
else:
df_docentes_envolvidos.to_csv(csv_docentes_envolvidos, index=False,
header=cabecalho_docentes_envolvidos_extensao, encoding=encoding,
sep=separador)

```

```

carrega_docentes_envolvidos_extensao = PythonOperator(
task_id='carrega_docentes_envolvidos_extensao',
python_callable=cria_planilha_docentes_envolvidos_extensao
)

```

```

""" def cria_planilha_total_docentes_por_campus(ti):
#Obtém o resultado da query
result_total_docentes_por_campus =
ti.xcom_pull(task_ids='extraí_total_docentes_por_campus')
#Cria um dataframe pandas
df_docentes_por_campus =
pd.DataFrame(result_total_docentes_por_campus)
#Mapea o cabeçalho do arquivo CSV
cabecalho_docentes_por_campus = ['Câmpus', 'CEP', 'Docentes']
#Exporta para CSV
if df_docentes_por_campus.empty:
df_docentes_por_campus =
pd.DataFrame(columns=cabecalho_docentes_por_campus)
df_docentes_por_campus.to_csv(csv_docentes_por_campus, index=False,
encoding=encoding, sep=separador)
else:
df_docentes_por_campus.to_csv(csv_docentes_por_campus, index=False,
header=cabecalho_docentes_por_campus, encoding=encoding, sep=separador)

```

```

carrega_total_docentes_por_campus = PythonOperator(
task_id='carrega_total_docentes_por_campus',
python_callable=cria_planilha_total_docentes_por_campus
) """

```

```

def cria_planilha_percentual_docentes_por_campus(ti):
#Obtém o resultado da query

```

```

        result_percentual_docentes_na_extensao =
ti.xcom_pull(task_ids='transforma_percentual_docentes_na_extensao')
        #Cria um dataframe pandas
        df_percentual_docentes_na_extensao =
pd.DataFrame(result_percentual_docentes_na_extensao)
        #Mapea o cabeçalho do arquivo CSV
        cabecalho_percentual_docentes_na_extensao =
['Câmpus','CEP','Ano','Docentes na Extensão(%)']
        #Exporta para CSV
        if df_percentual_docentes_na_extensao.empty:
            df_percentual_docentes_na_extensao =
pd.DataFrame(columns=cabecalho_percentual_docentes_na_extensao)

df_percentual_docentes_na_extensao.to_csv(csv_percentual_docentes_na_extensao, index=False, encoding=encoding, sep=separador)
        else:

df_percentual_docentes_na_extensao.to_csv(csv_percentual_docentes_na_extensao, index=False, header=cabecalho_percentual_docentes_na_extensao, encoding=encoding, sep=separador)

        carrega_percentual_docentes_por_campus = PythonOperator(
            task_id='carrega_percentual_docentes_por_campus',
            python_callable=cria_planilha_percentual_docentes_por_campus,
        )

        """ envia_arquivos_email = EmailOperator(
            task_id='envia_arquivos_email',
            to=Variable.get("IFSC_email_extensao"),# email definido em Admin ->
Variables
            subject='Indicadores de Extensão - Docentes',
            html_content="Segue em anexo os arquivos",
            files=lista_arquivos,
            conn_id="IFSC_SMTP", # Conexão SMTP definida em Admin -> Connections
        ) """

        # Definir as dependências entre as tasks
        [extrai_docentes_voluntarios_envolvidos_extensao,
        extrai_docentes_bolsistas_envolvidos_extensao,
        extrai_docentes_coordenadores_envolvidos_extensao,
        extrai_docentes_orientadores_envolvidos_extensao,
        extrai_docentes_que_submetem_extensao,
        extrai_docentes_envolvidos_extensao,
        extrai_total_docentes_por_campus] >>
transforma_percentual_docentes_na_extensao >>
cria_datalake_docentes_extensao >>
[carrega_docentes_voluntarios_envolvidos_extensao,

carrega_docentes_bolsistas_envolvidos_extensao,

```

carrega_docentes_coordenadores_envolvidos_extensao,
carrega_docentes_orientadores_envolvidos_extensao,
carrega_docentes_que_submetem_extensao,
carrega_docentes_envolvidos_extensao,
#carrega_total_docentes_por_campus,
carrega_percentual_docentes_por_campus]

TAES EM ATIVIDADES DE EXTENSÃO

```
from airflow import DAG
from airflow.operators.python import PythonOperator
from airflow.providers.postgres.operators.postgres import PostgresOperator
from airflow.providers.smtp.operators.smtp import EmailOperator
from airflow.models import Variable
from airflow.utils.dates import days_ago
from datetime import datetime
import pandas as pd
import numpy as np
import os

# Caminho do arquivo CSV de saída
datalake_extensao_dir = Variable.get("datalake")+'extensao/'
datalake_taes_dir = datalake_extensao_dir+'taes/'
encoding = "utf-8"
separador = ","
csv_taes_envolvidos =
datalake_taes_dir+'IFSC_EXTENSAO_SIGAA_TAES_ENVOLVIDOS.csv'
csv_taes_voluntarios_envolvidos =
datalake_taes_dir+'IFSC_EXTENSAO_SIGAA_TAES_VOLUNTARIOS_ENVOLVIDO
S.csv'
csv_taes_bolsistas_envolvidos =
datalake_taes_dir+'IFSC_EXTENSAO_SIGAA_TAES_BOLSISTAS_ENVOLVIDOS.c
sv'
csv_taes_coordenadores_envolvidos =
datalake_taes_dir+'IFSC_EXTENSAO_SIGAA_TAES_COORDENADORES_ENVOL
VIDOS.csv'
#csv_taes_por_campus =
datalake_taes_dir+'IFSC_EXTENSAO_SIGAA_TAES_POR_CAMPUS.csv'
csv_percentual_taes_na_extensao =
datalake_taes_dir+'IFSC_EXTENSAO_SIGAA_PERCENTUAL_TAES.csv'
lista_arquivos =
[csv_taes_envolvidos,csv_taes_voluntarios_envolvidos,csv_taes_bolsistas_envolvid
os,csv_taes_coordenadores_envolvidos, csv_percentual_taes_na_extensao]

with DAG(
    dag_id='IFSC_EXTENSAO_SIGAA_TAES',
    start_date=days_ago(1),
    schedule_interval='@daily',
    catchup=False,
) as dag:

    # taes em atividades de extensão com papel de voluntário por campus e ano
    extra_taes_voluntarios_envolvidos_extensao = PostgresOperator(
        task_id='extra_taes_voluntarios_envolvidos_extensao',
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
```

```

    sql =
Variable.get("script_SIGAA_IFSC_consulta_taes_voluntarios_envolvidos_extensao"),
# script definido em Admin -> Variables
    #do_xcom_push=True,
)

# taes em atividades de extensão com papel de bolsista por campus e ano
extrai_taes_bolsistas_envolvidos_extensao = PostgresOperator(
    task_id='extrai_taes_bolsistas_envolvidos_extensao',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql =
Variable.get("script_SIGAA_IFSC_consulta_quantidade_taes_bolsistas_envolvidos_extensao"), # script definido em Admin -> Variables
    #do_xcom_push=True,
)

# taes em atividades de extensão com papel de coordenador por campus e
ano
extrai_taes_coordenadores_envolvidos_extensao = PostgresOperator(
    task_id='extrai_taes_coordenadores_envolvidos_extensao',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql =
Variable.get("script_SIGAA_IFSC_consulta_quantidade_taes_coordenadores_envolv
idos_extensao"), # script definido em Admin -> Variables
    #do_xcom_push=True,
)

# taes envolvidos em atividades de extensão por campus e ano
extrai_taes_envolvidos_extensao = PostgresOperator(
    task_id='extrai_taes_envolvidos_extensao',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql =
Variable.get("script_SIGAA_IFSC_consulta_quantidade_taes_envolvidos_extensao")
, # script definido em Admin -> Variables
    #do_xcom_push=True,
)

# quantitativo total de servidores taes por campus para percentual (não há
registro histórico no SIGAA para agrupar com precisão por ano, está consultando o
quantitativo do ano corrente)
extrai_total_taes_por_campus = PostgresOperator(
    task_id='extrai_total_taes_por_campus',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql = Variable.get("script_SIGAA_IFSC_consulta_total_taes_por_campus"), #
script definido em Admin -> Variables

```

```

#do_xcom_push=True,
)

def calcula_percentual_taes_na_extensao(ti) -> pd.DataFrame:
    #Obtém o resultado das queries
    result_total_taes_por_campus =
    ti.xcom_pull(task_ids='extrai_total_taes_por_campus')
    result_taes_envolvidos_extensao =
    ti.xcom_pull(task_ids='extrai_taes_envolvidos_extensao')
    #Cria dataframes pandas a partir das queries
    df_result_total_taes_por_campus =
    pd.DataFrame(result_total_taes_por_campus,
    columns=['campus','cep','taes_campus'])
    df_taes_envolvidos_extensao =
    pd.DataFrame(result_taes_envolvidos_extensao,
    columns=['campus','cep','ano','taes_extensao'])
    #Mescla os dois dataframes com base e cep preservando
    df_taes_envolvidos_extensao
    df_merged = pd.merge(
    df_taes_envolvidos_extensao,
    df_result_total_taes_por_campus,
    on=['campus', 'cep'],
    how='left'
    ).fillna(0)
    #Calcula o percentual de TAEs na extensão por campus
    df_merged['percentual_taes_na_extensao'] = (
    (df_merged['taes_extensao'] * 100) / df_merged['taes_campus']
    ).fillna(0)
    # formata o percentual para até 2 casas decimais
    df_merged['percentual_taes_na_extensao'] =
    df_merged['percentual_taes_na_extensao'].round(2)
    # Garante que os anos que não tiveram TAES na extensão tenha como
    resultado 0%
    df_merged.loc[df_merged['taes_extensao'] == 0,
    'percentual_taes_na_extensao'] = 0
    # substitui valores infinitos por <0.01
    df_merged.replace([np.inf, -np.inf], 0, inplace=True)
    # Cria um novo dataframe apenas com as colunas interessadas
    df_resultado_percentual_taes_na_extensao = df_merged[['campus', 'cep',
    'ano', 'percentual_taes_na_extensao']]
    return df_resultado_percentual_taes_na_extensao

transforma_percentual_taes_na_extensao = PythonOperator(
    task_id='transforma_percentual_taes_na_extensao',
    python_callable=calcula_percentual_taes_na_extensao
)

def cria_diretorio():
    #cria os diretórios se não existirem

```

```

if not os.path.exists(datalake_extensao_dir):
    os.makedirs(datalake_extensao_dir)
elif not os.path.exists(datalake_taes_dir):
    os.makedirs(datalake_taes_dir)

cria_datalake_taes_extensao = PythonOperator(
    task_id='cria_datalake_taes_extensao',
    python_callable=cria_diretorio,
)

def cria_planilha_taes_voluntarios_envolvidos_extensao(ti):
    #Obtém o resultado da query
    result_taes_voluntarios_envolvidos_extensao =
ti.xcom_pull(task_ids='extrai_taes_voluntarios_envolvidos_extensao')
    #Cria um dataframe pandas
    df_taes_voluntarios =
pd.DataFrame(result_taes_voluntarios_envolvidos_extensao)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_taes_voluntarios_extensao = ['Câmpus', 'CEP', 'Ano', 'TAEs
Voluntários']
    #Exporta para CSV
    if df_taes_voluntarios.empty:
        df_taes_voluntarios =
pd.DataFrame(columns=cabecalho_taes_voluntarios_extensao)
        df_taes_voluntarios.to_csv(csv_taes_voluntarios_envolvidos, index=False,
encoding=encoding, sep=separador)
    else:
        df_taes_voluntarios.to_csv(csv_taes_voluntarios_envolvidos, index=False,
header=cabecalho_taes_voluntarios_extensao, encoding=encoding, sep=separador)

carrega_taes_voluntarios_envolvidos_extensao = PythonOperator(
    task_id='carrega_taes_voluntarios_envolvidos_extensao',
    python_callable=cria_planilha_taes_voluntarios_envolvidos_extensao,
)

def cria_planilha_taes_bolsistas_envolvidos_extensao(ti):
    #Obtém o resultado da query
    result_taes_bolsistas_envolvidos_extensao =
ti.xcom_pull(task_ids='extrai_taes_bolsistas_envolvidos_extensao')
    #Cria um dataframe pandas
    df_taes_bolsistas =
pd.DataFrame(result_taes_bolsistas_envolvidos_extensao)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_taes_bolsistas_extensao = ['Câmpus', 'CEP', 'Ano', 'TAEs Bolsistas']
    #Exporta para CSV
    if df_taes_bolsistas.empty:
        df_taes_bolsistas =
pd.DataFrame(columns=cabecalho_taes_bolsistas_extensao)

```

```

df_taes_bolsistas.to_csv(csv_taes_bolsistas_envolvidos, index=False,
encoding=encoding, sep=separador)
else:
df_taes_bolsistas.to_csv(csv_taes_bolsistas_envolvidos, index=False,
header=cabecalho_taes_bolsistas_extensao, encoding=encoding, sep=separador)

```

```

carrega_taes_bolsistas_envolvidos_extensao = PythonOperator(
task_id='carrega_taes_bolsistas_envolvidos_extensao',
python_callable=cria_planilha_taes_bolsistas_envolvidos_extensao,
)

```

```

def cria_planilha_taes_coordenadores_envolvidos_extensao(ti):
#Obtém o resultado da query
result_taes_coordenadores_envolvidos_extensao =
ti.xcom_pull(task_ids='extrai_taes_coordenadores_envolvidos_extensao')
#Cria um dataframe pandas
df_taes_coordenadores =
pd.DataFrame(result_taes_coordenadores_envolvidos_extensao)
#Mapea o cabeçalho do arquivo CSV
cabecalho_taes_coordenadores_extensao = ['Câmpus', 'CEP','Ano','TAEs
Coordenadores']
#Exporta para CSV
if df_taes_coordenadores.empty:
df_taes_coordenadores =
pd.DataFrame(columns=cabecalho_taes_coordenadores_extensao)
df_taes_coordenadores.to_csv(csv_taes_coordenadores_envolvidos,
index=False, encoding=encoding, sep=separador)
else:
df_taes_coordenadores.to_csv(csv_taes_coordenadores_envolvidos,
index=False, header=cabecalho_taes_coordenadores_extensao,
encoding=encoding, sep=separador)

```

```

carrega_taes_coordenadores_envolvidos_extensao = PythonOperator(
task_id='carrega_taes_coordenadores_envolvidos_extensao',
python_callable=cria_planilha_taes_coordenadores_envolvidos_extensao,
)

```

```

def cria_planilha_taes_envolvidos_extensao(ti):
#Obtém o resultado da query
result_taes_envolvidos_extensao =
ti.xcom_pull(task_ids='extrai_taes_envolvidos_extensao')
#Cria um dataframe pandas
df_taes_envolvidos = pd.DataFrame(result_taes_envolvidos_extensao)
#Mapea o cabeçalho do arquivo CSV
cabecalho_taes_envolvidos_extensao = ['Câmpus', 'CEP','Ano','TAEs
Envolvidos na Extensão']
#Exporta para CSV
if df_taes_envolvidos.empty:

```



```

df_taes_envolvidos =
pd.DataFrame(columns=cabecalho_taes_envolvidos_extensao)
df_taes_envolvidos.to_csv(csv_taes_envolvidos, index=False,
encoding=encoding, sep=separador)
else:
df_taes_envolvidos.to_csv(csv_taes_envolvidos, index=False,
header=cabecalho_taes_envolvidos_extensao, encoding=encoding, sep=separador)

```

```

carrega_taes_envolvidos_extensao = PythonOperator(
task_id='carrega_taes_envolvidos_extensao',
python_callable=cria_planilha_taes_envolvidos_extensao,
)

```

```

""" def cria_planilha_total_taes_por_campus(ti):
#Obtém o resultado da query
result_total_taes_por_campus =
ti.xcom_pull(task_ids='extrai_total_taes_por_campus')
#Cria um dataframe pandas
df_taes_por_campus = pd.DataFrame(result_total_taes_por_campus)
#Mapea o cabeçalho do arquivo CSV
cabecalho_taes_por_campus = ['Câmpus', 'CEP','TAEs']
#Exporta para CSV
if df_taes_por_campus.empty:
df_taes_por_campus =
pd.DataFrame(columns=cabecalho_taes_por_campus)
df_taes_por_campus.to_csv(csv_taes_por_campus, index=False,
encoding=encoding, sep=separador)
else:
df_taes_por_campus.to_csv(csv_taes_por_campus, index=False,
header=cabecalho_taes_por_campus, encoding=encoding, sep=separador)

```

```

carrega_total_taes_por_campus = PythonOperator(
task_id='carrega_total_taes_por_campus',
python_callable=cria_planilha_total_taes_por_campus,
) """

```

```

def cria_planilha_percentual_taes_por_campus(ti):
#Obtém o resultado da query
result_percentual_taes_na_extensao =
ti.xcom_pull(task_ids='transforma_percentual_taes_na_extensao')
#Cria um dataframe pandas
df_percentual_taes_na_extensao =
pd.DataFrame(result_percentual_taes_na_extensao)
#Mapea o cabeçalho do arquivo CSV
cabecalho_percentual_taes_na_extensao = ['Câmpus','CEP','Ano','TAEs na
Extensão(%)']
#Exporta para CSV
if df_percentual_taes_na_extensao.empty:

```

```

df_percentual_taes_na_extensao =
pd.DataFrame(columns=cabecalho_percentual_taes_na_extensao)
df_percentual_taes_na_extensao.to_csv(csv_percentual_taes_na_extensao,
index=False, encoding=encoding, sep=separador)
else:
df_percentual_taes_na_extensao.to_csv(csv_percentual_taes_na_extensao,
index=False, header=cabecalho_percentual_taes_na_extensao, encoding=encoding,
sep=separador)

```

```

carrega_percentual_taes_por_campus = PythonOperator(
task_id='carrega_percentual_taes_por_campus',
python_callable=cria_planilha_percentual_taes_por_campus,
)

```

```

""" envia_arquivos_email = EmailOperator(
task_id='envia_arquivos_email',
to=Variable.get("IFSC_email_extensao"),# email definido em Admin ->

```

Variables

```

subject='Indicadores de Extensão - TAEs',
html_content="Segue em anexo os arquivos",
files=lista_arquivos,
conn_id="IFSC_SMTP", # Conexão SMTP definida em Admin -> Connections
) """

```

Definir as dependências entre as tasks

```

[extrai_taes_voluntarios_envolvidos_extensao,
extrai_taes_bolsistas_envolvidos_extensao,
extrai_taes_coordenadores_envolvidos_extensao,
extrai_taes_envolvidos_extensao,
extrai_total_taes_por_campus] >> transforma_percentual_taes_na_extensao
>> cria_datalake_taes_extensao >>
[carrega_taes_voluntarios_envolvidos_extensao,
carrega_taes_bolsistas_envolvidos_extensao,

```

```

carrega_taes_coordenadores_envolvidos_extensao,
#carrega_total_taes_por_campus,
carrega_taes_envolvidos_extensao,
carrega_percentual_taes_por_campus]

```

DISCENTES EM ATIVIDADES DE EXTENSÃO

```
from airflow import DAG
from airflow.operators.python import PythonOperator
from airflow.providers.postgres.operators.postgres import PostgresOperator
from airflow.providers.smtp.operators.smtp import EmailOperator
from airflow.models import Variable
from airflow.utils.dates import days_ago
from datetime import datetime
import pandas as pd
import numpy as np
import os

# Caminho do arquivo CSV de saída
datalake_extensao_dir = Variable.get("datalake")+ 'extensao/'
datalake_discentes_dir = datalake_extensao_dir+ 'discentes/'
encoding = "utf-8"
separador = ","
csv_discentes_envolvidos =
datalake_discentes_dir+'IFSC_EXTENSAO_SIGAA_DISCENTES_ENVOLVIDOS.csv'
csv_discentes_voluntarios_envolvidos =
datalake_discentes_dir+'IFSC_EXTENSAO_SIGAA_DISCENTES_VOLUNTARIOS_ENVOLVIDOS.csv'
csv_discentes_bolsistas_envolvidos =
datalake_discentes_dir+'IFSC_EXTENSAO_SIGAA_DISCENTES_BOLSISTAS_ENVOLVIDOS.csv'
csv_discentes_alem_creditos_obrigatorios =
datalake_discentes_dir+'IFSC_EXTENSAO_SIGAA_DISCENTES_ENVOLVIDOS_ALEM_CREDITOS_OBRIGATORIOS.csv'
#csv_discentes_por_campus =
datalake_discentes_dir+'IFSC_EXTENSAO_SIGAA_DISCENTES_POR_CAMPUS.csv'
csv_percentual_discentes_na_extensao =
datalake_discentes_dir+'IFSC_EXTENSAO_SIGAA_PERCENTUAL_DISCENTES.csv'
lista_arquivos =
[csv_discentes_envolvidos,csv_discentes_voluntarios_envolvidos,csv_discentes_bolsistas_envolvidos,csv_discentes_alem_creditos_obrigatorios,csv_percentual_discentes_na_extensao]

with DAG(
    dag_id='IFSC_EXTENSAO_SIGAA_DISCENTES',
    start_date=days_ago(1),
    schedule_interval='@daily',
    catchup=False,
) as dag:
```

```

# discentes em atividades de extensão com papel de voluntário por campus,
ano e tipo de curso
extraí_discentes_voluntarios_envolvidos_extensao = PostgresOperator(
    task_id='extraí_discentes_voluntarios_envolvidos_extensao',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql =
Variable.get("script_SIGAA_IFSC_consulta_discentes_voluntarios_envolvidos_exten
sao"), # script definido em Admin -> Variables
    #do_xcom_push=True,
)

```

```

# discentes em atividades de extensão com papel de bolsista por campus,
ano e tipo de curso
extraí_discentes_bolsistas_envolvidos_extensao = PostgresOperator(
    task_id='extraí_discentes_bolsistas_envolvidos_extensao',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql =
Variable.get("script_SIGAA_IFSC_consulta_quantidade_discentes_bolsistas_envolvi
dos_extensao"), # script definido em Admin -> Variables
    #do_xcom_push=True,
)

```

```

# discentes em atividades de extensão por campus, ano e tipo de curso
extraí_discentes_envolvidos_extensao = PostgresOperator(
    task_id='extraí_discentes_envolvidos_extensao',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql =
Variable.get("script_SIGAA_IFSC_consulta_quantidade_discentes_envolvidos_exten
sao"), # script definido em Admin -> Variables
    #do_xcom_push=True,
)

```

```

# quantitativo de discentes envolvidos em atividades de extensão além dos
créditos obrigatórios por campus, ano e tipo de curso
extraí_quantidade_discentes_envolvidos_alem_creditos_obrigatorios =
PostgresOperator(

task_id='extraí_quantidade_discentes_envolvidos_alem_creditos_obrigatorios',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql =
Variable.get("script_SIGAA_IFSC_consulta_quantidade_discentes_envolvidos_alem
_creditos_obrigatorios"), # script definido em Admin -> Variables
    #do_xcom_push=True,
)

```

```

# quantitativo total de discentes por campus, ano de ingresso e tipo de curso
para percentual
    extrai_total_discentes_por_campus = PostgresOperator(
        task_id='extrai_total_discentes_por_campus',
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
        sql =
Variable.get("script_SIGAA_IFSC_consulta_total_discentes_por_campus"), # script
definido em Admin -> Variables
        #do_xcom_push=True,
    )

```

```

def calcula_percentual_discentes_na_extensao(ti) -> pd.DataFrame:
    #Obtém o resultado das queries
    result_total_discentes_por_campus =
ti.xcom_pull(task_ids='extrai_total_discentes_por_campus')
    result_discentes_envolvidos_extensao =
ti.xcom_pull(task_ids='extrai_discentes_envolvidos_extensao')
    #Cria dataframes pandas a partir das queries
    df_result_total_discentes_por_campus =
pd.DataFrame(result_total_discentes_por_campus,
columns=['campus','cep','ano','tipo_curso','discentes_campus'])
    df_discentes_envolvidos_extensao =
pd.DataFrame(result_discentes_envolvidos_extensao,
columns=['campus','cep','ano','tipo_curso','discentes_extensao'])
    #Mescla os dois dataframes com base e cep preservando
df_discentes_envolvidos_extensao
    df_merged = pd.merge(
        df_discentes_envolvidos_extensao,
        df_result_total_discentes_por_campus,
        on=['campus','cep','ano','tipo_curso'],
        how='left'
    ).fillna(0)
    #Calcula o percentual de discentes na extensão por campus
    df_merged['percentual_discentes_na_extensao'] = (
        (df_merged['discentes_extensao'] * 100) /
df_merged['discentes_campus']
    ).fillna(0)
    # formata o percentual para até 2 casas decimais
    df_merged['percentual_discentes_na_extensao'] =
df_merged['percentual_discentes_na_extensao'].round(2)
    # Garante que os anos que não tiveram discentes na extensão tenha como
resultado 0%
    df_merged.loc[df_merged['discentes_extensao'] == 0,
'percentual_discentes_na_extensao'] = 0
    # substitui valores infinitos por <0.01
    df_merged.replace([np.inf, -np.inf], 0, inplace=True)
    # Cria um novo dataframe apenas com as colunas interessadas

```

```

df_resultado_percentual_discentes_na_extensao =
df_merged[['campus','cep','ano','tipo_curso','percentual_discentes_na_extensao']]
return df_resultado_percentual_discentes_na_extensao

```

```

transforma_percentual_discentes_na_extensao = PythonOperator(
task_id='transforma_percentual_discentes_na_extensao',
python_callable=calcula_percentual_discentes_na_extensao
)

```

```

def cria_diretorio():
#cria os diretórios se não existirem
if not os.path.exists(datalake_extensao_dir):
os.makedirs(datalake_extensao_dir)
elif not os.path.exists(datalake_discentes_dir):
os.makedirs(datalake_discentes_dir)

```

```

cria_datalake_discentes_extensao = PythonOperator(
task_id='cria_datalake_discentes_extensao',
python_callable=cria_diretorio,
)

```

```

def cria_planilha_discentes_voluntarios_envolvidos_extensao(ti):
#Obtém o resultado da query
result_discentes_voluntarios_envolvidos_extensao =
ti.xcom_pull(task_ids='extrai_discentes_voluntarios_envolvidos_extensao')
#Cria um dataframe pandas
df_discentes_voluntarios =
pd.DataFrame(result_discentes_voluntarios_envolvidos_extensao)
#Mapea o cabeçalho do arquivo CSV
cabecalho_discentes_voluntarios_extensao = ['Câmpus', 'CEP','Ano','Tipo de
Curso','Discentes Voluntários']
#Exporta para CSV
if df_discentes_voluntarios.empty:
df_discentes_voluntarios =
pd.DataFrame(columns=cabecalho_discentes_voluntarios_extensao)
df_discentes_voluntarios.to_csv(csv_discentes_voluntarios_envolvidos,
index=False, encoding=encoding, sep=separador)
else:
df_discentes_voluntarios.to_csv(csv_discentes_voluntarios_envolvidos,
index=False, header=cabecalho_discentes_voluntarios_extensao,
encoding=encoding, sep=separador)

```

```

carrega_discentes_voluntarios_envolvidos_extensao = PythonOperator(
task_id='carrega_discentes_voluntarios_envolvidos_extensao',
python_callable=cria_planilha_discentes_voluntarios_envolvidos_extensao
)

```

```

def cria_planilha_discentes_bolsistas_envolvidos_extensao(ti):
#Obtém o resultado da query

```

```

        result_discentes_bolsistas_envolvidos_extensao =
ti.xcom_pull(task_ids='extrai_discentes_bolsistas_envolvidos_extensao')
        #Cria um dataframe pandas
        df_discentes_bolsistas =
pd.DataFrame(result_discentes_bolsistas_envolvidos_extensao)
        #Mapea o cabeçalho do arquivo CSV
        cabecalho_discentes_bolsistas_extensao = ['Câmpus', 'CEP','Ano','Tipo de
Curso','Discentes Bolsistas']
        #Exporta para CSV
        if df_discentes_bolsistas.empty:
            df_discentes_bolsistas =
pd.DataFrame(columns=cabecalho_discentes_bolsistas_extensao)
            df_discentes_bolsistas.to_csv(csv_discentes_bolsistas_envolvidos,
index=False, sep=separador)
        else:
            df_discentes_bolsistas.to_csv(csv_discentes_bolsistas_envolvidos,
index=False, header=cabecalho_discentes_bolsistas_extensao, encoding=encoding,
sep=separador)

        carrega_discentes_bolsistas_envolvidos_extensao = PythonOperator(
            task_id='carrega_discentes_bolsistas_envolvidos_extensao',
            python_callable=cria_planilha_discentes_bolsistas_envolvidos_extensao
        )

```

```

def
cria_planilha_quantidade_discentes_envolvidos_alem_creditos_obrigatorios(ti)
:
    #Obtém o resultado da query
    result_quantidade_discentes_envolvidos_alem_creditos_obrigatorios =
ti.xcom_pull(task_ids='extrai_quantidade_discentes_envolvidos_alem_creditos_obrig
atorios')
    #Cria um dataframe pandas
    df_discentes_alem_creditos_obrigatorios =
pd.DataFrame(result_quantidade_discentes_envolvidos_alem_creditos_obrigatorios)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_discentes_alem_creditos_obrigatorios = ['Câmpus',
'CEP','Ano','Tipo de Curso','Discentes em Ações de Extensão além dos Créditos
Obrigatórios']
    #Exporta para CSV
    if df_discentes_alem_creditos_obrigatorios.empty:
        df_discentes_alem_creditos_obrigatorios =
pd.DataFrame(columns=cabecalho_discentes_alem_creditos_obrigatorios)

    df_discentes_alem_creditos_obrigatorios.to_csv(csv_discentes_alem_creditos_obrig
atorios, index=False, encoding=encoding, sep=separador)
    else:

        df_discentes_alem_creditos_obrigatorios.to_csv(csv_discentes_alem_creditos_obrig

```

```
atorios, index=False, header=cabecalho_discentes_alem_creditos_obrigatorios,
encoding=encoding, sep=separador)
```

```
carrega_quantidade_discentes_envolvidos_alem_creditos_obrigatorios =
PythonOperator(
```

```
task_id='carrega_quantidade_discentes_envolvidos_alem_creditos_obrigatorios',
```

```
python_callable=cria_planilha_quantidade_discentes_envolvidos_alem_creditos_obri
gatorios
)
```

```
def cria_planilha_discentes_envolvidos_extensao(ti):
    #Obtém o resultado da query
    result_discentes_envolvidos_extensao =
ti.xcom_pull(task_ids='extrai_discentes_envolvidos_extensao')
    #Cria um dataframe pandas
    df_discentes_envolvidos =
pd.DataFrame(result_discentes_envolvidos_extensao)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_discentes_envolvidos_extensao = ['Câmpus', 'CEP', 'Ano', 'Tipo de
Curso', 'Discentes em Ações']
    #Exporta para CSV
    if df_discentes_envolvidos.empty:
        df_discentes_envolvidos =
pd.DataFrame(columns=cabecalho_discentes_envolvidos_extensao)
        df_discentes_envolvidos.to_csv(csv_discentes_envolvidos, index=False,
sep=separador)
    else:
        df_discentes_envolvidos.to_csv(csv_discentes_envolvidos, index=False,
header=cabecalho_discentes_envolvidos_extensao, encoding=encoding,
sep=separador)
```

```
carrega_discentes_envolvidos_extensao = PythonOperator(
task_id='carrega_discentes_envolvidos_extensao',
python_callable=cria_planilha_discentes_envolvidos_extensao
)
```

```
""" def cria_planilha_total_discentes_por_campus(ti):
    #Obtém o resultado da query
    result_total_discentes_por_campus =
ti.xcom_pull(task_ids='extrai_total_discentes_por_campus')
    #Cria um dataframe pandas
    df_discentes_por_campus =
pd.DataFrame(result_total_discentes_por_campus)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_discentes_por_campus = ['Câmpus', 'CEP', 'Ano', 'Tipo de
Curso', 'Discentes']
    #Exporta para CSV
```



```

        if df_discentes_por_campus.empty:
            df_discentes_por_campus =
pd.DataFrame(columns=cabecalho_discentes_por_campus)
            df_discentes_por_campus.to_csv(csv_discentes_por_campus, index=False,
sep=separador)
        else:
            df_discentes_por_campus.to_csv(csv_discentes_por_campus, index=False,
header=cabecalho_discentes_por_campus, encoding=encoding, sep=separador)

```

```

carrega_total_discentes_por_campus = PythonOperator(
    task_id='carrega_total_discentes_por_campus',
    python_callable=cria_planilha_total_discentes_por_campus
) """

```

```

def cria_planilha_percentual_discentes_por_campus(ti):
    #Obtém o resultado da query
    result_percentual_discentes_na_extensao =
ti.xcom_pull(task_ids='transforma_percentual_discentes_na_extensao')
    #Cria um dataframe pandas
    df_percentual_discentes_na_extensao =
pd.DataFrame(result_percentual_discentes_na_extensao)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_percentual_discentes_na_extensao = ['Câmpus','CEP','Ano','Tipo
de Curso','Discentes na Extensão(%)']
    #Exporta para CSV
    if df_percentual_discentes_na_extensao.empty:
        df_percentual_discentes_na_extensao =
pd.DataFrame(columns=cabecalho_percentual_discentes_na_extensao)

df_percentual_discentes_na_extensao.to_csv(csv_percentual_discentes_na_extens
ao, index=False, encoding=encoding, sep=separador)
    else:

```

```

df_percentual_discentes_na_extensao.to_csv(csv_percentual_discentes_na_extens
ao, index=False, header=cabecalho_percentual_discentes_na_extensao,
encoding=encoding, sep=separador)

```

```

carrega_percentual_discentes_por_campus = PythonOperator(
    task_id='carrega_percentual_discentes_por_campus',
    python_callable=cria_planilha_percentual_discentes_por_campus,
)

```

```

""" envia_arquivos_email = EmailOperator(
    task_id='envia_arquivos_email',
    to=Variable.get("IFSC_email_extensao"),# email definido em Admin ->

```

Variables

```

    subject='Indicadores de Extensão - Discentes',
    html_content="Segue em anexo os arquivos",
    files=lista_arquivos,

```

```
conn_id="IFSC_SMTP", # Conexão SMTP definida em Admin -> Connections
) ""
```

Definir as dependências entre as tasks

```
[extrai_discentes_voluntarios_envolvidos_extensao,
extrai_discentes_bolsistas_envolvidos_extensao,
extrai_quantidade_discentes_envolvidos_alem_creditos_obrigatorios,
extrai_discentes_envolvidos_extensao,
extrai_total_discentes_por_campus] >>
transforma_percentual_discentes_na_extensao >>
cria_datalake_discentes_extensao >>
[carrega_discentes_voluntarios_envolvidos_extensao,

carrega_discentes_bolsistas_envolvidos_extensao,

carrega_quantidade_discentes_envolvidos_alem_creditos_obrigatorios,

carrega_discentes_envolvidos_extensao,

#carrega_total_discentes_por_campus,

carrega_percentual_discentes_por_campus]
```

PARTICIPANTES

```
from airflow import DAG
from airflow.operators.python import PythonOperator
from airflow.providers.postgres.operators.postgres import PostgresOperator
from airflow.providers.smtp.operators.smtp import EmailOperator
from airflow.models import Variable
from airflow.utils.dates import days_ago
from datetime import datetime
import pandas as pd
import numpy as np
import os
import logging

# Caminho do arquivo CSV de saída
datalake_extensao_dir = Variable.get("datalake")+ 'extensao/'
datalake_participantes_dir = datalake_extensao_dir+'participantes/'
encoding = "utf-8"
separador = ","
csv_participantes_servidores_envolvidos =
datalake_participantes_dir+'IFSC_EXTENSAO_SIGAA_PARTICIPANTES_SERVIDO
RES_ENVOLVIDOS.csv'
csv_participantes_externos_envolvidos =
datalake_participantes_dir+'IFSC_EXTENSAO_SIGAA_PARTICIPANTES_EXTERN
OS_ENVOLVIDOS.csv'
csv_participantes_externos_voluntarios_envolvidos =
datalake_participantes_dir+'IFSC_EXTENSAO_SIGAA_PARTICIPANTES_EXTERN
OS_VOLUNTARIOS_ENVOLVIDOS.csv'
csv_participantes_externos_envolvidos_por_papel =
datalake_participantes_dir+'IFSC_EXTENSAO_SIGAA_PARTICIPANTES_EXTERN
OS_ENVOLVIDOS_POR_PAPEL.csv'
csv_participantes_externos_envolvidos_por_status =
datalake_participantes_dir+'IFSC_EXTENSAO_SIGAA_PARTICIPANTES_EXTERN
OS_ENVOLVIDOS_POR_STATUS.csv'
csv_participantes_externos_por_atividade_extensao =
datalake_participantes_dir+'IFSC_EXTENSAO_SIGAA_PARTICIPANTES_EXTERN
OS_POR_ATIVIDADES_EXTENSAO.csv'
#csv_participantes_por_campus =
datalake_participantes_dir+'IFSC_EXTENSAO_SIGAA_PARTICIPANTES_POR_CA
MPUS.csv'
#csv_servidores_por_campus =
datalake_participantes_dir+'IFSC_EXTENSAO_SIGAA_SERVIDORES_ATIVOS_PO
R_CAMPUS.csv'
csv_participantes_externos_por_atividade_extensao =
datalake_participantes_dir+'IFSC_EXTENSAO_SIGAA_PARTICIPANTES_EXTERN
OS_POR_ATIVIDADE.csv'
csv_participantes_externos_por_total_participantes =
datalake_participantes_dir+'IFSC_EXTENSAO_SIGAA_PARTICIPANTES_EXTERN
OS_POR_TOTAL_PARTICIPANTES.csv'
```

```

csv_percentual_servidores_envolvidos =
datalake_participantes_dir+'IFSC_EXTENSAO_SIGAA_PARTICIPANTES_PERCEN
TUAL_SERVIDORES_ENVOLVIDOS.csv'
lista_arquivos=[csv_participantes_servidores_envolvidos,csv_participantes_externos
_envolvidos,csv_participantes_externos_voluntarios_envolvidos,csv_participantes_e
xternos_envolvidos_por_papel,csv_participantes_externos_envolvidos_por_status,cs
v_participantes_externos_por_atividade_extensao,csv_participantes_externos_por_t
otal_participantes,csv_percentual_servidores_envolvidos]

```

```

with DAG(
    dag_id='IFSC_EXTENSAO_SIGAA_PARTICIPANTES',
    start_date=days_ago(1),
    schedule_interval='@daily',
    catchup=False,
) as dag:

```

```

    # quantitativo de servidores envolvidos em atividades de extensão por
campus e ano
    extrai_participantes_servidores_envolvidos_extensao = PostgresOperator(
        task_id='extrai_participantes_servidores_envolvidos_extensao',
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
        sql =
Variable.get("script_SIGAA_IFSC_consulta_quantidade_participantes_servidores_en
volvidos_extensao"), # script definido em Admin -> Variables
        #do_xcom_push=True,
    )

```

```

    # quantitativo de participantes externos envolvidos em atividades de extensão
por campus e ano
    extrai_participantes_externos_envolvidos_extensao = PostgresOperator(
        task_id='extrai_participantes_externos_envolvidos_extensao',
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
        sql =
Variable.get("script_SIGAA_IFSC_consulta_quantidade_participantes_externos_env
olvidos_extensao"), # script definido em Admin -> Variables
        #do_xcom_push=True,
    )

```

```

    # quantitativo de participantes externos com papel de voluntario envolvidos
em atividades de extensão por campus e ano
    extrai_participantes_externos_voluntarios_envolvidos_extensao =
PostgresOperator(
        task_id='extrai_participantes_externos_voluntarios_envolvidos_extensao',
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections

```

```

    sql =
Variable.get("script_SIGAA_IFSC_consulta_participantes_externos_voluntarios_env
olvidos_extensao"), # script definido em Admin -> Variables
    #do_xcom_push=True,
)

# quantitativo de participantes externos envolvidos em atividades de extensão
por campus, ano e papel
    extrai_participantes_externos_envolvidos_extensao_por_papel =
PostgresOperator(
    task_id='extrai_participantes_externos_envolvidos_extensao_por_papel',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql =
Variable.get("script_SIGAA_IFSC_consulta_participantes_externos_envolvidos_exte
nsao_por_papel"), # script definido em Admin -> Variables
    #do_xcom_push=True,
)

# quantitativo de participantes externos envolvidos em atividades de extensão
por campus, ano e status
    extrai_participantes_externos_envolvidos_extensao_por_status =
PostgresOperator(
    task_id='extrai_participantes_externos_envolvidos_extensao_por_status',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql =
Variable.get("script_SIGAA_IFSC_consulta_participantes_externos_envolvidos_exte
nsao_por_status"), # script definido em Admin -> Variables
    #do_xcom_push=True,
)

# proporção de participantes externos por total de atividade de extensão
#(usar tabela de quantitativo total de participantes externos e tabela de
quantitativo total de atividades de extensão da categoria ATIVIDADES DE
EXTENSÃO)
    extrai_quantitativo_atividades = PostgresOperator(
    task_id='extrai_quantitativo_atividades',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql =
Variable.get("script_SIGAA_IFSC_consulta_quantitativo_atividades_extensao"), #
script definido em Admin -> Variables
    #do_xcom_push=True,
)

# proporção de participantes externos por total de participantes (retorna total
de participantes de extensão por campus e ano)
    extrai_total_participantes_por_campus = PostgresOperator(

```

```

        task_id='extraí_total_participantes_por_campus',
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
        sql =
Variable.get("script_SIGAA_IFSC_consulta_total_participantes_por_campus"), #
script definido em Admin -> Variables
        #do_xcom_push=True,
    )

```

percentual de servidores envolvidos em atividades de extensão (retorna total de servidores por campus e ano, considerar o quantitativo de servidores envolvidos em extensão)

```

        extraí_total_servidores_por_campus = PostgresOperator(
        task_id='extraí_total_servidores_por_campus',
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
        sql =
Variable.get("script_SIGAA_IFSC_consulta_total_servidores_por_campus"), # script
definido em Admin -> Variables
        #do_xcom_push=True,
    )

```

def

calcula_proporcao_participantes_externos_por_atividades_extensao(ti) -> **pd.DataFrame**:

```

    # Obtém o resultado das queries
    result_participantes_externos =
ti.xcom_pull(task_ids='extraí_participantes_externos_envolvidos_extensao')
    result_quantitativo_atividades =
ti.xcom_pull(task_ids='extraí_quantitativo_atividades')
    # Cria os dataframes pandas
    df_participantes_externos = pd.DataFrame(result_participantes_externos,
columns=['campus','cep','ano','participantes_externos'])
    df_quantitativo_atividades = pd.DataFrame(result_quantitativo_atividades,
columns=['campus','cep','ano','atividades'])
    #Mescla os dois dataframes com base campus,cep e ano preservando
df_participantes_externos
    df_merged = pd.merge(
        df_participantes_externos,
        df_quantitativo_atividades,
        on=['campus','cep','ano'],
        how='left'
    ).fillna(0)
    # Calcula a proporção de participantes externos por total de atividade de
extensão
    df_merged['proporcao_participantes_externos_por_atividade'] = (
        (df_merged['participantes_externos']) / df_merged['atividades']
    ).fillna(0)
    # formata o percentual para até 2 casas decimais

```

```

df_merged['proporcao_participantes_externos_por_atividade'] =
df_merged['proporcao_participantes_externos_por_atividade'].round(2)
# substitui valores infinitos por <0.01
df_merged.replace([np.inf, -np.inf], 0, inplace=True)
# Cria um novo dataframe apenas com as colunas interessadas
df_resultado_proporcao = df_merged[['campus', 'cep',
'ano', 'proporcao_participantes_externos_por_atividade']]
return df_resultado_proporcao

```

```

# proporção de participantes externos por total de atividade de extensão
transforma_proporcao_participantes_externos_por_atividades_extensao =
PythonOperator(

```

```

task_id='transforma_proporcao_participantes_externos_por_atividades_extensao',

python_callable=calcula_proporcao_participantes_externos_por_atividades_extensa
o
)

```

```

def
calcula_proporcao_participantes_externos_por_total_participantes(ti) ->
pd.DataFrame:
    # Obtém o resultado das queries
    result_participantes_externos =
ti.xcom_pull(task_ids='extrai_participantes_externos_envolvidos_extensao')
    result_total_participantes =
ti.xcom_pull(task_ids='extrai_total_participantes_por_campus')
    # Cria os dataframes pandas
    df_participantes_externos = pd.DataFrame(result_participantes_externos,
columns=['campus', 'cep', 'ano', 'participantes_externos'])
    df_total_participantes = pd.DataFrame(result_total_participantes,
columns=['campus', 'cep', 'ano', 'participantes'])
    #Mescla os dois dataframes com base campus, cep e ano preservando
df_participantes_externos
    df_merged = pd.merge(
df_participantes_externos,
df_total_participantes,
on=['campus', 'cep', 'ano'],
how='left'
).fillna(0)
    # Calcula a proporção de participantes externos por total de participantes
df_merged['proporcao_participantes_externos_por_total_participantes'] = (
(df_merged['participantes_externos']) / df_merged['participantes']
).fillna(0)
    # formata o percentual para até 2 casas decimais
df_merged['proporcao_participantes_externos_por_total_participantes'] =
df_merged['proporcao_participantes_externos_por_total_participantes'].round(2)
    # substitui valores infinitos por <0.01
df_merged.replace([np.inf, -np.inf], 0, inplace=True)

```

```

# Cria um novo dataframe apenas com as colunas interessadas
df_resultado_proporcao = df_merged[['campus', 'cep',
'ano','proporcao_participantes_externos_por_total_participantes']]
return df_resultado_proporcao

# proporção de participantes externos por total de participantes
transforma_proporcao_participantes_externos_por_total_participantes =
PythonOperator(

task_id='transforma_proporcao_participantes_externos_por_total_participantes',

python_callable=calcula_proporcao_participantes_externos_por_total_participantes
)

```

```

def calcula_percentual_servidores_envolvidos_extensao(ti) ->
pd.DataFrame:
# Obtém resultado das queries
result_servidores_por_campus =
ti.xcom_pull(task_ids='extrai_total_servidores_por_campus')
result_servidores_participantes =
ti.xcom_pull(task_ids='extrai_participantes_servidores_envolvidos_extensao')
# Cria os dataframes pandas
df_servidores_por_campus = pd.DataFrame(result_servidores_por_campus,
columns=['campus','cep','servidores_campus'])
df_servidores_participantes = pd.DataFrame(result_servidores_participantes,
columns=['campus','cep','ano','servidores_participantes'])
# Mescla os dois dataframes com base em campus e cep preservando
df_servidores_participantes
df_merged = pd.merge(
df_servidores_participantes,
df_servidores_por_campus,
on=['campus','cep'],
how='left'
).fillna(0)
# Calcula o percentual de servidores envolvidos em atividades de extensão
df_merged['percentual_servidores_participantes'] = (
(df_merged['servidores_participantes'] * 100)/ df_merged['servidores_campus']
).fillna(0)
# formata o percentual para até 2 casas decimais
df_merged['percentual_servidores_participantes'] =
df_merged['percentual_servidores_participantes'].round(2)
# Garante que os anos que não tiveram servidores na extensão tenha como
resultado 0%
df_merged.loc[df_merged['servidores_participantes'] == 0,
'percentual_servidores_participantes'] = 0
# substitui valores infinitos por <0.01
df_merged.replace([np.inf, -np.inf], 0, inplace=True)
# Cria um novo dataframe apenas com as colunas interessadas

```



```

df_resultado_percentual =
df_merged[['campus','cep','ano','percentual_servidores_participantes']]
return df_resultado_percentual

```

```

# percentual de servidores envolvidos em atividades de extensão
transforma_percentual_servidores_envolvidos_extensao = PythonOperator(
task_id='transforma_percentual_servidores_envolvidos_extensao',
python_callable=calcula_percentual_servidores_envolvidos_extensao
)

```

```

def transformacao_info():
logging.info("Iniciando Tranformação de Dados")

```

```

inicia_transformacao = PythonOperator(
task_id='inicia_transformacao',
python_callable=transformacao_info
)

```

```

def cria_diretorio():
#cria os diretórios se não existirem
if not os.path.exists(datalake_extensao_dir):
os.makedirs(datalake_extensao_dir)
elif not os.path.exists(datalake_participantes_dir):
os.makedirs(datalake_participantes_dir)

```

```

cria_datalake_participantes_extensao = PythonOperator(
task_id='cria_datalake_participantes_extensao',
python_callable=cria_diretorio,
)

```

```

def cria_planilha_participantes_servidores_envolvidos_extensao(ti):
#Obtém o resultado da query
result_participantes_servidores_envolvidos_extensao =
ti.xcom_pull(task_ids='extrai_participantes_servidores_envolvidos_extensao')
#Cria um dataframe pandas
df_participantes_servidores =
pd.DataFrame(result_participantes_servidores_envolvidos_extensao)
#Mapea o cabeçalho do arquivo CSV
cabecalho_participantes_servidores = ['Câmpus', 'CEP', 'Ano', 'Servidores
Envolvidos em Ações']
#Exporta para CSV
if df_participantes_servidores.empty:
df_participantes_servidores =
pd.DataFrame(columns=cabecalho_participantes_servidores)
df_participantes_servidores.to_csv(csv_participantes_servidores_envolvidos,
index=False, encoding=encoding, sep=separador)
else:

```

```
df_participantes_servidores.to_csv(csv_participantes_servidores_envolvidos,
index=False, header=cabecalho_participantes_servidores, encoding=encoding,
sep=separador)
```

```
carrega_participantes_servidores_envolvidos_extensao = PythonOperator(
task_id='carrega_participantes_servidores_envolvidos_extensao',
python_callable=cria_planilha_participantes_servidores_envolvidos_extensao
)
```

```
def cria_planilha_participantes_externos_envolvidos_extensao(ti):
#Obtém o resultado da query
result_participantes_externos_envolvidos_extensao =
ti.xcom_pull(task_ids='extraí_participantes_externos_envolvidos_extensao')
#Cria um dataframe pandas
df_participantes_externos =
pd.DataFrame(result_participantes_externos_envolvidos_extensao)
#Mapea o cabeçalho do arquivo CSV
cabecalho_participantes_externos = ['Câmpus', 'CEP', 'Ano', 'Participantes
Externos em Ações']
#Exporta para CSV
if df_participantes_externos.empty:
df_participantes_externos =
pd.DataFrame(columns=cabecalho_participantes_externos)
df_participantes_externos.to_csv(csv_participantes_externos_envolvidos,
index=False, encoding=encoding, sep=separador)
else:
df_participantes_externos.to_csv(csv_participantes_externos_envolvidos,
index=False, header=cabecalho_participantes_externos, encoding=encoding,
sep=separador)
```

```
carrega_participantes_externos_envolvidos_extensao = PythonOperator(
task_id='carrega_participantes_externos_envolvidos_extensao',
python_callable=cria_planilha_participantes_externos_envolvidos_extensao
)
```

```
def
cria_planilha_participantes_externos_voluntarios_envolvidos_extensao(ti):
#Obtém o resultado da query
result_participantes_externos_voluntarios_envolvidos_extensao =
ti.xcom_pull(task_ids='extraí_participantes_externos_voluntarios_envolvidos_extensao')
#Cria um dataframe pandas
df_participantes_externos_voluntarios =
pd.DataFrame(result_participantes_externos_voluntarios_envolvidos_extensao)
#Mapea o cabeçalho do arquivo CSV
cabecalho_participantes_externos_voluntarios = ['Câmpus',
'CEP', 'Ano', 'Participantes Externos Voluntários em Ações']
#Exporta para CSV
if df_participantes_externos_voluntarios.empty:
```

```

df_participantes_externos_voluntarios =
pd.DataFrame(columns=cabecalho_participantes_externos_voluntarios)

df_participantes_externos_voluntarios.to_csv(csv_participantes_externos_envolvidos
, index=False, encoding=encoding, sep=separador)
else:

df_participantes_externos_voluntarios.to_csv(csv_participantes_externos_voluntarios
s_envolvidos, index=False, header=cabecalho_participantes_externos_voluntarios,
encoding=encoding, sep=separador)

carrega_participantes_externos_voluntarios_envolvidos_extensao =
PythonOperator(
    task_id='carrega_participantes_externos_voluntarios_envolvidos_extensao',

python_callable=cria_planilha_participantes_externos_voluntarios_envolvidos_exten
sao
)

def
cria_planilha_participantes_externos_envolvidos_extensao_por_papel(ti):
    #Obtém o resultado da query
    result_participantes_externos_envolvidos_extensao_por_papel =
ti.xcom_pull(task_ids='extrai_participantes_externos_envolvidos_extensao_por_pap
el')
    #Cria um dataframe pandas
    df_participantes_externos_por_papel =
pd.DataFrame(result_participantes_externos_envolvidos_extensao_por_papel)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_participantes_externos_por_papel = ['Câmpus',
'CEP', 'Ano', 'Papel', 'Participantes Externos em Ações']
    #Exporta para CSV
    if df_participantes_externos_por_papel.empty:
        df_participantes_externos_por_papel =
pd.DataFrame(columns=cabecalho_participantes_externos_por_papel)

df_participantes_externos_por_papel.to_csv(csv_participantes_externos_envolvidos
_por_papel, index=False, encoding=encoding, sep=separador)
else:

df_participantes_externos_por_papel.to_csv(csv_participantes_externos_envolvidos
_por_papel, index=False, header=cabecalho_participantes_externos_por_papel,
encoding=encoding, sep=separador)

carrega_participantes_externos_envolvidos_extensao_por_papel =
PythonOperator(
    task_id='carrega_participantes_externos_envolvidos_extensao_por_papel',

```

```

python_callable=cria_planilha_participantes_externos_envolvidos_extensao_por_papel
    )

    def
cria_planilha_participantes_externos_envolvidos_extensao_por_status(ti):
        #Obtém o resultado da query
        result_participantes_externos_envolvidos_extensao_por_status =
ti.xcom_pull(task_ids='extrai_participantes_externos_envolvidos_extensao_por_status')
        #Cria um dataframe pandas
        df_participantes_externos_por_status =
pd.DataFrame(result_participantes_externos_envolvidos_extensao_por_status)
        #Mapea o cabeçalho do arquivo CSV
        cabecalho_participantes_externos_por_status = ['Câmpus',
'CEP', 'Ano', 'Status', 'Participantes Externos em Ações']
        #Exporta para CSV
        if df_participantes_externos_por_status.empty:
            df_participantes_externos_por_status =
pd.DataFrame(columns=cabecalho_participantes_externos_por_status)

        df_participantes_externos_por_status.to_csv(csv_participantes_externos_envolvidos_extensao_por_status, index=False, encoding=encoding, sep=separador)
        else:

        df_participantes_externos_por_status.to_csv(csv_participantes_externos_envolvidos_extensao_por_status, index=False, header=cabecalho_participantes_externos_por_status, encoding=encoding, sep=separador)

        carrega_participantes_externos_envolvidos_extensao_por_status =
PythonOperator(
            task_id='carrega_participantes_externos_envolvidos_extensao_por_status',

python_callable=cria_planilha_participantes_externos_envolvidos_extensao_por_status
    )

    """ def cria_planilha_total_participantes_por_campus(ti):
        #Obtém o resultado da query
        result_total_participantes_por_campus =
ti.xcom_pull(task_ids='extrai_total_participantes_por_campus')
        #Cria um dataframe pandas
        df_participantes_por_campus =
pd.DataFrame(result_total_participantes_por_campus)
        #Mapea o cabeçalho do arquivo CSV
        cabecalho_participantes_por_campus = ['Câmpus', 'CEP', 'Ano', 'Participantes em Ações']
        #Exporta para CSV

```

```

        if df_participantes_por_campus.empty:
            df_participantes_por_campus =
pd.DataFrame(columns=cabecalho_participantes_por_campus)
            df_participantes_por_campus.to_csv(csv_participantes_por_campus,
index=False, encoding=encoding, sep=separador)
        else:
            df_participantes_por_campus.to_csv(csv_participantes_por_campus,
index=False, header=cabecalho_participantes_por_campus)

```

```

carrega_total_participantes_por_campus = PythonOperator(
    task_id='carrega_total_participantes_por_campus',
    python_callable=cria_planilha_total_participantes_por_campus
)

```

```

def cria_planilha_total_servidores_por_campus(ti):
    #Obtém o resultado da query
    result_total_servidores_por_campus =
ti.xcom_pull(task_ids='extraí_total_servidores_por_campus')
    #Cria um dataframe pandas
    df_servidores_por_campus =
pd.DataFrame(result_total_servidores_por_campus)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_servidores_por_campus = ['Câmpus', 'CEP', 'Servidores Ativos']
    #Exporta para CSV
    if df_servidores_por_campus.empty:
        df_servidores_por_campus =
pd.DataFrame(columns=cabecalho_servidores_por_campus)
        df_servidores_por_campus.to_csv(csv_servidores_por_campus, index=False,
encoding=encoding, sep=separador)
    else:
        df_servidores_por_campus.to_csv(csv_servidores_por_campus, index=False,
header=cabecalho_servidores_por_campus, encoding=encoding, sep=separador)

```

```

carrega_total_servidores_por_campus = PythonOperator(
    task_id='carrega_total_servidores_por_campus',
    python_callable=cria_planilha_total_servidores_por_campus
)

```

```
def
```

```

cria_planilha_proporcao_participantes_externos_por_atividades_extensao(ti):
    #Obtém o resultado da query
    result_proporcao_participantes_externos_por_atividades =
ti.xcom_pull(task_ids='transforma_proporcao_participantes_externos_por_atividades
_extensao')
    #Cria um dataframe pandas
    df_proporcao_participantes_externos_por_atividades =
pd.DataFrame(result_proporcao_participantes_externos_por_atividades)
    #Mapea o cabeçalho do arquivo CSV

```

```

        cabecalho_proporcao_participantes_externos_por_atividades =
['Câmpus','CEP','Ano','Propoção de Participante Externo por Ação de Extensão']
        #Exporta para CSV
        if df_proporcao_participantes_externos_por_atividades.empty:
            df_proporcao_participantes_externos_por_atividades =
pd.DataFrame(columns=cabecalho_proporcao_participantes_externos_por_atividades)

df_proporcao_participantes_externos_por_atividades.to_csv(csv_participantes_externos_por_atividade_extensao, index=False, encoding=encoding, sep=separador)
        else:

df_proporcao_participantes_externos_por_atividades.to_csv(csv_participantes_externos_por_atividade_extensao, index=False,
header=cabecalho_proporcao_participantes_externos_por_atividades,
encoding=encoding, sep=separador)

        carrega_proporcao_participantes_externos_por_atividades_extensao =
PythonOperator(

task_id='carrega_proporcao_participantes_externos_por_atividades_extensao',

python_callable=cria_planilha_proporcao_participantes_externos_por_atividades_extensao
    )

    def
cria_planilha_proporcao_participantes_externos_por_total_participantes(ti):
        #Obtém o resultado da query
        result_proporcao_participantes_externos_por_total_participantes =
ti.xcom_pull(task_ids='transforma_proporcao_participantes_externos_por_total_participantes')
        #Cria um dataframe pandas
        df_proporcao_participantes_externos_por_total_participantes =
pd.DataFrame(result_proporcao_participantes_externos_por_total_participantes)
        #Mapea o cabeçalho do arquivo CSV
        cabecalho_proporcao_participantes_externos_por_total_participantes =
['Câmpus','CEP','Ano','Propoção de Participantes Externos por Total de Participantes']
        #Exporta para CSV
        if df_proporcao_participantes_externos_por_total_participantes.empty:
            df_proporcao_participantes_externos_por_total_participantes =
pd.DataFrame(columns=cabecalho_proporcao_participantes_externos_por_total_participantes)

df_proporcao_participantes_externos_por_total_participantes.to_csv(csv_participantes_externos_por_total_participantes, index=False, encoding=encoding,
sep=separador)
        else:

```

```
df_proporcao_participantes_externos_por_total_participantes.to_csv(csv_participantes_externos_por_total_participantes, index=False, header=cabecalho_proporcao_participantes_externos_por_total_participantes, encoding=encoding, sep=separador)
```

```
carrega_proporcao_participantes_externos_por_total_participantes = PythonOperator(
    task_id='carrega_proporcao_participantes_externos_por_total_participantes',
    python_callable=cria_planilha_proporcao_participantes_externos_por_total_participantes
)
```

```
def cria_planilha_percentual_servidores_envolvidos_extensao(ti):
    #Obtém o resultado da query
    result_percentual_servidores_envolvidos_extensao =
ti.xcom_pull(task_ids='transforma_percentual_servidores_envolvidos_extensao')
    #Cria um dataframe pandas
    df_percentual_servidores_envolvidos_extensao =
pd.DataFrame(result_percentual_servidores_envolvidos_extensao)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_percentual_servidores_envolvidos_extensao =
['Câmpus','CEP','Ano','Servidores Envolvidos em Ações(%)']
    #Exporta para CSV
    if df_percentual_servidores_envolvidos_extensao.empty:
        df_percentual_servidores_envolvidos_extensao =
pd.DataFrame(columns=cabecalho_percentual_servidores_envolvidos_extensao)
```

```
df_percentual_servidores_envolvidos_extensao.to_csv(csv_percentual_servidores_envolvidos_extensao, index=False, encoding=encoding, sep=separador)
else:
```

```
df_percentual_servidores_envolvidos_extensao.to_csv(csv_percentual_servidores_envolvidos_extensao, index=False, header=cabecalho_percentual_servidores_envolvidos_extensao, encoding=encoding, sep=separador)
```

```
carrega_percentual_servidores_envolvidos_extensao = PythonOperator(
    task_id='carrega_percentual_servidores_envolvidos_extensao',
    python_callable=cria_planilha_percentual_servidores_envolvidos_extensao
)
```

```
""" envia_arquivos_email = EmailOperator(
    task_id='envia_arquivos_email',
    to=Variable.get("IFSC_email_extensao"),# email definido em Admin ->
Variables
    subject='Indicadores de Extensão - Participantes',
    html_content="Segue em anexo os arquivos",
```

```
files=lista_arquivos,  
conn_id="IFSC_SMTP", # Conexão SMTP definida em Admin -> Connections  
) ""
```

```
# Definir as dependências entre as tasks  
[extraí_participantes_servidores_envolvidos_extensao,  
extraí_participantes_externos_envolvidos_extensao,  
extraí_participantes_externos_voluntarios_envolvidos_extensao,  
extraí_participantes_externos_envolvidos_extensao_por_papel,  
extraí_participantes_externos_envolvidos_extensao_por_status,  
#extraí_participantes_extensao_por_atividades_extensao,  
extraí_total_participantes_por_campus,  
extraí_total_servidores_por_campus,  
extraí_quantitativo_atividades] >> inicia_transformacao >>  
[transforma_proporcao_participantes_externos_por_atividades_extensao,  
  
transforma_proporcao_participantes_externos_por_total_participantes,  
transforma_percentual_servidores_envolvidos_extensao]  
>> cria_datalake_participantes_extensao >>  
[carrega_participantes_servidores_envolvidos_extensao,  
  
carrega_participantes_externos_envolvidos_extensao,  
  
carrega_participantes_externos_voluntarios_envolvidos_extensao,  
  
carrega_participantes_externos_envolvidos_extensao_por_papel,  
  
carrega_participantes_externos_envolvidos_extensao_por_status,  
  
#carrega_participantes_extensao_por_atividades_extensao,  
  
#carrega_total_participantes_por_campus,  
  
#carrega_total_servidores_por_campus,  
  
carrega_proporcao_participantes_externos_por_atividades_extensao,  
  
carrega_proporcao_participantes_externos_por_total_participantes,  
  
carrega_percentual_servidores_envolvidos_extensao]
```


MODALIDADES DE ATIVIDADES DE EXTENSÃO

```
from airflow import DAG
from airflow.operators.python import PythonOperator
from airflow.providers.postgres.operators.postgres import PostgresOperator
from airflow.providers.smtp.operators.smtp import EmailOperator
from airflow.models import Variable
from airflow.utils.dates import days_ago
from datetime import datetime
import pandas as pd
import numpy as np
import os
import logging

# Caminho do arquivo CSV de saída
datalake_extensao_dir = Variable.get("datalake")+ 'extensao/'
datalake_modalidades_dir = datalake_extensao_dir+'modalidades/'
encoding = "utf-8"
separador = ","
csv_atividades_por_modalidades =
datalake_modalidades_dir+'IFSC_EXTENSAO_SIGAA_ATIVIDADES_POR_MODALIDADES.csv'
#csv_matriculas_ativas_por_programas_projetos =
datalake_modalidades_dir+'IFSC_EXTENSAO_SIGAA_MATRICULAS_ATIVAS_POR_PROGRAMAS_E_PROJETOS.csv'
csv_programas_projetos =
datalake_modalidades_dir+'IFSC_EXTENSAO_SIGAA_PROGRAMAS_E_PROJETOS.csv'
csv_matriculas_ativas_por_modalidade =
datalake_modalidades_dir+'IFSC_EXTENSAO_SIGAA_MATRICULAS_ATIVAS_POR_MODALIDADE.csv'
#csv_matriculas_ativas =
datalake_modalidades_dir+'IFSC_EXTENSAO_SIGAA_MATRICULAS_ATIVAS.csv'
csv_incubadoras =
datalake_modalidades_dir+'IFSC_EXTENSAO_SIGAA_INCUBADORAS.csv'
csv_incubadoras_graduadas =
datalake_modalidades_dir+'IFSC_EXTENSAO_SIGAA_INCUBADORAS_GRADUADAS.csv'
csv_cooperativas_graduadas =
datalake_modalidades_dir+'IFSC_EXTENSAO_SIGAA_COOPERATIVAS_GRADUADAS.csv'
csv_proporcao_discentes_programas_projetos_por_campus =
datalake_modalidades_dir+'IFSC_EXTENSAO_SIGAA_PROPORCAO_MATRICULAS_ATIVAS_POR_PROGRAMAS_E_PROJETOS.csv'
csv_proporcao_discentes_por_modalidade =
datalake_modalidades_dir+'IFSC_EXTENSAO_SIGAA_PROPORCAO_MATRICULAS_ATIVAS_POR_MODALIDADE.csv'
```

```

csv_proporcao_atividades_por_modalidade =
datalake_modalidades_dir+'IFSC_EXTENSAO_SIGAA_PROPORCAO_ATIVIDADE
S_EXTENSAO_POR_MODALIDADE.csv'
lista_arquivos = [csv_atividades_por_modalidades,
csv_programas_projetos,csv_matriculas_ativas_por_modalidade, csv_incubadoras,
csv_incubadoras_graduadas, csv_cooperativas_graduadas,
csv_proporcao_discentes_programas_projetos_por_campus,
csv_proporcao_discentes_por_modalidade,
csv_proporcao_atividades_por_modalidade]

```

```

with DAG(
    dag_id='IFSC_EXTENSAO_SIGAA_MODALIDADES',
    start_date=days_ago(1),
    schedule_interval='@daily',
    catchup=False,

```

```

) as dag:

```

```

    # quantitativo de curso extensão por campus e ano
    # quantitativo de evento por campus e ano
    # quantitativo de programa por campus e ano
    # quantitativo de produto por campus e ano
    # quantitativo de projeto por campus e ano
    # quantitativo de prestação de serviço por campus e ano
    extrai_atividades_extensao_por_modalidades = PostgresOperator(
        task_id='extrai_atividades_extensao_por_modalidades',
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em

```

```

Admin -> Connections

```

```

    sql =

```

```

Variable.get("script_SIGAA_IFSC_consulta_atividades_extensao_por_modalidades")
, # script definido em Admin -> Variables
    #do_xcom_push=True,
)

```

```

    # proporção de discentes com matrícula ativa por programas e projetos por
campus e ano (retorna quantitativo de discentes com matrícula ativa em programas
e projetos)

```

```

    extrai_discentes_matricula_ativa_programas_projetos = PostgresOperator(
        task_id='extrai_discentes_matricula_ativa_programas_projetos',
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em

```

```

Admin -> Connections

```

```

    sql =

```

```

Variable.get("script_SIGAA_IFSC_consulta_discentes_matricula_ativa_programas_p
rojetos"), # script definido em Admin -> Variables
    #do_xcom_push=True,
)

```

```

    # quantitativo de programas e projetos por campus e ano
    extrai_programas_projetos = PostgresOperator(

```

```

        task_id='extrai_programas_projetos',
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
        sql = Variable.get("script_SIGAA_IFSC_consulta_programas_projetos"), #
script definido em Admin -> Variables
        #do_xcom_push=True,
    )

    # quantitativo de discentes com matrícula ativa por programas por campus e
ano
    # quantitativo de discentes com matrícula ativa por projetos por campus e ano
    # quantitativo de discentes com matrícula ativa por produtos por campus e
ano
    # quantitativo de discentes com matrícula ativa por prestação de serviço por
campus e ano
    # quantitativo de discentes com matrícula ativa por curso extensão por
campus e ano
    # quantitativo de discentes com matrícula ativa por evento por campus e ano
    extrai_discentes_matricula_ativa_por_modalidade = PostgresOperator(
        task_id='extrai_discentes_matricula_ativa_por_modalidade',
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
        sql =
Variable.get("script_SIGAA_IFSC_consulta_discentes_matricula_ativa_por_modalida
de"), # script definido em Admin -> Variables
        #do_xcom_push=True,
    )

    # proporção de discentes com matrícula ativa por programas por campus e
ano
    # proporção de discentes com matrícula ativa por projetos por campus e ano
    # proporção de discentes com matrícula ativa por produtos por campus e ano
    # proporção de discentes com matrícula ativa por prestação de serviço por
campus e ano
    # retorna a quantidade de matrículas ativas por campus e ano de ingresso
para cálculo
    extrai_discentes_matricula_ativa = PostgresOperator(
        task_id='extrai_discentes_matricula_ativa',
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
        sql = Variable.get("script_SIGAA_IFSC_consulta_discentes_matricula_ativa"),
# script definido em Admin -> Variables
        #do_xcom_push=True,
    )
    # proporção de atividades de extensão no ano por modalidade projeto por
campus e ano
    # proporção de atividades de extensão no ano por modalidade curso por
campus e ano
    # retorna quantitativo total de atividades de extensão por câmpus e ano

```

```

        extrai_quantitativo_atividades = PostgresOperator(
            task_id='extrai_quantitativo_atividades',
            postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
            sql =
Variable.get("script_SIGAA_IFSC_consulta_quantitativo_atividades_extensao"), #
script definido em Admin -> Variables
            #do_xcom_push=True,
        )

```

```

        # quantitativo de incubadoras por campus e ano
        extrai_incubadoras = PostgresOperator(
            task_id='extrai_incubadoras',
            postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
            sql = Variable.get("script_SIGAA_IFSC_consulta_incubadoras"), # script
definido em Admin -> Variables
            #do_xcom_push=True,
        )

```

```

        # quantitativo de empreendimentos graduados em incubadoras resultados da
extensão por campus e ano
        extrai_incubadoras_graduadas = PostgresOperator(
            task_id='extrai_incubadoras_graduadas',
            postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
            sql = Variable.get("script_SIGAA_IFSC_consulta_incubadoras_graduadas"), #
script definido em Admin -> Variables
            #do_xcom_push=True,
        )

```

```

        # quantitativo de cooperativas populares graduadas em incubadoras
resultadas da extensão por campus e ano
        extrai_cooperativas_graduadas = PostgresOperator(
            task_id='extrai_cooperativas_graduadas',
            postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
            sql = Variable.get("script_SIGAA_IFSC_consulta_cooperativas_graduadas"), #
script definido em Admin -> Variables
            #do_xcom_push=True,
        )

```

```

def calcula_proporcao_discentes_programas_projetos_por_campus(ti)
-> pd.DataFrame:
    #Obtém resultado das queries
    result_discentes_matricula_ativa_programas_projetos =
ti.xcom_pull(task_ids='extrai_discentes_matricula_ativa_programas_projetos')
    result_discentes_matricula_ativa =
ti.xcom_pull(task_ids='extrai_discentes_matricula_ativa')

```

```

# Cria os dataframes pandas para manipulação
df_discentes_matricula_ativa_programas_projetos =
pd.DataFrame(result_discentes_matricula_ativa_programas_projetos,
columns=['campus','cep','ano','discentes_programas_projetos'])
df_discentes_matricula_ativa =
pd.DataFrame(result_discentes_matricula_ativa,
columns=['campus','cep','ano','discentes'])
# mescla os dataframes preservando
df_discentes_matricula_ativa_programas_projetos
df_merged = pd.merge(
df_discentes_matricula_ativa_programas_projetos,
df_discentes_matricula_ativa,
on=['campus','cep','ano'],
how='left'
).fillna(0)
#Calcula a proporção de discentes com matrículas ativas por programas e
projetos
df_merged['proporcao_discentes_ativos'] = (
(df_merged['discentes_programas_projetos']) / df_merged['discentes']
).fillna(0)
# formata o percentual para até 2 casas decimais
df_merged['proporcao_discentes_ativos'] =
df_merged['proporcao_discentes_ativos'].round(2)
# substitui valores infinitos por <0.01
df_merged.replace([np.inf, -np.inf], 0, inplace=True)
# Cria um novo dataframe apenas com as colunas interessadas
df_resultado = df_merged[['campus', 'cep', 'ano',
'proporcao_discentes_ativos']]
return df_resultado

```

```

# proporção de discentes com matrícula ativa por programas e projetos por
campus e ano
transforma_proporcao_discentes_programas_projetos_por_campus =
PythonOperator(
task_id='transforma_proporcao_discentes_programas_projetos_por_campus',

python_callable=calcula_proporcao_discentes_programas_projetos_por_campus
)

```

```

def calcula_proporcao_discentes_por_modalidade(ti) -> pd.DataFrame:
#Obtém resultado das queries
result_discentes_matricula_ativa =
ti.xcom_pull(task_ids='extrai_discentes_matricula_ativa')
result_discentes_matricula_ativa_por_modalidade =
ti.xcom_pull(task_ids='extrai_discentes_matricula_ativa_por_modalidade')
# Cria os dataframes pandas para manipulação
df_discentes_matricula_ativa =
pd.DataFrame(result_discentes_matricula_ativa,
columns=['campus','cep','ano','discentes'])

```

```

df_discentes_matricula_ativa_por_modalidade =
pd.DataFrame(result_discentes_matricula_ativa_por_modalidade,
columns=['campus','cep','ano','modalidade','discentes_extensao'])
# mescla os dataframes preservando
df_discentes_matricula_ativa_por_modalidade
df_merged = pd.merge(
df_discentes_matricula_ativa_por_modalidade,
df_discentes_matricula_ativa,
on=['campus','cep','ano'],
how='left'
).fillna(0)
#Calcula oa proporção de discentes com matrículas ativas por modalidade
df_merged['proporcao_discentes_ativos_modalidade'] = (
(df_merged['discentes_extensao']) / df_merged['discentes']
).fillna(0)
# formata o percentual para até 2 casas decimais
df_merged['proporcao_discentes_ativos_modalidade'] =
df_merged['proporcao_discentes_ativos_modalidade'].round(2)
# substitui valores infinitos por <0.01
df_merged.replace([np.inf, -np.inf], 0, inplace=True)
# Cria um novo dataframe apenas com as colunas interessadas
df_resultado = df_merged[['campus', 'cep',
'ano','modalidade','proporcao_discentes_ativos_modalidade']]
return df_resultado

```

```

# proporção de discentes com matrícula ativa por programas por campus e
ano
# proporção de discentes com matrícula ativa por projetos por campus e ano
# proporção de discentes com matrícula ativa por produtos por campus e ano
# proporção de discentes com matrícula ativa por prestação de serviço por
campus e ano
transforma_proporcao_discentes_por_modalidade = PythonOperator(
task_id='transforma_proporcao_discentes_por_modalidade',
python_callable=calcula_proporcao_discentes_por_modalidade
)

```

```

def calcula_proporcao_atividades_por_modalidade(ti) -> pd.DataFrame:
#Obtém resultado das queries
result_atividades_por_modalidades =
ti.xcom_pull(task_ids='extrai_atividades_extensao_por_modalidades')
result_quantitativo_atividades =
ti.xcom_pull(task_ids='extrai_quantitativo_atividades')
# Cria os dataframes pandas para manipulação
df_atividades_por_modalidades =
pd.DataFrame(result_atividades_por_modalidades,
columns=['campus','cep','ano','modalidade','atividade_modalidade'])
df_quantitativo_atividades = pd.DataFrame(result_quantitativo_atividades,
columns=['campus','cep','ano','atividade'])
# mescla os dataframes preservando df_atividades_por_modalidades

```

```

df_merged = pd.merge(
    df_atividades_por_modalidades,
    df_quantitativo_atividades,
    on=['campus', 'cep', 'ano'],
    how='left'
).fillna(0)
#Calcula a proporção de atividades de extensão por modalidade
df_merged['proporcao_atividade_modalidade'] = (
    (df_merged['atividade_modalidade']) / df_merged['atividade']
).fillna(0)
# formata o percentual para até 2 casas decimais
df_merged['proporcao_atividade_modalidade'] =
df_merged['proporcao_atividade_modalidade'].round(2)
# substitui valores infinitos por <0.01
df_merged.replace([np.inf, -np.inf], 0, inplace=True)
# Cria um novo dataframe apenas com as colunas interessadas
df_resultado =
df_merged[['campus', 'cep', 'ano', 'modalidade', 'proporcao_atividade_modalidade']]
return df_resultado

```

proporção de atividades de extensão no ano por modalidade projeto por campus e ano

proporção de atividades de extensão no ano por modalidade curso por campus e ano

```

transforma_proporcao_atividades_por_modalidade = PythonOperator(
    task_id = 'transforma_proporcao_atividades_por_modalidade',
    python_callable = calcula_proporcao_atividades_por_modalidade
)

```

```

def transformacao_info():
    logging.info("Iniciando Transformação de Dados")

```

```

inicia_transformacao = PythonOperator(
    task_id='inicia_transformacao',
    python_callable=transformacao_info
)

```

```

def cria_diretorio():
    #cria os diretórios se não existirem
    if not os.path.exists(datalake_extensao_dir):
        os.makedirs(datalake_extensao_dir)
    elif not os.path.exists(datalake_modalidades_dir):
        os.makedirs(datalake_modalidades_dir)

```

```

cria_datalake_modalidades_extensao = PythonOperator(
    task_id='cria_datalake_modalidades_extensao',
    python_callable=cria_diretorio,
)

```



```

def cria_planilha_atividades_extensao_por_modalidades(ti):
    #Obtém o resultado da query
    result_atividades_por_modalidades =
ti.xcom_pull(task_ids='extraí_atividades_extensao_por_modalidades')
    #Cria um dataframe pandas
    df_atividades_por_modalidades =
pd.DataFrame(result_atividades_por_modalidades)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_atividades_por_modalidades = ['Câmpus',
'CEP', 'Ano', 'Modalidade', 'Ações de Extensão']
    #Exporta para CSV
    if df_atividades_por_modalidades.empty:
        df_atividades_por_modalidades =
pd.DataFrame(columns=cabecalho_atividades_por_modalidades)
        df_atividades_por_modalidades.to_csv(csv_atividades_por_modalidades,
index=False, encoding=encoding, sep=separador)
    else:
        df_atividades_por_modalidades.to_csv(csv_atividades_por_modalidades,
index=False, header=cabecalho_atividades_por_modalidades, encoding=encoding,
sep=separador)

```

```

carrega_atividades_extensao_por_modalidades = PythonOperator(
    task_id='carrega_atividades_extensao_por_modalidades',
    python_callable=cria_planilha_atividades_extensao_por_modalidades
)

```

```

""" def cria_planilha_discentes_matricula_ativa_programas_projetos(ti):
    #Obtém o resultado da query
    result_discentes_matricula_ativa_programas_projetos =
ti.xcom_pull(task_ids='extraí_discentes_matricula_ativa_programas_projetos')
    #Cria um dataframe pandas
    df_matriculas_ativas_programas_projetos =
pd.DataFrame(result_discentes_matricula_ativa_programas_projetos)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_matriculas_ativas_programas_projetos = ['Câmpus', 'CEP', 'Ano',
'Discentes com Matrícula Ativa em Programas e Projetos']
    #Exporta para CSV
    if df_matriculas_ativas_programas_projetos.empty:
        df_matriculas_ativas_programas_projetos =
pd.DataFrame(columns=cabecalho_matriculas_ativas_programas_projetos)

df_matriculas_ativas_programas_projetos.to_csv(csv_matriculas_ativas_por_progra
mas_projetos, index=False, encoding=encoding, sep=separador)
    else:

df_matriculas_ativas_programas_projetos.to_csv(csv_matriculas_ativas_por_progra
mas_projetos, index=False,

```



```
header=cabecalho_matriculas_ativas_programas_projetos, encoding=encoding,  
sep=separador)
```

```
carrega_discentes_matricula_ativa_programas_projetos = PythonOperator(  
task_id='carrega_discentes_matricula_ativa_programas_projetos',  
python_callable=cria_planilha_discentes_matricula_ativa_programas_projetos  
) """
```

```
def cria_planilha_programas_projetos(ti):  
    #Obtém o resultado da query  
    result_programas_projetos =  
ti.xcom_pull(task_ids='extrai_programas_projetos')  
    #Cria um dataframe pandas  
    df_programas_projetos = pd.DataFrame(result_programas_projetos)  
    #Mapea o cabeçalho do arquivo CSV  
    cabecalho_programas_projetos = ['Câmpus', 'CEP', 'Ano', 'Programas e  
Projetos']  
    #Exporta para CSV  
    if df_programas_projetos.empty:  
        df_programas_projetos =  
pd.DataFrame(columns=cabecalho_programas_projetos)  
        df_programas_projetos.to_csv(csv_programas_projetos, index=False,  
encoding=encoding, sep=separador)  
    else:  
        df_programas_projetos.to_csv(csv_programas_projetos, index=False,  
header=cabecalho_programas_projetos, encoding=encoding, sep=separador)
```

```
carrega_programas_projetos = PythonOperator(  
task_id='carrega_programas_projetos',  
python_callable=cria_planilha_programas_projetos  
)
```

```
def cria_planilha_discentes_matricula_ativa_por_modalidade(ti):  
    #Obtém o resultado da query  
    result_discentes_matricula_ativa_por_modalidade =  
ti.xcom_pull(task_ids='extrai_discentes_matricula_ativa_por_modalidade')  
    #Cria um dataframe pandas  
    df_discentes_matricula_ativa_por_modalidade =  
pd.DataFrame(result_discentes_matricula_ativa_por_modalidade)  
    #Mapea o cabeçalho do arquivo CSV  
    cabecalho_discentes_matricula_ativa_por_modalidade = ['Câmpus',  
'CEP', 'Ano', 'Modalidade', 'Discentes com Matrícula Ativa em Ações']  
    #Exporta para CSV  
    if df_discentes_matricula_ativa_por_modalidade.empty:  
        df_discentes_matricula_ativa_por_modalidade =  
pd.DataFrame(columns=cabecalho_discentes_matricula_ativa_por_modalidade)  
  
df_discentes_matricula_ativa_por_modalidade.to_csv(csv_matriculas_ativas_por_m  
odalidade, index=False, encoding=encoding, sep=separador)
```

else:

```
df_discentes_matricula_ativa_por_modalidade.to_csv(csv_matriculas_ativas_por_m
odalidade, index=False,
header=cabecalho_discentes_matricula_ativa_por_modalidade, encoding=encoding,
sep=separador)
```

```
carrega_discentes_matricula_ativa_por_modalidade = PythonOperator(
task_id='carrega_discentes_matricula_ativa_por_modalidade',
python_callable=cria_planilha_discentes_matricula_ativa_por_modalidade
)
```

```
""" def cria_planilha_discentes_matricula_ativa(ti):
#Obtém o resultado da query
result_discentes_matricula_ativa =
ti.xcom_pull(task_ids='extraí_discentes_matricula_ativa')
#Cria um dataframe pandas
df_discentes_matricula_ativa =
pd.DataFrame(result_discentes_matricula_ativa)
#Mapea o cabeçalho do arquivo CSV
cabecalho_discentes_matricula_ativa = ['Câmpus', 'CEP', 'Ano de
Ingresso', 'Discentes com Matrícula Ativa em Ações']
#Exporta para CSV
if df_discentes_matricula_ativa.empty:
df_discentes_matricula_ativa =
pd.DataFrame(columns=cabecalho_discentes_matricula_ativa)
df_discentes_matricula_ativa.to_csv(csv_matriculas_ativas, index=False,
encoding=encoding, sep=separador)
else:
df_discentes_matricula_ativa.to_csv(csv_matriculas_ativas, index=False,
header=cabecalho_discentes_matricula_ativa)
```

```
carrega_discentes_matricula_ativa = PythonOperator(
task_id='carrega_discentes_matricula_ativa',
python_callable=cria_planilha_discentes_matricula_ativa
) """
```

```
def cria_planilha_incubadoras(ti):
#Obtém o resultado da query
result_incubadoras = ti.xcom_pull(task_ids='extraí_incubadoras')
#Cria um dataframe pandas
df_incubadoras = pd.DataFrame(result_incubadoras)
#Mapea o cabeçalho do arquivo CSV
cabecalho_incubadoras = ['Câmpus', 'CEP', 'Ano', 'Incubadoras']
#Exporta para CSV
if df_incubadoras.empty:
df_incubadoras = pd.DataFrame(columns=cabecalho_incubadoras)
df_incubadoras.to_csv(csv_incubadoras, index=False, encoding=encoding,
sep=separador)
```

```

else:
    df_incubadoras.to_csv(csv_incubadoras, index=False,
header=cabecalho_incubadoras, encoding=encoding, sep=separador)

carrega_incubadoras = PythonOperator(
    task_id='carrega_incubadoras',
    python_callable=cria_planilha_incubadoras
)

def cria_planilha_incubadoras_graduadas(ti):
    #Obtém o resultado da query
    result_incubadoras_graduadas =
ti.xcom_pull(task_ids='extraí_incubadoras_graduadas')
    #Cria um dataframe pandas
    df_incubadoras_graduadas = pd.DataFrame(result_incubadoras_graduadas)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_incubadoras_graduadas = ['Câmpus', 'CEP', 'Ano', 'Incubadoras
Graduadas']
    #Exporta para CSV
    if df_incubadoras_graduadas.empty:
        df_incubadoras_graduadas =
pd.DataFrame(columns=cabecalho_incubadoras_graduadas)
        df_incubadoras_graduadas.to_csv(csv_incubadoras_graduadas, index=False,
encoding=encoding, sep=separador)
    else:
        df_incubadoras_graduadas.to_csv(csv_incubadoras_graduadas, index=False,
header=cabecalho_incubadoras_graduadas, encoding=encoding, sep=separador)

carrega_incubadoras_graduadas = PythonOperator(
    task_id='carrega_incubadoras_graduadas',
    python_callable=cria_planilha_incubadoras_graduadas
)

def cria_planilha_cooperativas_graduadas(ti):
    #Obtém o resultado da query
    result_cooperativas_graduadas =
ti.xcom_pull(task_ids='extraí_cooperativas_graduadas')
    #Cria um dataframe pandas
    df_cooperativas_graduadas = pd.DataFrame(result_cooperativas_graduadas)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_cooperativas_graduadas = ['Câmpus', 'CEP', 'Ano', 'Cooperativas
Graduadas']
    #Exporta para CSV
    if df_cooperativas_graduadas.empty:
        df_cooperativas_graduadas =
pd.DataFrame(columns=cabecalho_cooperativas_graduadas)
        df_cooperativas_graduadas.to_csv(csv_cooperativas_graduadas,
index=False, encoding=encoding, sep=separador)
    else:

```

```
df_cooperativas_graduadas.to_csv(csv_cooperativas_graduadas,
index=False, header=cabecalho_cooperativas_graduadas, encoding=encoding,
sep=separador)
```

```
carrega_cooperativas_graduadas = PythonOperator(
task_id='carrega_cooperativas_graduadas',
python_callable=cria_planilha_cooperativas_graduadas
)
```

```
def
```

```
cria_planilha_proporcao_discentes_programas_projetos_por_campus(ti):
```

```
#Obtém o resultado da query
```

```
result_proporcao_discentes_programas_projetos_por_campus =
ti.xcom_pull(task_ids='transforma_proporcao_discentes_programas_projetos_por_ca
mpus')
```

```
#Cria um dataframe pandas
```

```
df_proporcao_discentes_programas_projetos_por_campus =
pd.DataFrame(result_proporcao_discentes_programas_projetos_por_campus)
```

```
#Mapea o cabeçalho do arquivo CSV
```

```
cabecalho_proporcao_discentes_programas_projetos_por_campus =
['Câmpus', 'CEP', 'Ano', 'Proporção de Discentes com Matrícula Ativa em Programas
e Projetos']
```

```
#Exporta para CSV
```

```
if df_proporcao_discentes_programas_projetos_por_campus.empty:
```

```
df_proporcao_discentes_programas_projetos_por_campus =
pd.DataFrame(columns=cabecalho_proporcao_discentes_programas_projetos_por_c
ampus)
```

```
df_proporcao_discentes_programas_projetos_por_campus.to_csv(csv_proporcao_di
scentes_programas_projetos_por_campus, index=False, encoding=encoding,
sep=separador)
```

```
else:
```

```
df_proporcao_discentes_programas_projetos_por_campus.to_csv(csv_proporcao_di
scentes_programas_projetos_por_campus, index=False,
header=cabecalho_proporcao_discentes_programas_projetos_por_campus,
encoding=encoding, sep=separador)
```

```
carrega_proporcao_discentes_programas_projetos_por_campus =
```

```
PythonOperator(
```

```
task_id='carrega_proporcao_discentes_programas_projetos_por_campus',
```

```
python_callable=cria_planilha_proporcao_discentes_programas_projetos_por_camp
us
```

```
)
```

```
def cria_planilha_proporcao_discentes_por_modalidade(ti):
```

```
#Obtém o resultado da query
```

```

        result_proporcao_discentes_por_modalidade =
ti.xcom_pull(task_ids='transforma_proporcao_discentes_por_modalidade')
        #Cria um dataframe pandas
        df_proporcao_discentes_por_modalidade =
pd.DataFrame(result_proporcao_discentes_por_modalidade)
        #Mapea o cabeçalho do arquivo CSV
        cabecalho_proporcao_discentes_por_modalidade = ['Câmpus', 'CEP',
'Ano','Modalidade','Proporção de Discentes com Matrícula Ativa por Modalidade']
        #Exporta para CSV
        if df_proporcao_discentes_por_modalidade.empty:
            df_proporcao_discentes_por_modalidade =
pd.DataFrame(columns=cabecalho_proporcao_discentes_por_modalidade)

df_proporcao_discentes_por_modalidade.to_csv(csv_proporcao_discentes_por_mod
alidade, index=False, encoding=encoding, sep=separador)
        else:

df_proporcao_discentes_por_modalidade.to_csv(csv_proporcao_discentes_por_mod
alidade, index=False, header=cabecalho_proporcao_discentes_por_modalidade,
encoding=encoding, sep=separador)

        carrega_proporcao_discentes_por_modalidade = PythonOperator(
            task_id='carrega_proporcao_discentes_por_modalidade',
            python_callable=cria_planilha_proporcao_discentes_por_modalidade
        )

def cria_planilha_proporcao_atividades_por_modalidade(ti):
    #Obtém o resultado da query
    result_proporcao_atividades_por_modalidade =
ti.xcom_pull(task_ids='transforma_proporcao_atividades_por_modalidade')
    #Cria um dataframe pandas
    df_proporcao_atividades_por_modalidade =
pd.DataFrame(result_proporcao_atividades_por_modalidade)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_proporcao_atividades_por_modalidade = ['Câmpus', 'CEP',
'Ano','Modalidade','Proporção de Ações de Extensão por Modalidade']
    #Exporta para CSV
    if df_proporcao_atividades_por_modalidade.empty:
        df_proporcao_atividades_por_modalidade =
pd.DataFrame(columns=cabecalho_proporcao_atividades_por_modalidade)

df_proporcao_atividades_por_modalidade.to_csv(csv_proporcao_atividades_por_mo
dalidade, index=False, encoding=encoding, sep=separador)
        else:

df_proporcao_atividades_por_modalidade.to_csv(csv_proporcao_atividades_por_mo
dalidade, index=False, header=cabecalho_proporcao_atividades_por_modalidade,
encoding=encoding, sep=separador)

```

```
carrega_proporcao_atividades_por_modalidade = PythonOperator(
task_id='carrega_proporcao_atividades_por_modalidade',
python_callable=cria_planilha_proporcao_atividades_por_modalidade
)
```

```
""" envia_arquivos_email = EmailOperator(
task_id='envia_arquivos_email',
to=Variable.get("IFSC_email_extensao"),# email definido em Admin ->
```

Variables

```
subject='Indicadores de Extensão - Modalidades',
html_content="Segue em anexo os arquivos",
files=lista_arquivos,
conn_id="IFSC_SMTP", # Conexão SMTP definida em Admin -> Connections
) """
```

Definir as dependências entre as tasks

```
[extrai_atividades_extensao_por_modalidades,
extrai_discentes_matricula_ativa_programas_projetos,
extrai_programas_projetos,
extrai_discentes_matricula_ativa_por_modalidade,
extrai_discentes_matricula_ativa,
extrai_incubadoras,
extrai_incubadoras_graduadas,
extrai_cooperativas_graduadas,
extrai_quantitativo_atividades] >> inicia_transformacao >>
[transforma_proporcao_discentes_programas_projetos_por_campus,
```

```
transforma_proporcao_discentes_por_modalidade,
```

```
transforma_proporcao_atividades_por_modalidade] >>
```

```
cria_datalake_modalidades_extensao >>
```

```
[carrega_atividades_extensao_por_modalidades,
```

```
#carrega_discentes_matricula_ativa_programas_projetos,
```

```
carrega_programas_projetos,
```

```
carrega_discentes_matricula_ativa_por_modalidade,
```

```
#carrega_discentes_matricula_ativa,
```

```
carrega_incubadoras,
```

```
carrega_incubadoras_graduadas,
```

```
carrega_cooperativas_graduadas,
```

```
carrega_proporcao_discentes_programas_projetos_por_campus,
```

```
carrega_proporcao_discentes_por_modalidade,  
carrega_proporcao_atividades_por_modalidade]
```

CURRICULARIZAÇÃO

```
from airflow import DAG
from airflow.operators.python import PythonOperator
from airflow.providers.postgres.operators.postgres import PostgresOperator
from airflow.providers.smtp.operators.smtp import EmailOperator
from airflow.models import Variable
from airflow.utils.dates import days_ago
from datetime import datetime
import pandas as pd
import numpy as np
import os
import logging

# Caminho do arquivo CSV de saída
datalake_extensao_dir = Variable.get("datalake")+'extensao/'
datalake_curricularizacao_dir = datalake_extensao_dir+'curricularizacao/'
encoding = "utf-8"
separador = ","
csv_discentes_em_atividade_curricular_extensao =
datalake_curricularizacao_dir+'IFSC_EXTENSAO_SIGAA_DISCENTES_EM_ATIVID
ADE_CURRICULAR_EXTENSAO.csv'
#csv_credits_ch_curriculo_extensao_por_tipo_curso =
datalake_curricularizacao_dir+'IFSC_EXTENSAO_SIGAA_CREDITOS_CH_CURRI
CULO_EXTENSAO_POR_TIPO_CURSO.csv'
#csv_total_credits_ch_curriculo_por_tipo_curso =
datalake_curricularizacao_dir+'IFSC_EXTENSAO_SIGAA_TOTAL_CREDITOS_CH_
CURRICULO_POR_TIPO_CURSO.csv'
#csv_credits_ch_curriculo_extensao =
datalake_curricularizacao_dir+'IFSC_EXTENSAO_SIGAA_CREDITOS_CH_CURRI
CULO_EXTENSAO.csv'
#csv_total_credits_ch_curriculo =
datalake_curricularizacao_dir+'IFSC_EXTENSAO_SIGAA_TOTAL_CREDITOS_CH_
CURRICULO.csv'
csv_atividades_extensao_curriculo_por_tipo_curso =
datalake_curricularizacao_dir+'IFSC_EXTENSAO_SIGAA_ATIVIDADES_EXTENSA
O_CURRICULO_POR_TIPO_CURSO.csv'
#csv_cursos_ativos_com_extensao_ppc =
datalake_curricularizacao_dir+'IFSC_EXTENSAO_SIGAA_CURSOS_ATIVOS_COM
_EXTENSAO_PPC.CSV'
#csv_total_ppcs_cursos_ativos =
datalake_curricularizacao_dir+'IFSC_EXTENSAO_SIGAA_TOTAL_PPCS_CURSOS
_ATIVOS.CSV'
csv_percentual_credits_ch_curriculo_extensao_por_tipo_curso =
datalake_curricularizacao_dir+'IFSC_EXTENSAO_SIGAA_PERCENTUAL_CREDIT
OS_CH_CURRICULO_EXTENSAO_POR_TIPO_CURSO.csv'
csv_percentual_credits_ch_curriculo_extensao =
datalake_curricularizacao_dir+'IFSC_EXTENSAO_SIGAA_PERCENTUAL_CREDIT
OS_CH_CURRICULO_EXTENSAO.csv'
```



```

csv_percentual_ppcs_extensao =
datalake_curricularizacao_dir+'IFSC_EXTENSAO_SIGAA_PERCENTUAL_CURSOS
_ATIVOS_COM_EXTENSAO_PPC.CSV'
lista_arquivos=[csv_discentes_em_atividade_curricular_extensao,
csv_percentual_creditos_ch_curriculo_extensao_por_tipo_curso,
csv_atividades_extensao_curriculo_por_tipo_curso, csv_percentual_ppcs_extensao,
csv_percentual_creditos_ch_curriculo_extensao]

```

```

with DAG(
    dag_id='IFSC_EXTENSAO_SIGAA_CURRICULARIZACAO',
    start_date=days_ago(1),
    schedule_interval='@daily',
    catchup=False,
) as dag:

```

```

    # quantitativo de curso extensão por campus e ano
    extrai_discentes_em_atividade_curricular_extensao = PostgresOperator(
        task_id='extrai_discentes_em_atividade_curricular_extensao',
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
        sql =
Variable.get("script_SIGAA_IFSC_consulta_discentes_em_atividade_curricular_exte
nsao"), # script definido em Admin -> Variables
        #do_xcom_push=True,
    )

```

```

    # percentual de créditos ou carga horária destinados à extensão no currículo
de nível médio técnico por campus e ano
    # percentual de créditos ou carga horária destinados à extensão no currículo
de graduação por campus e ano
    # percentual de créditos ou carga horária destinados à extensão no currículo
de especialização por campus e ano
    # percentual de créditos ou carga horária destinados à extensão no currículo
de mestrado por campus e ano
    # percentual de créditos ou carga horária destinados à extensão no currículo
de doutorado por campus e ano
    # percentual de créditos ou carga horária destinados à extensão no currículo
de pós-graduação (mestrado e doutorado) por campus e ano
    # retorna o quantitativo de créditos e carga horária destinados à extensão
    extrai_creditos_ch_curriculo_extensao_por_tipo_curso = PostgresOperator(
        task_id='extrai_creditos_ch_curriculo_extensao_por_tipo_curso',
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
        sql =
Variable.get("script_SIGAA_IFSC_consulta_creditos_ch_curriculo_extensao_por_tip
o_curso"), # script definido em Admin -> Variables
        #do_xcom_push=True,
    )

```

```

# percentual de créditos ou carga horária destinados à extensão no currículo
de nível médio técnico por campus e ano
# percentual de créditos ou carga horária destinados à extensão no currículo
de graduação por campus e ano
# percentual de créditos ou carga horária destinados à extensão no currículo
de especialização por campus e ano
# percentual de créditos ou carga horária destinados à extensão no currículo
de mestrado por campus e ano
# percentual de créditos ou carga horária destinados à extensão no currículo
de doutorado por campus e ano
# percentual de créditos ou carga horária destinados à extensão no currículo
de pós-graduação (mestrado e doutorado) por campus e ano
# retorna o quantitativo de créditos e carga horária total dos cursos
extrai_total_credits_ch_curriculo_por_tipo_curso = PostgresOperator(
    task_id='extrai_total_credits_ch_curriculo_por_tipo_curso',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql =
Variable.get("script_SIGAA_IFSC_consulta_total_credits_ch_curriculo_por_tipo_curso"), # script definido em Admin -> Variables
    #do_xcom_push=True,
)

# percentual de créditos ou carga horária de ações de extensão no currículo
de cursos por campus e ano
# retorna o quantitativo destinados à extensão
extrai_credits_ch_curriculo_extensao = PostgresOperator(
    task_id='extrai_credits_ch_curriculo_extensao',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql =
Variable.get("script_SIGAA_IFSC_consulta_credits_ch_curriculo_extensao"), #
script definido em Admin -> Variables
    #do_xcom_push=True,
)

# percentual de créditos ou carga horária de ações de extensão no currículo
de cursos por campus e ano
# retorna o quantitativo de créditos e carga horária total dos cursos
extrai_total_credits_ch_curriculo = PostgresOperator(
    task_id='extrai_total_credits_ch_curriculo',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql =
Variable.get("script_SIGAA_IFSC_consulta_total_credits_ch_curriculo"), # script
definido em Admin -> Variables
    #do_xcom_push=True,
)

```

```

# quantitativo de atividades de extensão por tipo de curso por campus e ano
extraí_atividades_extensao_curriculo_por_tipo_curso = PostgresOperator(
    task_id='extraí_atividades_extensao_curriculo_por_tipo_curso',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql =
Variable.get("script_SIGAA_IFSC_consulta_atividades_extensao_curriculo_por_tipo
_curso"), # script definido em Admin -> Variables
    #do_xcom_push=True,
)

```

```

# Percentual de PPCs com curricularização da extensão e atividades
complementares por campus e ano
# retorna o total de cursos ativos em que há curricularização da extensão
extraí_cursos_com_extensao_ppcs = PostgresOperator(
    task_id='extraí_cursos_com_extensao_ppcs',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql =
Variable.get("script_SIGAA_IFSC_consulta_cursos_com_extensao_ppcs"), # script
definido em Admin -> Variables
    #do_xcom_push=True,
)

```

```

# Percentual de PPCs com curricularização da extensão e atividades
complementares por campus e ano
# retorna o total de cursos ativos cadastrados
extraí_total_ppcs_cursos_ativos = PostgresOperator(
    task_id='extraí_total_ppcs_cursos_ativos',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql = Variable.get("script_SIGAA_IFSC_consulta_total_ppcs_cursos_ativos"),
# script definido em Admin -> Variables
    #do_xcom_push=True,
)

```

```

def calcula_percentual_cr_ch_por_tipo_curso(ti) -> pd.DataFrame:
    #Obtém o resultado das queries
    result_cr_ch_extensao =
ti.xcom_pull(task_ids='extraí_creditos_ch_curriculo_extensao_por_tipo_curso')
    result_cr_ch_curriculo =
ti.xcom_pull(task_ids='extraí_total_creditos_ch_curriculo_por_tipo_curso')
    #Cria dataframes pandas
    df_cr_ch_extensao = pd.DataFrame(result_cr_ch_extensao,
columns=['campus','cep','ano','tipo_curso','cr_extensao','ch_extensao'])
    df_cr_ch_curriculo = pd.DataFrame(result_cr_ch_curriculo,
columns=['campus','cep','ano','tipo_curso','cr','ch'])
    #Mescla os dois dataframes

```

```

df_merged = pd.merge(
    df_cr_ch_extensao,
    df_cr_ch_curriculo,
    on=['campus','cep','ano','tipo_curso'],
    how='left'
).fillna(0)
#Calcula o percentual
df_merged['percentual_cr_extensao'] = (
    (df_merged['cr_extensao'] * 100) / df_merged['cr']
).fillna(0)
df_merged['percentual_ch_extensao'] = (
    (df_merged['ch_extensao'] * 100) / df_merged['ch']
).fillna(0)
# formata o percentual para até 2 casas decimais
df_merged['percentual_cr_extensao'] =
df_merged['percentual_cr_extensao'].round(2)
df_merged['percentual_ch_extensao'] =
df_merged['percentual_ch_extensao'].round(2)
# substitui valores infinitos por <0.01
df_merged.replace([np.inf, -np.inf], 0, inplace=True)
# separa as colunas de interesse
df_resultado =
df_merged[['campus','cep','ano','tipo_curso','percentual_cr_extensao','percentual_ch_
extensao']]
return df_resultado

```

percentual de créditos ou carga horária destinados à extensão no currículo de nível médio técnico por campus e ano

percentual de créditos ou carga horária destinados à extensão no currículo de graduação por campus e ano

percentual de créditos ou carga horária destinados à extensão no currículo de especialização por campus e ano

percentual de créditos ou carga horária destinados à extensão no currículo de mestrado por campus e ano

percentual de créditos ou carga horária destinados à extensão no currículo de doutorado por campus e ano

percentual de créditos ou carga horária destinados à extensão no currículo de pós-graduação (mestrado e doutorado) por campus e ano

```

transforma_percentual_cr_ch_por_tipo_curso = PythonOperator(
    task_id = 'transforma_percentual_cr_ch_por_tipo_curso',
    python_callable=calcula_percentual_cr_ch_por_tipo_curso
)

```

def calcula_percentual_cr_ch(ti) -> pd.DataFrame:

#Obtém o resultado das queries

result_cr_ch_curriculo =

ti.xcom_pull(task_ids='extrai_total_creditos_ch_curriculo')

```

        result_cr_ch_extensao =
ti.xcom_pull(task_ids='extrai_creditos_ch_curriculo_extensao')
        #Cria dataframes pandas
        df_cr_ch_extensao = pd.DataFrame(result_cr_ch_extensao,
columns=['campus','cep','ano','cr_extensao','ch_extensao'])
        df_cr_ch_curriculo = pd.DataFrame(result_cr_ch_curriculo,
columns=['campus','cep','ano','cr','ch'])
        #Mescla os dois dataframes
        df_merged = pd.merge(
        df_cr_ch_extensao,
        df_cr_ch_curriculo,
        on=['campus','cep','ano'],
        how='left'
        ).fillna(0)
        #Calcula o percentual
        df_merged['percentual_cr_extensao'] = (
            (df_merged['cr_extensao'] * 100) / df_merged['cr']
        ).fillna(0)
        df_merged['percentual_ch_extensao'] = (
            (df_merged['ch_extensao'] * 100) / df_merged['ch']
        ).fillna(0)
        # formata o percentual para até 2 casas decimais
        df_merged['percentual_cr_extensao'] =
df_merged['percentual_cr_extensao'].round(2)
        df_merged['percentual_ch_extensao'] =
df_merged['percentual_ch_extensao'].round(2)
        # substitui valores infinitos por <0.01
        df_merged.replace([np.inf, -np.inf], 0, inplace=True)
        # separa as colunas de interesse
        df_resultado =
df_merged[['campus','cep','ano','percentual_cr_extensao','percentual_ch_extensao']]
        return df_resultado

```

percentual de créditos ou carga horária de ações de extensão no currículo de cursos por campus e ano

```

transforma_percentual_cr_ch = PythonOperator(
    task_id = 'transforma_percentual_cr_ch',
    python_callable=calcula_percentual_cr_ch
)

```

def calcula_percentual_ppcs_extensao(ti) -> pd.DataFrame:

#Obtém o resultado da query

result_ppcs_extensao =

```
ti.xcom_pull(task_ids='extrai_cursos_com_extensao_ppcs')
```

```
result_ppcs_total = ti.xcom_pull(task_ids='extrai_total_ppcs_cursos_ativos')
```

#Cria um dataframe pandas

```

df_ppcs_extensao = pd.DataFrame(result_ppcs_extensao,
columns=['campus','cep','ano','tipo_curso','ppc_extensao'])

```

```

df_ppcs_total = pd.DataFrame(result_ppcs_total,
columns=['campus','cep','ano','tipo_curso','ppc'])
#Mapeia o cabeçalho do arquivo CSV
#Mescla os dois dataframes
df_merged_ppc = pd.merge(
df_ppcs_extensao,
df_ppcs_total,
on=['campus','cep','ano','tipo_curso'],
how='left'
).fillna(0)
#Calcula o percentual
df_merged_ppc['percentual_ppc'] = (
(df_merged_ppc['ppc_extensao'] * 100) / df_merged_ppc['ppc']
).fillna(0)
# formata o percentual para até 2 casas decimais
df_merged_ppc['percentual_ppc'] = df_merged_ppc['percentual_ppc'].round(2)
# substitui valores infinitos por <0.01
df_merged_ppc.replace([np.inf, -np.inf], 0, inplace=True)
# separa as colunas de interesse
df_resultado_ppc =
df_merged_ppc[['campus','cep','ano','tipo_curso','percentual_ppc']]
return df_resultado_ppc

```

Percentual de PPCs com curricularização da extensão e atividades complementares por campus e ano

```

transforma_percentual_ppcs_extensao = PythonOperator(
task_id='transforma_percentual_ppcs_extensao',
python_callable=calcula_percentual_ppcs_extensao
)

```

```

def transformacao_info():
logging.info("Iniciando Transformação de Dados")

```

```

inicia_transformacao = PythonOperator(
task_id='inicia_transformacao',
python_callable=transformacao_info
)

```

```

def cria_diretorio():
#cria os diretórios se não existirem
if not os.path.exists(datalake_extensao_dir):
os.makedirs(datalake_extensao_dir)
elif not os.path.exists(datalake_curricularizacao_dir):
os.makedirs(datalake_curricularizacao_dir)

```

```

cria_datalake_curricularizacao_extensao = PythonOperator(
task_id='cria_datalake_curricularizacao_extensao',
python_callable=cria_diretorio,
)

```

```

def cria_planilha_discentes_em_atividade_curricular_extensao(ti):
    #Obtém o resultado da query
    result_discentes_em_atividade_curricular_extensao =
ti.xcom_pull(task_ids='extraí_discentes_em_atividade_curricular_extensao')
    #Cria um dataframe pandas
    df_discentes_em_atividade_curricular_extensao =
pd.DataFrame(result_discentes_em_atividade_curricular_extensao)
    #Mapeia o cabeçalho do arquivo CSV
    cabecalho_atividades_por_curricularizacao = ['Câmpus','CEP','Ano','Discentes
em Atividade Curricular Extensionista']
    #Exporta para CSV
    if df_discentes_em_atividade_curricular_extensao.empty:
        df_discentes_em_atividade_curricular_extensao =
pd.DataFrame(columns=cabecalho_atividades_por_curricularizacao)

df_discentes_em_atividade_curricular_extensao.to_csv(csv_discentes_em_atividade
_curricular_extensao, index=False, encoding=encoding, sep=separador)
    else:

df_discentes_em_atividade_curricular_extensao.to_csv(csv_discentes_em_atividade
_curricular_extensao, index=False,
header=cabecalho_atividades_por_curricularizacao, encoding=encoding,
sep=separador)

carrega_discentes_em_atividade_curricular_extensao = PythonOperator(
    task_id='carrega_discentes_em_atividade_curricular_extensao',
    python_callable=cria_planilha_discentes_em_atividade_curricular_extensao
)

""" def cria_planilha_creditos_ch_curriculo_extensao_por_tipo_curso(ti):
    #Obtém o resultado da query
    result_creditos_ch_curriculo_extensao_por_tipo_curso =
ti.xcom_pull(task_ids='extraí_creditos_ch_curriculo_extensao_por_tipo_curso')
    #Cria um dataframe pandas
    df_creditos_ch_curriculo_extensao_por_tipo_curso =
pd.DataFrame(result_creditos_ch_curriculo_extensao_por_tipo_curso)
    #Mapeia o cabeçalho do arquivo CSV
    cabecalho_creditos_ch_curriculo_extensao_por_tipo_curso = ['Câmpus',
'CEP','Ano','Tipo de Curso','Créditos destinados a Extensão','Carga Horária
destinada a Extensão']
    #Exporta para CSV
    if df_creditos_ch_curriculo_extensao_por_tipo_curso.empty:
        df_creditos_ch_curriculo_extensao_por_tipo_curso =
pd.DataFrame(columns=cabecalho_creditos_ch_curriculo_extensao_por_tipo_curso)

df_creditos_ch_curriculo_extensao_por_tipo_curso.to_csv(csv_creditos_ch_curriculo
_extensao_por_tipo_curso, index=False, encoding=encoding, sep=separador)
    else:

```



```
df_credits_ch_curriculo_extensao_por_tipo_curso.to_csv(csv_credits_ch_curriculo_extensao_por_tipo_curso, index=False, header=cabecalho_credits_ch_curriculo_extensao_por_tipo_curso, encoding=encoding, sep=separador)
```

```
carrega_credits_ch_curriculo_extensao_por_tipo_curso = PythonOperator(task_id='carrega_credits_ch_curriculo_extensao_por_tipo_curso',
```

```
python_callable=cria_planilha_credits_ch_curriculo_extensao_por_tipo_curso) """
```

```
""" def cria_planilha_total_credits_ch_curriculo_por_tipo_curso(ti):  
    #Obtém o resultado da query  
    result_total_credits_ch_curriculo_por_tipo_curso =  
ti.xcom_pull(task_ids='extraí_total_credits_ch_curriculo_por_tipo_curso')  
    #Cria um dataframe pandas  
    df_total_credits_ch_curriculo_por_tipo_curso =  
pd.DataFrame(result_total_credits_ch_curriculo_por_tipo_curso)  
    #Mapeia o cabeçalho do arquivo CSV  
    cabecalho_total_credits_ch_curriculo_por_tipo_curso = ['Câmpus',  
'CEP', 'Ano', 'Tipo de Curso', 'Créditos', 'Carga Horária']  
    #Exporta para CSV  
    if df_total_credits_ch_curriculo_por_tipo_curso.empty:  
        df_total_credits_ch_curriculo_por_tipo_curso =  
pd.DataFrame(columns=cabecalho_total_credits_ch_curriculo_por_tipo_curso)
```

```
df_total_credits_ch_curriculo_por_tipo_curso.to_csv(csv_total_credits_ch_curriculo_por_tipo_curso, index=False, encoding=encoding, sep=separador)  
else:
```

```
df_total_credits_ch_curriculo_por_tipo_curso.to_csv(csv_total_credits_ch_curriculo_por_tipo_curso, index=False, header=cabecalho_total_credits_ch_curriculo_por_tipo_curso, encoding=encoding, sep=separador)
```

```
carrega_total_credits_ch_curriculo_por_tipo_curso = PythonOperator(task_id='carrega_total_credits_ch_curriculo_por_tipo_curso',  
python_callable=cria_planilha_total_credits_ch_curriculo_por_tipo_curso) """
```

```
""" def cria_planilha_credits_ch_curriculo_extensao(ti):  
    #Obtém o resultado da query  
    result_credits_ch_curriculo_extensao =  
ti.xcom_pull(task_ids='extraí_credits_ch_curriculo_extensao')  
    #Cria um dataframe pandas  
    df_credits_ch_curriculo_extensao =  
pd.DataFrame(result_credits_ch_curriculo_extensao)  
    #Mapeia o cabeçalho do arquivo CSV
```



```
cabecalho_credits_ch_curriculo_extensao = ['Câmpus', 'CEP', 'Ano', 'Créditos  
destinados a Extensão', 'Carga Horária destinada a Extensão']
```

```
#Exporta para CSV
```

```
if df_credits_ch_curriculo_extensao.empty:
```

```
df_credits_ch_curriculo_extensao =
```

```
pd.DataFrame(columns=cabecalho_credits_ch_curriculo_extensao)
```

```
df_credits_ch_curriculo_extensao.to_csv(csv_credits_ch_curriculo_extensao,  
index=False, encoding=encoding, sep=separador)
```

```
else:
```

```
df_credits_ch_curriculo_extensao.to_csv(csv_credits_ch_curriculo_extensao,  
index=False, header=cabecalho_credits_ch_curriculo_extensao,  
encoding=encoding, sep=separador)
```

```
carrega_credits_ch_curriculo_extensao = PythonOperator(  
task_id='carrega_credits_ch_curriculo_extensao',  
python_callable=cria_planilha_credits_ch_curriculo_extensao  
) """
```

```
""" def cria_planilha_total_credits_ch_curriculo(ti):  
#Obtém o resultado da query  
result_total_credits_ch_curriculo =  
ti.xcom_pull(task_ids='extrai_total_credits_ch_curriculo')  
#Cria um dataframe pandas  
df_total_credits_ch_curriculo =  
pd.DataFrame(result_total_credits_ch_curriculo)  
#Mapeia o cabeçalho do arquivo CSV  
cabecalho_total_credits_ch_curriculo = ['Câmpus',  
'CEP', 'Ano', 'Créditos', 'Carga Horária']  
#Exporta para CSV  
if df_total_credits_ch_curriculo.empty:  
df_total_credits_ch_curriculo =  
pd.DataFrame(columns=cabecalho_total_credits_ch_curriculo)  
df_total_credits_ch_curriculo.to_csv(csv_total_credits_ch_curriculo,  
index=False, encoding=encoding, sep=separador)  
else:  
df_total_credits_ch_curriculo.to_csv(csv_total_credits_ch_curriculo,  
index=False, header=cabecalho_total_credits_ch_curriculo, encoding=encoding,  
sep=separador)
```

```
carrega_total_credits_ch_curriculo = PythonOperator(  

```

```
task_id='carrega_credits_ch_curriculo_carrega_total_credits_ch_curriculoextensa  
o',
```

```
python_callable=cria_planilha_total_credits_ch_curriculo  
) """
```

```
def cria_planilha_atividades_extensao_curriculo_por_tipo_curso(ti):
```

```

        #Obtém o resultado da query
        result_atividades_extensao_curriculo_por_tipo_curso =
ti.xcom_pull(task_ids='extraí_atividades_extensao_curriculo_por_tipo_curso')
        #Cria um dataframe pandas
        df_atividades_extensao_curriculo_por_tipo_curso =
pd.DataFrame(result_atividades_extensao_curriculo_por_tipo_curso)
        #Mapeia o cabeçalho do arquivo CSV
        cabecalho_total_credits_ch_curriculo = ['Câmpus', 'CEP', 'Ano', 'Tipo de
Curso', 'Ações de Extensão no Currículo']
        #Exporta para CSV
        if df_atividades_extensao_curriculo_por_tipo_curso.empty:
            df_atividades_extensao_curriculo_por_tipo_curso =
pd.DataFrame(columns=cabecalho_total_credits_ch_curriculo)
        else:

df_atividades_extensao_curriculo_por_tipo_curso.to_csv(csv_atividades_extensao_c
urriculo_por_tipo_curso, index=False,
header=cabecalho_total_credits_ch_curriculo, encoding=encoding, sep=separador)

        carrega_atividades_extensao_curriculo_por_tipo_curso = PythonOperator(
            task_id='carrega_atividades_extensao_curriculo_por_tipo_curso',
            python_callable=cria_planilha_atividades_extensao_curriculo_por_tipo_curso
        )

        """ def cria_planilha_cursos_com_extensao_ppcs(ti):
        #Obtém o resultado da query
        result_cursos_com_extensao_ppcs =
ti.xcom_pull(task_ids='extraí_cursos_com_extensao_ppcs')
        #Cria um dataframe pandas
        df_cursos_com_extensao_ppcs =
pd.DataFrame(result_cursos_com_extensao_ppcs)
        #Mapeia o cabeçalho do arquivo CSV
        cabecalho_cursos_com_extensao_ppcs = ['Câmpus', 'CEP', 'Ano', 'Tipo de
Curso', 'PPCs com Extensão']
        #Exporta para CSV
        if df_cursos_com_extensao_ppcs.empty:
            df_cursos_com_extensao_ppcs =
pd.DataFrame(columns=cabecalho_cursos_com_extensao_ppcs)

        df_cursos_com_extensao_ppcs.to_csv(csv_cursos_ativos_com_extensao_ppc,
index=False, encoding=encoding, sep=separador)
        else:

df_cursos_com_extensao_ppcs.to_csv(csv_cursos_ativos_com_extensao_ppc,
index=False, header=cabecalho_cursos_com_extensao_ppcs, encoding=encoding,
sep=separador)

        carrega_cursos_com_extensao_ppcs = PythonOperator(
            task_id='carrega_cursos_com_extensao_ppcs',

```

```

python_callable=cria_planilha_cursos_com_extensao_ppcs
) """

""" def cria_planilha_total_ppcs_cursos_ativos(ti):
#Obtém o resultado da query
result_total_ppcs_cursos_ativos =
ti.xcom_pull(task_ids='extraí_total_ppcs_cursos_ativos')
#Cria um dataframe pandas
df_total_ppcs_cursos_ativos =
pd.DataFrame(result_total_ppcs_cursos_ativos)
#Mapeia o cabeçalho do arquivo CSV
cabecalho_total_ppcs_cursos_ativos = ['Câmpus','CEP','Ano de
Cadastro','Tipo de Curso','PPCs']
#Exporta para CSV
if df_total_ppcs_cursos_ativos.empty:
df_total_ppcs_cursos_ativos =
pd.DataFrame(columns=cabecalho_total_ppcs_cursos_ativos)
df_total_ppcs_cursos_ativos.to_csv(csv_total_ppcs_cursos_ativos,
index=False, encoding=encoding, sep=separador)
else:
df_total_ppcs_cursos_ativos.to_csv(csv_total_ppcs_cursos_ativos,
index=False, header=cabecalho_total_ppcs_cursos_ativos, encoding=encoding,
sep=separador)

carrega_total_ppcs_cursos_ativos = PythonOperator(
task_id='carrega_total_ppcs_cursos_ativos',
python_callable=cria_planilha_total_ppcs_cursos_ativos
) """

def cria_planilha_percentual_cr_ch_por_tipo_curso(ti):
#Obtém o resultado da query
result_percentual_cr_ch_por_tipo_curso =
ti.xcom_pull(task_ids='transforma_percentual_cr_ch_por_tipo_curso')
#Cria um dataframe pandas
df_percentual_cr_ch_por_tipo_curso =
pd.DataFrame(result_percentual_cr_ch_por_tipo_curso)
#Mapeia o cabeçalho do arquivo CSV
cabecalho_percentual_cr_ch_por_tipo_curso = ['Câmpus', 'CEP','Ano','Tipo de
Curso','Créditos destinados a Extensão(%)','Carga Horária destinada a
Extensão(%)']
#Exporta para CSV
if df_percentual_cr_ch_por_tipo_curso.empty:
df_percentual_cr_ch_por_tipo_curso =
pd.DataFrame(columns=cabecalho_percentual_cr_ch_por_tipo_curso)

df_percentual_cr_ch_por_tipo_curso.to_csv(csv_percentual_creditos_ch_curriculo_e
xtensao_por_tipo_curso, index=False, encoding=encoding, sep=separador)
else:

```

```
df_percentual_cr_ch_por_tipo_curso.to_csv(csv_percentual_creditos_ch_curriculo_e
xtensao_por_tipo_curso, index=False,
header=cabecalho_percentual_cr_ch_por_tipo_curso, encoding=encoding,
sep=separador)
```

```
carrega_percentual_cr_ch_por_tipo_curso = PythonOperator(
task_id='carrega_percentual_cr_ch_por_tipo_curso',
python_callable=cria_planilha_percentual_cr_ch_por_tipo_curso
)
```

```
def cria_planilha_percentual_cr_ch(ti):
#Obtém o resultado da query
result_percentual_cr_ch =
ti.xcom_pull(task_ids='transforma_percentual_cr_ch')
#Cria um dataframe pandas
df_percentual_cr_ch = pd.DataFrame(result_percentual_cr_ch)
#Mapeia o cabeçalho do arquivo CSV
cabecalho_percentual_cr_ch = ['Câmpus', 'CEP', 'Ano', 'Créditos destinados a
Extensão(%)', 'Carga Horária destinada a Extensão(%)']
#Exporta para CSV
if df_percentual_cr_ch.empty:
df_percentual_cr_ch = pd.DataFrame(columns=cabecalho_percentual_cr_ch)
df_percentual_cr_ch.to_csv(csv_percentual_creditos_ch_curriculo_extensao,
index=False, encoding=encoding, sep=separador)
else:
df_percentual_cr_ch.to_csv(csv_percentual_creditos_ch_curriculo_extensao,
index=False, header=cabecalho_percentual_cr_ch, encoding=encoding,
sep=separador)
```

```
carrega_percentual_cr_ch = PythonOperator(
task_id='carrega_percentual_cr_ch',
python_callable=cria_planilha_percentual_cr_ch
)
```

```
def cria_planilha_percentual_ppcs_extensao(ti):
#Obtém o resultado da query
result_percentual_ppcs_extensao =
ti.xcom_pull(task_ids='transforma_percentual_ppcs_extensao')
#Cria um dataframe pandas
df_percentual_ppcs_extensao =
pd.DataFrame(result_percentual_ppcs_extensao)
#Mapeia o cabeçalho do arquivo CSV
cabecalho_percentual_ppcs_extensao = ['Câmpus', 'CEP', 'Ano', 'Tipo de
Curso', 'PPCs com Extensão(%)']
#Exporta para CSV
if df_percentual_ppcs_extensao.empty:
df_percentual_ppcs_extensao =
pd.DataFrame(columns=cabecalho_percentual_ppcs_extensao)
```

```

df_percentual_ppcs_extensao.to_csv(csv_percentual_ppcs_extensao,
index=False, encoding=encoding, sep=separador)
else:
df_percentual_ppcs_extensao.to_csv(csv_percentual_ppcs_extensao,
index=False, header=cabecalho_percentual_ppcs_extensao, encoding=encoding,
sep=separador)

```

```

carrega_percentual_ppcs_extensao = PythonOperator(
task_id='carrega_percentual_ppcs_extensao',
python_callable=cria_planilha_percentual_ppcs_extensao
)

```

```

""" envia_arquivos_email = EmailOperator(
task_id='envia_arquivos_email',
to=Variable.get("IFSC_email_extensao"),# email definido em Admin ->

```

Variables

```

subject='Indicadores de Extensão - Curricularização',
html_content="Segue em anexo os arquivos",
files=lista_arquivos,
conn_id="IFSC_SMTP", # Conexão SMTP definida em Admin -> Connections
) """

```

Definir as dependências entre as tasks

```

[extrai_discentes_em_atividade_curricular_extensao,
extrai_creditos_ch_curriculo_extensao_por_tipo_curso,
extrai_total_creditos_ch_curriculo_por_tipo_curso,
extrai_creditos_ch_curriculo_extensao,
extrai_total_creditos_ch_curriculo,
extrai_atividades_extensao_curriculo_por_tipo_curso,
extrai_cursos_com_extensao_ppcs,
extrai_total_ppcs_cursos_ativos] >> inicia_transformacao >>
[transforma_percentual_cr_ch_por_tipo_curso,
transforma_percentual_cr_ch,
transforma_percentual_ppcs_extensao] >>

```

```

cria_datalake_curricularizacao_extensao >>

```

```

[carrega_discentes_em_atividade_curricular_extensao,

```

```

#carrega_creditos_ch_curriculo_extensao_por_tipo_curso,

```

```

#carrega_total_creditos_ch_curriculo_por_tipo_curso,

```

```

#carrega_creditos_ch_curriculo_extensao,

```

```

#carrega_total_creditos_ch_curriculo,

```

```

carrega_atividades_extensao_curriculo_por_tipo_curso,

```

```

#carrega_cursos_com_extensao_ppcs,

```

```
#carrega_total_ppcs_cursos_ativos,  
  
carrega_percentual_cr_ch_por_tipo_curso,  
carrega_percentual_cr_ch,  
  
carrega_percentual_ppcs_extensao]
```

ORÇAMENTO

```
from airflow import DAG
from airflow.operators.python import PythonOperator
from airflow.providers.postgres.operators.postgres import PostgresOperator
from airflow.providers.smtp.operators.smtp import EmailOperator
from airflow.models import Variable
from airflow.utils.dates import days_ago
from datetime import datetime
import pandas as pd
import os
```

```
# Caminho do arquivo CSV de saída
datalake_extensao_dir = Variable.get("datalake")+ 'extensao/'
datalake_orcamento_dir = datalake_extensao_dir+'orcamento/'
encoding = "utf-8"
separador = ","
lista_arquivos = []
```

```
with DAG(
    dag_id='IFSC_EXTENSAO_SIGAA_ORCAMENTO',
    start_date=days_ago(1),
    schedule_interval='@daily',
    catchup=False,
) as dag:
```

```
    # NÃO TEM REGISTRO NO SIGAA
    # percentual de recursos financeiros do orçamento anual público aplicados
    em extensão
    # quantitativo de recurso financeiro orçamentário total por câmpus e ano
    # quantitativo de recursos financeiros do orçamento anual público obtido para
    extensão por câmpus e ano
    # quantitativo de recursos financeiros do orçamento anual público aplicados
    em extensão por câmpus e ano
    # quantitativo de recursos para extensão captados via prestação de serviços
    acadêmicos especializados por câmpus e ano
    # quantitativo de recursos para extensão captados via editais externos por
    câmpus e ano
    # quantitativo de recurso aplicado por edital interno de extensão por câmpus e
    ano
    # quantitativo de recurso repassado aos coordenadores de atividade de
    extensão por câmpus e ano
    # quantitativo de recurso repassado aos bolsistas discentes por câmpus e ano
```

```
def cria_diretorio():
    #cria os diretórios se não existirem
    if not os.path.exists(datalake_extensao_dir):
        os.makedirs(datalake_extensao_dir)
    elif not os.path.exists(datalake_orcamento_dir):
```

```

os.makedirs(datalake_orcamento_dir)

cria_datalake_orcamento_extensao = PythonOperator(
    task_id='cria_datalake_orcamento_extensao',
    python_callable=cria_diretorio,
)

# envia_arquivos_email = EmailOperator(
#     task_id='envia_arquivos_email',
#     to=Variable.get("IFSC_email_extensao"),# email definido em Admin ->
Variables
#     subject='Indicadores de Extensão - Orçamento',
#     html_content="Segue em anexo os arquivos",
#     files=lista_arquivos,
#     conn_id="IFSC_SMTP", # Conexão SMTP definida em Admin ->
Connections
# )

# Definir as dependências entre as tasks
cria_datalake_orcamento_extensao

```


PÚBLICO ALVO

```
from airflow import DAG
from airflow.operators.python import PythonOperator
from airflow.providers.postgres.operators.postgres import PostgresOperator
from airflow.providers.smtp.operators.smtp import EmailOperator
from airflow.models import Variable
from airflow.utils.dates import days_ago
from datetime import datetime
import pandas as pd
import numpy as np
import logging
import os

# Caminho do arquivo CSV de saída
datalake_extensao_dir = Variable.get("datalake")+ 'extensao/'
datalake_publico_alvo_dir = datalake_extensao_dir+'publico_alvo/'
encoding = "utf-8"
separador = ","
csv_demanda_publico_externo =
datalake_publico_alvo_dir+'IFSC_EXTENSAO_SIGAA_DEMANDA_PUBLICO_EXT
ERNO.csv'
csv_publico_envolvido =
datalake_publico_alvo_dir+'IFSC_EXTENSAO_SIGAA_PUBLICO_ENVOLVIDO.csv'
csv_publico_envolvido_por_prestacao_servico =
datalake_publico_alvo_dir+'IFSC_EXTENSAO_SIGAA_PUBLICO_ENVOLVIDO_PO
R_PRESTACAO_SERVICO.csv'
csv_publico_envolvido_por_programa_projeto =
datalake_publico_alvo_dir+'IFSC_EXTENSAO_SIGAA_PUBLICO_ENVOLVIDO_PO
R_PROGRAMA_PROJETO.csv'
csv_publico_envolvido_por_curso_evento =
datalake_publico_alvo_dir+'IFSC_EXTENSAO_SIGAA_PUBLICO_ENVOLVIDO_PO
R_CURSO_EVENTO.csv'
csv_capitacao_publico_EJA =
datalake_publico_alvo_dir+'IFSC_EXTENSAO_SIGAA_CAPACITACAO_PUBLICO_
EJA.csv'
csv_capitacao_publico_PCD =
datalake_publico_alvo_dir+'IFSC_EXTENSAO_SIGAA_CAPACITACAO_PUBLICO_
PCD.csv'
#csv_discentes_graduacao =
datalake_publico_alvo_dir+'IFSC_EXTENSAO_SIGAA_DISCENTES_GRADUACAO.
csv'
csv_docentes_rede_publica_capacitados =
datalake_publico_alvo_dir+'IFSC_EXTENSAO_SIGAA_DOCENTES_REDE_PUBLI
CA_CAPACITADOS.csv'
#csv_atividades_destinadas_publico_vulneravel =
datalake_publico_alvo_dir+'IFSC_EXTENSAO_SIGAA_ATIVIDADES_DESTINADAS
_PUBLICO_VULNERAVEL.csv'
```

```

csv_percentual_atividades_destinadas_publico_vulneravel =
datalake_publico_alvo_dir+'IFSC_EXTENSAO_SIGAA_PERCENTUAL_ATIVIDADE
S_DESTINADAS_PUBLICO_VULNERAVEL.csv'
#csv_atividades_destinadas_escolas_publicas =
datalake_publico_alvo_dir+'IFSC_EXTENSAO_SIGAA_ATIVIDADES_DESTINADAS
_ESCOLAS_PUBLICAS.csv'
csv_percentual_atividades_destinadas_escolas_publicas =
datalake_publico_alvo_dir+'IFSC_EXTENSAO_SIGAA_PERCENTUAL_ATIVIDADE
S_DESTINADAS_ESCOLAS_PUBLICAS.csv'
csv_municipios_envolvidos =
datalake_publico_alvo_dir+'IFSC_EXTENSAO_SIGAA_MUNICIPIOS_ENVOLVIDOS
.csv'
csv_proporcao_publico_extensao_por_discente_graduacao =
datalake_publico_alvo_dir+'IFSC_EXTENSAO_SIGAA_PUBLICO_ENVOLVIDO_PO
R_DISCENTE_GRADUACAO.csv'
lista_arquivos = [csv_demanda_publico_externo, csv_publico_envolvido,
csv_publico_envolvido_por_prestacao_servico,
csv_publico_envolvido_por_programa_projeto,
csv_publico_envolvido_por_curso_evento, csv_capacitacao_publico_EJA,
csv_capacitacao_publico_PCD, csv_docentes_rede_publica_capacitados,
csv_percentual_atividades_destinadas_publico_vulneravel,
csv_percentual_atividades_destinadas_escolas_publicas,
csv_municipios_envolvidos,csv_proporcao_publico_extensao_por_discente_graduac
ao]

```

```

with DAG(
    dag_id='IFSC_EXTENSAO_SIGAA_PUBLICO_ALVO',
    start_date=days_ago(1),
    schedule_interval='@daily',
    catchup=False,
) as dag:

```

quantitativo de ações de extensão demandadas pelo público externo por
câmpus e ano

```

    extrai_demanda_publico_externo = PostgresOperator(
        task_id='extrai_demanda_publico_externo',
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
        sql = Variable.get("script_SIGAA_IFSC_consulta_demanda_publico_externo"),
# script definido em Admin -> Variables
        #do_xcom_push=True,
    )

```

quantitativo de pessoas envolvidas por atividades de extensão por câmpus
e ano

```

    extrai_publico_envolvido = PostgresOperator(
        task_id='extrai_publico_envolvido',
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections

```

```
sql = Variable.get("script_SIGAA_IFSC_consulta_publico_envolvido"), # script
definido em Admin -> Variables
#do_xcom_push=True,
)
```

quantitativo de pessoas alcançadas por atividades de modalidade prestação de serviço por câmpus e ano

```
extrai_publico_envolvido_por_prestacao_servico = PostgresOperator(
task_id='extrai_publico_envolvido_por_prestacao_servico',
postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
sql =
Variable.get("script_SIGAA_IFSC_consulta_publico_envolvido_por_prestacao_servic
o"), # script definido em Admin -> Variables
#do_xcom_push=True,
)
```

quantitativo de pessoas alcançadas por atividades de modalidade programas e projetos por câmpus e ano

```
extrai_publico_envolvido_por_programa_projeto = PostgresOperator(
task_id='extrai_publico_envolvido_por_programa_projeto',
postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
sql =
Variable.get("script_SIGAA_IFSC_consulta_publico_envolvido_por_programa_projet
o"), # script definido em Admin -> Variables
#do_xcom_push=True,
)
```

quantitativo de pessoas alcançadas por atividades de modalidade cursos e eventos por câmpus e ano

```
extrai_publico_envolvido_por_curso_evento = PostgresOperator(
task_id='extrai_publico_envolvido_por_curso_evento',
postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
sql =
Variable.get("script_SIGAA_IFSC_consulta_publico_envolvido_por_curso_evento"),
# script definido em Admin -> Variables
#do_xcom_push=True,
)
```

quantitativo de ações de extensão para capacitação de população EJA por câmpus e ano

```
extrai_capitacao_publico_EJA = PostgresOperator(
task_id='extrai_capitacao_publico_EJA',
postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
sql = Variable.get("script_SIGAA_IFSC_consulta_capitacao_publico_EJA"),
# script definido em Admin -> Variables
```

```

#do_xcom_push=True,
)

# quantitativo de ações de extensão para capacitação de população PCD por
câmpus e ano
extrai_capitacao_publico_PCD = PostgresOperator(
    task_id='extrai_capitacao_publico_PCD',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql = Variable.get("script_SIGAA_IFSC_consulta_capitacao_publico_PCD"),
# script definido em Admin -> Variables
    #do_xcom_push=True,
)

# proporção de pessoas envolvidas por discentes de graduação por meio de
ações de extensão por câmpus e ano (considerar o quantitativo de público envolvido
pela consulta abaixo de discentes de graduacao)
extrai_discentes_graduacao = PostgresOperator(
    task_id='extrai_discentes_graduacao',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql = Variable.get("script_SIGAA_IFSC_consulta_discentes_graduacao"), #
script definido em Admin -> Variables
    #do_xcom_push=True,
)

# quantitativo de docentes da rede pública capacitados por programas e
projetos de formação continuada por câmpus e ano
extrai_docentes_rede_publica_capitados = PostgresOperator(
    task_id='extrai_docentes_rede_publica_capitados',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql =
Variable.get("script_SIGAA_IFSC_consulta_docentes_rede_publica_capitados"), #
script definido em Admin -> Variables
    #do_xcom_push=True,
)

# percentual de ações de extensão destinadas à inclusão de população
vulnerável por câmpus e ano (consultar a quantidade de ações de extensão na
categoria ATIVIDADES DE EXTENSÃO)
extrai_atividades_destinadas_publico_vulneravel = PostgresOperator(
    task_id='extrai_atividades_destinadas_publico_vulneravel',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql =
Variable.get("script_SIGAA_IFSC_consulta_atividades_destinadas_publico_vulnerav
el"), # script definido em Admin -> Variables
    #do_xcom_push=True,

```

```

)

# percentual de ações de extensão dirigidas às escolas públicas por câmpus
e ano (consultar a quantidade de ações de extensão na categoria ATIVIDADES DE
EXTENSÃO)
extrai_atividades_destinadas_escolas_publicas = PostgresOperator(
    task_id='extrai_atividades_destinadas_escolas_publicas',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql =
Variable.get("script_SIGAA_IFSC_consulta_atividades_destinadas_escolas_publicas
"), # script definido em Admin -> Variables
    #do_xcom_push=True,
)

# quantitativo total de atividades de extensão por câmpus e ano
extrai_quantitativo_atividades = PostgresOperator(
    task_id='extrai_quantitativo_atividades',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql =
Variable.get("script_SIGAA_IFSC_consulta_quantitativo_atividades_extensao"), #
script definido em Admin -> Variables
    #do_xcom_push=True,
)

# percentual de municípios envolvidos pelas ações extensionistas em relação
ao total de municípios que são abrangidos pela respectiva IES
# possível resgatar somente o quantitativo de municípios
extrai_municipios_envolvidos = PostgresOperator(
    task_id='extrai_municipios_envolvidos',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql =
Variable.get("script_SIGAA_IFSC_consulta_municipios_envolvidos_extensao"), #
script definido em Admin -> Variables
    #do_xcom_push=True,
)

def calcula_proporcao_publico_extensao_por_discente_graduacao(ti) ->
pd.DataFrame:
    #Obtém o resultado das queries
    result_publico_envolvido = ti.xcom_pull(task_ids='extrai_publico_envolvido')
    result_discentes_graduacao =
ti.xcom_pull(task_ids='extrai_discentes_graduacao')
    #Cria dataframes pandas
    df_publico_envolvido = pd.DataFrame(result_publico_envolvido,
columns=['campus','cep','ano','publico_extensao'])

```

```

df_discentes_graduacao = pd.DataFrame(result_discentes_graduacao,
columns=['campus','cep','ano','tipo_curso','discente_graduacao'])
#mescla os dataframes
df_merged = pd.merge(
df_discentes_graduacao,
df_publico_envolvido,
on=['campus','cep','ano'],
how='left'
).fillna(0)
#Calcula o percentual
df_merged['proporcao_publico_extensao_discente_graduacao'] = (
(df_merged['publico_extensao']) / df_merged['discente_graduacao']
).fillna(0)
# formata o percentual para até 2 casas decimais
df_merged['proporcao_publico_extensao_discente_graduacao'] =
df_merged['proporcao_publico_extensao_discente_graduacao'].round(2)
# substitui valores infinitos por <0.01
df_merged.replace([np.inf, -np.inf], 0, inplace=True)
# Cria um novo dataframe apenas com as colunas interessadas
df_resultado = df_merged[['campus', 'cep', 'ano','tipo_curso',
'proporcao_publico_extensao_discente_graduacao']]
return df_resultado

```

```

# proporção de pessoas envolvidas por discentes de graduação por meio de
ações de extensão por câmpus e ano
transforma_proporcao_publico_extensao_por_discente_graduacao =
PythonOperator(
task_id='transforma_proporcao_publico_extensao_por_discente_graduacao',

python_callable=calcula_proporcao_publico_extensao_por_discente_graduacao
)

```

```

def calcula_percentual_extensao_populacao_vulneravel(ti) ->
pd.DataFrame:
#Obtém o resultado da query
result_atividades_destinadas_publico_vulneravel =
ti.xcom_pull(task_ids='extrai_atividades_destinadas_publico_vulneravel')
result_quantitativo_atividades =
ti.xcom_pull(task_ids='extrai_quantitativo_atividades')
#Cria um dataframe pandas
df_atividades_vulneravel =
pd.DataFrame(result_atividades_destinadas_publico_vulneravel,
columns=['campus','cep','ano','atividades_vulneravel'])
df_atividades = pd.DataFrame(result_quantitativo_atividades,
columns=['campus','cep','ano','atividades'])
#mescla os dataframes
df_merged = pd.merge(
df_atividades_vulneravel,
df_atividades,

```

```

on=['campus','cep','ano'],
how='left'
).fillna(0)
#Calcula o percentual
df_merged['percentual_atividades_populacao_vulneravel'] = (
    (df_merged['atividades_vulneravel'] * 100) / df_merged['atividades']
).fillna(0)
# formata o percentual para até 2 casas decimais
df_merged['percentual_atividades_populacao_vulneravel'] =
df_merged['percentual_atividades_populacao_vulneravel'].round(2)
# substitui valores infinitos por <0.01
df_merged.replace([np.inf, -np.inf], 0, inplace=True)
# Cria um novo dataframe apenas com as colunas interessadas
df_resultado =
df_merged[['campus','cep','ano','percentual_atividades_populacao_vulneravel']]
return df_resultado

```

```

# percentual de ações de extensão destinadas à inclusão de população
vulnerável por câmpus e ano
transforma_percentual_extensao_populacao_vulneravel = PythonOperator(
    task_id='transforma_percentual_extensao_populacao_vulneravel',
    python_callable=calcula_percentual_extensao_populacao_vulneravel
)

```

```

def calcula_percentual_extensao_escolas_publicas(ti) -> pd.DataFrame:
    #Obtém o resultado da query
    result_atividades_escolas =
ti.xcom_pull(task_ids='extraí_atividades_destinadas_escolas_publicas')
    result_quantitativo_atividades =
ti.xcom_pull(task_ids='extraí_quantitativo_atividades')
    #Cria um dataframe pandas
    df_atividades_escolas= pd.DataFrame(result_atividades_escolas,
columns=['campus','cep','ano','atividades_escola'])
    df_atividades = pd.DataFrame(result_quantitativo_atividades,
columns=['campus','cep','ano','atividades'])
    #mescla os dataframes
    df_merged = pd.merge(
df_atividades_escolas,
df_atividades,
on=['campus','cep','ano'],
how='left'
).fillna(0)
    #Calcula o percentual
df_merged['percentual_atividades_escolas_publicas'] = (
    (df_merged['atividades_escola'] * 100) / df_merged['atividades']
).fillna(0)
    # formata o percentual para até 2 casas decimais

```



```

df_merged['percentual_atividades_escolas_publicas'] =
df_merged['percentual_atividades_escolas_publicas'].round(2)
# substitui valores infinitos por <0.01
df_merged.replace([np.inf, -np.inf], 0, inplace=True)
# Cria um novo dataframe apenas com as colunas interessadas
df_resultado =
df_merged[['campus','cep','ano','percentual_atividades_escolas_publicas']]
return df_resultado

```

percentual de ações de extensão dirigidas às escolas públicas por câmpus e ano

```

transforma_percentual_extensao_escolas_publicas = PythonOperator(
task_id='transforma_percentual_extensao_escolas_publicas',
python_callable=calcula_percentual_extensao_escolas_publicas
)

```

```

def transformacao_info():
logging.info("Iniciando Tranformação de Dados")

```

```

inicia_transformacao = PythonOperator(
task_id='inicia_transformacao',
python_callable=transformacao_info
)

```

```

def cria_diretorio():
#cria os diretórios se não existirem
if not os.path.exists(datalake_extensao_dir):
os.makedirs(datalake_extensao_dir)
elif not os.path.exists(datalake_publico_alvo_dir):
os.makedirs(datalake_publico_alvo_dir)

```

```

cria_datalake_publico_alvo_extensao = PythonOperator(
task_id='cria_datalake_publico_alvo_extensao',
python_callable=cria_diretorio,
)

```

```

def cria_planilhas_demanda_publico_externo(ti):
#Obtém o resultado da query
result_demanda_publico_externo =
ti.xcom_pull(task_ids='extrai_demanda_publico_externo')
#Cria um dataframe pandas
df_demanda_publico_externo =
pd.DataFrame(result_demanda_publico_externo)
#Mapea o cabeçalho do arquivo CSV
cabecalho_demanda_publico_externo = ['Câmpus', 'CEP','Ano','Ações
Demandadas pelo Público Externo']

```



```

#Exporta para CSV
if df_demanda_publico_externo.empty:
    df_demanda_publico_externo =
pd.DataFrame(columns=cabecalho_demanda_publico_externo)
    df_demanda_publico_externo.to_csv(csv_demanda_publico_externo,
index=False, encoding=encoding, sep=separador)
else:
    df_demanda_publico_externo.to_csv(csv_demanda_publico_externo,
index=False, header=cabecalho_demanda_publico_externo, encoding=encoding,
sep=separador)

```

```

carrega_demanda_publico_externo = PythonOperator(
task_id='carrega_demanda_publico_externo',
python_callable=cria_planilhas_demanda_publico_externo,
)

```

```

def cria_planilhas_publico_envolvido(ti):
#Obtém o resultado da query
result_publico_envolvido = ti.xcom_pull(task_ids='extrai_publico_envolvido')
#Cria um dataframe pandas
df_publico_envolvido = pd.DataFrame(result_publico_envolvido)
#Mapea o cabeçalho do arquivo CSV
cabecalho_publico_envolvido = ['Câmpus', 'CEP', 'Ano', 'Público Envolvido em
Ações']

```

```

#Exporta para CSV
if df_publico_envolvido.empty:
    df_publico_envolvido =
pd.DataFrame(columns=cabecalho_publico_envolvido)
    df_publico_envolvido.to_csv(csv_publico_envolvido, index=False,
encoding=encoding, sep=separador)
else:
    df_publico_envolvido.to_csv(csv_publico_envolvido, index=False,
header=cabecalho_publico_envolvido, encoding=encoding, sep=separador)

```

```

carrega_publico_envolvido = PythonOperator(
task_id='carrega_publico_envolvido',
python_callable=cria_planilhas_publico_envolvido,
)

```

```

def cria_planilhas_publico_envolvido_por_prestacao_servico(ti):
#Obtém o resultado da query
result_publico_envolvido_por_prestacao_servico =
ti.xcom_pull(task_ids='extrai_publico_envolvido_por_prestacao_servico')
#Cria um dataframe pandas
df_publico_envolvido_por_prestacao_servico =
pd.DataFrame(result_publico_envolvido_por_prestacao_servico)
#Mapea o cabeçalho do arquivo CSV
cabecalho_publico_envolvido_por_prestacao_servico = ['Câmpus',
'CEP', 'Ano', 'Público Envolvido em Prestações de Serviço']

```

```

#Exporta para CSV
if df_publico_envolvido_por_prestacao_servico.empty:
    df_publico_envolvido_por_prestacao_servico =
pd.DataFrame(columns=cabecalho_publico_envolvido_por_prestacao_servico)

df_publico_envolvido_por_prestacao_servico.to_csv(csv_publico_envolvido_por_pre
stacao_servico, index=False, encoding=encoding, sep=separador)
else:

df_publico_envolvido_por_prestacao_servico.to_csv(csv_publico_envolvido_por_pre
stacao_servico, index=False,
header=cabecalho_publico_envolvido_por_prestacao_servico, encoding=encoding,
sep=separador)

carrega_publico_envolvido_por_prestacao_servico = PythonOperator(
    task_id='carrega_publico_envolvido_por_prestacao_servico',
    python_callable=cria_planilhas_publico_envolvido_por_prestacao_servico,
)

def cria_planilhas_publico_envolvido_por_programa_projeto(ti):
    #Obtém o resultado da query
    result_publico_envolvido_por_programa_projeto =
ti.xcom_pull(task_ids='extrai_publico_envolvido_por_programa_projeto')
    #Cria um dataframe pandas
    df_publico_envolvido_por_programa_projeto =
pd.DataFrame(result_publico_envolvido_por_programa_projeto)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_publico_envolvido_por_programa_projeto = ['Câmpus',
'CEP','Ano','Público Envolvido em Programas e Projetos']
    #Exporta para CSV
    if df_publico_envolvido_por_programa_projeto.empty:
        df_publico_envolvido_por_programa_projeto =
pd.DataFrame(columns=cabecalho_publico_envolvido_por_programa_projeto)

df_publico_envolvido_por_programa_projeto.to_csv(csv_publico_envolvido_por_pro
grama_projeto, index=False, encoding=encoding, sep=separador)
else:

df_publico_envolvido_por_programa_projeto.to_csv(csv_publico_envolvido_por_pro
grama_projeto, index=False,
header=cabecalho_publico_envolvido_por_programa_projeto, encoding=encoding,
sep=separador)

carrega_publico_envolvido_por_programa_projeto = PythonOperator(
    task_id='carrega_publico_envolvido_por_programa_projeto',
    python_callable=cria_planilhas_publico_envolvido_por_programa_projeto,
)

def cria_planilhas_publico_envolvido_por_curso_evento(ti):

```

```

#Obtém o resultado da query
result_publico_envolvido_por_curso_evento =
ti.xcom_pull(task_ids='extrai_publico_envolvido_por_curso_evento')
#Cria um dataframe pandas
df_publico_envolvido_por_curso_evento =
pd.DataFrame(result_publico_envolvido_por_curso_evento)
#Mapea o cabeçalho do arquivo CSV
cabecalho_publico_envolvido_por_curso_evento = ['Câmpus',
'CEP','Ano','Público Envolvido em Cursos e Eventos']
#Exporta para CSV
if df_publico_envolvido_por_curso_evento.empty:
df_publico_envolvido_por_curso_evento =
pd.DataFrame(columns=cabecalho_publico_envolvido_por_curso_evento)

df_publico_envolvido_por_curso_evento.to_csv(csv_publico_envolvido_por_curso_e
vento, index=False, encoding=encoding, sep=separador)
else:

df_publico_envolvido_por_curso_evento.to_csv(csv_publico_envolvido_por_curso_e
vento, index=False, header=cabecalho_publico_envolvido_por_curso_evento,
encoding=encoding, sep=separador)

```

```

carrega_publico_envolvido_por_curso_evento = PythonOperator(
task_id='carrega_publico_envolvido_por_curso_evento',
python_callable=cria_planilhas_publico_envolvido_por_curso_evento,
)

def cria_planilhas_capacitacao_publico_EJA(ti):
#Obtém o resultado da query
result_capacitacao_publico_EJA =
ti.xcom_pull(task_ids='extrai_capacitacao_publico_EJA')
#Cria um dataframe pandas
df_capacitacao_publico_EJA =
pd.DataFrame(result_capacitacao_publico_EJA)
#Mapea o cabeçalho do arquivo CSV
cabecalho_capacitacao_publico_EJA = ['Câmpus', 'CEP','Ano','Público EJA
Capacitado com Extensão']
#Exporta para CSV
if df_capacitacao_publico_EJA.empty:
df_capacitacao_publico_EJA =
pd.DataFrame(columns=cabecalho_capacitacao_publico_EJA)
df_capacitacao_publico_EJA.to_csv(csv_capacitacao_publico_EJA,
index=False, encoding=encoding, sep=separador)
else:
df_capacitacao_publico_EJA.to_csv(csv_capacitacao_publico_EJA,
index=False, header=cabecalho_capacitacao_publico_EJA, encoding=encoding,
sep=separador)

```

```

carrega_capacitacao_publico_EJA = PythonOperator(
    task_id='carrega_capacitacao_publico_EJA',
    python_callable=cria_planilhas_capacitacao_publico_EJA,
)

def cria_planilhas_capacitacao_publico_PCD(ti):
    #Obtém o resultado da query
    result_capacitacao_publico_PCD =
ti.xcom_pull(task_ids='extraí_capacitacao_publico_PCD')
    #Cria um dataframe pandas
    df_capacitacao_publico_PCD =
pd.DataFrame(result_capacitacao_publico_PCD)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_capacitacao_publico_PCD = ['Câmpus', 'CEP','Ano','Público PCD
Capacitado com Extensão']
    #Exporta para CSV
    if df_capacitacao_publico_PCD.empty:
        df_capacitacao_publico_PCD =
pd.DataFrame(columns=cabecalho_capacitacao_publico_PCD)
        df_capacitacao_publico_PCD.to_csv(csv_capacitacao_publico_PCD,
index=False, encoding=encoding, sep=separador)
    else:
        df_capacitacao_publico_PCD.to_csv(csv_capacitacao_publico_PCD,
index=False, header=cabecalho_capacitacao_publico_PCD, encoding=encoding,
sep=separador)

carrega_capacitacao_publico_PCD = PythonOperator(
    task_id='carrega_capacitacao_publico_PCD',
    python_callable=cria_planilhas_capacitacao_publico_PCD,
)

""" def cria_planilhas_discentes_graduacao(ti):
    #Obtém o resultado da query
    result_discentes_graduacao =
ti.xcom_pull(task_ids='extraí_discentes_graduacao')
    #Cria um dataframe pandas
    df_discentes_graduacao = pd.DataFrame(result_discentes_graduacao)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_discentes_graduacao = ['Câmpus', 'CEP','Ano de Ingresso','Tipo
de Curso de Graduação', 'Discentes']
    #Exporta para CSV
    if df_discentes_graduacao.empty:
        df_discentes_graduacao =
pd.DataFrame(columns=cabecalho_discentes_graduacao)
        df_discentes_graduacao.to_csv(csv_discentes_graduacao, index=False,
encoding=encoding, sep=separador)
    else:
        df_discentes_graduacao.to_csv(csv_discentes_graduacao, index=False,
header=cabecalho_discentes_graduacao, encoding=encoding, sep=separador)

```

```

carrega_discentes_graduacao = PythonOperator(
    task_id='carrega_discentes_graduacao',
    python_callable=cria_planilhas_discentes_graduacao,
) """

def cria_planilhas_docentes_rede_publica_capacitados(ti):
    #Obtém o resultado da query
    result_docentes_rede_publica_capacitados =
ti.xcom_pull(task_ids='extraí_docentes_rede_publica_capacitados')
    #Cria um dataframe pandas
    df_docentes_rede_publica_capacitados =
pd.DataFrame(result_docentes_rede_publica_capacitados)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_docentes_rede_publica_capacitados = ['Câmpus', 'CEP', 'Ano',
'Docentes da Rede Pública Capacitados por Programas ou Projetos']
    #Exporta para CSV
    if df_docentes_rede_publica_capacitados.empty:
        df_docentes_rede_publica_capacitados =
pd.DataFrame(columns=cabecalho_docentes_rede_publica_capacitados)

df_docentes_rede_publica_capacitados.to_csv(csv_docentes_rede_publica_capacit
ados, index=False, encoding=encoding, sep=separador)
    else:

df_docentes_rede_publica_capacitados.to_csv(csv_docentes_rede_publica_capacit
ados, index=False, header=cabecalho_docentes_rede_publica_capacitados,
encoding=encoding, sep=separador)

carrega_docentes_rede_publica_capacitados = PythonOperator(
    task_id='carrega_docentes_rede_publica_capacitados',
    python_callable=cria_planilhas_docentes_rede_publica_capacitados,
)

""" def cria_planilhas_atividades_destinadas_publico_vulneravel(ti):
    #Obtém o resultado da query
    result_atividades_destinadas_publico_vulneravel =
ti.xcom_pull(task_ids='extraí_atividades_destinadas_publico_vulneravel')
    #Cria um dataframe pandas
    df_atividades_destinadas_publico_vulneravel =
pd.DataFrame(result_atividades_destinadas_publico_vulneravel)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_atividades_destinadas_publico_vulneravel = ['Câmpus',
'CEP', 'Ano', 'Ações Extensionistas Destinadas à População em Vulnerabilidade']
    #Exporta para CSV
    if df_atividades_destinadas_publico_vulneravel.empty:
        df_atividades_destinadas_publico_vulneravel =
pd.DataFrame(columns=cabecalho_atividades_destinadas_publico_vulneravel)

```

```
df_atividades_destinadas_publico_vulneravel.to_csv(csv_atividades_destinadas_publico_vulneravel, index=False, encoding=encoding, sep=separador)
else:
```

```
df_atividades_destinadas_publico_vulneravel.to_csv(csv_atividades_destinadas_publico_vulneravel, index=False,
header=cabecalho_atividades_destinadas_publico_vulneravel, encoding=encoding,
sep=separador)
```

```
carrega_atividades_destinadas_publico_vulneravel = PythonOperator(
    task_id='carrega_atividades_destinadas_publico_vulneravel',
    python_callable=cria_planilhas_atividades_destinadas_publico_vulneravel,
) """
```

```
""" def cria_planilhas_atividades_destinadas_escolas_publicas(ti):
    #Obtém o resultado da query
    result_atividades_destinadas_escolas_publicas =
ti.xcom_pull(task_ids='extraí_atividades_destinadas_escolas_publicas')
    #Cria um dataframe pandas
    df_atividades_destinadas_escolas_publicas =
pd.DataFrame(result_atividades_destinadas_escolas_publicas)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_atividades_destinadas_escolas_publicas = ['Câmpus', 'CEP', 'Ano',
'Ações Extensionistas Destinadas à Escolas Públicas']
    #Exporta para CSV
    if df_atividades_destinadas_escolas_publicas.empty:
        df_atividades_destinadas_escolas_publicas =
pd.DataFrame(columns=cabecalho_atividades_destinadas_escolas_publicas)
```

```
df_atividades_destinadas_escolas_publicas.to_csv(csv_atividades_destinadas_escolas_publicas, index=False, encoding=encoding, sep=separador)
else:
```

```
df_atividades_destinadas_escolas_publicas.to_csv(csv_atividades_destinadas_escolas_publicas, index=False,
header=cabecalho_atividades_destinadas_escolas_publicas, encoding=encoding,
sep=separador)
```

```
carrega_atividades_destinadas_escolas_publicas = PythonOperator(
    task_id='carrega_atividades_destinadas_escolas_publicas',
    python_callable=cria_planilhas_atividades_destinadas_escolas_publicas,
) """
```

```
def cria_planilhas_municipios_envolvidos(ti):
    #Obtém o resultado da query
    result_municipios_envolvidos =
ti.xcom_pull(task_ids='extraí_municipios_envolvidos')
    #Cria um dataframe pandas
```

```

df_municipios_envolvidos = pd.DataFrame(result_municipios_envolvidos)
#Mapea o cabeçalho do arquivo CSV
cabecalho_municipios_envolvidos = ['Câmpus', 'CEP', 'Ano', 'Municípios
Envolvidos em Ações de Extensão']
#Exporta para CSV
if df_municipios_envolvidos.empty:
df_municipios_envolvidos =
pd.DataFrame(columns=cabecalho_municipios_envolvidos)
df_municipios_envolvidos.to_csv(csv_municipios_envolvidos, index=False,
encoding=encoding, sep=separador)
else:
df_municipios_envolvidos.to_csv(csv_municipios_envolvidos, index=False,
header=cabecalho_municipios_envolvidos, encoding=encoding, sep=separador)

carrega_municipios_envolvidos = PythonOperator(
task_id='carrega_municipios_envolvidos',
python_callable=cria_planilhas_municipios_envolvidos,
)

def cria_planilha_publico_extensao_por_discente_graduacao(ti):
#Obtém o resultado da query
result_publico_por_discente_graduacao =
ti.xcom_pull(task_ids='transforma_proporcao_publico_extensao_por_discente_gradu
acao')
#Cria um dataframe pandas
df_publico_por_discente_graduacao =
pd.DataFrame(result_publico_por_discente_graduacao)
#Mapea o cabeçalho do arquivo CSV
cabecalho_publico_por_discente_graduacao = ['Câmpus', 'CEP', 'Ano', 'Tipo de
Curso', 'Proporção de Público Externo Envolvido por Discente de Graduação']
#Exporta para CSV
if df_publico_por_discente_graduacao.empty:
df_publico_por_discente_graduacao =
pd.DataFrame(columns=cabecalho_publico_por_discente_graduacao)

df_publico_por_discente_graduacao.to_csv(csv_proporcao_publico_extensao_por_d
iscente_graduacao, index=False, encoding=encoding, sep=separador)
else:

df_publico_por_discente_graduacao.to_csv(csv_proporcao_publico_extensao_por_d
iscente_graduacao, index=False,
header=cabecalho_publico_por_discente_graduacao, encoding=encoding,
sep=separador)

carrega_proporcao_publico_extensao_por_discente_graduacao =
PythonOperator(
task_id='carrega_proporcao_publico_extensao_por_discente_graduacao',
python_callable=cria_planilha_publico_extensao_por_discente_graduacao,
)

```



```

def cria_planilha_percentual_extensao_populacao_vulneravel(ti):
    #Obtém o resultado da query
    result_percentual_extensao_populacao_vulneravel =
ti.xcom_pull(task_ids='transforma_percentual_extensao_populacao_vulneravel')
    #Cria um dataframe pandas
    df_percentual_extensao_populacao_vulneravel =
pd.DataFrame(result_percentual_extensao_populacao_vulneravel)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_percentual_extensao_populacao_vulneravel =
['Câmpus','CEP','Ano','Ações Destinadas a Atender População Vulnerável(%)']
    #Exporta para CSV
    if df_percentual_extensao_populacao_vulneravel.empty:
        df_percentual_extensao_populacao_vulneravel =
pd.DataFrame(columns=cabecalho_percentual_extensao_populacao_vulneravel)

df_percentual_extensao_populacao_vulneravel.to_csv(csv_percentual_atividades_d
estinadas_publico_vulneravel, index=False, encoding=encoding, sep=separador)
    else:

df_percentual_extensao_populacao_vulneravel.to_csv(csv_percentual_atividades_d
estinadas_publico_vulneravel, index=False,
header=cabecalho_percentual_extensao_populacao_vulneravel,
encoding=encoding, sep=separador)

carrega_percentual_extensao_populacao_vulneravel = PythonOperator(
    task_id='carrega_percentual_extensao_populacao_vulneravel',
    python_callable=cria_planilha_percentual_extensao_populacao_vulneravel,
)

def cria_planilha_percentual_extensao_escolas_publicas(ti):
    #Obtém o resultado da query
    result_percentual_extensao_escolas_publicas =
ti.xcom_pull(task_ids='transforma_percentual_extensao_escolas_publicas')
    #Cria um dataframe pandas
    df_percentual_extensao_escolas_publicas =
pd.DataFrame(result_percentual_extensao_escolas_publicas)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_percentual_extensao_escolas_publicas =
['Câmpus','CEP','Ano','Ações Destinadas a Atender Escolas Públicas(%)']
    #Exporta para CSV
    if df_percentual_extensao_escolas_publicas.empty:
        df_percentual_extensao_escolas_publicas =
pd.DataFrame(columns=cabecalho_percentual_extensao_escolas_publicas)

df_percentual_extensao_escolas_publicas.to_csv(csv_percentual_atividades_destin
adas_escolas_publicas, index=False, encoding=encoding, sep=separador)
    else:

```



```
df_percentual_extensao_escolas_publicas.to_csv(csv_percentual_atividades_destinadas_escolas_publicas, index=False, header=cabecalho_percentual_extensao_escolas_publicas, encoding=encoding, sep=separador)
```

```
carrega_percentual_extensao_escolas_publicas = PythonOperator(
    task_id='carrega_percentual_extensao_escolas_publicas',
    python_callable=cria_planilha_percentual_extensao_escolas_publicas,
)
```

```
"" envia_arquivos_email = EmailOperator(
task_id='envia_arquivos_email',
to=Variable.get("IFSC_email_extensao"),# email definido em Admin ->
```

Variables

```
subject='Indicadores de Extensão - Público Alvo',
html_content="Segue em anexo os arquivos",
files=lista_arquivos,
conn_id="IFSC_SMTP", # Conexão SMTP definida em Admin -> Connections
) ""
```

Definir as dependências entre as tasks

```
[extrai_demanda_publico_externo,
extrai_publico_envolvido,
extrai_publico_envolvido_por_prestacao_servico,
extrai_publico_envolvido_por_programa_projeto,
extrai_publico_envolvido_por_curso_evento,
extrai_capacitacao_publico_EJA,
extrai_capacitacao_publico_PCD,
extrai_discentes_graduacao,
extrai_docentes_rede_publica_capacitados,
extrai_atividades_destinadas_publico_vulneravel,
extrai_atividades_destinadas_escolas_publicas,
extrai_municipios_envolvidos,
extrai_quantitativo_atividades] >> inicia_transformacao >>
[transforma_proporcao_publico_extensao_por_discente_graduacao,
```

transforma percentual extensao populacao vulneravel,

```
transforma_percentual_extensao_escolas_publicas] >>
cria_datalake_publico_alvo_extensao >> [carrega_demanda_publico_externo,
```

carrega publico envolvido por prestacao servico,

carrega publico envolvido por programa projeto,

carrega_publico_envolvido_por_curso_evento, carrega_capacitacao_publico_EJA,

carrega_capitacao_publico_PCD,
#carrega_discentes_graduacao,

carrega_docentes_rede_publica_capacitados,

#carrega_atividades_destinadas_publico_vulneravel,

#carrega_atividades_destinadas_escolas_publicas,
carrega_municipios_envolvidos,

carrega_proporcao_publico_extensao_por_discente_graduacao,

carrega_percentual_extensao_populacao_vulneravel,

carrega_percentual_extensao_escolas_publicas]

PARCERIAS

```
from airflow import DAG
from airflow.operators.python import PythonOperator
from airflow.providers.postgres.operators.postgres import PostgresOperator
from airflow.providers.smtp.operators.smtp import EmailOperator
from airflow.models import Variable
from airflow.utils.dates import days_ago
from datetime import datetime
import pandas as pd
import numpy as np
import logging
import os

# Caminho do arquivo CSV de saída
datalake_extensao_dir = Variable.get("datalake")+ 'extensao/'
datalake_parcerias_dir = datalake_extensao_dir+'parcerias/'
encoding = "utf-8"
separador = ","
#csv_parcerias_extensao =
datalake_parcerias_dir+'IFSC_EXTENSAO_SIGAA_PARCERIAS_EXTENSAO.csv'
csv_percentual_parcerias_extensao =
datalake_parcerias_dir+'IFSC_EXTENSAO_SIGAA_PERCENTUAL_PARCERIAS_E
XTENSAO.csv'
#csv_quantitativo_atividades_financiamento_externo =
datalake_parcerias_dir+'IFSC_EXTENSAO_SIGAA_QUANTITATIVO_ATIVIDADES_
COM_FINANCIAMENTO_EXTERNO.csv'
csv_percentual_atividades_financiamento_externo =
datalake_parcerias_dir+'IFSC_EXTENSAO_SIGAA_PERCENTUAL_ATIVIDADES_C
OM_FINANCIAMENTO_EXTERNO.csv'
csv_listagem_atividades_financiamento_externo =
datalake_parcerias_dir+'IFSC_EXTENSAO_SIGAA_LISTAGEM_ATIVIDADES_COM
_FINANCIAMENTO_EXTERNO.csv'
csv_quantitativo_instituicoes_parceiras_extensao =
datalake_parcerias_dir+'IFSC_EXTENSAO_SIGAA_QUANTITATIVO_INSTITUICOE
S_PARCEIRAS_EXTENSAO.csv'
csv_percentual_instituicoes_parceiras_extensao =
datalake_parcerias_dir+'IFSC_EXTENSAO_SIGAA_PERCENTUAL_INSTITUICOES
_PARCEIRAS_EXTENSAO.csv'
lista_arquivos =
[csv_percentual_parcerias_extensao,csv_percentual_atividades_financiamento_exte
rno,csv_listagem_atividades_financiamento_externo,csv_quantitativo_instituicoes_p
arceiras_extensao,csv_percentual_instituicoes_parceiras_extensao]

with DAG(
    dag_id='IFSC_EXTENSAO_SIGAA_PARCERIAS',
    start_date=days_ago(1),
    schedule_interval='@daily',
    catchup=False,
```

) as dag:

percentual de ações de extensão com parcerias institucionais por câmpus e ano (considerar a consulta de total de atividades de extensão por câmpus e ano na categoria ATIVIDADES DE EXTENSÃO)

quantitativo de ações de extensão com parcerias institucionais por câmpus e ano

```
extrai_quantitativo_parcerias_extensao = PostgresOperator(  
    task_id='extrai_quantitativo_parcerias_extensao',  
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em  
Admin -> Connections  
    sql = Variable.get("script_SIGAA_IFSC_consulta_quantitativo_parcerias"), #  
script definido em Admin -> Variables  
    #do_xcom_push=True,  
)
```

taxa de aprovação de propostas de extensão em editais externos por câmpus e ano (considerar quantitativo total de atividades na categoria ATIVIDADES DE EXTENSÃO)

```
extrai_quantitativo_atividades_financiamento_externo = PostgresOperator(  
    task_id='extrai_quantitativo_atividades_financiamento_externo',  
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em  
Admin -> Connections  
    sql =  
Variable.get("script_SIGAA_IFSC_consulta_quantitativo_atividades_financiamento_e  
xterno"), # script definido em Admin -> Variables  
    #do_xcom_push=True,  
)
```

listagem nominal das parcerias institucionais com ações de extensão por câmpus e ano

```
extrai_listagem_atividades_financiamento_externo = PostgresOperator(  
    task_id='extrai_listagem_atividades_financiamento_externo',  
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em  
Admin -> Connections  
    sql =  
Variable.get("script_SIGAA_IFSC_consulta_listagem_atividades_financiamento_exte  
rno"), # script definido em Admin -> Variables  
    #do_xcom_push=True,  
)
```

quantitativo de instituições ativas nos fóruns de extensão e relações externas nos câmpus por ano.

```
extrai_quantitativo_instituicoes_parceiras_extensao = PostgresOperator(  
    task_id='extrai_quantitativo_instituicoes_parceiras_extensao',  
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em  
Admin -> Connections
```

```

    sql =
Variable.get("script_SIGAA_IFSC_consulta_quantitativo_instituicoes_parceiras_exte
nsao"), # script definido em Admin -> Variables
    #do_xcom_push=True,
)

    # percentual de representação oficial da instituição junto à sociedade civil por
câmpus e ano
    # (considerar o quantitativo de instituições ativas nos fóruns como parcerias
oficiais por quantitativo total de atividades na categoria ATIVIDADES DE
EXTENSÃO)
    # quantitativo total de atividades de extensão por câmpus e ano
    extrai_quantitativo_atividades = PostgresOperator(
    task_id='extrai_quantitativo_atividades',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql =
Variable.get("script_SIGAA_IFSC_consulta_quantitativo_atividades_extensao"), #
script definido em Admin -> Variables
    #do_xcom_push=True,
)

```

```

def calcula_percentual_atividades_com_parcerias(ti) -> pd.DataFrame:
    #Obtém o resultado das queries
    result_quantitativo_parcerias_extensao =
ti.xcom_pull(task_ids='extrai_quantitativo_parcerias_extensao')
    result_quantitativo_atividades =
ti.xcom_pull(task_ids='extrai_quantitativo_atividades')
    #Cria um dataframe pandas
    df_quantitativo_parcerias_extensao =
pd.DataFrame(result_quantitativo_parcerias_extensao,
columns=['campus','cep','ano','atividades_com_parceria'])
    df_atividades = pd.DataFrame(result_quantitativo_atividades,
columns=['campus','cep','ano','atividades'])
    #mescla os dataframes
    df_merged = pd.merge(
    df_quantitativo_parcerias_extensao,
    df_atividades,
    on=['campus','cep','ano'],
    how='left'
    ).fillna(0)
    #Calcula o percentual
    df_merged['percentual_atividades_com_parcerias'] = (
        (df_merged['atividades_com_parceria'] * 100) / df_merged['atividades']
    ).fillna(0)
    # formata o percentual para até 2 casas decimais
    df_merged['percentual_atividades_com_parcerias'] =
df_merged['percentual_atividades_com_parcerias'].round(2)
    # substitui valores infinitos por <0.01

```

```

df_merged.replace([np.inf, -np.inf], 0, inplace=True)
# Cria um novo dataframe apenas com as colunas interessadas
df_resultado =
df_merged[['campus','cep','ano','percentual_atividades_com_parcerias']]
return df_resultado

ano

# percentual de ações de extensão com parcerias institucionais por câmpus e
ano
transforma_percentual_atividades_com_parcerias = PythonOperator(
task_id='transforma_percentual_atividades_com_parcerias',
python_callable=calcula_percentual_atividades_com_parcerias
)

def calcula_percentual_atividades_financiamento_externo(ti) ->
pd.DataFrame:
#Obtém o resultado da queries
result_quantitativo_atividades_financiamento_externo =
ti.xcom_pull(task_ids='extraí_quantitativo_atividades_financiamento_externo')
result_quantitativo_atividades =
ti.xcom_pull(task_ids='extraí_quantitativo_atividades')
#Cria um dataframe pandas
df_quantitativo_atividades_financiamento_externo =
pd.DataFrame(result_quantitativo_atividades_financiamento_externo,
columns=['campus','cep','ano','atividades_financiamento_externo'])
df_atividades = pd.DataFrame(result_quantitativo_atividades,
columns=['campus','cep','ano','atividades'])
#mescla os dataframes
df_merged = pd.merge(
df_quantitativo_atividades_financiamento_externo,
df_atividades,
on=['campus','cep','ano'],
how='left'
).fillna(0)
#Calcula o percentual
df_merged['percentual_atividades_com_financiamento_externo'] = (
(df_merged['atividades_financiamento_externo'] * 100) /
df_merged['atividades']
).fillna(0)
# formata o percentual para até 2 casas decimais
df_merged['percentual_atividades_com_financiamento_externo'] =
df_merged['percentual_atividades_com_financiamento_externo'].round(2)
# substitui valores infinitos por <0.01
df_merged.replace([np.inf, -np.inf], 0, inplace=True)
# Cria um novo dataframe apenas com as colunas interessadas
df_resultado =
df_merged[['campus','cep','ano','percentual_atividades_com_financiamento_externo']]
return df_resultado

```

taxa de aprovação de propostas de extensão em editais externos por
câmpus e ano

```
transforma_percentual_atividades_financiamento_externo = PythonOperator(  
    task_id='transforma_percentual_atividades_financiamento_externo',  
    python_callable=calcula_percentual_atividades_financiamento_externo  
)
```

```
def calcula_percentual_instituicoes_parceiras_extensao(ti) ->  
pd.DataFrame:  
    #Obtém o resultado da queries  
    result_quantitativo_instituicoes_parceiras =  
    ti.xcom_pull(task_ids='extrai_quantitativo_instituicoes_parceiras_extensao')  
    result_quantitativo_atividades =  
    ti.xcom_pull(task_ids='extrai_quantitativo_atividades')  
    #Cria um dataframe pandas  
    df_quantitativo_instituicoes_parceiras =  
    pd.DataFrame(result_quantitativo_instituicoes_parceiras,  
        columns=['campus','cep','ano','atividades_com_instituicoes_parceiras'])  
    df_atividades = pd.DataFrame(result_quantitativo_atividades,  
        columns=['campus','cep','ano','atividades'])  
    #mescla os dataframes  
    df_merged = pd.merge(  
        df_quantitativo_instituicoes_parceiras,  
        df_atividades,  
        on=['campus','cep','ano'],  
        how='left'  
    ).fillna(0)  
    #Calcula o percentual  
    df_merged['percentual_atividades_com_instituicoes_parceiras'] = (  
        (df_merged['atividades_com_instituicoes_parceiras'] * 100) /  
        df_merged['atividades']  
    ).fillna(0)  
    # formata o percentual para até 2 casas decimais  
    df_merged['percentual_atividades_com_instituicoes_parceiras'] =  
    df_merged['percentual_atividades_com_instituicoes_parceiras'].round(2)  
    # substitui valores infinitos por <0.01  
    df_merged.replace([np.inf, -np.inf], 0, inplace=True)  
    # Cria um novo dataframe apenas com as colunas interessadas  
    df_resultado =  
    df_merged[['campus','cep','ano','percentual_atividades_com_instituicoes_parceiras']]  
    return df_resultado
```

percentual de representação oficial da instituição junto à sociedade civil por
câmpus e ano

```
transforma_percentual_instituicoes_parceiras_extensao = PythonOperator(  
    task_id='transforma_percentual_instituicoes_parceiras_extensao',  
    python_callable=calcula_percentual_instituicoes_parceiras_extensao
```

)

```
def transformacao_info():  
    logging.info("Iniciando Tranformação de Dados")
```

```
    inicia_transformacao = PythonOperator(  
        task_id='inicia_transformacao',  
        python_callable=transformacao_info  
    )
```

```
def cria_diretorio():  
    #cria os diretórios se não existirem  
    if not os.path.exists(datalake_extensao_dir):  
        os.makedirs(datalake_extensao_dir)  
    elif not os.path.exists(datalake_parcerias_dir):  
        os.makedirs(datalake_parcerias_dir)
```

```
    cria_datalake_parcerias_extensao = PythonOperator(  
        task_id='cria_datalake_parcerias_extensao',  
        python_callable=cria_diretorio,  
    )
```

```
    """ def cria_planilha_quantitativo_parcerias_extensao(ti):  
        #Obtém o resultado da query  
        result_quantitativo_parcerias_extensao =  
        ti.xcom_pull(task_ids='extrai_quantitativo_parcerias_extensao')  
        #Cria um dataframe pandas  
        df_quantitativo_parcerias_extensao =  
        pd.DataFrame(result_quantitativo_parcerias_extensao)  
        #Mapea o cabeçalho do arquivo CSV  
        cabecalho_quantitativo_parcerias_extensao = ['Câmpus',  
        'CEP', 'Ano', 'Parcerias de Extensão']  
        #Exporta para CSV  
        if df_quantitativo_parcerias_extensao.empty:  
            df_quantitativo_parcerias_extensao =  
            pd.DataFrame(columns=cabecalho_quantitativo_parcerias_extensao)  
            df_quantitativo_parcerias_extensao.to_csv(csv_parcerias_extensao,  
            index=False, encoding=encoding, sep=separador)  
        else:  
            df_quantitativo_parcerias_extensao.to_csv(csv_parcerias_extensao,  
            index=False, header=cabecalho_quantitativo_parcerias_extensao,  
            encoding=encoding, sep=separador)
```

```
    carrega_quantitativo_parcerias_extensao = PythonOperator(  
        task_id='carrega_quantitativo_parcerias_extensao',  
        python_callable=cria_planilha_quantitativo_parcerias_extensao  
    ) """
```

```
    """ def cria_planilha_quantitativo_atividades_financiamento_externo(ti):
```



```

        #Obtém o resultado da query
        result_quantitativo_atividades_financiamento_externo =
ti.xcom_pull(task_ids='extrai_quantitativo_atividades_financiamento_externo')
        #Cria um dataframe pandas
        df_quantitativo_atividades_financiamento_externo =
pd.DataFrame(result_quantitativo_atividades_financiamento_externo)
        #Mapea o cabeçalho do arquivo CSV
        cabecalho_quantitativo_atividades_financiamento_externo = ['Câmpus',
'CEP','Ano','Ações com Financiamento Externo']
        #Exporta para CSV
        if df_quantitativo_atividades_financiamento_externo.empty:
            df_quantitativo_atividades_financiamento_externo =
pd.DataFrame(columns=cabecalho_quantitativo_atividades_financiamento_externo)

df_quantitativo_atividades_financiamento_externo.to_csv(csv_quantitativo_atividade
s_financiamento_externo, index=False, encoding=encoding, sep=separador)
        else:

df_quantitativo_atividades_financiamento_externo.to_csv(csv_quantitativo_atividade
s_financiamento_externo, index=False,
header=cabecalho_quantitativo_atividades_financiamento_externo,
encoding=encoding, sep=separador)

```

```

carrega_quantitativo_atividades_financiamento_externo = PythonOperator(
    task_id='carrega_quantitativo_atividades_financiamento_externo',
    python_callable=cria_planilha_quantitativo_atividades_financiamento_externo
) """

```

```

def cria_planilha_listagem_atividades_financiamento_externo(ti):
    #Obtém o resultado da query
    result_listagem_atividades_financiamento_externo =
ti.xcom_pull(task_ids='extrai_listagem_atividades_financiamento_externo')
    #Cria um dataframe pandas
    df_listagem_atividades_financiamento_externo =
pd.DataFrame(result_listagem_atividades_financiamento_externo)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_listagem_atividades_financiamento_externo = ['Câmpus',
'CEP','Ano','Ações com Financiamento Externo']
    #Exporta para CSV
    if df_listagem_atividades_financiamento_externo.empty:
        df_listagem_atividades_financiamento_externo =
pd.DataFrame(columns=cabecalho_listagem_atividades_financiamento_externo)

df_listagem_atividades_financiamento_externo.to_csv(csv_listagem_atividades_fina
nciamento_externo, index=False, encoding=encoding, sep=separador)
    else:

df_listagem_atividades_financiamento_externo.to_csv(csv_listagem_atividades_fina
nciamento_externo, index=False,

```

```
header=cabecalho_listagem_atividades_financiamento_externo,  
encoding=encoding, sep=separador)
```

```
carrega_listagem_atividades_financiamento_externo = PythonOperator(  
    task_id='carrega_listagem_atividades_financiamento_externo',  
    python_callable=cria_planilha_listagem_atividades_financiamento_externo  
)
```

```
def cria_planilha_quantitativo_instituicoes_parceiras_extensao(ti):  
    #Obtém o resultado da query  
    result_quantitativo_instituicoes_parceiras_extensao =  
    ti.xcom_pull(task_ids='extrai_quantitativo_instituicoes_parceiras_extensao')  
    #Cria um dataframe pandas  
    df_quantitativo_instituicoes_parceiras_extensao =  
    pd.DataFrame(result_quantitativo_instituicoes_parceiras_extensao)  
    #Mapea o cabeçalho do arquivo CSV  
    cabecalho_quantitativo_instituicoes_parceiras_extensao = ['Câmpus',  
'CEP', 'Ano', 'Ações com Parcerias Autorizadas']  
    #Exporta para CSV  
    if df_quantitativo_instituicoes_parceiras_extensao.empty:  
        df_quantitativo_instituicoes_parceiras_extensao =  
        pd.DataFrame(columns=cabecalho_quantitativo_instituicoes_parceiras_extensao)  
  
    df_quantitativo_instituicoes_parceiras_extensao.to_csv(csv_quantitativo_instituicoes_  
_parceiras_extensao, index=False, encoding=encoding, sep=separador)  
    else:  
  
    df_quantitativo_instituicoes_parceiras_extensao.to_csv(csv_quantitativo_instituicoes_  
_parceiras_extensao, index=False,  
header=cabecalho_quantitativo_instituicoes_parceiras_extensao,  
encoding=encoding, sep=separador)
```

```
carrega_quantitativo_instituicoes_parceiras_extensao = PythonOperator(  
    task_id='carrega_quantitativo_instituicoes_parceiras_extensao',  
    python_callable=cria_planilha_quantitativo_instituicoes_parceiras_extensao  
)
```

```
def cria_planilha_percentual_atividades_com_parcerias(ti):  
    #Obtém o resultado da query  
    result_percentual_atividades_com_parcerias =  
    ti.xcom_pull(task_ids='transforma_percentual_atividades_com_parcerias')  
    #Cria um dataframe pandas  
    df_percentual_atividades_com_parcerias =  
    pd.DataFrame(result_percentual_atividades_com_parcerias)  
    #Mapea o cabeçalho do arquivo CSV  
    cabecalho_percentual_atividades_com_parcerias = ['Câmpus',  
'CEP', 'Ano', 'Ações com Parcerias(%)']  
    #Exporta para CSV  
    if df_percentual_atividades_com_parcerias.empty:
```

```

df_percentual_atividades_com_parcerias =
pd.DataFrame(columns=cabecalho_percentual_atividades_com_parcerias)

df_percentual_atividades_com_parcerias.to_csv(csv_percentual_parcerias_extensao
, index=False, encoding=encoding, sep=separador)
else:

df_percentual_atividades_com_parcerias.to_csv(csv_percentual_parcerias_extensao
, index=False, header=cabecalho_percentual_atividades_com_parcerias,
encoding=encoding, sep=separador)

carrega_percentual_atividades_com_parcerias = PythonOperator(
task_id='carrega_percentual_atividades_com_parcerias',
python_callable=cria_planilha_percentual_atividades_com_parcerias
)

def cria_planilha_percentual_atividades_financiamento_externo(ti):
#Obtém o resultado da query
result_percentual_atividades_financiamento_externo =
ti.xcom_pull(task_ids='transforma_percentual_atividades_financiamento_externo')
#Cria um dataframe pandas
df_percentual_atividades_financiamento_externo =
pd.DataFrame(result_percentual_atividades_financiamento_externo)
#Mapea o cabeçalho do arquivo CSV
cabecalho_percentual_atividades_financiamento_externo = ['Câmpus',
'CEP','Ano','Ações com Financiamento Externo(%)']
#Exporta para CSV
if df_percentual_atividades_financiamento_externo.empty:
df_percentual_atividades_financiamento_externo =
pd.DataFrame(columns=cabecalho_percentual_atividades_financiamento_externo)

df_percentual_atividades_financiamento_externo.to_csv(csv_percentual_atividades_
financiamento_externo, index=False, encoding=encoding, sep=separador)
else:

df_percentual_atividades_financiamento_externo.to_csv(csv_percentual_atividades_
financiamento_externo, index=False,
header=cabecalho_percentual_atividades_financiamento_externo,
encoding=encoding, sep=separador)

carrega_percentual_atividades_financiamento_externo = PythonOperator(
task_id='carrega_percentual_atividades_financiamento_externo',
python_callable=cria_planilha_percentual_atividades_financiamento_externo
)

def cria_planilha_percentual_instituicoes_parceiras_extensao(ti):
#Obtém o resultado da query
result_percentual_instituicoes_parceiras_extensao =
ti.xcom_pull(task_ids='transforma_percentual_instituicoes_parceiras_extensao')

```

```

#Cria um dataframe pandas
df_percentual_instituicoes_parceiras_extensao =
pd.DataFrame(result_percentual_instituicoes_parceiras_extensao)
#Mapea o cabeçalho do arquivo CSV
cabecalho_percentual_instituicoes_parceiras_extensao = ['Câmpus',
'CEP','Ano','Ações com Parcerias Autorizadas(%)']
#Exporta para CSV
if df_percentual_instituicoes_parceiras_extensao.empty:
df_percentual_instituicoes_parceiras_extensao =
pd.DataFrame(columns=cabecalho_percentual_instituicoes_parceiras_extensao)

df_percentual_instituicoes_parceiras_extensao.to_csv(csv_percentual_instituicoes_p
arceiras_extensao, index=False, encoding=encoding, sep=separador)
else:

df_percentual_instituicoes_parceiras_extensao.to_csv(csv_percentual_instituicoes_p
arceiras_extensao, index=False,
header=cabecalho_percentual_instituicoes_parceiras_extensao,
encoding=encoding, sep=separador)

carrega_percentual_instituicoes_parceiras_extensao = PythonOperator(
task_id='carrega_percentual_instituicoes_parceiras_extensao',
python_callable=cria_planilha_percentual_instituicoes_parceiras_extensao
)

""" envia_arquivos_email = EmailOperator(
task_id='envia_arquivos_email',
to=Variable.get("IFSC_email_extensao"),# email definido em Admin ->
Variables
subject='Indicadores de Extensão - Parcerias',
html_content="Segue em anexo os arquivos",
files=lista_arquivos,
conn_id="IFSC_SMTP", # Conexão SMTP definida em Admin -> Connections
) """

# Definir as dependências entre as tasks
[extrai_quantitativo_parcerias_extensao,
extrai_quantitativo_atividades_financiamento_externo,
extrai_listagem_atividades_financiamento_externo,
extrai_quantitativo_instituicoes_parceiras_extensao,
extrai_quantitativo_atividades] >> inicia_transformacao >>
[transforma_percentual_atividades_com_parcerias,

transforma_percentual_atividades_financiamento_externo,

transforma_percentual_instituicoes_parceiras_extensao] >>
cria_datalake_parcerias_extensao >>
[carrega_listagem_atividades_financiamento_externo,

```

#carrega_quantitativo_parcerias_extensao,
#carrega_quantitativo_atividades_financiamento_externo,
carrega_quantitativo_instituicoes_parceiras_extensao,
carrega_percentual_atividades_com_parcerias,
carrega_percentual_atividades_financiamento_externo,
carrega_percentual_instituicoes_parceiras_extensao]

PRODUÇÃO EXTENSIONISTA

```
from typing import Optional
from airflow import DAG
from airflow.operators.python import PythonOperator
from airflow.providers.postgres.operators.postgres import PostgresOperator
from airflow.providers.smtp.operators.smtp import EmailOperator
from airflow.models import Variable
from airflow.utils.dates import days_ago
from datetime import datetime
import pandas as pd
import os

# Caminho do arquivo CSV de saída
datalake_extensao_dir = Variable.get("datalake")+'extensao/'
datalake_producoes_dir = datalake_extensao_dir+'producoes/'
encoding = "utf-8"
separador = ","
csv_quantitativo_producoes_extensionistas =
datalake_producoes_dir+'IFSC_EXTENSAO_SIGAA_QUANTITATIVO_PRODUCOE
S_EXTENSIONISTAS.csv'
lista_arquivos = [csv_quantitativo_producoes_extensionistas]

with DAG(
    dag_id='IFSC_EXTENSAO_SIGAA_PRODUCOES',
    start_date=days_ago(1),
    schedule_interval='@daily',
    catchup=False,
) as dag:

    # quantitativo de revistas e periódicos institucionais por câmpus e ano
    # quantitativo de produção de livros ou capítulos com base em resultados da
    extensão por câmpus e ano
    # quantitativo de publicação de artigos em periódicos com base em resultados
    da extensão por câmpus e ano
    # quantitativo de mídias de comunicações em eventos com base em
    resultados da extensão por câmpus e ano
    # quantitativo de produções audiovisuais resultadas da extensão por câmpus
    e ano
    # quantitativo de produções artísticas (exposições, espetáculos, outros)
    resultadas da extensão por câmpus e ano
    # retorna os resultados quantitativos e qualitativos dos relatórios de ações
    extensão
    extrai_resultados_relatorio_extensao = PostgresOperator(
        task_id='extrai_resultados_relatorio_extensao',
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
        Admin -> Connections
```

```

sql =
Variable.get("script_SIGAA_IFSC_consulta_resultados_relatorio_extensao"), # script
definido em Admin -> Variables
#do_xcom_push=True,
)

def categorizar_produto(texto, palavras_chave) -> Optional[str]:
if pd.isnull(texto):
return None
if pd.isna(texto):
return None
texto = texto.lower()
for categoria, palavras in palavras_chave.items():
if any(palavra in texto for palavra in palavras):
return categoria
return None

def classifica_producoes_de_relatorios_extensao(ti) -> pd.DataFrame:
#Obtém o resultado da query
result_resultados_relatorio_extensao =
ti.xcom_pull(task_ids='extraí_resultados_relatorio_extensao')
#Cria um dataframe pandas
df_relatorio = pd.DataFrame(result_resultados_relatorio_extensao,
columns=['campus','cep','ano','resultados_qualitativos','resultados_quantitativos'])
#dicionário de categoria: palavras-chave
palavras_chave = {
    "Revistas e Periódicos": ["revista", "periódico", "journal"],
    "Livros ou Capítulos": ["livro", "capítulo", "book", "chapter"],
    "Artigos": ["artigo", "paper", "article"],
    "Mídias de Comunicações em Eventos":
["evento", "comunicação", "comunicacao"],
    "Produções Audiovisuais": ["audiovisual"],
    "Produções Artísticas": ["artística", "arte"]
}
#categoriza conforme a coluna de resultados_qualitativos
df_relatorio['tipo_produto_qualitativo'] = df_relatorio.apply(lambda x:
categorizar_produto(x['resultados_qualitativos'], palavras_chave), axis=1)
#categoriza conforme a coluna de resultados_quantitativos
df_relatorio['tipo_produto_quantitativo'] = df_relatorio.apply(lambda x:
categorizar_produto(x['resultados_quantitativos'], palavras_chave), axis=1)
#faz a junção das colunas tipo_produto_qualitativo e
tipo_produto_quantitativo conforme a categoria
df_relatorio["tipo_produto"] =
df_relatorio["tipo_produto_qualitativo"].combine_first(df_relatorio["tipo_produto_quant
itativo"])
# Agrupar e contar em um novo dataframe
df_quantitativo_producoes = df_relatorio.groupby(['campus', 'cep', 'ano',
'tipo_produto']).size().reset_index(name='quantidade')
return df_quantitativo_producoes

```

```

transforma_quantitativo_producoes_extensao = PythonOperator(
    task_id='transforma_quantitativo_producoes_extensao',
    python_callable=classifica_producoes_de_relatorios_extensao
)

def cria_diretorio():
    #cria os diretórios se não existirem
    if not os.path.exists(datalake_extensao_dir):
        os.makedirs(datalake_extensao_dir)
    elif not os.path.exists(datalake_producoes_dir):
        os.makedirs(datalake_producoes_dir)

cria_datalake_producoes_extensao = PythonOperator(
    task_id='cria_datalake_producoes_extensao',
    python_callable=cria_diretorio,
)

def cria_planilha_quantitativo_producoes_extensao(ti):
    #Obtém o resultado da transformação
    result_quantitativo_producoes_extensao =
ti.xcom_pull(task_ids='transforma_quantitativo_producoes_extensao')
    #Cria um dataframe pandas
    df_quantitativo_producoes_extensao =
pd.DataFrame(result_quantitativo_producoes_extensao)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_quantitativo_producoes_extensao =
['Câmpus','CEP','Ano','Categoria','Produções Extensionistas']
    #Exporta para CSV
    if df_quantitativo_producoes_extensao.empty:
        df_quantitativo_producoes_extensao =
pd.DataFrame(columns=cabecalho_quantitativo_producoes_extensao)

df_quantitativo_producoes_extensao.to_csv(csv_quantitativo_producoes_extensioni
stas, index=False, encoding=encoding, sep=separador)
    else:

df_quantitativo_producoes_extensao.to_csv(csv_quantitativo_producoes_extensioni
stas, index=False, header=cabecalho_quantitativo_producoes_extensao,
encoding=encoding, sep=separador)

carrega_quantitativo_producoes_extensao = PythonOperator(
    task_id='carrega_quantitativo_producoes_extensao',
    python_callable=cria_planilha_quantitativo_producoes_extensao
)

""" envia_arquivos_email = EmailOperator(
    task_id='envia_arquivos_email',

```



```
to=Variable.get("IFSC_email_extensao"),# email definido em Admin ->  
Variables  
subject='Indicadores de Extensão - Produção Extensionista',  
html_content="Segue em anexo os arquivos",  
files=lista_arquivos,  
conn_id="IFSC_SMTP", # Conexão SMTP definida em Admin -> Connections  
) ""
```

```
# Definir as dependências entre as tasks  
extrai_resultados_relatorio_extensao >>  
transforma_quantitativo_producoes_extensao >>  
cria_datalake_producoes_extensao >> carrega_quantitativo_producoes_extensao
```

ÁREAS TEMÁTICAS DA EXTENSÃO

```
from airflow import DAG
from airflow.operators.python import PythonOperator
from airflow.providers.postgres.operators.postgres import PostgresOperator
from airflow.providers.smtp.operators.smtp import EmailOperator
from airflow.models import Variable
from airflow.utils.dates import days_ago
from datetime import datetime
import pandas as pd
import os

# Caminho do arquivo CSV de saída
datalake_extensao_dir = Variable.get("datalake")+ 'extensao/'
datalake_areas_tematicas_dir = datalake_extensao_dir+'areas_tematicas/'
encoding = "utf-8"
separador = ","
csv_quantitativo_areas_tematicas_extensao =
datalake_areas_tematicas_dir+'IFSC_EXTENSAO_SIGAA_QUANTITATIVO_AREAS
_TEMATICAS_EXTENSAO.csv'
lista_arquivos = [csv_quantitativo_areas_tematicas_extensao]

with DAG(
    dag_id='IFSC_EXTENSAO_SIGAA_AREAS_TEMATICAS',
    start_date=days_ago(1),
    schedule_interval='@daily',
    catchup=False,
) as dag:

    # quantitativo de atividades de extensão por área da comunicação por
    # campus e ano
    # quantitativo de atividades de extensão por área da cultura por campus e
    # ano
    # quantitativo de atividades de extensão por área dos direitos humanos e
    # justiça por campus e ano
    # quantitativo de atividades de extensão por área da tecnologia e produção
    # por campus e ano
    # quantitativo de atividades de extensão por área do trabalho por campus e
    # ano

    extrai_quantitativo_areas_tematicas_extensao = PostgresOperator(
        task_id='extrai_quantitativo_areas_tematicas_extensao',
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
        Admin -> Connections
    )
    sql =
    Variable.get("script_SIGAA_IFSC_consulta_quantitativo_areas_tematicas_extensao"
    ), # script definido em Admin -> Variables
    #do_xcom_push=True,
    )
```

```

def cria_diretorio():
    #cria os diretórios se não existirem
    if not os.path.exists(datalake_extensao_dir):
        os.makedirs(datalake_extensao_dir)
    elif not os.path.exists(datalake_areas_tematicas_dir):
        os.makedirs(datalake_areas_tematicas_dir)

    cria_datalake_areas_tematicas_extensao = PythonOperator(
        task_id='cria_datalake_areas_tematicas_extensao',
        python_callable=cria_diretorio,
    )

def cria_planilha_quantitativo_areas_tematicas_extensao(ti):
    #Obtém o resultado da transformação
    result_quantitativo_areas_tematicas_extensao =
ti.xcom_pull(task_ids='extrai_quantitativo_areas_tematicas_extensao')
    #Cria um dataframe pandas
    df_quantitativo_areas_tematicas_extensao =
pd.DataFrame(result_quantitativo_areas_tematicas_extensao)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_quantitativo_areas_tematicas_extensao =
['Câmpus', 'CEP', 'Ano', 'Área Temática', 'Ações de Extensão']
    #Exporta para CSV
    if df_quantitativo_areas_tematicas_extensao.empty:
        df_quantitativo_areas_tematicas_extensao =
pd.DataFrame(columns=cabecalho_quantitativo_areas_tematicas_extensao)

df_quantitativo_areas_tematicas_extensao.to_csv(csv_quantitativo_areas_tematicas_extensao, index=False, encoding=encoding, sep=separador)
    else:

df_quantitativo_areas_tematicas_extensao.to_csv(csv_quantitativo_areas_tematicas_extensao, index=False,
header=cabecalho_quantitativo_areas_tematicas_extensao, encoding=encoding,
sep=separador)

carrega_quantitativo_areas_tematicas_extensao = PythonOperator(
    task_id='carrega_quantitativo_areas_tematicas_extensao',
    python_callable=cria_planilha_quantitativo_areas_tematicas_extensao
)

""" envia_arquivos_email = EmailOperator(
    task_id='envia_arquivos_email',
    to=Variable.get("IFSC_email_extensao"),# email definido em Admin ->
Variables
    subject='Indicadores de Extensão - Áreas Temáticas',
    html_content="Segue em anexo os arquivos",
    files=lista_arquivos,

```

```
conn_id="IFSC_SMTP", # Conexão SMTP definida em Admin -> Connections  
) ""
```

Definir as dependências entre as tasks

```
    extrai_quantitativo_areas_tematicas_extensao >>  
cria_datalake_areas_tematicas_extensao >>  
carrega_quantitativo_areas_tematicas_extensao
```

INFRAESTRUTURA E RECURSOS HUMANOS

```
from airflow import DAG
from airflow.operators.python import PythonOperator
from airflow.providers.postgres.operators.postgres import PostgresOperator
from airflow.providers.smtp.operators.smtp import EmailOperator
from airflow.models import Variable
from airflow.utils.dates import days_ago
from datetime import datetime
import pandas as pd
import numpy as np
import os

# Caminho do arquivo CSV de saída
datalake_extensao_dir = Variable.get("datalake")+'extensao/'
datalake_infraestrutura_rh_dir = datalake_extensao_dir+'infraestrutura_rh/'
encoding = "utf-8"
separador = ","
csv_quantitativo_unidades_proponentes_extensao =
datalake_infraestrutura_rh_dir+'IFSC_EXTENSAO_SIGAA_QUANTITATIVO_UNIDA
DES_PROponentes_EXTENSAO.csv'
csv_listagem_unidades_proponentes_extensao =
datalake_infraestrutura_rh_dir+'IFSC_EXTENSAO_SIGAA_LISTAGEM_UNIDADES
_PROponentes_EXTENSAO.csv'
csv_percentual_servidores_envolvidos_instituicoes_parceiras =
datalake_infraestrutura_rh_dir+'IFSC_EXTENSAO_SIGAA_PERCENTUAL_SERVID
ORES_ENVOLVIDOS_INSTITUICOES_PARCEIRAS.csv'
lista_arquivos =
[csv_quantitativo_unidades_proponentes_extensao,csv_listagem_unidades_propone
ntes_extensao,csv_percentual_servidores_envolvidos_instituicoes_parceiras]

with DAG(
    dag_id='IFSC_EXTENSAO_SIGAA_INFRAESTRUTURA_RH',
    start_date=days_ago(1),
    schedule_interval='@daily',
    catchup=False,
) as dag:

    # proporção de horas de treinamento por servidor da gestão extensionista por
    # câmpus e ano
    # proporção de participação em eventos por servidor da gestão extensionista
    # por câmpus e ano
    # proporção de participação em eventos promovidos ou apoiados pela
    # pró-reitoria de extensão por servidor da gestão extensionista por câmpus e ano
    # proporção espaço físico (área) por servidor por câmpus e ano
    # percentual de capacidade de atendimento de demandas de transporte
    # logístico para ações de extensão por câmpus e ano
```

quantitativo de unidades organizacionais proponentes de ação de extensão por câmpus e ano

```
extrai_quantitativo_unidades_proponentes_extensao = PostgresOperator(  
    task_id='extrai_quantitativo_unidades_proponentes_extensao',  
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em  
Admin -> Connections  
    sql =  
Variable.get("script_SIGAA_IFSC_consulta_quantitativo_unidades_proponentes_extensao"), # script definido em Admin -> Variables  
    #do_xcom_push=True,  
)
```

listagem nominal das unidades organizacionais proponentes de ação de extensão por câmpus e ano

```
extrai_listagem_unidades_proponentes_extensao = PostgresOperator(  
    task_id='extrai_listagem_unidades_proponentes_extensao',  
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em  
Admin -> Connections  
    sql =  
Variable.get("script_SIGAA_IFSC_consulta_listagem_unidades_proponentes_extensao"), # script definido em Admin -> Variables  
    #do_xcom_push=True,  
)
```

percentual de servidores em representação oficial da instituição junto à sociedade civil por câmpus e ano

(considerar o quantitativo de servidores em atividades envolvidas com instituições ativas em fóruns por quantitativo total de servidores na categoria PARTICIPANTES DA EXTENSÃO)

quantitativo de servidores envolvidos em atividades com instituições ativas nos fóruns de extensão por campus e ano

```
extrai_participantes_servidores_envolvidos_atividades_com_instituicoes_parceiras =  
PostgresOperator(  

```

```
    task_id='extrai_participantes_servidores_envolvidos_atividades_com_instituicoes_parceiras',  

```

```
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em  
Admin -> Connections  

```

```
    sql =  
Variable.get("script_SIGAA_IFSC_consulta_quantidade_participantes_servidores_envolvidos_atividades_com_instituicoes_parceiras"), # script definido em Admin -> Variables  
    #do_xcom_push=True,  
)
```

quantitativo de servidores envolvidos em atividades de extensão por campus e ano

```

        extrai_participantes_servidores_envolvidos_extensao = PostgresOperator(
            task_id='extrai_participantes_servidores_envolvidos_extensao',
            postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
        sql =
Variable.get("script_SIGAA_IFSC_consulta_quantidade_participantes_servidores_en
volvidos_extensao"), # script definido em Admin -> Variables
        #do_xcom_push=True,
    )

```

```

def calcula_percentual_servidores_envolvidos_instituicoes_parceiras(ti)
-> pd.DataFrame:
    #Obtém o resultado da query
    result_servidores_instituicoes_parceiras =
ti.xcom_pull(task_ids='extrai_participantes_servidores_envolvidos_atividades_com_i
nstituicoes_parceiras')
    result_servidores_extensao =
ti.xcom_pull(task_ids='extrai_participantes_servidores_envolvidos_extensao')
    #Cria um dataframe pandas
    df_servidores_instituicoes_parceiras =
pd.DataFrame(result_servidores_instituicoes_parceiras,
columns=['campus','cep','ano','participantes_servidores_instituicoes_parceiras'])
    df_participantes_servidores = pd.DataFrame(result_servidores_extensao,
columns=['campus','cep','ano','participantes_servidores'])
    #mescla os dataframes
    df_merged = pd.merge(
        df_servidores_instituicoes_parceiras,
        df_participantes_servidores,
        on=['campus','cep','ano'],
        how='left'
    ).fillna(0)
    #Calcula o percentual
    df_merged['percentual_servidores_envolvidos_instituicoes_parceiras'] = (
        (df_merged['participantes_servidores_instituicoes_parceiras'] * 100) /
df_merged['participantes_servidores']
    ).fillna(0)
    # formata o percentual para até 2 casas decimais
    df_merged['percentual_servidores_envolvidos_instituicoes_parceiras'] =
df_merged['percentual_servidores_envolvidos_instituicoes_parceiras'].round(2)
    # substitui valores infinitos por <0.01
    df_merged.replace([np.inf, -np.inf], 0, inplace=True)
    # Cria um novo dataframe apenas com as colunas interessadas
    df_resultado =
df_merged[['campus','cep','ano','percentual_servidores_envolvidos_instituicoes_parc
eiras']]
    return df_resultado

```

```

        # percentual de servidores em representação oficial da instituição junto à
        sociedade civil por câmpus e ano
        transforma_percentual_servidores_envolvidos_instituicoes_parceiras =
        PythonOperator(

task_id='transforma_percentual_servidores_envolvidos_instituicoes_parceiras',

python_callable=calcula_percentual_servidores_envolvidos_instituicoes_parceiras
)

```

```

def cria_diretorio():
    #cria os diretórios se não existirem
    if not os.path.exists(datalake_extensao_dir):
        os.makedirs(datalake_extensao_dir)
    elif not os.path.exists(datalake_infraestrutura_rh_dir):
        os.makedirs(datalake_infraestrutura_rh_dir)

```

```

    cria_datalake_infraestrutura_rh_extensao = PythonOperator(
        task_id='cria_datalake_infraestrutura_rh_extensao',
        python_callable=cria_diretorio,
    )

```

```

def cria_planilha_quantitativo_unidades_proponentes_extensao(ti):
    #Obtém o resultado da transformação
    result_quantitativo_unidades_proponentes_extensao =
    ti.xcom_pull(task_ids='extraí_quantitativo_unidades_proponentes_extensao')
    #Cria um dataframe pandas
    df_quantitativo_unidades_proponentes_extensao =
    pd.DataFrame(result_quantitativo_unidades_proponentes_extensao)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_quantitativo_unidades_proponentes_extensao =
    ['Câmpus','CEP','Ano','Unidades Proponentes de Extensão']
    #Exporta para CSV
    if df_quantitativo_unidades_proponentes_extensao.empty:
        df_quantitativo_unidades_proponentes_extensao =
        pd.DataFrame(columns=cabecalho_quantitativo_unidades_proponentes_extensao)

    df_quantitativo_unidades_proponentes_extensao.to_csv(csv_quantitativo_unidades_
    proponentes_extensao, index=False, encoding=encoding, sep=separador)
    else:

    df_quantitativo_unidades_proponentes_extensao.to_csv(csv_quantitativo_unidades_
    proponentes_extensao, index=False,
    header=cabecalho_quantitativo_unidades_proponentes_extensao,
    encoding=encoding, sep=separador)

```

```

    carrega_quantitativo_unidades_proponentes_extensao = PythonOperator(
        task_id='carrega_quantitativo_unidades_proponentes_extensao',
        python_callable=cria_planilha_quantitativo_unidades_proponentes_extensao
    )

```



```

)

def cria_planilha_listagem_unidades_proponentes_extensao(ti):
    #Obtém o resultado da transformação
    result_listagem_unidades_proponentes_extensao =
ti.xcom_pull(task_ids='extraí_listagem_unidades_proponentes_extensao')
    #Cria um dataframe pandas
    df_listagem_unidades_proponentes_extensao =
pd.DataFrame(result_listagem_unidades_proponentes_extensao)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_listagem_unidades_proponentes_extensao =
['Câmpus','CEP','Ano','Unidade Proponente de Extensão']
    #Exporta para CSV
    if df_listagem_unidades_proponentes_extensao.empty:
        df_listagem_unidades_proponentes_extensao =
pd.DataFrame(columns=cabecalho_listagem_unidades_proponentes_extensao)

df_listagem_unidades_proponentes_extensao.to_csv(csv_listagem_unidades_propo
nentes_extensao, index=False, encoding=encoding, sep=separador)
    else:

df_listagem_unidades_proponentes_extensao.to_csv(csv_listagem_unidades_propo
nentes_extensao, index=False,
header=cabecalho_listagem_unidades_proponentes_extensao, encoding=encoding,
sep=separador)

    carrega_listagem_unidades_proponentes_extensao = PythonOperator(
        task_id='carrega_listagem_unidades_proponentes_extensao',
        python_callable=cria_planilha_listagem_unidades_proponentes_extensao
    )

def
cria_planilha_percentual_servidores_envolvidos_instituicoes_parceiras(ti):
    #Obtém o resultado da transformação
    result_percentual_servidores_envolvidos_instituicoes_parceiras =
ti.xcom_pull(task_ids='transforma_percentual_servidores_envolvidos_instituicoes_pa
rceiras')
    #Cria um dataframe pandas
    df_percentual_servidores_envolvidos_instituicoes_parceiras =
pd.DataFrame(result_percentual_servidores_envolvidos_instituicoes_parceiras)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_percentual_servidores_envolvidos_instituicoes_parceiras =
['Câmpus','CEP','Ano','Servidores em Representação Oficial junto à Parcerias da
Sociedade Civil(%)']
    #Exporta para CSV
    if df_percentual_servidores_envolvidos_instituicoes_parceiras.empty:
        df_percentual_servidores_envolvidos_instituicoes_parceiras =
pd.DataFrame(columns=cabecalho_percentual_servidores_envolvidos_instituicoes_
parceiras)

```

```
df_percentual_servidores_envolvidos_instituicoes_parceiras.to_csv(csv_percentual_servidores_envolvidos_instituicoes_parceiras, index=False, encoding=encoding, sep=separador)
```

```
else:
```

```
df_percentual_servidores_envolvidos_instituicoes_parceiras.to_csv(csv_percentual_servidores_envolvidos_instituicoes_parceiras, index=False, header=cabecalho_percentual_servidores_envolvidos_instituicoes_parceiras, encoding=encoding, sep=separador)
```

```
carrega_percentual_servidores_envolvidos_instituicoes_parceiras = PythonOperator(
```

```
    task_id='carrega_percentual_servidores_envolvidos_instituicoes_parceiras',
```

```
    python_callable=cria_planilha_percentual_servidores_envolvidos_instituicoes_parceiras
```

```
)
```

```
""" envia_arquivos_email = EmailOperator(
    task_id='envia_arquivos_email',
    to=Variable.get("IFSC_email_extensao"),# email definido em Admin ->
```

```
Variables
```

```
    subject='Indicadores de Extensão - Infraestrutura e Recursos Humanos',
    html_content="Segue em anexo os arquivos",
    files=lista_arquivos,
    conn_id="IFSC_SMTP", # Conexão SMTP definida em Admin -> Connections
) """
```

```
# Definir as dependências entre as tasks
```

```
[extrai_quantitativo_unidades_proponentes_extensao,
extrai_listagem_unidades_proponentes_extensao,
```

```
extrai_participantes_servidores_envolvidos_atividades_com_instituicoes_parceiras,
extrai_participantes_servidores_envolvidos_extensao] >>
```

```
transforma_percentual_servidores_envolvidos_instituicoes_parceiras >>
```

```
cria_datalake_infraestrutura_rh_extensao >>
```

```
[carrega_quantitativo_unidades_proponentes_extensao,
```

```
carrega_listagem_unidades_proponentes_extensao,
```

```
carrega_percentual_servidores_envolvidos_instituicoes_parceiras]
```

ATIVIDADES DE EXTENSÃO

```
from airflow import DAG
from airflow.operators.python import PythonOperator
from airflow.providers.postgres.operators.postgres import PostgresOperator
from airflow.providers.smtp.operators.smtp import EmailOperator
from airflow.models import Variable
from airflow.utils.dates import days_ago
from datetime import datetime
import pandas as pd
import numpy as np
import logging
import os

# Caminho do arquivo CSV de saída
datalake_extensao_dir = Variable.get("datalake")+ 'extensao/'
datalake_atividades_dir = datalake_extensao_dir+'atividades/'
encoding = "utf-8"
separador = ","
csv_quantitativo_atividades_extensao =
datalake_atividades_dir+'IFSC_EXTENSAO_SIGAA_QUANTITATIVO_ATIVIDADES_
_EXTENSAO.csv'
csv_quantitativo_atividades_continuas_extensao =
datalake_atividades_dir+'IFSC_EXTENSAO_SIGAA_QUANTITATIVO_ATIVIDADES_
_CONTINUAS_EXTENSAO.csv'
#csv_quantitativo_atividades_extensao_aprovadas =
datalake_atividades_dir+'IFSC_EXTENSAO_SIGAA_QUANTITATIVO_ATIVIDADES_
_APROVADAS_EXTENSAO.csv'
#csv_quantitativo_atividades_extensao_submetidas =
datalake_atividades_dir+'IFSC_EXTENSAO_SIGAA_QUANTITATIVO_ATIVIDADES_
_SUBMETIDAS_EXTENSAO.csv'
csv_quantitativo_atividades_extensao_favoraveis =
datalake_atividades_dir+'IFSC_EXTENSAO_SIGAA_QUANTITATIVO_ATIVIDADES_
_FAVORAVEIS_EXTENSAO.csv'
csv_quantitativo_atividades_extensao_desfavoraveis =
datalake_atividades_dir+'IFSC_EXTENSAO_SIGAA_QUANTITATIVO_ATIVIDADES_
_DESFAVORAVEIS_EXTENSAO.csv'
csv_quantitativo_atividades_extensao_por_status =
datalake_atividades_dir+'IFSC_EXTENSAO_SIGAA_QUANTITATIVO_ATIVIDADES_
_EXTENSAO_POR_STATUS.csv'
#csv_quantitativo_atividades_articuladas_pesquisa =
datalake_atividades_dir+'IFSC_EXTENSAO_SIGAA_QUANTITATIVO_ATIVIDADES_
_EXTENSAO_ARTICULADAS_PESQUISA.csv'
csv_percentual_atividades_articuladas_pesquisa =
datalake_atividades_dir+'IFSC_EXTENSAO_SIGAA_PERCENTUAL_ATIVIDADES_
_EXTENSAO_ARTICULADAS_PESQUISA.csv'
#csv_quantitativo_atividades_articuladas_ensino =
datalake_atividades_dir+'IFSC_EXTENSAO_SIGAA_QUANTITATIVO_ATIVIDADES_
_EXTENSAO_ARTICULADAS_ENSINO.csv'
```

```

csv_percentual_atividades_articuladas_ensino =
datalake_atividades_dir+'IFSC_EXTENSAO_SIGAA_PERCENTUAL_ATIVIDADES_
EXTENSAO_ARTICULADAS_ENSINO.csv'
csv_quantitativo_atividades_extensao_por_objetivo_sustentavel =
datalake_atividades_dir+'IFSC_EXTENSAO_SIGAA_QUANTITATIVO_ATIVIDADES_
_EXTENSAO_POR_OBJETIVO_SUSTENTAVEL.csv'
csv_proporcao_atividades_aprovadas_por_servidor_participante =
datalake_atividades_dir+'IFSC_EXTENSAO_SIGAA_PROPORCAO_ATIVIDADES_
APROVADAS_POR_SERVIDOR_PARTICIPANTE.csv'
csv_proporcao_atividades_submetidas_por_aprovadas =
datalake_atividades_dir+'IFSC_EXTENSAO_SIGAA_PROPORCAO_ATIVIDADES_
SUBMETIDAS_POR_APROVADAS.csv'
csv_listagem_editais_internos =
datalake_atividades_dir+'IFSC_EXTENSAO_SIGAA_LISTAGEM_EDITAIS_INTERN
OS.csv'
lista_arquivos = [csv_quantitativo_atividades_extensao,
csv_quantitativo_atividades_continuas_extensao,
csv_proporcao_atividades_submetidas_por_aprovadas,
csv_quantitativo_atividades_extensao_favoraveis,csv_quantitativo_atividades_exten
sao_desfavoraveis, csv_quantitativo_atividades_extensao_por_status,
csv_percentual_atividades_articuladas_pesquisa,
csv_percentual_atividades_articuladas_ensino, csv_listagem_editais_internos]

```

```

with DAG(
    dag_id='IFSC_EXTENSAO_SIGAA_ATIVIDADES',
    start_date=days_ago(1),
    schedule_interval='@daily',
    catchup=False,
) as dag:

```

```

    # quantitativo total de atividades de extensão por campus e ano
    extrai_quantitativo_atividades = PostgresOperator(
        task_id='extrai_quantitativo_atividades',
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
        sql =
Variable.get("script_SIGAA_IFSC_consulta_quantitativo_atividades_extensao"), #
script definido em Admin -> Variables
        #do_xcom_push=True,
    )

```

```

    # quantitativo de atividades de extensão contínuas por campus e ano
    extrai_quantitativo_atividades_continuas = PostgresOperator(
        task_id='extrai_quantitativo_atividades_continuas',
        postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections

```

```
sql =  
Variable.get("script_SIGAA_IFSC_consulta_quantitativo_atividades_continuas_exten  
sao"), # script definido em Admin -> Variables  
    #do_xcom_push=True,  
    )
```

proporção de ações de extensão com status aprovada por servidor por
câmpus e ano (considerar percentual de servidores envolvidos em atividades de
extensão (retorna total de servidores por campus e ano) da categoria
PARTICIPANTES)

```
extrai_quantitativo_atividades_aprovadas = PostgresOperator(  
    task_id='extrai_quantitativo_atividades_aprovadas',  
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em  
Admin -> Connections
```

```
sql =  
Variable.get("script_SIGAA_IFSC_consulta_quantitativo_atividades_aprovadas_exte  
nsao"), # script definido em Admin -> Variables  
    #do_xcom_push=True,  
    )
```

quantitativo de servidores envolvidos em atividades de extensão por
campus e ano

```
extrai_participantes_servidores_envolvidos_extensao = PostgresOperator(  
    task_id='extrai_participantes_servidores_envolvidos_extensao',  
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em  
Admin -> Connections
```

```
sql =  
Variable.get("script_SIGAA_IFSC_consulta_quantidade_participantes_servidores_en  
volvidos_extensao"), # script definido em Admin -> Variables  
    #do_xcom_push=True,  
    )
```

proporção de ações de extensão submetidas por aprovadas por câmpus e
ano (considerar a consulta anterior)

retorna quantitativo de ações de extensão submetidas

```
extrai_quantitativo_atividades_submetidas = PostgresOperator(  
    task_id='extrai_quantitativo_atividades_submetidas',  
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em  
Admin -> Connections
```

```
sql =  
Variable.get("script_SIGAA_IFSC_consulta_quantitativo_atividades_submetidas_ext  
ensao"), # script definido em Admin -> Variables  
    #do_xcom_push=True,  
    )
```

quantitativo de atividades de extensão com status favorável (segundo o
SIGAA) por câmpus e ano

```
extrai_quantitativo_atividades_favoraveis = PostgresOperator(  
    task_id='extrai_quantitativo_atividades_favoraveis',
```

```
postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
sql =
Variable.get("script_SIGAA_IFSC_consulta_quantitativo_atividades_favoraveis_exte
nsao"), # script definido em Admin -> Variables
#do_xcom_push=True,
)
```

```
# quantitativo de atividades de extensão com status desfavorável (segundo o
SIGAA) por câmpus e ano
extraí_quantitativo_atividades_desfavoraveis = PostgresOperator(
task_id='extraí_quantitativo_atividades_desfavoraveis',
postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
sql =
Variable.get("script_SIGAA_IFSC_consulta_quantitativo_atividades_desfavoraveis_e
xtensao"), # script definido em Admin -> Variables
#do_xcom_push=True,
)
```

```
# quantitativo de atividade de extensão por status por câmpus e ano
extraí_quantitativo_atividades_por_status = PostgresOperator(
task_id='extraí_quantitativo_atividades_por_status',
postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
sql =
Variable.get("script_SIGAA_IFSC_consulta_quantitativo_atividades_extensao_por_st
atus"), # script definido em Admin -> Variables
#do_xcom_push=True,
)
```

```
# percentual de atividade de extensao que articularam com a pesquisa por
câmpus e ano (considerar quantitativo total de atividades de extensão por câmpus e
ano)
```

```
# retorna o quantitativo de atividades de extensão que articularam com a
pesquisa por câmpus e ano
extraí_quantitativo_atividades_articuladas_pesquisa = PostgresOperator(
task_id='extraí_quantitativo_atividades_articuladas_pesquisa',
postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
sql =
Variable.get("script_SIGAA_IFSC_consulta_quantitativo_atividades_extensao_articul
adas_pesquisa"), # script definido em Admin -> Variables
#do_xcom_push=True,
)
```

```
# percentual de atividade de extensao que articularam com o ensino por
câmpus e ano (considerar quantitativo total de atividades de extensão por câmpus e
ano)
```

```

# retorna o quantitativo de atividades de extensão que articularam com o
ensino por câmpus e ano
extraí_quantitativo_atividades_articuladas_ensino = PostgresOperator(
    task_id='extraí_quantitativo_atividades_articuladas_ensino',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql =
Variable.get("script_SIGAA_IFSC_consulta_quantitativo_atividades_extensao_articuladas_ensino"), # script definido em Admin -> Variables
    #do_xcom_push=True,
)

```

```

# quantitativo de atividades de extensão por objetivo sustentável da ONU por
câmpus e ano (não há na base de dados o registro dos objetivos sustentáveis, mas
é possível retornar o quantitativo)
extraí_quantitativo_atividades_por_objetivo_sustentavel = PostgresOperator(
    task_id='extraí_quantitativo_atividades_por_objetivo_sustentavel',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql =
Variable.get("script_SIGAA_IFSC_consulta_quantitativo_atividades_extensao_por_objetivo_sustentavel"), # script definido em Admin -> Variables
    #do_xcom_push=True,
)

```

```

# quantitativo de atividades de extensão por objetivo sustentável da ONU por
câmpus e ano (não há na base de dados o registro dos objetivos sustentáveis, mas
é possível retornar o quantitativo)
extraí_listagem_editais_internos = PostgresOperator(
    task_id='extraí_listagem_editais_internos',
    postgres_conn_id='IFSC_SIGAA', # Conexão de banco de dados definida em
Admin -> Connections
    sql = Variable.get("script_SIGAA_IFSC_consulta_listagem_editais_internos"),
# script definido em Admin -> Variables
    #do_xcom_push=True,
)

```

```

def
calcula_proporcao_atividades_aprovadas_por_servidor_participante(ti) ->
pd.DataFrame:
    # Obtém resultado das queries
    result_quantitativo_atividades_aprovadas =
ti.xcom_pull(task_ids='extraí_quantitativo_atividades_aprovadas')
    result_servidores_participantes =
ti.xcom_pull(task_ids='extraí_participantes_servidores_envolvidos_extensao')
    # Cria os dataframes pandas
    df_quantitativo_atividades_aprovadas =
pd.DataFrame(result_quantitativo_atividades_aprovadas,
columns=['campus','cep','ano','atividades_aprovadas'])

```



```

df_servidores_participantes = pd.DataFrame(result_servidores_participantes,
columns=['campus','cep','ano','servidores_participantes'])
#mescla os dataframes
df_merged = pd.merge(
df_quantitativo_atividades_aprovadas,
df_servidores_participantes,
on=['campus','cep','ano'],
how='left'
).fillna(0)
#Calcula o percentual
df_merged['proporcao_atividades_aprovadas_por_servidor_participante'] = (
(df_merged['atividades_aprovadas']) /
df_merged['servidores_participantes']
).fillna(0)
# formata o percentual para até 2 casas decimais
df_merged['proporcao_atividades_aprovadas_por_servidor_participante'] =
df_merged['proporcao_atividades_aprovadas_por_servidor_participante'].round(2)
# substitui valores infinitos por <0.01
df_merged.replace([np.inf, -np.inf], 0, inplace=True)
# Cria um novo dataframe apenas com as colunas interessadas
df_resultado =
df_merged[['campus','cep','ano','proporcao_atividades_aprovadas_por_servidor_parti
cipante']]
return df_resultado

```

proporção de ações de extensão com status aprovada por servidor por
câmpus e ano (considerar percentual de servidores envolvidos em atividades de
extensão

transforma_proporcao_atividades_aprovadas_por_servidor_participante =
PythonOperator(

task_id='transforma_proporcao_atividades_aprovadas_por_servidor_participante',

python_callable=calcula_proporcao_atividades_aprovadas_por_servidor_participant
e

)

def calcula_proporcao_atividades_submetidas_por_aprovadas(ti) ->
pd.DataFrame:

```

# Obtém resultado das queries
result_quantitativo_atividades_submetidas =
ti.xcom_pull(task_ids='extrai_quantitativo_atividades_submetidas')
result_quantitativo_atividades_aprovadas =
ti.xcom_pull(task_ids='extrai_quantitativo_atividades_aprovadas')
# Cria os dataframes pandas
df_quantitativo_atividades_submetidas =
pd.DataFrame(result_quantitativo_atividades_submetidas,
columns=['campus','cep','ano','atividades_submetidas'])

```



```

df_quantitativo_atividades_aprovadas =
pd.DataFrame(result_quantitativo_atividades_aprovadas,
columns=['campus','cep','ano','atividades_aprovadas'])

#mescla os dataframes
df_merged = pd.merge(
df_quantitativo_atividades_submetidas,
df_quantitativo_atividades_aprovadas,
on=['campus','cep','ano'],
how='left'
).fillna(0)
#Calcula o percentual
df_merged['proporcao_atividades_submetidas_por_aprovadas'] = (
(df_merged['atividades_submetidas']) /
df_merged['atividades_aprovadas']
).fillna(0)
# formata o percentual para até 2 casas decimais
df_merged['proporcao_atividades_submetidas_por_aprovadas'] =
df_merged['proporcao_atividades_submetidas_por_aprovadas'].round(2)
# substitui valores infinitos por <0.01
df_merged.replace([np.inf, -np.inf], 0, inplace=True)
# Cria um novo dataframe apenas com as colunas interessadas
df_resultado =
df_merged[['campus','cep','ano','proporcao_atividades_submetidas_por_aprovadas']]
return df_resultado

```

proporção de ações de extensão submetidas por aprovadas por câmpus e ano

```

transforma_proporcao_atividades_submetidas_por_aprovadas =
PythonOperator(
task_id='transforma_proporcao_atividades_submetidas_por_aprovadas',
python_callable=calcula_proporcao_atividades_submetidas_por_aprovadas
)

```

def

calcula_percentual_atividades_extensao_articularam_com_pesquisa(ti) ->

pd.DataFrame:

```

# Obtém resultado das queries
result_quantitativo_atividades_pesquisa =
ti.xcom_pull(task_ids='extrai_quantitativo_atividades_articuladas_pesquisa')
result_quantitativo_atividades =
ti.xcom_pull(task_ids='extrai_quantitativo_atividades')
# Cria os dataframes pandas
df_quantitativo_atividades_pesquisa =
pd.DataFrame(result_quantitativo_atividades_pesquisa,
columns=['campus','cep','ano','atividades_pesquisa'])
df_quantitativo_atividades = pd.DataFrame(result_quantitativo_atividades,
columns=['campus','cep','ano','atividades'])

```

```

#mescla os dataframes
df_merged = pd.merge(
df_quantitativo_atividades_pesquisa,
df_quantitativo_atividades,
on=['campus','cep','ano'],
how='left'
).fillna(0)
#Calcula o percentual
df_merged['percentual_atividades_extensao_com_pesquisa'] = (
(df_merged['atividades_pesquisa'] * 100) / df_merged['atividades']
).fillna(0)
# formata o percentual para até 2 casas decimais
df_merged['percentual_atividades_extensao_com_pesquisa'] =
df_merged['percentual_atividades_extensao_com_pesquisa'].round(2)
# substitui valores infinitos por <0.01
df_merged.replace([np.inf, -np.inf], 0, inplace=True)
# Cria um novo dataframe apenas com as colunas interessadas
df_resultado =
df_merged[['campus','cep','ano','percentual_atividades_extensao_com_pesquisa']]
return df_resultado

```

percentual de atividade de extensao que articularam com a pesquisa por câmpus e ano

transforma_percentual_atividades_extensao_articularam_com_pesquisa = PythonOperator(

task_id='transforma_percentual_atividades_extensao_articularam_com_pesquisa',

python_callable=calcula_percentual_atividades_extensao_articularam_com_pesquis
a

)

def calcula_percentual_atividades_extensao_articularam_com_ensino(ti)
-> pd.DataFrame:

```

# Obtém resultado das queries
result_quantitativo_atividades_ensino =
ti.xcom_pull(task_ids='extrai_quantitativo_atividades_articuladas_ensino')
result_quantitativo_atividades =

```

```

ti.xcom_pull(task_ids='extrai_quantitativo_atividades')
# Cria os dataframes pandas
df_quantitativo_atividades_ensino =

```

```

pd.DataFrame(result_quantitativo_atividades_ensino,
columns=['campus','cep','ano','atividades_ensino'])
df_quantitativo_atividades = pd.DataFrame(result_quantitativo_atividades,
columns=['campus','cep','ano','atividades'])

```

#mescla os dataframes

```
df_merged = pd.merge(
```

```

df_quantitativo_atividades_ensino,
df_quantitativo_atividades,
on=['campus','cep','ano'],
how='left'
).fillna(0)
#Calcula o percentual
df_merged['percentual_atividades_extensao_com_ensino'] = (
    (df_merged['atividades_ensino'] * 100) / df_merged['atividades']
).fillna(0)
# formata o percentual para até 2 casas decimais
df_merged['percentual_atividades_extensao_com_ensino'] =
df_merged['percentual_atividades_extensao_com_ensino'].round(2)
# substitui valores infinitos por <0.01
df_merged.replace([np.inf, -np.inf], 0, inplace=True)
# Cria um novo dataframe apenas com as colunas interessadas
df_resultado =
df_merged[['campus','cep','ano','percentual_atividades_extensao_com_ensino']]
return df_resultado

# percentual de atividade de extensao que articularam com o ensino por câmpus e
ano
    transforma_percentual_atividades_extensao_articularam_com_ensino =
PythonOperator(

task_id='transforma_percentual_atividades_extensao_articularam_com_ensino',

python_callable=calcula_percentual_atividades_extensao_articularam_com_ensino
)

```

```

def transformacao_info():
    logging.info("Iniciando Tranformação de Dados")

```

```

inicia_transformacao = PythonOperator(
    task_id='inicia_transformacao',
    python_callable=transformacao_info
)

```

```

def cria_diretorio():
    #cria os diretórios se não existirem
    if not os.path.exists(datalake_extensao_dir):
        os.makedirs(datalake_extensao_dir)
    elif not os.path.exists(datalake_atividades_dir):
        os.makedirs(datalake_atividades_dir)

```

```

cria_datalake_atividades_extensao = PythonOperator(
    task_id='cria_datalake_atividades_extensao',
    python_callable=cria_diretorio,
)

```

```

def cria_planilha_quantitativo_atividades(ti):
    #Obtém o resultado da transformação
    result_quantitativo_atividades =
ti.xcom_pull(task_ids='extrai_quantitativo_atividades')
    #Cria um dataframe pandas
    df_quantitativo_atividades = pd.DataFrame(result_quantitativo_atividades)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_quantitativo_atividades = ['Câmpus','CEP','Ano','Ações de
Extensão']
    #Exporta para CSV
    if df_quantitativo_atividades.empty:
        df_quantitativo_atividades =
pd.DataFrame(columns=cabecalho_quantitativo_atividades)
        df_quantitativo_atividades.to_csv(csv_quantitativo_atividades_extensao,
index=False, encoding=encoding, sep=separador)
    else:
        df_quantitativo_atividades.to_csv(csv_quantitativo_atividades_extensao,
index=False, header=cabecalho_quantitativo_atividades, encoding=encoding,
sep=separador)

```

```

carrega_quantitativo_atividades = PythonOperator(
    task_id='carrega_quantitativo_atividades',
    python_callable=cria_planilha_quantitativo_atividades
)

```

```

def cria_planilha_quantitativo_atividades_continuas(ti):
    #Obtém o resultado da transformação
    result_quantitativo_atividades_continuas =
ti.xcom_pull(task_ids='extrai_quantitativo_atividades_continuas')
    #Cria um dataframe pandas
    df_quantitativo_atividades_continuas =
pd.DataFrame(result_quantitativo_atividades_continuas)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_quantitativo_atividades_continuas = ['Câmpus','CEP','Ano','Ações
Contínuas de Extensão']
    #Exporta para CSV
    if df_quantitativo_atividades_continuas.empty:
        df_quantitativo_atividades_continuas =
pd.DataFrame(columns=cabecalho_quantitativo_atividades_continuas)

    df_quantitativo_atividades_continuas.to_csv(csv_quantitativo_atividades_continuas_
extensao, index=False, encoding=encoding, sep=separador)
    else:

        df_quantitativo_atividades_continuas.to_csv(csv_quantitativo_atividades_continuas_
extensao, index=False, header=cabecalho_quantitativo_atividades_continuas,
encoding=encoding, sep=separador)

```

```

carrega_quantitativo_atividades_continuas = PythonOperator(
    task_id='carrega_quantitativo_atividades_continuas',
    python_callable=cria_planilha_quantitativo_atividades_continuas
)

""" def cria_planilha_quantitativo_atividades_aprovadas(ti):
    #Obtém o resultado da transformação
    result_quantitativo_atividades_aprovadas =
ti.xcom_pull(task_ids='extraí_quantitativo_atividades_aprovadas')
    #Cria um dataframe pandas
    df_quantitativo_atividades_aprovadas =
pd.DataFrame(result_quantitativo_atividades_aprovadas)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_quantitativo_atividades_aprovadas = ['Câmpus','CEP','Ano','Ações
de Extensão Aprovadas']
    #Exporta para CSV
    if df_quantitativo_atividades_aprovadas.empty:
        df_quantitativo_atividades_aprovadas =
pd.DataFrame(columns=cabecalho_quantitativo_atividades_aprovadas)

df_quantitativo_atividades_aprovadas.to_csv(csv_quantitativo_atividades_extensao_
aprovadas, index=False, encoding=encoding, sep=separador)
    else:

df_quantitativo_atividades_aprovadas.to_csv(csv_quantitativo_atividades_extensao_
aprovadas, index=False, header=cabecalho_quantitativo_atividades_aprovadas,
encoding=encoding, sep=separador)

carrega_quantitativo_atividades_aprovadas = PythonOperator(
    task_id='carrega_quantitativo_atividades_aprovadas',
    python_callable=cria_planilha_quantitativo_atividades_aprovadas
) """

""" def cria_planilha_quantitativo_atividades_submetidas(ti):
    #Obtém o resultado da transformação
    result_quantitativo_atividades_submetidas =
ti.xcom_pull(task_ids='extraí_quantitativo_atividades_submetidas')
    #Cria um dataframe pandas
    df_quantitativo_atividades_submetidas =
pd.DataFrame(result_quantitativo_atividades_submetidas)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_quantitativo_atividades_submetidas =
['Câmpus','CEP','Ano','Ações de Extensão Submetidas']
    #Exporta para CSV
    if df_quantitativo_atividades_submetidas.empty:
        df_quantitativo_atividades_submetidas =
pd.DataFrame(columns=cabecalho_quantitativo_atividades_submetidas)

```

```
df_quantitativo_atividades_submetidas.to_csv(csv_quantitativo_atividades_extensao_
_submetidas, index=False, encoding=encoding, sep=separador)
else:
```

```
df_quantitativo_atividades_submetidas.to_csv(csv_quantitativo_atividades_extensao_
_submetidas, index=False, header=cabecalho_quantitativo_atividades_submetidas,
encoding=encoding, sep=separador)
```

```
carrega_quantitativo_atividades_submetidas = PythonOperator(
task_id='carrega_quantitativo_atividades_submetidas',
python_callable=cria_planilha_quantitativo_atividades_submetidas
) """
```

```
def cria_planilha_quantitativo_atividades_favoraveis(ti):
#Obtém o resultado da transformação
result_quantitativo_atividades_favoraveis =
ti.xcom_pull(task_ids='extrai_quantitativo_atividades_favoraveis')
#Cria um dataframe pandas
df_quantitativo_atividades_favoraveis =
pd.DataFrame(result_quantitativo_atividades_favoraveis)
#Mapea o cabeçalho do arquivo CSV
cabecalho_quantitativo_atividades_favoraveis = ['Câmpus','CEP','Ano','Ações
de Extensão Favoráveis']
#Exporta para CSV
if df_quantitativo_atividades_favoraveis.empty:
df_quantitativo_atividades_favoraveis =
pd.DataFrame(columns=cabecalho_quantitativo_atividades_favoraveis)

df_quantitativo_atividades_favoraveis.to_csv(csv_quantitativo_atividades_extensao_
favoraveis, index=False, encoding=encoding, sep=separador)
else:
```

```
df_quantitativo_atividades_favoraveis.to_csv(csv_quantitativo_atividades_extensao_
favoraveis, index=False, header=cabecalho_quantitativo_atividades_favoraveis,
encoding=encoding, sep=separador)
```

```
carrega_quantitativo_atividades_favoraveis = PythonOperator(
task_id='carrega_quantitativo_atividades_favoraveis',
python_callable=cria_planilha_quantitativo_atividades_favoraveis
)
```

```
def cria_planilha_quantitativo_atividades_desfavoraveis(ti):
#Obtém o resultado da transformação
result_quantitativo_atividades_desfavoraveis =
ti.xcom_pull(task_ids='extrai_quantitativo_atividades_desfavoraveis')
#Cria um dataframe pandas
df_quantitativo_atividades_desfavoraveis =
pd.DataFrame(result_quantitativo_atividades_desfavoraveis)
```

```

        #Mapea o cabeçalho do arquivo CSV
        cabecalho_quantitativo_atividades_desfavoraveis =
['Câmpus','CEP','Ano','Ações de Extensão Desfavoráveis']
        #Exporta para CSV
        if df_quantitativo_atividades_desfavoraveis.empty:
            df_quantitativo_atividades_desfavoraveis =
pd.DataFrame(columns=cabecalho_quantitativo_atividades_desfavoraveis)

df_quantitativo_atividades_desfavoraveis.to_csv(csv_quantitativo_atividades_extens
ao_desfavoraveis, index=False, encoding=encoding, sep=separador)
        else:

df_quantitativo_atividades_desfavoraveis.to_csv(csv_quantitativo_atividades_extens
ao_desfavoraveis, index=False,
header=cabecalho_quantitativo_atividades_desfavoraveis, encoding=encoding,
sep=separador)

        carrega_quantitativo_atividades_desfavoraveis = PythonOperator(
            task_id='carrega_quantitativo_atividades_desfavoraveis',
            python_callable=cria_planilha_quantitativo_atividades_desfavoraveis
        )

def cria_planilha_quantitativo_atividades_por_status(ti):
    #Obtém o resultado da transformação
    result_quantitativo_atividades_por_status =
ti.xcom_pull(task_ids='extrai_quantitativo_atividades_por_status')
    #Cria um dataframe pandas
    df_quantitativo_atividades_por_status =
pd.DataFrame(result_quantitativo_atividades_por_status)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_quantitativo_atividades_por_status =
['Câmpus','CEP','Ano','Status','Ações de Extensão']
    #Exporta para CSV
    if df_quantitativo_atividades_por_status.empty:
        df_quantitativo_atividades_por_status =
pd.DataFrame(columns=cabecalho_quantitativo_atividades_por_status)

df_quantitativo_atividades_por_status.to_csv(csv_quantitativo_atividades_extensao_
por_status, index=False, encoding=encoding, sep=separador)
        else:

df_quantitativo_atividades_por_status.to_csv(csv_quantitativo_atividades_extensao_
por_status, index=False, header=cabecalho_quantitativo_atividades_por_status,
encoding=encoding, sep=separador)

        carrega_quantitativo_atividades_por_status = PythonOperator(
            task_id='carrega_quantitativo_atividades_por_status',
            python_callable=cria_planilha_quantitativo_atividades_por_status
        )

```



```

        """ def cria_planilha_quantitativo_atividades_articuladas_pesquisa(ti):
        #Obtém o resultado da transformação
        result_quantitativo_atividades_articuladas_pesquisa =
ti.xcom_pull(task_ids='extrai_quantitativo_atividades_articuladas_pesquisa')
        #Cria um dataframe pandas
        df_quantitativo_atividades_articuladas_pesquisa =
pd.DataFrame(result_quantitativo_atividades_articuladas_pesquisa)
        #Mapea o cabeçalho do arquivo CSV
        cabecalho_quantitativo_atividades_articuladas_pesquisa =
['Câmpus','CEP','Ano','Ações de Extensão Articuladas com a Pesquisa']
        #Exporta para CSV
        if df_quantitativo_atividades_articuladas_pesquisa.empty:
            df_quantitativo_atividades_articuladas_pesquisa =
pd.DataFrame(columns=cabecalho_quantitativo_atividades_articuladas_pesquisa)

df_quantitativo_atividades_articuladas_pesquisa.to_csv(csv_quantitativo_atividades
_articuladas_pesquisa, index=False, encoding=encoding, sep=separador)
        else:

df_quantitativo_atividades_articuladas_pesquisa.to_csv(csv_quantitativo_atividades
_articuladas_pesquisa, index=False,
header=cabecalho_quantitativo_atividades_articuladas_pesquisa,
encoding=encoding, sep=separador)

```

```

        carrega_quantitativo_atividades_articuladas_pesquisa = PythonOperator(
        task_id='carrega_quantitativo_atividades_articuladas_pesquisa',
        python_callable=cria_planilha_quantitativo_atividades_articuladas_pesquisa
        ) """

```

```

        """ def cria_planilha_quantitativo_atividades_articuladas_ensino(ti):
        #Obtém o resultado da transformação
        result_quantitativo_atividades_articuladas_ensino =
ti.xcom_pull(task_ids='extrai_quantitativo_atividades_articuladas_ensino')
        #Cria um dataframe pandas
        df_quantitativo_atividades_articuladas_ensino =
pd.DataFrame(result_quantitativo_atividades_articuladas_ensino)
        #Mapea o cabeçalho do arquivo CSV
        cabecalho_quantitativo_atividades_articuladas_ensino =
['Câmpus','CEP','Ano','Ações de Extensão Articuladas com o Ensino']
        #Exporta para CSV
        if df_quantitativo_atividades_articuladas_ensino.empty:
            df_quantitativo_atividades_articuladas_ensino =
pd.DataFrame(columns=cabecalho_quantitativo_atividades_articuladas_ensino)

df_quantitativo_atividades_articuladas_ensino.to_csv(csv_quantitativo_atividades_ar
ticuladas_ensino, index=False, encoding=encoding, sep=separador)
        else:

```



```
df_quantitativo_atividades_articuladas_ensino.to_csv(csv_quantitativo_atividades_ar
ticuladas_ensino, index=False,
header=cabecalho_quantitativo_atividades_articuladas_ensino, encoding=encoding,
sep=separador)
```

```
carrega_quantitativo_atividades_articuladas_ensino = PythonOperator(
task_id='carrega_quantitativo_atividades_articuladas_ensino',
python_callable=cria_planilha_quantitativo_atividades_articuladas_ensino
) """
```

```
def cria_planilha_quantitativo_atividades_por_objetivo_sustentavel(ti):
#Obtém o resultado da transformação
result_quantitativo_atividades_por_objetivo_sustentavel =
ti.xcom_pull(task_ids='extrai_quantitativo_atividades_por_objetivo_sustentavel')
#Cria um dataframe pandas
df_quantitativo_atividades_por_objetivo_sustentavel =
pd.DataFrame(result_quantitativo_atividades_por_objetivo_sustentavel)
#Mapea o cabeçalho do arquivo CSV
cabecalho_quantitativo_atividades_por_objetivo_sustentavel =
['Câmpus','CEP','Ano','Ações de Extensão com Objetivos Sustentáveis da ONU']
#Exporta para CSV
if df_quantitativo_atividades_por_objetivo_sustentavel.empty:
df_quantitativo_atividades_por_objetivo_sustentavel =
pd.DataFrame(columns=cabecalho_quantitativo_atividades_por_objetivo_sustentave
l)
```

```
df_quantitativo_atividades_por_objetivo_sustentavel.to_csv(csv_quantitativo_ativida
des_extensao_por_objetivo_sustentavel, index=False, encoding=encoding,
sep=separador)
else:
```

```
df_quantitativo_atividades_por_objetivo_sustentavel.to_csv(csv_quantitativo_ativida
des_extensao_por_objetivo_sustentavel, index=False,
header=cabecalho_quantitativo_atividades_por_objetivo_sustentavel,
encoding=encoding, sep=separador)
```

```
carrega_quantitativo_atividades_por_objetivo_sustentavel = PythonOperator(
task_id='carrega_quantitativo_atividades_por_objetivo_sustentavel',

python_callable=cria_planilha_quantitativo_atividades_por_objetivo_sustentavel
)
```

```
def
cria_planilha_proporcao_atividades_aprovadas_por_servidor_participante(ti):
#Obtém o resultado da transformação
result_proporcao_atividades_aprovadas_por_servidor_participante =
ti.xcom_pull(task_ids='transforma_proporcao_atividades_aprovadas_por_servidor_p
articipante')
```

```

#Cria um dataframe pandas
df_proporcao_atividades_aprovadas_por_servidor_participante =
pd.DataFrame(result_proporcao_atividades_aprovadas_por_servidor_participante)
#Mapea o cabeçalho do arquivo CSV
cabecalho_proporcao_atividades_aprovadas_por_servidor_participante =
['Câmpus','CEP','Ano','Proporção de Ações de Extensão Aprovadas por Servidores
Participantes']
#Exporta para CSV
if df_proporcao_atividades_aprovadas_por_servidor_participante.empty:
df_proporcao_atividades_aprovadas_por_servidor_participante =
pd.DataFrame(columns=cabecalho_proporcao_atividades_aprovadas_por_servidor_
participante)

df_proporcao_atividades_aprovadas_por_servidor_participante.to_csv(csv_proporcao
_atividades_aprovadas_por_servidor_participante, index=False,
encoding=encoding, sep=separador)
else:

df_proporcao_atividades_aprovadas_por_servidor_participante.to_csv(csv_proporcao
_atividades_aprovadas_por_servidor_participante, index=False,
header=cabecalho_proporcao_atividades_aprovadas_por_servidor_participante,
encoding=encoding, sep=separador)

carrega_proporcao_atividades_aprovadas_por_servidor_participante =
PythonOperator(

task_id='carrega_proporcao_atividades_aprovadas_por_servidor_participante',

python_callable=cria_planilha_proporcao_atividades_aprovadas_por_servidor_partic
ipante
)

def cria_planilha_proporcao_atividades_submetidas_por_aprovadas(ti):
#Obtém o resultado da transformação
result_proporcao_atividades_submetidas_por_aprovadas =
ti.xcom_pull(task_ids='transforma_proporcao_atividades_submetidas_por_aprovada
s')

#Cria um dataframe pandas
df_proporcao_atividades_submetidas_por_aprovadas =
pd.DataFrame(result_proporcao_atividades_submetidas_por_aprovadas)
#Mapea o cabeçalho do arquivo CSV
cabecalho_proporcao_atividades_submetidas_por_aprovadas =
['Câmpus','CEP','Ano','Proporção de Ações de Extensão Submetidas por Aprovadas']
#Exporta para CSV
if df_proporcao_atividades_submetidas_por_aprovadas.empty:
df_proporcao_atividades_submetidas_por_aprovadas =
pd.DataFrame(columns=cabecalho_proporcao_atividades_submetidas_por_aprova
das)

```

```
df_proporcao_atividades_submetidas_por_aprovadas.to_csv(csv_proporcao_ativida
des_submetidas_por_aprovadas, index=False, encoding=encoding, sep=separador)
    else:
```

```
df_proporcao_atividades_submetidas_por_aprovadas.to_csv(csv_proporcao_ativida
des_submetidas_por_aprovadas, index=False,
header=cabecalho_proporcao_atividades_submetidas_por_aprovadas,
encoding=encoding, sep=separador)
```

```
    carrega_proporcao_atividades_submetidas_por_aprovadas =
PythonOperator(
    task_id='carrega_proporcao_atividades_submetidas_por_aprovadas',

python_callable=cria_planilha_proporcao_atividades_submetidas_por_aprovadas
)
```

```
    def
cria_planilha_percentual_atividades_extensao_articularam_com_pesquisa(ti):
    #Obtém o resultado da transformação
    result_percentual_atividades_extensao_articularam_com_pesquisa =
ti.xcom_pull(task_ids='transforma_percentual_atividades_extensao_articularam_com
_pesquisa')
    #Cria um dataframe pandas
    df_percentual_atividades_extensao_articularam_com_pesquisa =
pd.DataFrame(result_percentual_atividades_extensao_articularam_com_pesquisa)
    #Mapea o cabeçalho do arquivo CSV
    cabecalho_percentual_atividades_extensao_articularam_com_pesquisa =
['Câmpus','CEP','Ano','Ações de Extensão Articuladas com a Pesquisa(%)']
    #Exporta para CSV
    if df_percentual_atividades_extensao_articularam_com_pesquisa.empty:
    df_percentual_atividades_extensao_articularam_com_pesquisa =
pd.DataFrame(columns=cabecalho_percentual_atividades_extensao_articularam_co
m_pesquisa)
```

```
df_percentual_atividades_extensao_articularam_com_pesquisa.to_csv(csv_percentu
al_atividades_articuladas_pesquisa, index=False, encoding=encoding,
sep=separador)
    else:
```

```
df_percentual_atividades_extensao_articularam_com_pesquisa.to_csv(csv_percentu
al_atividades_articuladas_pesquisa, index=False,
header=cabecalho_percentual_atividades_extensao_articularam_com_pesquisa,
encoding=encoding, sep=separador)
```

```
    carrega_percentual_atividades_extensao_articularam_com_pesquisa =
PythonOperator(

task_id='carrega_percentual_atividades_extensao_articularam_com_pesquisa',
```

```
python_callable=cria_planilha_percentual_atividades_extensao_articularam_com_pesquisa
    )
```

```
def
```

```
cria_planilha_percentual_atividades_extensao_articularam_com_ensino(ti):
```

```
    #Obtém o resultado da transformação
```

```
    result_percentual_atividades_extensao_articularam_com_ensino =
```

```
    ti.xcom_pull(task_ids='transforma_percentual_atividades_extensao_articularam_com_ensino')
```

```
    #Cria um dataframe pandas
```

```
    df_percentual_atividades_extensao_articularam_com_ensino =
```

```
    pd.DataFrame(result_percentual_atividades_extensao_articularam_com_ensino)
```

```
    #Mapea o cabeçalho do arquivo CSV
```

```
    cabecalho_percentual_atividades_extensao_articularam_com_ensino =
```

```
    ['Câmpus','CEP','Ano','Ações de Extensão Articuladas com o Ensino(%)']
```

```
    #Exporta para CSV
```

```
    if df_percentual_atividades_extensao_articularam_com_ensino.empty:
```

```
        df_percentual_atividades_extensao_articularam_com_ensino =
```

```
        pd.DataFrame(columns=cabecalho_percentual_atividades_extensao_articularam_com_ensino)
```

```
    df_percentual_atividades_extensao_articularam_com_ensino.to_csv(csv_percentual_atividades_articuladas_ensino, index=False, encoding=encoding, sep=separador)
```

```
    else:
```

```
    df_percentual_atividades_extensao_articularam_com_ensino.to_csv(csv_percentual_atividades_articuladas_ensino, index=False, header=cabecalho_percentual_atividades_extensao_articularam_com_ensino, encoding=encoding, sep=separador)
```

```
    carrega_percentual_atividades_extensao_articularam_com_ensino =
```

```
    PythonOperator(
```

```
        task_id='carrega_percentual_atividades_extensao_articularam_com_ensino',
```

```
python_callable=cria_planilha_percentual_atividades_extensao_articularam_com_ensino
```

```
)
```

```
def cria_planilha_listagem_editais_internos(ti):
```

```
    #Obtém o resultado da transformação
```

```
    result_listagem_editais_internos =
```

```
    ti.xcom_pull(task_ids='extrai_listagem_editais_internos')
```

```
    #Cria um dataframe pandas
```

```
    df_listagem_editais_internos =
```

```
    pd.DataFrame(result_listagem_editais_internos)
```

```
    #Mapea o cabeçalho do arquivo CSV
```

```

cabecalho_listagem_editais_internos =
['Câmpus','CEP','Ano','Descrição','Fonte Orçamentária','Financiamento']
#Exporta para CSV
if df_listagem_editais_internos.empty:
df_listagem_editais_internos =
pd.DataFrame(columns=cabecalho_listagem_editais_internos)
df_listagem_editais_internos.to_csv(csv_listagem_editais_internos,
index=False, encoding=encoding, sep=separador)
else:
df_listagem_editais_internos.to_csv(csv_listagem_editais_internos,
index=False, header=cabecalho_listagem_editais_internos, encoding=encoding,
sep=separador)

```

```

carrega_listagem_editais_internos = PythonOperator(
task_id='carrega_listagem_editais_internos',
python_callable=cria_planilha_listagem_editais_internos
)

```

```

""" envia_arquivos_email = EmailOperator(
task_id='envia_arquivos_email',
to=Variable.get("IFSC_email_extensao"),# email definido em Admin ->

```

Variables

```

subject='Indicadores de Extensão - Atividades',
html_content="Segue em anexo os arquivos",
files=lista_arquivos,
conn_id="IFSC_SMTP", # Conexão SMTP definida em Admin -> Connections
) """

```

Definir as dependências entre as tasks

```

[extrai_quantitativo_atividades,
extrai_quantitativo_atividades_continuas,
extrai_quantitativo_atividades_aprovadas,
extrai_quantitativo_atividades_submetidas,
extrai_quantitativo_atividades_favoraveis,
extrai_quantitativo_atividades_desfavoraveis,
extrai_quantitativo_atividades_por_status,
extrai_quantitativo_atividades_articuladas_pesquisa,
extrai_quantitativo_atividades_articuladas_ensino,
extrai_quantitativo_atividades_por_objetivo_sustentavel,
extrai_participantes_servidores_envolvidos_extensao,
extrai_listagem_editais_internos] >> inicia_transformacao >>
[transforma_proporcao_atividades_aprovadas_por_servidor_participante,

transforma_proporcao_atividades_submetidas_por_aprovadas,

transforma_percentual_atividades_extensao_articularam_com_pesquisa,

transforma_percentual_atividades_extensao_articularam_com_ensino] >>
cria_datalake_atividades_extensao >> [carrega_quantitativo_atividades,

```

carrega_quantitativo_atividades_continuas,
#carrega_quantitativo_atividades_aprovadas,
#carrega_quantitativo_atividades_submetidas,
carrega_quantitativo_atividades_favoraveis,
carrega_quantitativo_atividades_desfavoraveis,
carrega_quantitativo_atividades_por_status,
#carrega_quantitativo_atividades_articuladas_pesquisa,
#carrega_quantitativo_atividades_articuladas_ensino,
carrega_quantitativo_atividades_por_objetivo_sustentavel,
carrega_proporcao_atividades_aprovadas_por_servidor_participante,
carrega_proporcao_atividades_submetidas_por_aprovadas,
carrega_percentual_atividades_extensao_articularam_com_pesquisa,
carrega_percentual_atividades_extensao_articularam_com_ensino,
carrega_listagem_editais_internos]