

INSTITUTO FEDERAL DE SANTA CATARINA

RONEI SIMÕES CALISTRO

DESENVOLVIMENTO DE SOFTWARE: UM ESTUDO DE CASO EM UMA  
EMPRESA DE COSTURA

Caçador - SC

2022

RONEI SIMÕES CALISTRO

DESENVOLVIMENTO DE SOFTWARE: UM ESTUDO DE CASO EM UMA  
EMPRESA DE COSTURA

Monografia apresentada ao Curso de Sistemas de Informação do Câmpus Caçador do Instituto Federal de Santa Catarina para a obtenção do diploma de Bacharel em Sistemas de Informação.

Orientadora: Sayonara Varela, Ma.

Coorientador: Cristiano Mesquita Garcia, Me.

Caçador - SC

2022

Calistro, Ronei Simões  
C153d Desenvolvimento de software : um estudo de caso em uma empresa  
de costura / Ronei Simões Calistro ; orientadora: Sayonara Varela ;  
coorientador: Cristiano Mesquita Garcia. -- Caçador, SC, 2022.  
65 f.

Trabalho de Conclusão de Curso (Graduação)-Instituto Federal  
de Educação, Ciência e Tecnologia de Santa Catarina, Curso de  
Sistemas de Informação.

Inclui bibliografias

1. Sistemas de informação. 2. Gestão do conhecimento. 3. *Enterprise  
Resource Planning* (ERP). I. Varela, Sayonara. II. Garcia, Cristiano  
Mesquita. III. Instituto Federal de Educação, Ciência e Tecnologia de  
Santa Catarina. Curso de Sistemas de Informação. IV. Título.


CDD 658.4038

RONEI SIMÕES CALISTRO

## DESENVOLVIMENTO DE SOFTWARE: UM ESTUDO DE CASO EM UMA EMPRESA DE COSTURA

Este trabalho foi julgado adequado para obtenção do título de Bacharel em Sistemas de Informação, pelo Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina, e aprovado na sua forma final pela comissão avaliadora abaixo indicada.

Caçador - SC, 21 de novembro de 2022.

Documento assinado digitalmente  
 SAYONARA VARELA  
Data: 14/12/2022 08:41:24-0300  
Verifique em <https://verificador.iti.br>

---


**Sayonara Varela, Ma.**  
Orientador(a)  
Instituto Federal de Santa Catarina

**Cristiano Garcia**

Digitally signed by Cristiano Garcia  
DN: C=BR, O=Instituto Federal de Santa Catarina, CN=Cristiano Garcia, E=cristiano.garcia@ifsc.edu.br  
Reason: Estou aprovando este documento  
Location:  
Date: 2022.12.13 19:30:02-0300  
Foxit PDF Reader Version: 12.0.1

---

**Cristiano Mesquita Garcia, Me.**  
Coorientador(a)  
Instituto Federal de Santa Catarina

Documento assinado digitalmente  
 FABIO DOS SANTOS  
Data: 14/12/2022 17:13:11-0300  
Verifique em <https://verificador.iti.br>

---

**Fabio dos Santos, Me.**  
Banca Avaliadora  
UEPG/Mercado Livre

**Paulo Roberto  
Córdova**

Digitally signed by Paulo Roberto Córdova  
DN: C=BR, OU=Coordenadoria de Sistemas de Informação, O=Instituto Federal de Santa Catarina, CN=Paulo Roberto Córdova, E=paulo.cordova@ifsc.edu.br  
Reason: De acordo com o conteúdo e acurácia do documento.  
Location: Caçador  
Date: 2022.12.14 19:10:49-0300  
Foxit PhantomPDF Version: 10.1.0

---

**Paulo Córdova, Me.**  
Banca Avaliadora  
Instituto Federal de Santa Catarina

*“Even if life seems difficult, there is always something you can do to succeed in it.”*  
*(Stephen Hawking)*

# AGRADECIMENTOS

Agradecimento especial a todos os professores e colegas com qual adquiri algum conhecimento durante o curso, pois foram de grande importância e serão levados adiante em minha trajetória, agradecimento também aos membros da banca e meus orientadores Sayonara e Cristiano.

*“Se você quer ser bem-sucedido, precisa ter dedicação total, buscar seu último limite e dar o melhor de si.”*  
*(Ayrton Senna)*

# RESUMO

Ao passar do tempo, é visto que as empresas estão com menos tempo para organizar suas informações do dia a dia, devido ao grande trabalho demandado. Em contrapartida, é importante para todos, manter o gerenciamento e controle das informações, visando a organização de seus processos e estar por dentro de seus gastos e suas receitas, dessa forma, aumentando sua chance de gerar lucro. Sendo assim, por meio do mapeamento sistemático da literatura, considerando a especificidade que a área da costura apresenta, foi possível identificar que os Sistemas de Informação podem contribuir significativamente para solução de diversos problemas organizacionais. Dessa forma, este projeto tem como objetivo o desenvolvimento de um sistema informatizado de gestão, o qual possa aumentar a produtividade, lucratividade, acompanhamento dos processos, finanças e tomada de decisões de uma empresa da área de costura da cidade de Caçador/SC. Para isto, desenvolveu-se uma proposta de *software* que contasse com funcionalidades que suprissem as dificuldades da organização, o qual foi validado junto a mesma através de testes de aceitação, utilizando teste alpha e beta e testes de usabilidade e regressão. Apoiado a isso, chegou-se a conclusão de que o *Enterprise Resource Planning* (ERP) é o sistema mais adequado para o gerenciamento dentro da organização no contexto analisado.

Palavras-chave: Sistemas de Informação; Costura; Gestão.



# ABSTRACT

As time goes by, it is seen that companies have less time to organize their day-to-day information, due to the great amount of work demanded. On the other hand, it is important for everyone to maintain the management and control of information, aiming at the organization of their processes and to be aware of their expenses and revenues, thus increasing their chances of generating profit. Therefore, through the systematic mapping of the literature, considering the specificity that the sewing area presents, it was possible to identify that Information Systems can contribute significantly to the solution of several organizational problems. Thus, this project aims to develop a computerized management system, which can increase the productivity, profitability, monitoring of processes, finances and decision-making of a company in the sewing area in the city of Caçador/SC. For this, a proposal was developed for a software that had functionalities that would supply the organization's difficulties, which was validated through acceptance tests, using alpha and beta tests, and usability and regression tests. Supported by this, the conclusion was reached that the Enterprise Resource Planning (ERP) system is the most appropriate for the management of the organization in the analyzed context.

**Keywords:** Information Systems; Sewing; Management.

# LISTA DE ILUSTRAÇÕES

Figura 1 – Classificação de Sistemas. . . . .	16
Figura 2 – Utilização dos sistemas de informação. . . . .	17
Figura 3 – Sistema de planejamento dos recursos empresariais. . . . .	19
Figura 4 – Módulos de um sistema ERP. . . . .	21
Figura 5 – Exemplo diagrama de caso de uso. . . . .	23
Figura 6 – Exemplo diagrama de classes. . . . .	24
Figura 7 – Arquitetura de aplicações Web usando o padrão MTV. . . . .	24
Figura 8 – Diagrama de Caso de Uso. . . . .	35
Figura 9 – Diagrama de classes. . . . .	42
Figura 10 – Fórmula para encontrar o Tempo Estimado (tE). . . . .	42
Figura 11 – Protótipo - Template do sistema. . . . .	44
Figura 12 – Protótipo - Dashboard OS. . . . .	44
Figura 13 – Protótipo - Formulários. . . . .	45
Figura 14 – Protótipo - Tela de contas. . . . .	45
Figura 15 – Protótipo - Fluxo de caixa. . . . .	46
Figura 16 – Protótipo - Tela de relatórios. . . . .	46
Figura 17 – Implantação - Camada Model do app OS. . . . .	47
Figura 18 – Forms Django. . . . .	47
Figura 19 – Implantação - Templates. . . . .	48
Figura 20 – Implantação - Views. . . . .	49
Figura 21 – View OS. . . . .	49
Figura 22 – Implantação - Relatórios. . . . .	50
Figura 23 – Automação - Módulo contas. . . . .	50
Figura 24 – Análise SUS - Quantidade de resposta por pergunta. . . . .	57
Figura 25 – Análise SUS - Escala SUS de cada resposta. . . . .	57

# LISTA DE TABELAS

Tabela 1 – Tabela com palavras-chave e sinônimos. . . . .	26
Tabela 2 – Número de artigos importados e selecionados. . . . .	26
Tabela 3 – Tabela com os critérios de exclusão. . . . .	27
Tabela 4 – Tabela com os artigos selecionados. . . . .	27
Tabela 5 – Tabela com a prioridade dos requisitos. . . . .	32
Tabela 6 – Caso de uso UC001 - Cadastrar usuário. . . . .	36
Tabela 7 – Caso de uso UC002 - Cadastrar empresa. . . . .	36
Tabela 8 – Caso de uso UC003 - Cadastrar localização. . . . .	37
Tabela 9 – Caso de uso UC004 - Cadastrar status. . . . .	37
Tabela 10 – Caso de uso UC005 - Cadastrar OS. . . . .	38
Tabela 11 – Caso de uso UC006 - Cadastrar item da OS. . . . .	38
Tabela 12 – Caso de uso UC007 - Cadastrar conta. . . . .	39
Tabela 13 – Caso de uso UC008 - Alterar status da OS. . . . .	39
Tabela 14 – Caso de uso UC009 - Gerar relatório. . . . .	40
Tabela 15 – Caso de uso UC010 - Gerenciar conta. . . . .	40
Tabela 16 – Caso de uso UC011 - Gerenciar fluxo de caixa. . . . .	40
Tabela 17 – Caso de uso UC012 - Fazer login. . . . .	41
Tabela 18 – Caso de uso UC013 - Fazer logout. . . . .	41
Tabela 19 – Duração de cada etapa do projeto. . . . .	43
Tabela 20 – Público alvo teste alpha e beta. . . . .	51
Tabela 21 – Caso de teste TC001 - Realizar cadastro de usuários. . . . .	51
Tabela 22 – Caso de teste TC002 - Realizar cadastro de empresas. . . . .	51
Tabela 23 – Caso de teste TC003 - Realizar cadastro de localizações. . . . .	52
Tabela 24 – Caso de teste TC004 - Realizar cadastro de OS. . . . .	52
Tabela 25 – Caso de teste TC005 - Realizar cadastro de item OS. . . . .	53
Tabela 26 – Caso de teste TC006 - Realizar cadastro de contas. . . . .	53
Tabela 27 – Caso de teste TC007 - Realizar alteração de status da OS. . . . .	54
Tabela 28 – Caso de teste TC008 - Gerar relatórios. . . . .	54
Tabela 29 – Caso de teste TC009 - Realizar pesquisas de contas a receber/pagar. . . . .	54
Tabela 30 – Caso de teste TC010 - Realizar acesso ao Fluxo de caixa. . . . .	55
Tabela 31 – Caso de teste TC011 - Realizar login com usuário e senha inválidos. . . . .	55
Tabela 32 – Caso de teste TC012 - Realizar login com usuário e senha válidos. . . . .	55
Tabela 33 – Caso de teste TC013 - Realizar logout do sistema. . . . .	56

# LISTA DE ABREVIATURAS E SIGLAS

**ERP** - *Enterprise Resource Planning*

**ES** - Engenharia de Software

**MVC** - *Model-View-Controller*

**PERT** - *Program Evaluation and Review Technique*

**SAD** - Sistema de Apoio à Decisão

**SI** - Sistemas de Informação

**SIG** - Sistema de Informação Gerencial

**SPT** - Sistema de Processamento de Transações

**SQL** - *Structured Query Language*

**TI** - Tecnologia da Informação

**UML** - *Unified Modeling Language*

# SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>13</b>
<b>1.1</b>	<b>Problema de Pesquisa</b>	<b>14</b>
<b>1.2</b>	<b>Hipótese de Pesquisa</b>	<b>14</b>
<b>1.3</b>	<b>Objetivos</b>	<b>14</b>
1.3.1	Objetivo Geral	14
1.3.2	Objetivos Específicos	14
<b>1.4</b>	<b>Justificativa</b>	<b>15</b>
<b>1.5</b>	<b>Organização do texto</b>	<b>15</b>
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>16</b>
<b>2.1</b>	<b>Sistemas de Informação</b>	<b>16</b>
2.1.1	Sistemas empresariais básicos (operacionais)	17
2.1.2	Sistemas de informação gerencial (SIG)	18
2.1.3	Sistemas de Apoio à Decisão (SAD)	18
<b>2.2</b>	<b>O que é um ERP?</b>	<b>19</b>
2.2.1	Vantagens do ERP	19
2.2.2	Características dos sistemas ERP	20
2.2.3	Módulos ERP	20
2.2.3.1	ERP Financeiro	21
<b>2.3</b>	<b>Engenharia de Software</b>	<b>22</b>
2.3.1	Unified Modeling Language (UML)	22
2.3.1.1	Diagrama de Caso de Uso	23
2.3.1.2	Diagrama de Classes	23
2.3.2	Model-Template-View (MTV)	24
2.3.3	Testes de <i>software</i>	25
<b>3</b>	<b>ESTADO DA ARTE DA ÁREA PESQUISADA</b>	<b>26</b>
<b>3.1</b>	<b>Mapeamento Sistemático da Literatura</b>	<b>26</b>
3.1.1	Critérios de Inclusão	26
3.1.2	Critérios de Exclusão	27
<b>3.2</b>	<b>Análise dos trabalhos selecionados</b>	<b>27</b>
<b>4</b>	<b>METODOLOGIA</b>	<b>29</b>
<b>4.1</b>	<b>Recursos</b>	<b>29</b>
4.1.1	Ferramentas Utilizadas	29
4.1.1.1	Framework Django	29
4.1.1.2	PostgreSQL	30
4.1.1.3	yEd Graph Editor	30
4.1.1.4	Astah Community	30
4.1.1.5	pgAdmin	30
<b>5</b>	<b>RESULTADOS</b>	<b>31</b>
<b>5.1</b>	<b>Estudo de caso</b>	<b>31</b>

5.1.1	Contexto Empresarial . . . . .	31
5.1.1.1	Análise do processo atual . . . . .	32
5.1.2	Análise de requisitos . . . . .	32
5.1.2.1	Requisitos funcionais . . . . .	32
5.1.2.2	Requisitos não funcionais . . . . .	34
5.1.3	Diagramas . . . . .	35
5.1.3.1	Diagrama de Caso de Uso . . . . .	35
5.1.3.2	Diagrama de Classes . . . . .	35
5.1.4	Estimativas do projeto de <i>software</i> . . . . .	41
5.1.4.1	Esforço estimado . . . . .	42
5.1.4.2	Custo estimado . . . . .	43
5.1.5	Protótipo do <i>software</i> . . . . .	43
5.1.5.1	Metodologia do desenvolvimento . . . . .	43
5.1.5.2	Implantação do <i>software</i> . . . . .	46
5.1.5.3	<i>Model</i> . . . . .	46
5.1.5.4	<i>Template</i> . . . . .	47
5.1.5.5	<i>View</i> . . . . .	48
<b>5.2</b>	<b>Testes</b> . . . . .	<b>49</b>
5.2.1	Teste de aceitação . . . . .	50
5.2.1.1	Teste <i>alpha</i> e <i>beta</i> . . . . .	50
5.2.2	Teste de regressão . . . . .	52
5.2.3	Teste de usabilidade . . . . .	52
5.2.3.1	SUS . . . . .	53
<b>6</b>	<b>CONCLUSÕES</b> . . . . .	<b>58</b>
	<b>REFERÊNCIAS</b> . . . . .	<b>59</b>
	<b>APÊNDICE A – PERGUNTAS DA ENTREVISTA INICIAL</b> . . . . .	<b>61</b>
	<b>APÊNDICE B – SUS – QUESTIONÁRIO</b> . . . . .	<b>62</b>

# 1 INTRODUÇÃO

No decorrer dos anos, o uso de Sistemas de Informação por parte das organizações vem aumentando consideravelmente. [Monitora \(2021\)](#) relata que a tecnologia tornou-se necessária para suprir os desafios do mercado moderno e crescer diante de um grau cada vez maior de competitividade entre as empresas. Em decorrência da pandemia da COVID-19, essa tendência cresceu em maior velocidade e devido às vantagens que a inovação tecnológica traz como maior controle sobre as atividades, os resultados e a redução dos custos, o futuro tende a ser ainda mais digital.

Junto a isso, a dificuldade das empresas na parte de gestão tem demandado auxílio em todos os ramos. Dessa maneira, é necessário pensar em possíveis soluções em software que prestem o apoio necessário para a gestão da organização, sendo que se bem utilizados podem otimizar as tarefas da empresa e até mesmo alavancar o seu crescimento. De acordo com [Stair e Reynolds \(2016\)](#), os sistemas de informação geralmente têm seu desenvolvimento com foco no atendimento das necessidades de todo tipo de organização e pessoas. Ainda segundo [Batista \(2013\)](#), nas empresas em crescimento ou com condições para isso, o uso das inovações tecnológicas vem propiciando um melhor controle interno, pois disponibiliza as informações no tempo ideal e com confiabilidade necessária para permanecer no mercado.

Visando todos os benefícios trazidos pela Tecnologia da Informação (TI), como auxílio na gestão financeira e ajuda na tomada de decisões, por exemplo, é atrativo a sua inserção no ambiente das organizações. Atualmente no mercado, existem diversos tipos de softwares disponíveis, entretanto como destaca [Batista \(2013\)](#) alguns ramos podem necessitar alguns critérios específicos em seus sistemas, fazendo com que o desenvolvimento de uma solução específica para tal seja necessário.

Em [Batista \(2013\)](#) é visto que, empregando cenários e filtros apropriados no uso dos dados, são definidas informações que simbolizam a exceção e a maior relevância no processo organizacional. Sendo assim, em uma empresa que faz todo o processo manualmente pode obter muitos benefícios utilizando sistemas de informação. Existem diversos tipos de classificações de SI disponíveis, dentre eles destacam-se o Sistema de Informação Gerencial (SIG), Sistema de Apoio à Decisão (SAD) e o Sistema de Processamento de Transações (SPT), os quais possuem diversos recursos disponíveis em questão de informações.

Entretanto, no ramo de costura, principalmente em pequenas empresas, muitas vezes a equipe é composta por uma ou poucas pessoas, fazendo com que o tempo disponível para a realização das tarefas seja curto. Dessa forma, o uso de um sistema para a ajuda da sua gestão pode fazer com que o desenvolvimento das atividades seja mais eficaz, além de identificar possíveis falhas no processo de produção e colaborar trazendo índices de resultados ao final dos meses trabalhados, de acordo com os benefícios relatados por [Stair e Reynolds \(2016\)](#).

Assim sendo, este trabalho busca desenvolver uma solução de software que auxilie a gestão na parte financeira e de processos de uma pequena empresa no ramo de costura localizada em Caçador/SC, inicialmente identificando a principal dificuldade enfrentada pela organização e buscando supri-la a partir desse sistema. Para isso, serão utilizados conceitos de Engenharia de Software (ES) e Gestão, junto a ferramentas de programação, que facilitarão na detecção dos problemas e obtenção de resultados úteis para auxílio nos mesmos.

## 1.1 Problema de Pesquisa

A empresa em questão trabalha com o desenvolvimento de peças de costura para empresas terceiras e atualmente conta apenas com a proprietária realizando todo o trabalho, o que acaba deixando as tarefas de gerenciamento um pouco de lado, tanto da parte financeira, em levantamento de informações sobre receitas, despesas e lucro, quanto a parte de processos, em relação às ordens de serviço desenvolvidas pela organização, dessa forma, dificultando as tomadas de decisões.

No mercado, existem diversas opções de sistemas disponíveis para realização do gerenciamento financeiro, processos e tomada de decisões, entretanto, algumas empresas podem não encontrar soluções eficazes que supram suas necessidades sem que as mesmas precisem modificar seus processos consideravelmente para que possam se adaptar a utilização das tecnologias de Sistemas de Informação. Além disso, outro fator importante que pode acabar dificultando a implementação destes processos, é a especificidade que a área da costura exige, como por exemplo na organização em questão, onde apenas uma pessoa é a responsável pelo trabalho e gerenciamento da empresa. Dessa forma, se faz necessário uma abordagem detalhada para implantação de uma solução eficaz.

Durante o desenvolvimento deste projeto, pode-se destacar diversos problemas enfrentados pela empresa em relação ao seu gerenciamento:

- Mensuração sobre a renda bruta mensal;
- Falta de informações sobre os gastos mensais;
- Falha no controle de informações referentes aos clientes;
- Falha no acompanhamento de informações referente as ordens de serviço;

Visto os problemas de gerenciamento enfrentados pela empresa, a questão do problema a ser resolvida neste projeto é: Como solucionar os problemas de gestão financeira e de processos de uma pequena empresa no ramo de costura localizada em Caçador/SC utilizando Sistemas de Informação?

## 1.2 Hipótese de Pesquisa

A possível hipótese para o problema apresentado é que uma solução viável usando Sistemas de Informação para tornar mais efetivo o gerenciamento de processos e financeiro, pode ser a utilização de um software próprio.

## 1.3 Objetivos

### 1.3.1 Objetivo Geral

Este trabalho busca desenvolver um protótipo de software para a gestão financeira e de processos com informações úteis para tomada de decisões dos gestores de uma pequena empresa do ramo de costura da cidade de Caçador/SC.

### 1.3.2 Objetivos Específicos

- Mapear os processos desenvolvidos na empresa;
- Identificar os tipos de sistemas de informação utilizados para a solução do problema, por meio de revisão de literatura;



- Levantar requisitos conforme as necessidades da organização, junto às partes interessadas;
- Modelar o software a ser desenvolvido visando aspectos estruturais, comportamentais e informacionais, utilizando diagramas UML;
- Desenvolver um sistema WEB seguindo os requisitos propostos.
- Realizar testes para validação da eficiência e usabilidade do sistema, junto a empresa do projeto proposto.

## 1.4 Justificativa

Em pequenas empresas evidencia-se que o uso de sistemas de informação não seja tão comum, Prates e Ospina (2009) ressaltam em seu estudo que isso ocorre devido os recursos disponíveis serem menores, a falta de tempo ser considerada uma barreira e até mesmo uma resistência por parte da gestão à uma implantação de sistemas informatizados na organização.

De toda maneira, sabe-se a necessidade que as empresas têm de gerenciamento e controle de suas tarefas e a tecnologia da informação pode ser usada para suprir essa carência, beneficiando a organização e garantindo a entrega de resultados quando bem utilizada. Segundo (LAUDON; LAUDON, 2015, pg.11) “Das ferramentas de que os administradores dispõem, as tecnologias e os sistemas de informação estão entre as mais importantes para atingir altos níveis de eficiência e produtividade nas operações, especialmente quando combinadas com mudanças no comportamento da administração e nas práticas de negócio.”

Sendo assim, este trabalho surge a partir de uma demanda de software para ajudar na gestão financeira e de processos de uma pequena empresa no ramo de costura, onde hoje todos os processos de controle de tarefas e análise de resultados são feitos de forma manual, não tendo um controle de tempo e quantidade de tarefas que foram desenvolvidas no mês. Dessa forma, há uma dificuldade no gerenciamento da empresa, sendo que não existe uma base para verificação de gastos e lucro que a empresa obteve no mês, dificultando assim a tomada de decisões.

Embora já existam diversos sistemas informatizados de gestão disponíveis no mercado, nem todos possuem recursos específicos para atender a demanda da empresa em questão, sendo assim, o software que será desenvolvido busca suprir a carência tecnológica na área de gestão destacada e análise de resultados da organização, trazendo uma solução eficaz para o desenvolvimento da mesma.

## 1.5 Organização do texto

O restante deste trabalho está organizado da seguinte forma: No [Capítulo 2](#) são apresentados os principais conceitos relacionados a Sistemas de Informação, bem como as técnicas estudadas. No [Capítulo 3](#) são apresentados os resultados do mapeamento sistemático da literatura. No [Capítulo 4](#) são discutidos os procedimentos metodológicos. No [Capítulo 5](#) é apresentado o estudo de caso da modelagem de *software* realizada durante o projeto e os resultados dos testes realizados. Por fim, no [Capítulo 6](#) são apresentados as considerações finais acerca deste trabalho.

## 2 FUNDAMENTAÇÃO TEÓRICA

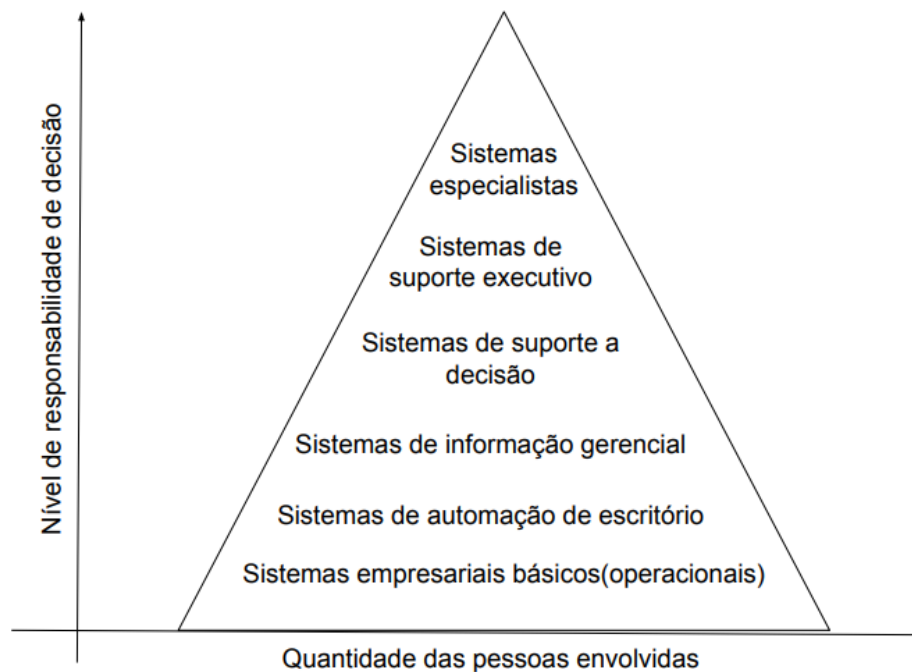
Neste capítulo são abordados os principais conceitos e tipos de sistemas de informação, os quais serão utilizados no desenvolvimento da proposta de software abordada, além das motivações para seu uso e maneiras de identificá-las. Ainda, serão apresentadas questões teóricas da área de sistemas de informação.

### 2.1 Sistemas de Informação

É possível afirmar que a Tecnologia da Informação (TI), está cada vez mais presente no dia a dia das pessoas e organizações. [Stair e Reynolds \(2016\)](#) dizem que os Sistemas de Informação (SI) são utilizados em quase todas as profissões imagináveis. Segundo [Prado e Souza \(2014\)](#), um SI pode ser entendido como um conjunto inter-relacionado de partes ou elementos que coleta, processa, armazena e distribui informações, de forma organizada e coordenada, para apoiar a tomada de decisões. Além disso, [Laudon e Laudon \(2015\)](#) afirmam que os SI também auxiliam os gerentes e trabalhadores a analisar problemas, visualizar assuntos complexos e criar novos produtos.

Visto a diversidade de sistemas existentes, imagina-se que sua classificação não seja em apenas um tipo e realmente existem diversos tipos de SI. Como [Batista \(2013\)](#) destaca, existem sistemas para as mais diversas aplicações e alguns têm uma visão voltada apenas para a gestão e coleta de dados e outros possuem uma abordagem muito mais ampla. Ainda [Batista \(2013\)](#) apresenta sua classificação de SI de acordo com a Figura 1.

Figura 1 – Classificação de Sistemas.



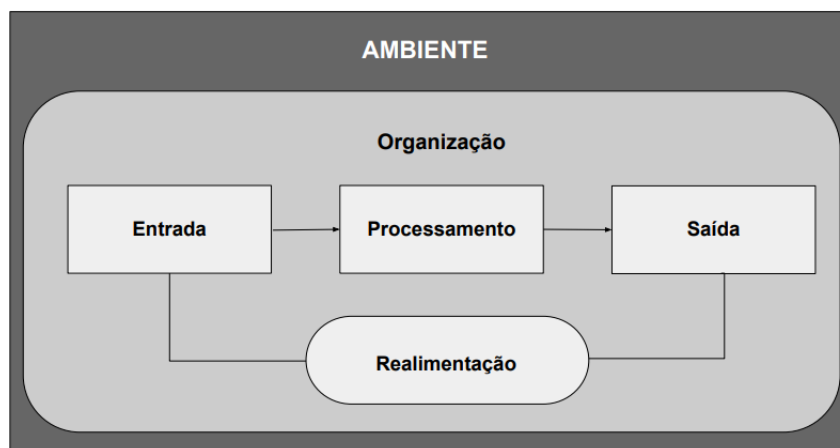
Fonte: Baseado em [Batista \(2013\)](#)

A partir da classificação de [Batista \(2013\)](#), posteriormente serão apresentados os sistemas empresariais básicos (operacionais), que visam atender justamente o nível operacional das organizações e os

Sistemas de Informação Gerencial (SIG), que busca atender os níveis de mais alto escalão das organizações, sendo o tático ou estratégico. Também citando outros SIs importantes utilizados no gerenciamento das organizações.

É possível verificar na Figura 2 a utilização dos SI dentro das organizações de acordo com [Batista \(2013\)](#), onde o quadro demonstra as fases que o sistema irá passar durante o seu uso.

Figura 2 – Utilização dos sistemas de informação.



Fonte: Baseado em [Batista \(2013\)](#)

Interpretando a Figura 2, estão presentes os seguintes elementos:

- Entrada de dados: Nesta fase acontece o recebimento de todos os dados gerados na empresa pelo desenvolvimento das suas atividades rotineiras.
- Processamento: Na segunda fase, ocorre o processo de transformação de todos os dados gerados em informações úteis para o processo de tomada de decisão.
- Saída de dados: Por fim, na saída ocorre a exibição de todos os dados que foram processados e que serão a base para as decisões gerenciais e estratégicas. Lembrando ainda que podem ocorrer casos onde os dados terão que voltar à primeira fase para uma realimentação que busca refina-los e avaliá-los de uma melhor forma.

### 2.1.1 Sistemas empresariais básicos (operacionais)

Os sistemas empresariais básicos são responsáveis por atender o nível mais primário da empresa, fazendo com que aumente o controle das atividades do dia a dia. Como destaca [Batista \(2013\)](#), algumas vezes pode não ser atribuído o devido valor a esse tipo de sistema dentro da organização, porém ele pode ser considerado essencial para o funcionamento da empresa de uma forma com que a mesma torne-se até mesmo dependente do seu uso, onde sua paralisação pode trazer prejuízos para a mesma.

Esse tipo de sistema, como o próprio nome diz, serve para executar serviços mais básicos dentro da empresa, que geralmente são realizados diariamente e são essenciais para a condução da organização. Ainda segundo [Batista \(2013\)](#), uma das principais funções dos sistemas empresariais básicos é justamente manipular e armazenar os dados básicos da empresa, buscando deixá-los disponíveis para o nível tático da organização.

Alguns exemplos comuns desse tipo de sistema destacados por [Batista \(2013\)](#) são os sistemas de emissão de notas fiscais, controle de recursos humanos ou departamento pessoal, basicamente todos os sistemas de controle de dados operacionais da organização.

### 2.1.2 Sistemas de informação gerencial (SIG)

Esse modelo de sistema, como detalha [Batista \(2013\)](#), é o responsável por auxiliar a empresa a alcançar seus objetivos, fornecendo às pessoas que tomam as decisões uma visão detalhada das informações para que elas possam controlar, organizar e planejar de forma mais ágil e eficaz. Também são conhecidos como MIS (*Management Information System* - Sistemas de Gerenciamento de Informação).

De acordo com [Stair e Reynolds \(2016, pg.443\)](#), SIG “é um conjunto integrado de pessoas, procedimentos, bancos de dados e dispositivos que fornece aos gestores e aos tomadores de decisão informações que ajudam a alcançar os objetivos organizacionais”. Além disso, eles darão uma vantagem competitiva às empresas, fornecendo informações certas no formato e tempo certos para as pessoas certas.

Ainda [Batista \(2013\)](#) ressalta que os SIGs oferecem conjuntos de relatórios resumidos sobre o desempenho da empresa, que são desenvolvidos especificamente para o negócio, possuem uma metodologia adequada, selecionando informações estratégicas e atividades críticas da empresa e utilizam ferramentas que vão permitir a viusao analítica dos dados/processos e gerar uma visão dos principais indicadores de desempenho da empresa.

Sendo assim, entende-se que todos os tipos de organizações podem utilizar os SIGs, incluindo empresas do ramo de costura, onde se é possível fazer um gerenciamento eficaz dos seus processos internos. Como exemplo de SIG utilizado para esse ramo, podemos listar os sistemas de controle de produção e sistemas de controle de estoque.

### 2.1.3 Sistemas de Apoio à Decisão (SAD)

Cada vez mais as empresas têm buscado ajuda para a tomada de decisões importantes para o futuro das mesmas, os SAD foram justamente pensados para essas situações. Segundo [Prado e Souza \(2014\)](#), são os SADs que vão apoiar as decisões estratégicas e construídos para apoio às decisões gerenciais semiestruturadas e não estruturadas. Esses sistemas possuem diversas características, entre elas destacadas por [Stair e Reynolds \(2016\)](#) estão:

- Rápido fornecimento a informações;
- Capacidade de lidar com grandes volumes de dados de diferentes fontes;
- Fornecimento de relatórios e flexibilidade na apresentação;
- Oferecimento de orientação tanto textual, quanto gráfica.
- Antevisão de oportunidades ou um problema futuro.

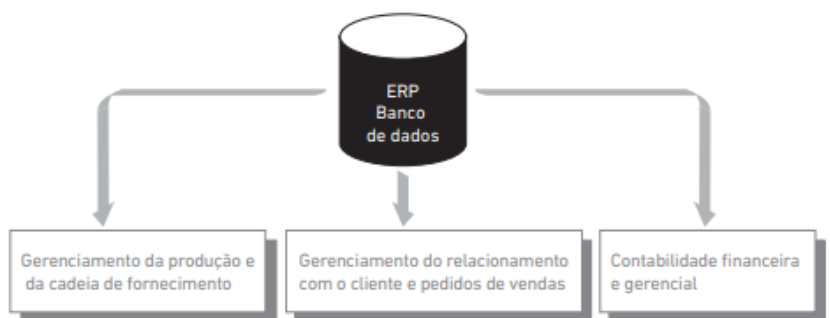
Visto isso, fica perceptível a grande vantagem que um sistema desse tipo pode trazer para a empresa, podendo oferecer também uma redução nos custos operacionais, além de trabalhar em diferentes níveis da organização.

Comparando os SAD com os SIG, pode-se ver diferenças relevantes. Segundo [Prado e Souza \(2014, pg.216\)](#), “Eles se diferenciam dos SIG, pois têm muito mais capacidade analítica, permitindo ao usuário empregar vários modelos diferentes para análise de informação”. Seguindo o raciocínio, [Stair e Reynolds \(2016, pg.468\)](#) destaca que essa diferença continua, “incluindo os tipos de problemas resolvidos, o apoio dado aos usuários, a ênfase e a abordagem da decisão, o tipo, a velocidade, o resultado e o desenvolvimento do sistema utilizado.”

## 2.2 O que é um ERP?

Sistemas ERP (*Enterprise Resource Planning*) ou Sistema de Gestão Integrado, são sistemas com tecnologia para auxiliar os gestores da organização a melhorar os processos internos e integrar atividades de diferentes setores, como finanças e RH, por exemplo. Como é apresentado na Figura 3, de acordo com [Stair e Reynolds \(2016, pg.410\)](#), “no centro do sistema ERP está um banco de dados compartilhado por todos os usuários para que todas as funções do negócio tenham acesso aos dados atuais e consistentes para a tomada de decisão operacional e planejamento”.

Figura 3 – Sistema de planejamento dos recursos empresariais.



Fonte: [Stair e Reynolds \(2016\)](#)

Como destacam [Prado e Souza \(2014, p. 39\)](#), os sistemas ERP “têm sido visualizados como potencial solução para as necessidades de processamento de informação nas organizações”. [Meireles \(2001, p. 69\)](#) ressalta que, “a solução de um ERP não é uma exclusividade das grandes corporações. Ela é acessível também as empresas de médio e pequeno portes”. Ainda segundo ele, uma de suas grandes vantagens é o fato de “adequar o software às necessidades atuais da empresa e permitir a sua evolução futura”. Dessa forma, faz-se possível sua aplicação na organização em questão deste projeto, pois apesar de possuir poucos recursos atualmente, no futuro é possível incrementar o software desenvolvido.

[Prado e Souza \(2014, p. 39\)](#) também ressalta que, “ao implantar um sistema ERP, as organizações globais têm em mente, além dos benefícios pretendidos de redução de custos, um ideal de controle e ampla visibilidade de seus negócios, enfatizado pelos processos padronizados e integrados”. Sendo assim, entende-se que as organizações que buscam a implantação de um ERP, procuram aumentar sua capacidade de governança nos processos internos.

### 2.2.1 Vantagens do ERP

Os sistemas ERP trazem vários benefícios para a organização, conforme serão apresentados na sequência, podendo trazer até mesmo uma vantagem competitiva para as mesmas, conforme destaca ([MEIRELES, 2001](#)).

Nas empresas que possuem o sistema de ERP, supõem-se que a média do grau de excelência das armas da competição influenciadas por ele deva ser significativamente maior do que nas empresas que não o possuem. Também, acredita-se que a média do grau de competitividade avaliado pela variação positiva do indicador financeiro, receita líquida, seja significativamente maior nas empresas que possuem tal sistema do que nas empresas que não o possuem ([MEIRELES, 2001, pg.72](#)).

Seguindo esse raciocínio, é possível ver diversos benefícios na implantação dos sistemas ERP nas organizações, dentre eles, [Stair e Reynolds \(2016\)](#) destacam os principais, elevação no acesso aos dados com qualidade para tomada de decisão operacional, eliminação de sistemas que não estão mais sendo

usados ou são ineficientes, melhora nos processos de trabalho e padronização da tecnologia, a seguir serão apresentados cada um:

- Melhoria de acesso aos dados com qualidade para a tomada de decisão operacional: Uma das grandes melhorias que a empresa que conta com um sistema ERP vai obter em seus processos internos é o aumento na precisão que os dados fornecidos por ele fornece para a tomada de decisões operacionais, eliminando diversos erros, pois os dados dentro do sistema são todos integrados no banco de dados.
- Eliminação de sistemas legados caros e inflexíveis: Existem empresas que possuem um ou muitos sistemas que muitas vezes são antigos e separados, onde torna-se muito difícil a adaptação dos mesmos com novas funcionalidades da empresa, seja porque os desenvolvedores não estão mais na empresa e/ou os sistemas não estão documentados. Dessa forma os sistemas ERP surgem para unificá-los e integrá-los para a empresa, suprimindo da melhor maneira as deficiências dos antigos sistemas.
- Melhoria nos processos de trabalho: Os sistemas ERP fornecem para as empresas uma melhor estrutura em seus processos de negócio. Através da realização de pesquisas, são reunidos requerimentos de diversas companhias líderes dentro do mesmo setor e combinam os mesmos com o que é descoberto em pesquisas realizadas por consultores ou instituições. Dessa forma, um sistema ERP assegura bons processos de trabalho com base nas melhores práticas.
- Atualização da infraestrutura da tecnologia: É interessante para toda organização padronizar as tecnologias de forma centralizada e organizada, dessa forma, elimina a precisão de várias plataformas de hardware, como banco de dados, por exemplo, além de reduzir custos. Com o uso de um sistema ERP, a organização terá a oportunidade de modernizar sua TI.

Visto os principais benefícios que um sistema ERP oferece às organizações e tendo em vista as necessidades que as mesmas têm de controlar e organizar seus processos e informações, fica-se claro a importância da implantação de um sistema desse tipo na maioria das empresas, incluindo empresas ingressantes no mercado.

### 2.2.2 Características dos sistemas ERP

Os sistemas de gestão integrados possuem também diversas características que os distinguem de outros pacotes e sistemas desenvolvidos internamente em outras empresas. Dentre eles, [Prado e Souza \(2014\)](#) destacam algumas:

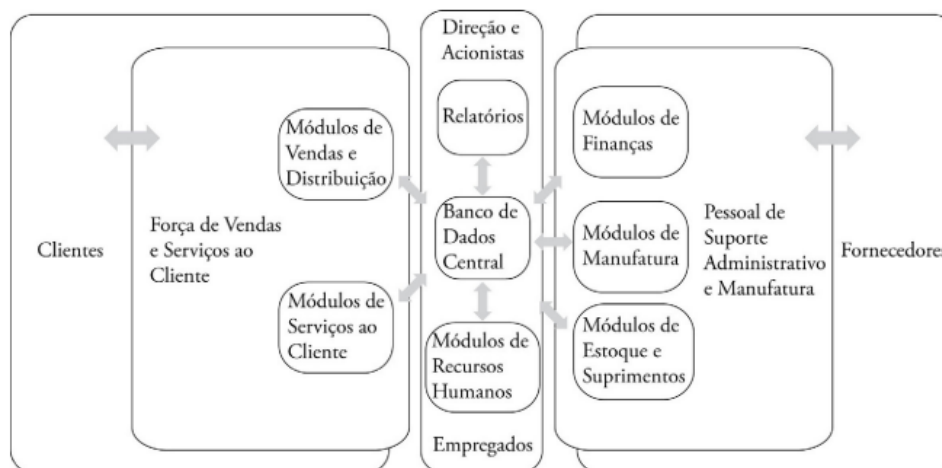
- São pacotes comerciais de software;
- Incorporam modelos de processos de negócios (*best practices*);
- São sistemas de informação integrados e utilizam um banco de dados corporativo;
- Possuem grande abrangência funcional;
- Requerem procedimentos de ajuste para que possam ser utilizados em determinada empresa.

### 2.2.3 Módulos ERP

Os sistemas ERP são compostos por diversos módulos, cada um deles relacionado a processos empresariais específicos, como finanças ou produção. Como visto em [Prado e Souza \(2014\)](#), o ERP

será composto pela conexão de todos esses módulos, o que gera uma fonte única de dados precisos e compartilhados entre os departamentos. Na Figura 4 está ilustrado os módulos geralmente contidos em um sistema ERP.

Figura 4 – Módulos de um sistema ERP.



Fonte: Prado e Souza (2014)

### 2.2.3.1 ERP Financeiro

Um dos módulos do ERP que é de grande importância para qualquer organização, sem dúvida alguma é o financeiro, como destaca Marquez (2019), ele pode aumentar a efetividade do controle financeiro da empresa, além de garantir um planejamento de recursos completo.

Marquez (2019) também comenta que um sistema ERP financeiro possui diversos recursos para auxiliar as organizações no seu gerenciamento, planejamento e controle, auxiliando também na tomada de decisões dos gestores. Ele ainda elenca um resumo com os principais benefícios de um ERP financeiro, alguns deles são:

- Confiabilidade nos dados financeiros;
- Identificação mais rápida de falhas;
- Visão geral dos negócios mais clara para o gestor;
- Mais organização e assertividade nas tomadas de decisões.

O ERP financeiro também é capaz de automatizar uma série de tarefas, facilitando a gestão do negócio, como a TOTVS (2021) relata, ele traz funcionalidades que possuem vários recursos com foco na coleta, armazenamento e processamento de informações. Algumas das principais funcionalidades são:

- Organização do fluxo de caixa;
- Geração de relatórios;
- Controle de contas a pagar e receber;
- Cadastro de clientes e fornecedores;

Visto os benefícios trazidos pelo ERP financeiro para organização, notasse a importância para o investimento no mesmo. Dessa forma, a [TOTVS \(2021\)](#) também cita alguns motivos para o investimento em um ERP financeiro, como segurança de dados e informações, padronização de atividades, auxílio na tomada de decisões, otimização de processos, mais organização, controle de prazos e redução de custos.

## 2.3 Engenharia de Software

A Engenharia de Software pode ser definida como o ramo da engenharia que tem o foco em todos os aspectos do desenvolvimento de um software, desde as fases iniciais até mesmo depois de pronto. Ela deve ser levada em consideração por parte dos desenvolvedores/empresas, pois possui diversos benefícios para ambos, como maior eficiência e diminuição de custos, por exemplo. Isso, sem dúvida é muito importante, um dos motivos é o descrito a seguir por ([SOMMERVILLE, 2011](#)).

Geralmente é mais barato, a longo prazo, usar métodos e técnicas da engenharia de software para sistemas de software, em vez de simplesmente escrever os programas como se fossem algum projeto pessoal. Para a maioria dos sistemas, a maior parte do custo é mudar o software depois que ele começa a ser usado ([SOMMERVILLE, 2011](#), pg.5).

Com base nisso, fica-se claro que a construção de um software é um processo que deve ser bem planejado. Pensando dessa maneira deverão ser seguidos alguns métodos no processo de desenvolvimento deste projeto. Em todos, serão levados em consideração recursos e técnicas da Engenharia de Software, como:

- Levantamento de requisitos: Essa é uma das etapas mais importantes, pois é nela que ocorre a descrição das funções que o sistema irá realizar, sendo que se mal executada por afetar na entrega do projeto final. Busca-se nela atender a demanda do cliente.
- Modelagem de sistemas: Na modelagem é onde são feitos todos os modelos abstratos do sistema que será produzido, onde cada um apresenta uma perspectiva diferente do sistema, conforme visto em ([SOMMERVILLE, 2011](#), pg.82). Neste projeto isso será feita com a ajuda da linguagem de notação UML, onde será possível ter uma visão detalhada do futuro sistema.

Dessa forma, no final da construção do software é possível fazer a validação da solução proposta, verificando se o mesmo atende a todas as demandas iniciais solicitadas pelo cliente, que foram ressaltadas durante o levantamento dos requisitos e desenvolvidos conforme a modelagem. A validação feita pelo cliente irá salientar se a proposta foi adequada para a resolução do problema.

### 2.3.1 Unified Modeling Language (UML)

A *Unified Modeling Language*, ou Linguagem Unificada de Modelagem, como destaca [Noletto \(2020\)](#) “é uma linguagem de notação utilizada para modelar e documentar as diversas fases do desenvolvimento de sistemas orientados a objetos”. Ou seja, com a ajuda de elementos gráficos irá ajudar as equipes de desenvolvimento a visualizarem os aspectos da aplicação a ser desenvolvida, facilitando a compreensão do seu funcionamento.

Esses gráficos podem ser chamados de diagramas, que vão representar os elementos do modelo. A linguagem UML em sua versão 2.0 é composta por 14 tipos de diagramas, sendo os principais: diagrama de caso de uso, de classes, de objeto, de pacotes, de sequência, de atividade.

Neste projeto, serão utilizados como auxílio os dois primeiros diagramas citados, o diagrama de caso de uso e o diagrama de classes.



### 2.3.1.1 Diagrama de Caso de Uso

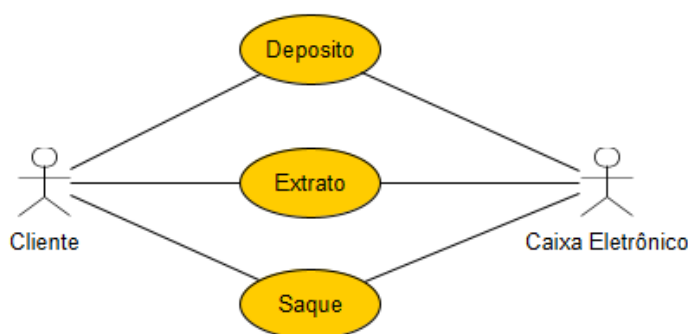
Esse tipo de diagrama é muito utilizado na fase de levantamento de requisitos. Essencialmente por ele é documentado e descrito todas as funcionalidades do sistema, além das interações do usuário com cada uma delas.

Como destaca [Leandro \(2012\)](#), os diagramas de casos de uso são compostos por quatro partes, descritas abaixo:

- Cenário: A sequência de eventos que acontecem quando um usuário interage com o sistema;
- Ator: O usuário do sistema;
- Caso de Uso: Cada tarefa ou funcionalidade realizada pelo ator;
- Comunicação: Liga um ator com um caso de uso.

Como exemplo, o diagrama da Figura 5 especifica quais atribuições de um cliente e um caixa eletrônico dentro de um banco.

Figura 5 – Exemplo diagrama de caso de uso.



Fonte: Elaborada pelo autor, 2022.

### 2.3.1.2 Diagrama de Classes

Os diagramas de classes são usados no desenvolvimento de modelos de sistemas orientados a objetos, buscando mostrar as classes de um sistema e as associações entre essas classes ([SOMMERVILLE, 2011](#)). Esse diagrama oferece diversos benefícios, como uma melhor visão geral dos esquemas da aplicação, sendo ela simples ou complexa. Além disso, seus gráficos são bem detalhados e fornecem uma descrição precisa sobre as funcionalidades. De acordo com [Douglas \(2018\)](#), o diagrama além de possuir relacionamentos, é composto por três partes:

- Parte superior: Onde irá conter o estereótipo e o nome da classe.
- Parte do meio: Parte que contém os atributos da classe. Geralmente onde será descrito as qualidades da mesma e só é necessária quando será feita essa descrição. Contém também o tipo de dado do atributo e a visibilidade dos mesmos, podendo ela ser pública, privada ou protegida.

- Parte inferior: Geralmente é o local onde ficam os métodos (operações) da classe. Sendo composta por uma lista onde cada método irá ocupar sua linha e vão descrever como a classe vai interagir com os dados.

Como exemplo, a Figura 6 especifica o diagrama de classes de um sistema de compras, com relacionamento entre os produtos e o carrinho. Como pode ser visto, a relação é que o carrinho possui vários produtos.

Figura 6 – Exemplo diagrama de classes.



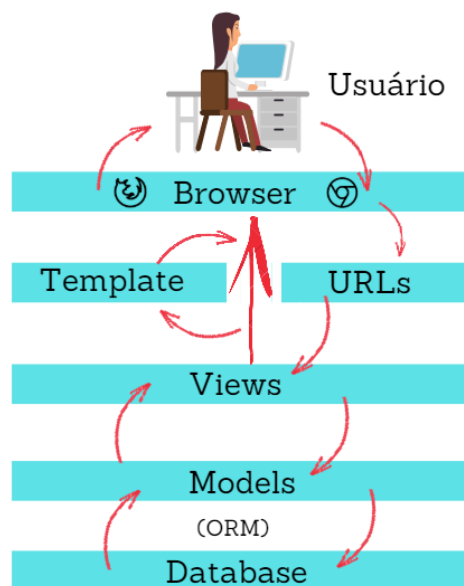
Fonte: Elaborada pelo autor, 2022.

### 2.3.2 Model-Template-View (MTV)

Se preocupando com a arquitetura da aplicação, será utilizado o MTV (Model, View e Template), um padrão utilizado pelo *framework* Django, responsável por contribuir na otimização da velocidade entre as requisições que são executadas pelos comandos dos usuários.

É um padrão que consiste em deixar o código mais legível e organizado, facilitando possíveis manutenções futuras. Possui semelhanças com o padrão MVC (Model, View e Controller), porém como é relatado na documentação [Foundation \(2013\)](#) pelos seus desenvolvedores, o *framework* prefere tratar o conteúdo de apresentação ao usuário como template e não view, conforme o primeiro.

Figura 7 – Arquitetura de aplicações Web usando o padrão MTV.



Fonte: [Pinheiro \(2019\)](#)

Pode-se verificar ainda a diferença detalhada entre cada uma das camadas da arquitetura Model, Template e View logo abaixo:

- Model: Sua responsabilidade é realizar o mapeamento do banco de dados para o projeto.
- Template: É a responsável por apresentar ao usuário as informações de forma visual. Ou seja, é onde será feita toda a interação com o usuário, usando recursos como botões e telas, por exemplo.
- View: Aqui vai acontecer a intermediação das requisições enviadas pelo Template com as respostas disponibilizadas pela Model. Nessa camada, será descrito quais dados serão apresentados ao usuário.

### 2.3.3 Testes de *software*

Os testes de *software* são utilizados para demonstração de como está o funcionamento do sistema e também para identificação de problemas do programa antes da liberação completa para uso, como visto em [Sommerville \(2011\)](#). O autor destaca ainda que existem diversos tipos de testes que podem ser aplicados para avaliação do sistema, entre eles temos:

- Testes de usuário.
- Testes de desenvolvimento.
- Teste unitário.
- Teste de componente.
- Teste de sistema.
- Testes de release.

[Stair e Reynolds \(2016\)](#) também salientam que se os testes forem inadequados podem muitas vezes resultar em problemas e erros, dessa forma, para garantir que um sistema de informação novo ou modificado funcione conforme o esperado, bons procedimentos de testes são essenciais.

## 3 ESTADO DA ARTE DA ÁREA PESQUISADA

A elaboração da pesquisa e seleção de trabalhos relacionados e conceitos foi realizada com foco em um mapeamento sistemático sobre artigos de pesquisas com propostas de resolução do problema de gerenciamento de tarefas em uma empresa do ramo de costura, utilizando para isso uma solução em software de Sistemas de Informação. Esta análise resultou na descoberta e seleção dos principais artigos de pesquisa no tema deste projeto de trabalho de conclusão de curso. Outro objetivo deste mapeamento sistemático foi verificar a aplicação de Sistemas de Informação, principalmente sistemas ERP, em micro ou pequenas empresas, visando sempre a parte de gestão das mesmas.

### 3.1 Mapeamento Sistemático da Literatura

Para realização do mapeamento sistemático, foi utilizado a ferramenta de pesquisa Google Acadêmico, a qual permite realizar buscas de acordo com diversos filtros e em diversos idiomas, simplificando a realização do mapeamento. Para iniciar a pesquisa na ferramenta, primeiramente foram definidos termos e palavras relacionadas que pudessem ajudar a montar uma *string* (sequência de caracteres) de busca para realizar as consultas nas bases de dados relacionadas. A questão definida foi:

- Como o desenvolvimento de um software pode auxiliar a gestão de uma pequena empresa do ramo de costura?

A partir dessa questão foi extraída as palavras chaves para montagem da *string* de busca, como é possível ver na Tabela 1.

Tabela 1 – Tabela com palavras-chave e sinônimos.

Palavra-chave	Sinônimo
Costura	Facção, Atelier, Malharia
Software	Desenvolvimento de Software, Solução de Software, Implantação de Software
Empresa	Microempresa, Pequena Empresa

Fonte: Elaborada pelo autor, 2022.

A Tabela 2 apresenta a quantidade de artigos retornados após a pesquisa utilizando a *string* de busca na base de dados Google Acadêmico. Fazendo a aplicação dos critérios descritos a seguir, somente três artigos estão realmente de acordo com a proposta desejada. Dessa forma, este trabalho levará em consideração a metodologia por eles abordada.

Tabela 2 – Número de artigos importados e selecionados.

Base de Dados	Artigos Importados	Artigos Selecionados
Google Acadêmico	93	3

Fonte: Elaborada pelo autor, 2022.

#### 3.1.1 Critérios de Inclusão

Para inclusão de um trabalho na pesquisa, o mesmo deveria atender a algum dos critérios de inclusão definidos a seguir:

- Conter a descrição e desenvolvimento de um sistema de gestão para empresas do ramo de costura;
- Artigos com idioma português brasileiro;
- Ter sido publicado a partir de 2016;
- Descrever um sistema ERP em micro ou pequenas empresas.

### 3.1.2 Critérios de Exclusão

Para realização da pesquisa nas bases de dados utilizando a *string* de pesquisa, ficou definido que seriam removidos trabalhos que constasse nos critérios de exclusão descritos na Tabela 3.

Tabela 3 – Tabela com os critérios de exclusão.

Critério de exclusão	Nº de artigos recusados
O estudo não refere a área da pesquisa	24
Estudos que não são artigos	11
Artigos com data de publicação anterior a 2016	55

Fonte: Elaborada pelo autor, 2022.

A seleção começou com 93 arquivos no total. Com a aplicação dos critérios de exclusão, observa-se um resultado de apenas 3 artigos. Isso ocorreu, pois 55 arquivos foram eliminados no critério “Artigos com data de publicação anterior a 2016”, que significa que os arquivos foram publicados em datas anteriores ao que este projeto busca. Outros 24 arquivos foram eliminados no critério “O estudo não refere a área da pesquisa”, que significa que eram estudos que não tinham relação com a área de pesquisa deste trabalho, fugindo por completo do tema. Foram encontrados também 11 arquivos que foram eliminados no critério “Estudos que não são artigos”, significando que se tratavam de trabalhos de conclusão de curso e afins, ou seja, não eram artigos.

## 3.2 Análise dos trabalhos selecionados

Para dar início a última etapa do Mapeamento Sistemático da Literatura, foram extraídos os dados dos artigos com a finalidade de compreender as principais técnicas e métodos utilizados para resolução de problemas similares aos propostos neste projeto. Na Tabela 4 estão listados os três artigos selecionados com seus respectivos autores.

Tabela 4 – Tabela com os artigos selecionados.

ID	Título do artigo	Autores/Autor(a)
A1	Implantação de software de eficiência produtiva em microempresa de confecção	Figueira e Orikasa (2021)
A2	Aplicação de ferramentas de controle de qualidade em uma empresa de confecção do interior do estado de São Paulo	Silva e Tarrento (2020)
A3	Os sistemas de informação como fator competitivo: Um estudo de caso em uma prestadora de serviços de costura	Cavichioli et al. (2016)

Fonte: Elaborada pelo autor, 2022.

Com foco nos artigos selecionados, pode-se identificar que a maioria dos autores utilizaram um sistema de informação na resolução do problema que diz respeito ao auxílio no gerenciamento de processos

na empresa, sendo o *software* mais indicado o ERP. A seguir serão apresentados aqueles que obtiveram os melhores resultados.

No trabalho de ID A1 e A3, os autores reforçam que para manter-se ou tornar-se competitivas no mercado, as empresas precisam investir em novas tecnologias. O trabalho A1 ressalta também, que microempresas sofrem dificuldades relativas à gestão e baixo investimento em informatização. Dessa forma, pode-se concluir que muitas delas optam por usar um sistema próprio como sua primeira experiência de uso, devido ao fato que sistemas com relevância no mercado geralmente possuem altos custos de implantação e manutenção.

No que se refere à uso de *softwares* como ferramenta para auxílio no gerenciamento dos processos da empresa, o trabalho de ID A2 afirma através do uso de dados fornecidos por um sistema ERP, que foram possíveis identificar problemas e buscar resultados satisfatórios como a redução de retrabalho.

Ainda no trabalho de ID A3, o autor além de firmar a importância dos SI conclui que o uso de sistemas de informação é uma ferramenta fundamental para sobrevivência e continuidade das organizações, pois conforme avaliou, o uso da mesma traz diversos benefícios tanto para gestão quanto para a operação das empresas.

Na sequência deste projeto, serão aprofundados os estudos considerando as técnicas citadas nesse mapeamento sistemático, como uma forma de aproveitar os esforços já realizados pelos pesquisadores desta área de pesquisa, buscando avançar o estado da arte e servir de base para trabalhos futuros.

## 4 METODOLOGIA

Inicialmente foi feita uma pesquisa exploratória da bibliografia, buscando compreender melhor os tipos de Sistemas de Informação que estão sendo utilizados pelas empresas para auxiliar em seus processos gerenciais. A partir dessa pesquisa, foi possível definir os modelos que serviram de base para o desenvolvimento deste projeto.

Com base nisso, foi levantado qual a principal dificuldade enfrentada pela organização em seu gerenciamento financeiro e desenvolvido um sistema que ajudasse na remoção da mesma. Para alcançar os objetivos desse projeto, foi adotado uma metodologia de trabalho que se baseou nas seguintes etapas:

- Realização de entrevistas com a responsável pela gestão da empresa, com objetivo de levantar o principal problema enfrentado nas atividades do setor financeiro e de processos;
- Mapeamento dos processos, visando identificar o que podia ser melhorado;
- Identificação dos principais tipos de Sistemas de Informação utilizados no gerenciamento de empresas aplicado ao ramo da costura, como primeiro objetivo utilizar o mapeamento sistemático apresentado no trabalho;
- Produção da modelagem a partir do levantamento de requisitos do sistema;
- Desenvolvimento do software considerando os requisitos previamente definidos e a modelagem feita;
- Validação do *software*, realizando testes aplicados junto a organização;

### 4.1 Recursos

Foram utilizados conceitos e técnicas básicas que fazem-se presentes na Engenharia de Software, com uso de alguns diagramas da linguagem de modelagem UML apresentadas no Capítulo 2. Junto a isso, para auxílio no desenvolvimento deste projeto, foi utilizado a metodologia MTV, presente no *framework* Django, visando a agilidade e padronização. Na sequência, serão mencionadas algumas das ferramentas que foram utilizadas para o desenvolvimento da aplicação.

#### 4.1.1 Ferramentas Utilizadas

A fim de auxiliar o processo de desenvolvimento deste projeto, foram propostos alguns softwares que servem de auxílio no planejamento e desenvolvimento da proposta de software. As ferramentas propostas estão listadas a seguir.

##### 4.1.1.1 Framework Django

O Django<sup>1</sup> trata-se de um *framework* de aplicações Web escrito em Python, gratuito e de código aberto. Possui a característica de incentivar o desenvolvimento rápido e o design limpo e pragmático, oferecendo ferramentas que agilizam o desenvolvimento do software e garantem a segurança do mesmo, além de utilizar a arquitetura de software semelhante ao MVC no seu desenvolvimento.

<sup>1</sup> Site Django <<https://www.djangoproject.com/>>.

#### 4.1.1.2 PostgreSQL

O PostgreSQL<sup>2</sup> é um dos sistemas gerenciadores de banco de dados (SGBD) objeto-relacional mais utilizados no mercado, gratuito e código aberto. Também é considerado um SGBD robusto e que possui muitos recursos disponíveis.

Uma das suas características marcantes é a possibilidade de fazer consultas complexas, além de suportar um intenso fluxo de dados garantindo estabilidade e segurança.

#### 4.1.1.3 yEd Graph Editor

O yEd Graph Editor<sup>3</sup> é um aplicativo desktop capaz de gerar diagramas de alta qualidade de uma forma rápida e eficaz. Com ele é possível criar diversos tipos de diagramas, entre eles o diagrama de classe UML, que será desenvolvido nesse projeto.

Entre suas principais características estão a possibilidade de importar dados de planilhas Excel ou XML e exportá-los em gráficos como PNG e JPG, além da criação de diagramas facilitados com uma interface intuitiva.

#### 4.1.1.4 Astah Community

O Astah Community<sup>4</sup> é um software para modelagem UML que disponibiliza diversos tipos de diagramas como o de Casos de Uso, desenvolvido neste projeto. Trata-se de um software destinado a auxiliar o trabalho de maneira rápida e fácil.

#### 4.1.1.5 pgAdmin

O pgAdmin<sup>5</sup> é uma ferramenta gerenciadora para PostgreSQL e bancos de dados relacionais derivados, que pode ser executada como um aplicativo web ou desktop. Ela permite fazer basicamente todas as tarefas necessárias de administração do banco de dados e geralmente vem instalada automaticamente durante o processo de instalação do PostgreSQL, que é o banco de dados que será utilizado neste projeto.

---

<sup>2</sup> Site PostgreSQL <<https://www.postgresql.org/>>.

<sup>3</sup> Site yEd Graph Editor <<https://www.yworks.com/products/yed>>.

<sup>4</sup> Site Astah Community <<https://astah.net/products/astah-community/>>.

<sup>5</sup> Site pgAdmin <<https://www.pgadmin.org/>>.



## 5 RESULTADOS

### 5.1 Estudo de caso

O principal objetivo deste capítulo é apresentar os resultados da pesquisa, apresentando um estudo de caso de modelagem de *software* em uma empresa do ramo de costura e os testes aplicados sobre o *software*. Para isso, serão apontados os requisitos necessários ao funcionamento do sistema, também, apresentado diagramas que mostram como deve funcionar o sistema desenvolvido, além do protótipo e os casos de teste.

O *software* desenvolvido tem como finalidade ajudar no gerenciamento financeiro da empresa, além de ampliar a visão da gestora ajudando na tomada de decisões. O sistema foi modelado levando em consideração as técnicas de UML, respeitando os padrões da linguagem e seguindo os requisitos levantados na seção abaixo.

#### 5.1.1 Contexto Empresarial

A empresa ora estudada trata-se de um ateliê de costura, situado no município de Caçador - SC. O negócio da empresa baseia-se no desenvolvimento de peças de costura para empresas terceiras, onde todo esse trabalho é feito pela atual proprietária da empresa.

Como todo o trabalho é centralizado sobre uma pessoa, a parte gerencial da empresa tem dificuldades no levantamento e controle de informações sobre o andamento da organização. Atualmente o fluxo de trabalho segue da seguinte forma:

- A empresa terceira entra em contato com ateliê, através de meios como Whatsapp;
- É feito o acordo e as peças são enviadas para o ateliê;
- É desenvolvido o trabalho de costura sobre as peças enviadas;
- As peças são entregues novamente para as empresas.

Em entrevista com a gestora do ateliê, foi constatado que atualmente, nem todos os dados de serviços realizados são arquivados ou organizados, sendo que a pouca parte guardada, é armazenada em pastas de papel, dificultando o controle ou geração de estatística sobre eles. Também foram relatados alguns problemas enfrentados pela empresa em relação a seu gerenciamento financeiro.

- Falta de informação sobre renda bruta mensal da empresa;
- Falta de informação sobre os gastos mensais do ateliê;
- Mensuração de lucros obtidos.

Como as ordens de serviço em geral são enviadas junto às peças e não há nenhum tipo de sistema informatizado que registre as informações contidas nela, caso não sejam arquivados os papéis não é possível fazer o controle das informações posteriormente.

### 5.1.1.1 Análise do processo atual

Como pode-se ver na descrição do processo do Item 5.1.1, observamos que a empresa possui um controle ineficaz sobre o levantamento de informações financeiras. O ateliê possui apenas parte das ordens de serviços desenvolvidas, não possuindo informações sobre a quantidade e lucro obtido sobre as mesmas ao final do mês de trabalho.

Manter todas essas informações é essencial para que a empresa identifique onde está o seu ponto forte e possa tomar decisões que façam com que a mesma cresça no mercado. Além disso, com eles será possível identificar por exemplo se é necessário a contratação de mais funcionários para desempenhar determinadas funções. Na próxima seção, será apresentada a modelagem de sistema baseada em estudos de caso, que pode auxiliar a empresa na organização da parte financeira.

### 5.1.2 Análise de requisitos

Após o estudo e análise do processo atual, foi realizado o levantamento de requisitos para assegurar a possibilidade de implementação do projeto. Para que isso fosse realizado, foram analisados os métodos existentes na organização, junto a discussões com a gestora da mesma. O principal objetivo deste processo é estabelecer os requisitos funcionais e não funcionais, tendo em vista a estruturação da arquitetura do sistema e a determinação de quais os requisitos devem ser destinados a quais elementos do sistema.

Levando em consideração o gerenciamento do escopo das etapas deste projeto, é preciso ter em mente que o grau de prioridade que um requisito terá no sistema pode ser definido conforme Tabela 5, que tem como objetivo identificar o grau conforme a necessidade da empresa, fazendo com que os requisitos sejam bem elaborados e não deixando com que a funcionalidade do sistema seja deficitária neste quesito. A Tabela se baseia no método MoSCoW, que segundo Pires (2019), trata-se de uma "técnica de priorização usada na gestão como um todo, análise de negócios, gestão de projetos e desenvolvimento de softwares com o intuito de encontrar um entendimento em comum entre as partes interessadas sobre a importância que elas atribuem a cada requisito."

Tabela 5 – Tabela com a prioridade dos requisitos.

Prioridade	Característica
Must Have (Tenho que fazer)	O sistema precisa dele para o seu funcionamento.
Should Have (Devo fazer)	Sem ele o sistema irá funcionar, porém não como deveria.
Could Have (Poderia fazer)	Não interfere no funcionamento do sistema, porém seria legal ter.
Won't Have (Não vou fazer)	Não será feito no momento, talvez no futuro.

Fonte: Pires (2019)

#### 5.1.2.1 Requisitos funcionais

Como destaca Sommerville (2011), requisitos funcionais são responsáveis por descrever o que o sistema deve fazer. Irão depender do tipo de software que será desenvolvido e de quem serão seus possíveis usuários. Para este estudo de caso, foram levantados os seguintes requisitos funcionais que o sistema deverá atender:

##### **RF 0001** Nível de prioridade: Devo fazer

**Nome:** Manter cadastro de usuários.

**Descrição:** O sistema deve manter o cadastro de usuários com permissões específicas para cada funcionalidade.

---

<b>RF 0001.1</b>	<b>Nível de prioridade: Poderia fazer</b>
<b>Nome:</b>	Histórico de alterações de usuários.
<b>Descrição:</b>	O sistema deve manter um histórico de alterações feitas no cadastro de um usuário.
<b>RF 0002</b>	<b>Nível de prioridade: Tenho que fazer</b>
<b>Nome:</b>	Manter cadastro de empresas/fornecedores.
<b>Descrição:</b>	O sistema deve manter o cadastro empresas e possíveis fornecedores.
<b>RF 0003</b>	<b>Nível de prioridade: Tenho que fazer</b>
<b>Nome:</b>	Manter cadastro de localizações.
<b>Descrição:</b>	O sistema deve manter um cadastro de localizações, com a cidade e estado de uma empresa/fornecedor.
<b>RF 0004</b>	<b>Nível de prioridade: Tenho que fazer</b>
<b>Nome:</b>	Fluxo de caixa mensal
<b>Descrição:</b>	O sistema deve conter a funcionalidade de fluxo de caixa mensal, visando manter o controle financeiro da empresa.
<b>RF 0004.1</b>	<b>Nível de prioridade: Devo fazer</b>
<b>Nome:</b>	Nível de acesso fluxo de caixa
<b>Descrição:</b>	O acesso ao fluxo de caixa deve ser feito apenas pelo usuário admin, não permitindo acesso a demais usuários do sistema, conforme hierarquia.
<b>RF 0005</b>	<b>Nível de prioridade: Tenho que fazer</b>
<b>Nome:</b>	Contas a receber
<b>Descrição:</b>	O sistema deve possuir a funcionalidade de contas a receber, onde serão visualizados todas as receitas da empresa.
<b>RF 0005.1</b>	<b>Nível de prioridade: Devo fazer</b>
<b>Nome:</b>	Nível de acesso contas a receber
<b>Descrição:</b>	O acesso a funcionalidade de contas a receber poderá ser realizado apenas pelo usuário admin.
<b>RF 0006</b>	<b>Nível de prioridade: Tenho que fazer</b>
<b>Nome:</b>	Contas a pagar
<b>Descrição:</b>	O sistema deve possuir a funcionalidade de contas a pagar, onde serão visualizados todas as dívidas da empresa.
<b>RF 0006.1</b>	<b>Nível de prioridade: Devo fazer</b>
<b>Nome:</b>	Nível de acesso contas a pagar
<b>Descrição:</b>	O acesso a funcionalidade de contas a pagar poderá ser realizado apenas pelo usuário admin.
<b>RF 0007</b>	<b>Nível de prioridade: Tenho que fazer</b>
<b>Nome:</b>	Manter cadastro de Contas
<b>Descrição:</b>	O sistema deve manter o cadastro de contas a receber/pagar, onde os dados serão responsáveis por alimentar o menu de contas.
<b>RF 0008</b>	<b>Nível de prioridade: Tenho que fazer</b>
<b>Nome:</b>	Relatórios
<b>Descrição:</b>	O sistema deve fornecer relatórios com informações relevantes para empresa, voltados a parte financeira.

**RF 0008.1 Nível de prioridade: Devo fazer****Nome:** Nível de acesso relatórios**Descrição:** O acesso aos relatórios poderá ser realizado apenas pelo usuário admin.**RF 0009 Nível de prioridade: Tenho que fazer****Nome:** Manter cadastro de OS**Descrição:** O sistema deve manter o cadastro de ordens de serviço, onde os dados serão responsáveis por alimentar outras funcionalidades do sistema.**RF 0009.1 Nível de prioridade: Tenho que fazer****Nome:** Manter cadastro de itens da OS**Descrição:** O sistema deve manter o cadastro de itens de uma ordem de serviço, caso existam.**RF 0009.2 Nível de prioridade: Tenho que fazer****Nome:** Manter cadastro de status**Descrição:** O sistema deve manter o cadastro de status, onde serão gravados os status que uma ordem de serviço se encontra.**RF 0009.3 Nível de prioridade: Poderia fazer****Nome:** Manter histórico de status**Descrição:** O sistema deve manter o histórico de alteração de status, onde serão gravados todas as alterações de status que uma ordem de serviço teve.

## 5.1.2.2 Requisitos não funcionais

Já os requisitos não funcionais, são definidos por [Sommerville \(2011\)](#), como requisitos que não estão diretamente relacionados às funcionalidades oferecidas pelo sistema. Eles basicamente irão especificar as características do sistema como um todo, por exemplo desempenho e disponibilidade. Dessa forma, os seguintes requisitos não funcionais deverão ser atendidos:

**RNF 0001 Nível de prioridade: Tenho que fazer****Nome:** Usabilidade e interface do sistema**Descrição:** O sistema deve conter uma interface intuitiva, facilitando o uso e entendimento do usuário.**RNF 0002 Nível de prioridade: Tenho que fazer****Nome:** Responsividade do sistema**Descrição:** O sistema deve funcionar como multiplataforma, em computadores, notebooks e celulares.**RNF 0003 Nível de prioridade: Tenho que fazer****Nome:** Segurança do sistema**Descrição:** O sistema deve conter métodos que protejam e evitem o acesso aos dados do usuário e empresa.**RNF 0004 Nível de prioridade: Tenho que fazer****Nome:** Confiabilidade do sistema**Descrição:** O sistema deve atender aos requisitos levantados e ser um sistema confiável.**RNF 0005 Nível de prioridade: Poderia fazer****Nome:** Framework do sistema**Descrição:** O sistema deve utilizar, preferencialmente, o Framework Django como forma de otimizar o desenvolvimento

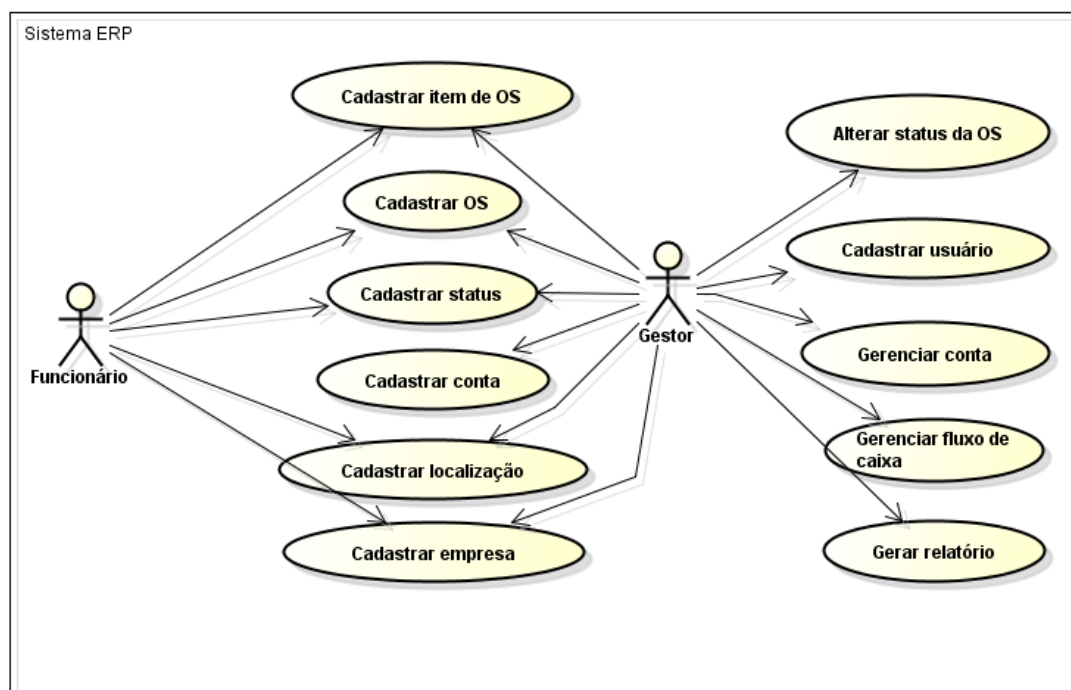
**RNF 0006 Nível de prioridade: Poderia fazer****Nome:** Banco de dados do sistema**Descrição:** O sistema deve utilizar, preferencialmente, o banco de dados PostgreSQL como Sistema de Gerenciamento de Banco de Dados(SGBD)**5.1.3 Diagramas**

Os diagramas UML servem para representar visualmente um sistema com seus principais atores, papéis, ações, artefatos ou classes, a fim de melhor compreender, alterar, manter ou documentar informações sobre o sistema (NOLETO, 2020). Neste projeto serão abordados o diagrama de caso de uso e diagrama de classe, os quais veremos a seguir.

**5.1.3.1 Diagrama de Caso de Uso**

Este tipo de diagrama é utilizado para realizar a demonstração das funcionalidades do sistema, junto a interação que cada ator terá com elas. A figura 8 representa os casos de uso identificados nesse estudo de caso e em seguida a descrição detalhada de cada um deles.

Figura 8 – Diagrama de Caso de Uso.



Fonte: Elaborada pelo autor, 2022.

**5.1.3.2 Diagrama de Classes**

Este tipo de diagrama é utilizado para realizar o mapeamento da estrutura do sistema, modelando a estrutura das classes, seus atributos, operações e relações. A Figura 9 demonstra as classes do sistema e suas interações com outras classes.

A classe Usuário tem o objetivo de armazenar os dados relativos aos usuários do sistema, mantendo seus dados comuns e atributos. A classe OS se relaciona com as classes, Empresa, ItensOS, Status e Conta. No relacionamento com a classe ItensOS, ela representa quais itens pertencem a qual OS, sendo que uma OS pode não conter itens. No relacionamento com a classe Conta, ela representa a qual OS a

Tabela 6 – Caso de uso UC001 - Cadastrar usuário.

Caso de uso:	UC001
Nome:	Cadastrar usuário
Ator(es):	Gestor
Descrição:	Permitir a inclusão, alteração ou exclusão de usuários.
Pré-condições:	Receber a identificação de tipo de operação e dados do usuário em caso de alteração ou exclusão.
Fluxo Básico:	<ol style="list-style-type: none"> <li>1. Manutenção do cadastro: <ol style="list-style-type: none"> <li>a. Em caso de inclusão, o sistema habilita a edição dos dados.</li> <li>b. Em caso de alteração, o sistema exibe os dados cadastrados e os habilita para edição.</li> <li>c. Em caso de exclusão, o sistema deleta o usuário, porém mantém um historico no log.</li> </ol> </li> <li>2. Em caso de inclusão ou alteração, o usuário informa os dados obrigatórios para cadastramento e demais campos caso possua.</li> <li>3. O usuário confirma a operação realizada.</li> <li>4. O sistema atualiza os dados cadastrais. <ol style="list-style-type: none"> <li>a. Em caso de inclusão, o sistema gera um número de identificação.</li> </ol> </li> </ol>
Exceções:	Não há.
Pós-condições:	O sistema será direcionado a listagem de usuários.
Requisitos de negócio:	<ol style="list-style-type: none"> <li>1. Todos os campos são obrigatórios.</li> <li>2. O sistema não poderá aceitar e-mails duplicados.</li> </ol>
Requisitos de interface:	Sem requisitos de interface.

Fonte: Elaborada pelo autor, 2022.

Tabela 7 – Caso de uso UC002 - Cadastrar empresa.

Caso de uso:	UC002
Nome:	Cadastrar empresa
Ator(es):	Gestor, Funcionário
Descrição:	Permitir a inclusão, alteração ou exclusão de empresas.
Pré-condições:	Receber a identificação de tipo de operação e dados da empresa em caso de alteração ou exclusão.
Fluxo Básico:	<ol style="list-style-type: none"> <li>1. Manutenção do cadastro: <ol style="list-style-type: none"> <li>a. Em caso de inclusão, o sistema habilita a edição dos dados.</li> <li>b. Em caso de alteração, o sistema exibe os dados cadastrados e os habilita para edição.</li> <li>c. Em caso de exclusão, o sistema deleta a empresa, porém mantém um historico no log.</li> </ol> </li> <li>2. Em caso de inclusão ou alteração, o usuário informa os dados obrigatórios para cadastramento e demais campos caso possua.</li> <li>3. O usuário confirma a operação realizada.</li> <li>4. O sistema atualiza os dados cadastrais. <ol style="list-style-type: none"> <li>a. Em caso de inclusão, o sistema gera um número de identificação.</li> </ol> </li> </ol>
Exceções:	Não há.
Pós-condições:	O sistema será direcionado a listagem de empresas.
Requisitos de negócio:	<ol style="list-style-type: none"> <li>1. Os campos cnpj, razão e telefone são obrigatórios.</li> <li>2. O sistema não poderá aceitar cnpj's duplicados.</li> </ol>
Requisitos de interface:	Sem requisitos de interface.

Fonte: Elaborada pelo autor, 2022.

Tabela 8 – Caso de uso UC003 - Cadastrar localização.

Caso de uso:	UC003
Nome:	Cadastrar localização
Ator(es):	Gestor, Funcionário
Descrição:	Permitir a inclusão, alteração ou exclusão de localização.
Pré-condições:	Receber a identificação de tipo de operação e dados da localização em caso de alteração ou exclusão.
Fluxo Básico:	<ol style="list-style-type: none"> <li>1. Manutenção do cadastro: <ol style="list-style-type: none"> <li>a. Em caso de inclusão, o sistema habilita a edição dos dados.</li> <li>b. Em caso de alteração, o sistema exibe os dados cadastrados e os habilita para edição.</li> <li>c. Em caso de exclusão, o sistema deleta a localização.</li> </ol> </li> <li>2. Em caso de inclusão ou alteração, o usuário informa os dados obrigatórios para cadastramento e demais campos caso possua.</li> <li>3. O usuário confirma a operação realizada.</li> <li>4. O sistema atualiza os dados cadastrais. <ol style="list-style-type: none"> <li>a. Em caso de inclusão, o sistema gera um número de identificação.</li> </ol> </li> </ol>
Exceções:	Não há.
Pós-condições:	O sistema será direcionado a listagem de localizações.
Requisitos de negócio:	1. Todos os campos são obrigatórios.
Requisitos de interface:	Sem requisitos de interface.

Fonte: Elaborada pelo autor, 2022.

Tabela 9 – Caso de uso UC004 - Cadastrar status.

Caso de uso:	UC004
Nome:	Cadastrar status
Ator(es):	Gestor
Descrição:	Permitir a inclusão, alteração ou exclusão de status.
Pré-condições:	Receber a identificação de tipo de operação e dados da status em caso de alteração ou exclusão.
Fluxo Básico:	<ol style="list-style-type: none"> <li>1. Manutenção do cadastro: <ol style="list-style-type: none"> <li>a. Em caso de inclusão, o sistema habilita a edição dos dados.</li> <li>b. Em caso de alteração, o sistema exibe os dados cadastrados e os habilita para edição.</li> <li>c. Em caso de exclusão, o sistema deleta o status.</li> </ol> </li> <li>2. Em caso de inclusão ou alteração, o usuário informa os dados obrigatórios para cadastramento e demais campos caso possua.</li> <li>3. O usuário confirma a operação realizada.</li> <li>4. O sistema atualiza os dados cadastrais. <ol style="list-style-type: none"> <li>a. Em caso de inclusão, o sistema gera um número de identificação.</li> </ol> </li> </ol>
Exceções:	Não há.
Pós-condições:	O sistema será direcionado a listagem de status.
Requisitos de negócio:	Não há.
Requisitos de interface:	Sem requisitos de interface.

Fonte: Elaborada pelo autor, 2022.

Tabela 10 – Caso de uso UC005 - Cadastrar OS.

Caso de uso:	UC005
Nome:	Cadastrar OS
Ator(es):	Gestor, Funcionário
Descrição:	Permitir a inclusão, alteração ou exclusão de ordens de serviço.
Pré-condições:	Receber a identificação de tipo de operação e dados da OS em caso de alteração.
Fluxo Básico:	<ol style="list-style-type: none"> <li>1. Manutenção do cadastro: <ol style="list-style-type: none"> <li>a. Em caso de inclusão, o sistema habilita a edição dos dados.</li> <li>b. Em caso de alteração, o sistema exibe os dados cadastrados e os habilita para edição.</li> </ol> </li> <li>2. Em caso de inclusão ou alteração, o usuário informa os dados obrigatórios para cadastramento e demais campos caso possua.</li> <li>3. O usuário confirma a operação realizada.</li> <li>4. O sistema atualiza os dados cadastrais. <ol style="list-style-type: none"> <li>a. Em caso de inclusão, o sistema gera um número de identificação.</li> </ol> </li> </ol>
Exceções:	Não há.
Pós-condições:	O sistema será direcionado a listagem de OS.
Requisitos de negócio:	Apenas os campos, numero de peças, remessa, referencia, valor e data de inicio poderão ser editados.
Requisitos de interface:	Trazer campo de seleção para empresas, calendário para os campos de data.

Fonte: Elaborada pelo autor, 2022.

Tabela 11 – Caso de uso UC006 - Cadastrar item da OS.

Caso de uso:	UC006
Nome:	Cadastrar item da OS
Ator(es):	Gestor, Funcionário
Descrição:	Permitir a inclusão, alteração ou exclusão de itens da OS.
Pré-condições:	Receber a identificação de tipo de operação e dados do item em caso de alteração.
Fluxo Básico:	<ol style="list-style-type: none"> <li>1. Manutenção do cadastro: <ol style="list-style-type: none"> <li>a. Em caso de inclusão, o sistema habilita a edição dos dados.</li> <li>b. Em caso de alteração, o sistema exibe os dados cadastrados e os habilita para edição.</li> <li>c. Em caso de exclusão, o sistema deleta o item.</li> </ol> </li> <li>2. Em caso de inclusão ou alteração, o usuário informa os dados obrigatórios para cadastramento e demais campos caso possua.</li> <li>3. O usuário confirma a operação realizada.</li> <li>4. O sistema atualiza os dados cadastrais. <ol style="list-style-type: none"> <li>a. Em caso de inclusão, o sistema gera um número de identificação.</li> </ol> </li> </ol>
Exceções:	Não há.
Pós-condições:	O sistema será direcionado a listagem de itens.
Requisitos de negócio:	Não há.
Requisitos de interface:	Não há.

Fonte: Elaborada pelo autor, 2022.



Tabela 12 – Caso de uso UC007 - Cadastrar conta.

Caso de uso:	UC007
Nome:	Cadastrar conta
Ator(es):	Gestor, Funcionário
Descrição:	Permitir a inclusão, alteração ou exclusão de contas.
Pré-condições:	Receber a identificação de tipo de operação e dados da conta em caso de alteração.
Fluxo Básico:	<ol style="list-style-type: none"> <li>1. Manutenção do cadastro: <ol style="list-style-type: none"> <li>a. Em caso de inclusão, o sistema habilita a edição dos dados.</li> <li>b. Em caso de alteração, o sistema exibe os dados cadastrados e os habilita para edição.</li> <li>c. Em caso de exclusão, o sistema deleta a conta. porém mantém um historico no log.</li> </ol> </li> <li>2. Em caso de inclusão ou alteração, o usuário informa os dados obrigatórios para cadastramento e demais campos caso possua.</li> <li>3. O usuário confirma a operação realizada.</li> <li>4. O sistema atualiza os dados cadastrais. <ol style="list-style-type: none"> <li>a. Em caso de inclusão, o sistema gera um número de identificação.</li> </ol> </li> </ol>
Exceções:	Não há.
Pós-condições:	O sistema será direcionado a listagem de contas.
Requisitos de negócio:	Não há.
Requisitos de interface:	Campo de seleção para o tipo de conta (receber/pagar).

Fonte: Elaborada pelo autor, 2022.

Tabela 13 – Caso de uso UC008 - Alterar status da OS.

Caso de uso:	UC008
Nome:	Alterar status da OS
Ator(es):	Gestor
Descrição:	Alterar Status da OS.
Pré-condições:	Não há.
Fluxo Básico:	<ol style="list-style-type: none"> <li>1. O gestor seleciona qual OS deseja mudar o status.</li> <li>2. Na tela de mudança o mesmo escolhe para qual status deseja alterar e o motivo.</li> <li>3. Clica em salvar.</li> </ol>
Exceções:	Não há.
Pós-condições:	O sistema será direcionado a listagem de OS.
Requisitos de negócio:	1. Não poderá ser alterado o status de OS finalizadas.
Requisitos de interface:	Campo de seleção para o status.

Fonte: Elaborada pelo autor, 2022.

Tabela 14 – Caso de uso UC009 - Gerar relatório.

Caso de uso:	UC009
Nome:	Gerar relatório
Ator(es):	Gestor
Descrição:	Permitir a geração de relatórios com informações gerenciais, referentes a parte financeira da empresa, além de processos realizados pela mesma.
Pré-condições:	Acesso a funcionalidade somente para usuário gestor.
Fluxo Básico:	1. O gestor seleciona a data em que deseja o relatório.  2. Ao clicar em gerar relatório, o sistema exibe o relatório solicitado.
Exceções:	1. Caso não tenha relatórios a emitir, o sistema exibe uma mensagem informativa.
Pós-condições:	Não há.
Requisitos de negócio:	Sem requisitos de negócio.
Requisitos de interface:	Sem requisitos de interface.

Fonte: Elaborada pelo autor, 2022.

Tabela 15 – Caso de uso UC010 - Gerenciar conta.

Caso de uso:	UC010
Nome:	Gerenciar conta
Ator(es):	Gestor
Descrição:	Painel de contas.
Pré-condições:	Não há.
Fluxo Básico:	1. O gestor seleciona o menu contas. 2. Sistema exibe painel com as contas a receber e a pagar que a empresa possui no mês.
Exceções:	Não há.
Pós-condições:	Não há.
Requisitos de negócio:	Conter filtros.
Requisitos de interface:	Conter gráficos informativos.

Fonte: Elaborada pelo autor, 2022.

Tabela 16 – Caso de uso UC011 - Gerenciar fluxo de caixa.

Caso de uso:	UC011
Nome:	Gerenciar fluxo de caixa
Ator(es):	Gestor
Descrição:	Painel de fluxo de caixa da empresa.
Pré-condições:	Não há.
Fluxo Básico:	1. O gestor seleciona o menu fluxo de caixa. 2. Sistema exibe painel com as informações financeiras da empresa, com valores recebidos e gastos e informações relevantes.
Exceções:	Não há.
Pós-condições:	Não há.
Requisitos de negócio:	Conter filtros.
Requisitos de interface:	Conter gráficos informativos.

Fonte: Elaborada pelo autor, 2022.

Tabela 17 – Caso de uso UC012 - Fazer login.

Caso de uso:	UC012
Nome:	Fazer login
Ator(es):	Gestor, Usuário
Descrição:	Realiza o login de usuários no sistema.
Pré-condições:	O usuário deve possuir cadastro no sistema.
Fluxo Básico:	<ol style="list-style-type: none"> <li>1. O usuário acessa o sistema e informa seu usuário e senha.</li> <li>2. O sistema verifica os dados informados pelo usuário.</li> <li>3. Caso os dados sejam válidos, direciona o usuário para próxima página.</li> <li>4. Caso os dados sejam inválidos, o sistema retorna uma mensagem de erro, direcionando para tela de login.</li> </ol>
Exceções:	Não há.
Pós-condições:	O sistema será direcionado a listagem de os.
Requisitos de negócio:	1. O sistema não deve permitir o acesso sem que as credenciais sejam validadas.
Requisitos de interface:	1. O campo senha deve ter máscara específica para não aparecer a senha do usuário.

Fonte: Elaborada pelo autor, 2022.

Tabela 18 – Caso de uso UC013 - Fazer logout.

Caso de uso:	UC013
Nome:	Fazer logout
Ator(es):	Gestor, Usuário
Descrição:	Realiza o logout de usuários no sistema.
Pré-condições:	O usuário deve estar logado no sistema.
Fluxo Básico:	<ol style="list-style-type: none"> <li>1. O usuário clica no botão logout.</li> <li>2. O sistema desloga o usuário do sistema e direciona para a homepage.</li> </ol>
Exceções:	Não há.
Pós-condições:	O sistema será direcionado a homepage.
Requisitos de negócio:	Não há.
Requisitos de interface:	1. O botão de logout só deve aparecer no menu caso o usuário esteja logado.

Fonte: Elaborada pelo autor, 2022.

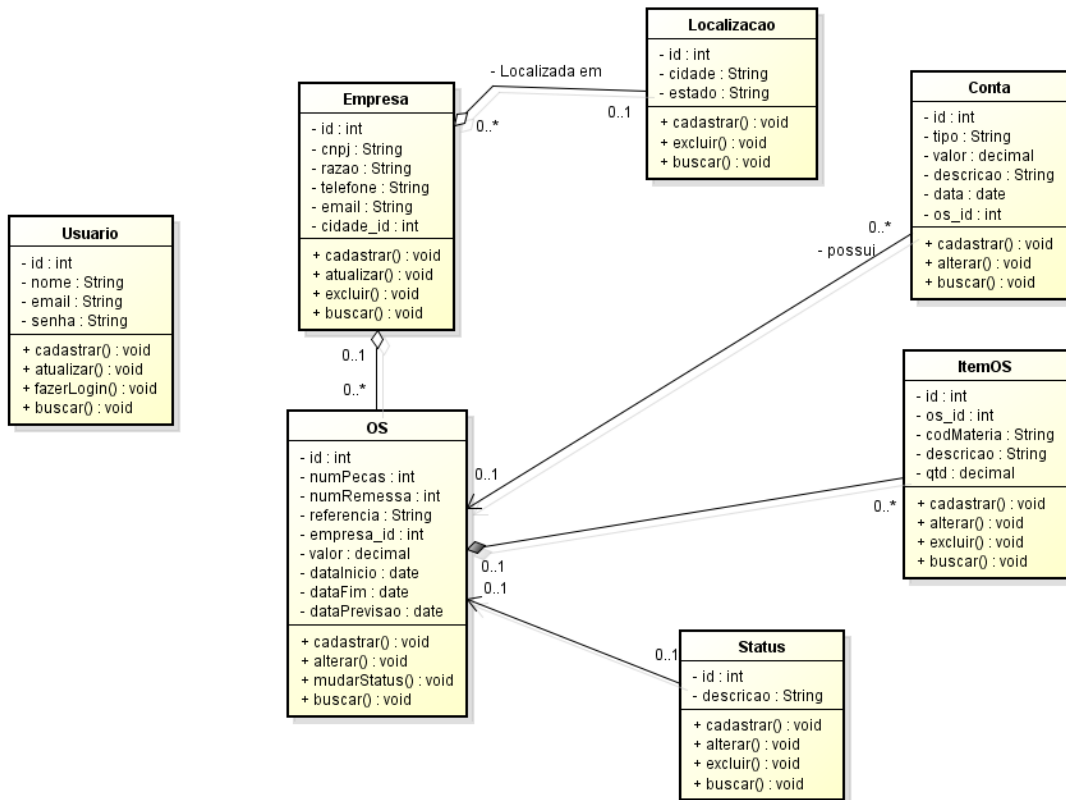
conta pertence, onde uma conta pode não possuir uma OS. No relacionamento com a classe Status, ela representa em qual status a OS se encontra. Já na classe Empresa, representa a qual empresa pertence a OS e a classe Localização, representa onde a empresa está localizada.

#### 5.1.4 Estimativas do projeto de *software*

Para a definição das estimativas deste projeto, foi utilizado a técnica *Program Evaluation and Review Technique* (PERT), muito utilizada na gestão de projetos, onde sua finalidade consiste em definir prazos mais assertivos em todas as etapas/atividades, se baseando basicamente em três estimativas possíveis para prever os prazos:

- Otimista (0): Neste cenário, tudo ocorreria de forma perfeita, sem atrasos e com menor tempo possível para conclusão de uma tarefa;

Figura 9 – Diagrama de classes.



Fonte: Elaborada pelo autor, 2022.

- Pessimista (P): Seria o pior cenário possível, onde o tempo para conclusão seria mais longo, considerando cenários desfavoráveis e riscos;
- Mais Provável (MP): O cenário mais realista, onde tudo ocorreria dentro da normalidade e satisfatória.

Com base nessas estimativas, será possível estimar o esforço, o tempo e o custo para a finalização do projeto. Para tal, além de definir as estimativas, é necessário realizar o cálculo do tempo estimado (tE), o qual será feito através da fórmula presente na Figura 19.

Figura 10 – Fórmula para encontrar o Tempo Estimado (tE).

$$PERT = (Pessimista + 4 \times Mais\ provável + Otimista) / 6$$

Fonte: Prata (2022)

#### 5.1.4.1 Esforço estimado

A determinação dos tempos otimistas, pessimistas e mais prováveis foram estimadas com base na necessidade atual da empresa junto a experiência de programação adquirida durante a faculdade e em projetos pessoais do programador.

A Tabela 19 ilustra todas as atividades presentes no projeto com seus respectivos tempos, incluindo o tempo estimado na última coluna, calculado conforme apresentado na Figura 19.

Tabela 19 – Duração de cada etapa do projeto.

Atividade	Duração Otimista (minutos)	Duração Provável (minutos)	Duração Pessimista (minutos)	Tempo Estimado (minutos)
Manter cadastro de usuários	240	310	360	307
Manter cadastro de empresas	180	235	300	237
Manter cadastro de localizações	150	185	210	183
Manter cadastro de Status	120	140	200	147
Manter cadastro de OS	480	590	720	594
Manter cadastro de item OS	270	360	480	365
Manter cadastro de contas	210	260	330	263
Gerar relatórios	235	285	330	284
Criar fluxo de caixa	300	430	540	427
Contas a receber/pagar	480	590	720	593
Design do sistema	660	790	900	787
* As atividades foram estimadas com base no Item 5.1.2				

Fonte: Elaborada pelo autor, 2022.

A partir disso, podemos determinar que o tempo do projeto é a soma dos tempos das atividades que devem ser executadas. Dessa forma, o tempo total estimado para a conclusão é de 4.187 minutos, aproximadamente 70 horas de desenvolvimento.

#### 5.1.4.2 Custo estimado

Considerando o esforço estimado e as tecnologias especificadas na análise de requisitos, deve ser levado em consideração que o *software* poderá ser desenvolvido por uma *Software House* (Empresas responsáveis por desenvolver e comercializar sistemas personalizados com base em uma demanda existente) que pode ser composta muitas vezes por vários profissionais de diferentes áreas e níveis.

Baseado nesta afirmação e considerando o custo médio de desenvolvimento a R\$ 250,00 (Duzentos e cinquenta reais) por hora, o custo total para desenvolvimento do *software* é de R\$ 17.425 (dezessete mil, quatrocentos e vinte e cinco reais) - valor estimado.

#### 5.1.5 Protótipo do *software*

##### 5.1.5.1 Metodologia do desenvolvimento

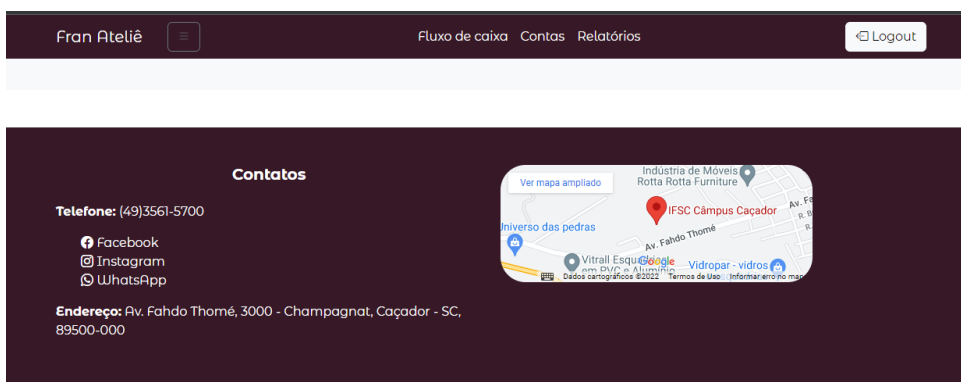
Para o início do projeto foi realizada uma breve entrevista com a gestora da empresa, para identificar os principais requisitos que o *software* deveria apresentar. Aliando o conhecimento em sistemas de informação e o conhecimento da gestora com a área de costura, foi possível definir os modelos que servirão de base para o desenvolvimento do *software*.

Para realização do desenvolvimento do sistema, foram levados em consideração os requisitos levantados conforme a Seção 5.1.2 e os modelos abstratos do sistema identificados na Seção 5.1.3.

Inicialmente foi definido um *template* para as páginas do sistema junto à gestora da empresa, onde todas as telas seguirão o mesmo, apresentando elementos em comum, o que irá facilitar a navegação dos usuários. Como pode ser visto na Figura 11, O *template* conta com um menu superior, onde ficam listadas as principais funcionalidades do sistema, Fluxo de Caixa, Contas e Relatórios, além de um botão para acesso às demais funcionalidades de cadastro, como usuários e empresas. Também conta com um

botão para fazer o logout, já a área em branco será destinada para o conteúdo específico de cada tela.

Figura 11 – Protótipo - Template do sistema.



Fonte: Elaborada pelo autor, 2022.

Após o usuário realizar o login no sistema, será direcionado para a tela de listagem de ordens de serviço, reproduzida na Figura 12. Nesta tela, é possível realizar o cadastro de ordens de serviço, visualizar detalhes e itens, além de acompanhar o status em que se encontram. Caso o perfil do usuário seja do grupo de Administradores, também terá a opção de alterar o status da ordem de serviço, sendo possível concluir ou cancelar a mesma, por exemplo. As ordens de serviço que forem alteradas para o status “concluída”, automaticamente irão gerar uma nova conta a receber. Durante o desenvolvimento, definiu-se que os status e localizações serão previamente cadastrados, dessa forma, não se faz necessário uma tela para cadastro de ambos.

Figura 12 – Protótipo - Dashboard OS.



Fonte: Elaborada pelo autor, 2022.

Para as telas de cadastro e edição, foi padronizado um template de formulário, onde todas as telas seguirão o mesmo como base, apenas diferenciando suas tratativas de negócio. A Figura 13 demonstra o formulário de cadastro de contas, exemplificando o layout de formulário utilizado pelas demais páginas.

A Figura 14 representa a tela de contas, nela ficarão listadas todas as contas que a empresa tem a receber ou recebidas e a pagar ou pagas. Usuários que não estão no grupo de administradores, poderão somente realizar o cadastro e busca de contas, já os administradores poderão além disso, editar, excluir ou

Figura 13 – Protótipo - Formulários.

**Cadastro de Contas**

Tipo\*  
Receber

Valor\*

Descrição

Data de previsão/pagamento  
dd/mm/aaaa

Paga

Cadastrar

Cancelar

Fonte: Elaborada pelo autor, 2022.

marcar as contas como pagas/recebidas. Dessa tela, irão sair os dados que vão alimentar a tela de fluxo de caixa e os relatórios gerados pelo sistema.

Figura 14 – Protótipo - Tela de contas.

**Lista de Contas**

Procure por descrição

Escolha uma opção...

Cadastrar Conta

#	Tipo	Valor	Descrição	Data/Previsão de Pagamento	Status	OS	Opções
25	Receber	946,56	Gerada via OS	5 de Novembro de 2022	A receber/Pagar	296616 - SLYWEA...	Editar Excluir
16	Receber	792,99	Gerada via OS	5 de Novembro de 2022	A receber/Pagar	304357 - SLYWEA...	Editar Excluir
17	Receber	792,99	Gerada via OS	5 de Novembro de 2022	A receber/Pagar	304357 - SLYWEA...	Editar Excluir
15	Receber	792,99	Gerada via OS	5 de Novembro de 2022	A receber/Pagar	304357 - SLYWEA...	Editar Excluir
14	Receber	790,56	Gerada via OS	2 de Novembro de 2022	A receber/Pagar	300198 - SLYWEA...	Editar Excluir

« 1 2 3 »

Fonte: Elaborada pelo autor, 2022.

A tela de fluxo de caixa, representada na Figura 15, possui um panorama geral do caixa da empresa, de acordo com o período filtrado, trazendo o total de receitas e despesas, além do lucro e % de lucratividade obtido, trazendo também a descrição das receitas e despesas consideradas no cálculo.

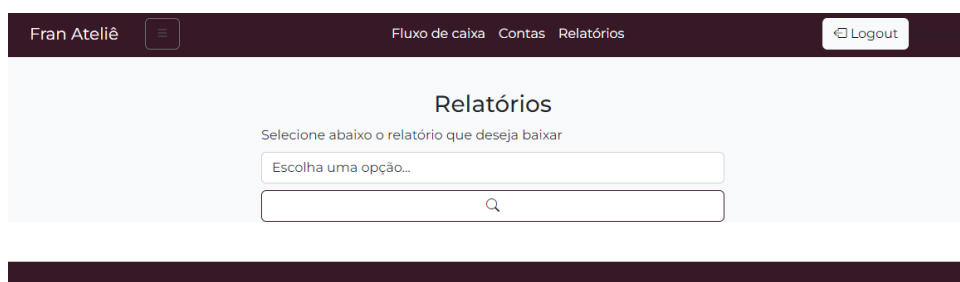
A Figura 16 ilustra a tela de relatórios, sendo que através dela é possível fazer o download de relatórios no formato csv. Dentre os relatórios disponíveis, estão os de contas a receber, pagar, recebidas e pagas. Uma das vantagens de dispor desse formato de relatórios no sistema, é a possibilidade de utilizá-los para realizar contas e extrair informações relevantes, além de ser possível utilizar o arquivo para importar em outras plataformas.

Figura 15 – Protótipo - Fluxo de caixa.



Fonte: Elaborada pelo autor, 2022.

Figura 16 – Protótipo - Tela de relatórios.



Fonte: Elaborada pelo autor, 2022.

### 5.1.5.2 Implantação do software

Para criação do *software*, utilizou-se o banco de dados PostgreSQL versão 14 e Django 4.1. O projeto segue o padrão Model-Template-View (MTV). Nesta seção, será apresentado uma breve visão das camadas model, template e view e como as mesmas foram desenvolvidas.

### 5.1.5.3 Model

Conforme análise de requisitos e o diagrama de classes visto na Figura 8, foi feita a criação dos modelos necessários para o desenvolvimento das funcionalidades do projeto, no Django ao realizar a modelagem das classes, conforme exemplo da Figura 17 e realizar as migrações, são criadas as tabelas no banco respectivamente conforme cada classe. Visando manter a organização do código e manutenções futuras, o projeto foi dividido em apps de Usuário, Empresas, Ordens de Serviço e o Módulo Financeiro. Dessa forma, cada app possui um arquivo chamado Models.py, responsável pela criação dos modelos do sistema.

O Django possui algumas funcionalidades interessantes, como a criação de arquivos de formulário. Com eles é possível criar formulários com base nas models, definindo as características e validações que o formulário deve ter, além de receber e processar as informações repassadas a ela, tudo isso criando um arquivo conforme exemplo da Figura 18, onde com base nas informações contidas na *model*, o Django automatiza várias linhas de desenvolvimento, otimizando o tempo perdido para criação de cada uma.

Dessa forma, todos os apps irão possuir um arquivo forms.py, com suas respectivas características



Figura 17 – Implantação - Camada Model do app OS.

```

models.py
ordemServico > models.py > ...
1  from django.db import models
2  from empresas.models import Empresa
3  from auditlog.registry import auditlog
4
5  class Status(models.Model):
6      descricao = models.CharField(max_length=30)
7
8      def __str__(self):
9          return self.descricao
10
11 class OrdemServico(models.Model):
12     numeroPecas = models.IntegerField()
13     numeroRemessa = models.IntegerField()
14     referencia = models.CharField(max_length=100)
15     valor = models.DecimalField(max_digits=7, decimal_places=2)
16     dataInicio = models.DateField()
17     dataFim = models.DateField(blank=True)
18     dataPrevisao = models.DateField(blank=True)
19     empresa = models.ForeignKey(Empresa, on_delete=models.CASCADE)
20     status = models.ForeignKey(Status, on_delete=models.CASCADE)
21
22     def __str__(self):
23         return f'{self.numeroRemessa} - {self.empresa}'
24
25     auditlog.register(OrdemServico)
26
27 class itensOS(models.Model):
28     codMateria = models.CharField(max_length=50)
29     descricao = models.CharField(max_length=100)
30     quantidade = models.FloatField()
31     ordemServico = models.ForeignKey(
32         OrdemServico, on_delete=models.CASCADE)
33

```

Fonte: Elaborada pelo autor, 2022.

Figura 18 – Forms Django.

```

ordemServico > forms.py > ...
1  from django import forms
2  from .models import OrdemServico, Status, statusHistoricoOS, itensOS
3
4  class DateInput(forms.DateInput):
5      input_type = 'date'
6
7
8  class OrdemServicoForm(forms.ModelForm):
9      class Meta:
10         model = OrdemServico
11         exclude = ['dataFim']
12
13         STATUS = (
14             (1, 'Não iniciada'),
15             (2, 'Em andamento'),
16             (3, 'Pausada'),
17             (4, 'Concluída'),
18             (5, 'Cancelada'),
19         )
20
21         widgets = {
22             'dataInicio': DateInput(),
23             'dataPrevisao': DateInput(),
24             'status': forms.Select(choices=STATUS),
25         }

```

Fonte: Elaborada pelo autor, 2022.

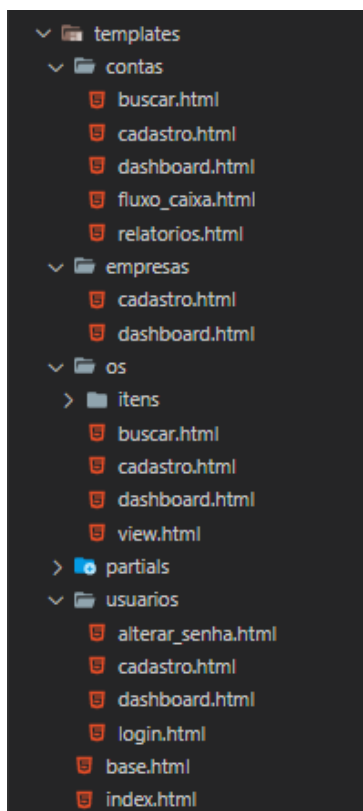
de criação.

#### 5.1.5.4 Template

A camada *template* no Django, é a responsável por apresentar a interface aos usuários, mostrando informações vindas da *view*. Para padronização da localização das páginas do projeto, foi criada uma pasta na raiz do mesmo chamada Templates, nela ficarão contidos todos os arquivos HTML do projeto.

Conforme visto na Figura 19, a pasta templates possui o arquivo index.html, responsável pela

Figura 19 – Implantação - Templates.



Fonte: Elaborada pelo autor, 2022.

home-page da empresa e o arquivo `base.html`, responsável pelo template padrão que as demais páginas devem utilizar, a pasta também conta com algumas subpastas, visando manter a organização e separando os arquivos respectivos a cada app, dessa forma deixando o código mais limpo e de fácil manutenção. A maioria das pastas vão possuir os arquivos referentes a cadastro e *dashboard*, além de algumas particularidades, como login e fluxo de caixa, por exemplo. Por fim a pasta `partials` armazena os arquivos que contêm partes de códigos reutilizáveis, dessa forma, evitando a duplicidade de códigos no sistema, neste projeto os arquivos utilizador foram `menu.html`, `footer.html` e `alertas.html`, além dos códigos de paginação e busca.

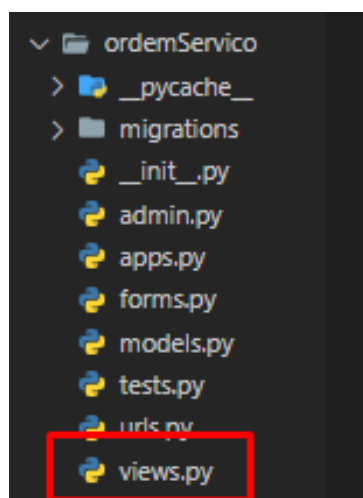
#### 5.1.5.5 View

A *view* no Django, é a camada responsável por fazer a ligação entre a model e o template, fazendo com que as informações recebidas nos templates sejam repassadas para as models e vice-versa. Assim como os models, as views ficaram divididas nos apps, conforme apresentado na Figura 20. Dessa forma, cada model apresentado anteriormente terá uma função específica na view.

A *view* basicamente, tem o papel da controller no padrão MVC, é nela que ficarão as funções que realizam as atividades do sistema. Nesse projeto, cada app possui suas respectivas funções no arquivo `views.py`. A Figura 21 exemplifica a função de cadastro e atualização de ordens de serviço, sendo possível visualizar o uso dos forms, models e a chamada dos arquivos da pasta `templates`.

Outra funcionalidade desenvolvida na *view* é a possibilidade de gerar relatórios no formato `.csv` (Arquivo de texto simples que armazena informações de planilhas e tabelas). Conforme ilustra a Figura 22, o sistema recebe a requisição de qual relatório o usuário deseja baixar e faz o levantamento dos registros e escrita no arquivo, disponibilizando o mesmo, permitindo que sejam criados diversos tipos de relatórios com informações úteis para tomada de decisão da gestora.

Figura 20 – Implantação - Views.



Fonte: Elaborada pelo autor, 2022.

Figura 21 – View OS.

```
ordemServico > views.py > cadastra_os
1  from django.shortcuts import get_object_or_404, render, redirect
2  from django.contrib import messages
3  from .forms import OrdemServicoForm, OrdemServicoEditForm, statusHistoricoOsForm, itensOsForm
4  from .models import OrdemServico, statusHistoricoOS, itensOS
5
6  # views OS
7  # Cadastro e atualização de OS
8  def cadastra_os(request, os_id=None):
9      if request.method == 'POST':
10         if os_id == None:
11             form = OrdemServicoForm(request.POST)
12         else:
13             os = OrdemServico.objects.get(pk=os_id)
14             form = OrdemServicoEditForm(request.POST, instance=os)
15         if form.is_valid():
16             form.save()
17         return redirect(['dashboard_os'])
18     else:
19         if os_id == None:
20             form = OrdemServicoForm()
21         else:
22             os = OrdemServico.objects.get(pk=os_id)
23             form = OrdemServicoEditForm(instance=os)
24     return render(request, 'os/cadastro.html', {'form': form})
25
26
```

Fonte: Elaborada pelo autor, 2022.

Visando facilitar o uso dos usuários, o sistema possui uma automatização nas ordens de serviço, onde ao uma ordem de serviço ter seu status alterada para concluída, o sistema automaticamente gera uma nova conta a receber, dessa forma, isentando do usuário a responsabilidade de cadastro em uma nova tela. Essa automatização é feita utilizando *Trigger Functions* disponíveis no Postgres, e seu esquema de criação é exemplificado conforme demonstra a Figura 23.

## 5.2 Testes

Após a conclusão do desenvolvimento do *software*, aplicou-se os testes para verificação da aceitação alcançada pelo sistema. Além disso, os testes também servem para identificação de possíveis alterações que devem ser feitas antes de uma possível versão final do produto.

Figura 22 – Implantação - Relatórios.

```

199
200 @login_required
201 def conta_csv(request):
202     tipoRelatorio = request.GET.get('rel')
203
204     if tipoRelatorio == 'c1':
205         response = HttpResponse(content_type='text/csv')
206         response['Content-Disposition'] = 'attachment; filename=contas.csv'
207
208         writer = csv.writer(response)
209
210         contas = Contas.objects.filter(tipo='Receber', paga=True)
211
212         writer.writerow(['tipo', 'descricao', 'valor', 'data', 'ordemServico'])
213
214         for conta in contas:
215             writer.writerow([conta.tipo, conta.descricao,
216                             conta.valor, conta.data, conta.ordemServico])
217
218         return response
219     elif tipoRelatorio == 'c2':

```

Fonte: Elaborada pelo autor, 2022.

Figura 23 – Automação - Módulo contas.

```

1 -- FUNCTION: public.fnc_grava_conta()
2
3 -- DROP FUNCTION IF EXISTS public.fnc_grava_conta();
4
5 CREATE OR REPLACE FUNCTION public.fnc_grava_conta()
6     RETURNS trigger
7     LANGUAGE 'plpgsql'
8     COST 100
9     VOLATILE NOT LEAKPROOF
10    AS $BODY$
11 DECLARE
12 BEGIN
13     IF NEW.status_id = 3 THEN
14         NEW."dataFim" = current_date;
15
16         INSERT INTO "moduloFinanceiro_contas"(tipo, valor, descricao, data, "ordemServico_id", paga)
17         VALUES ('Receber', NEW.valor, 'Gerada via OS', current_date+20, new.id, false);
18
19     END IF;
20
21     RETURN new;
22 END;
23 $BODY$;
24
25 ALTER FUNCTION public.fnc_grava_conta()
26     OWNER TO postgres;
27

```

Fonte: Elaborada pelo autor, 2022.

### 5.2.1 Teste de aceitação

Para realização do teste de aceitação, utilizou-se duas etapas. Durante a primeira etapa, foi utilizado o teste *alpha*, aplicado a um público fechado, tendo o objetivo de aperfeiçoar ainda mais o sistema, antes da disponibilização aos usuários. Já na segunda etapa, foi realizado o teste *beta*, que teve como público alvo um pequeno grupo de usuários que utilizarão o sistema em seu dia a dia. Esse é um dos últimos testes aplicados ao *software* para avaliar a aceitação do público.

Durante a fase de teste de aceitação, como destaca Leandro (2012), pode-se utilizar dos diagramas de caso de uso para geração de testes para o sistema. Junto a isso, a observação de interações e dificuldades do usuário com o sistema e a resposta do mesmo aos comportamentos do usuário devem ser observadas.

#### 5.2.1.1 Teste *alpha* e *beta*

Os dois testes foram aplicados manualmente dentro da empresa, por meio da aplicação *Railway*<sup>1</sup>, ferramenta que possibilita a implantação de um sistema para produção de maneira ágil e eficaz, simplificando

<sup>1</sup> Site Railway <<https://railway.app/>>.

o trabalho e reduzindo o tempo gasto, além de disponibilizar também o banco de dados Postgres. Dessa forma, o sistema foi configurado na plataforma e a alimentação das informações foi realizada pelos usuários.

Conforme visto na Tabela 20, o público alvo do teste *alpha* foi composto por 2 pessoas, já o público alvo do teste *beta* foi composto por 4 pessoas.

Tabela 20 – Público alvo teste alpha e beta.

Teste	Pessoa	Idade	Sexo	Ocupação	Facilidade com informática (escala 0 - 100)
Beta	Franciele	30	Feminino	CEO Ateliê	70
Alpha e Beta	Anthony	22	Masculino	Analista de Sistemas	90
Beta	Maikon	35	Masculino	Ajudante Ateliê	60
Alpha e Beta	Julio Cesar	22	Masculino	Analista de Sistemas	90

Fonte: Elaborada pelo autor.

Os cenários, tal qual, os casos de teste, foram elaborados utilizando como base os casos de uso apresentados no Capítulo 5. A seguir, as tabelas relacionam as especificações dos testes mapeados e seus respectivos resultados após a aplicação.

Tabela 21 – Caso de teste TC001 - Realizar cadastro de usuários.

Caso de teste	TC001	
Nome:	Realizar cadastro de usuários	
Descrição:	Tem como principal objetivo inserir novos usuários no sistema.	
Pré-condições:	O usuário que irá realizar o novo cadastro deve estar logado no sistema.	
Fluxo Básico:	Procedimento	Resultado esperado
	<ol style="list-style-type: none"> <li>1. Acessar o sistema;</li> <li>2. Abrir o menu e acessar a tela de usuários;</li> <li>3. Clicar em cadastrar novo;</li> <li>4. Informar os dados obrigatórios.</li> <li>5. Clicar em cadastrar.</li> </ol>	<ol style="list-style-type: none"> <li>1. Usuário incluído ao sistema.</li> </ol>
Resultado encontrado:	Os cadastros foram concluídos com sucesso.	
Resposta do usuário:	Não foram deixadas respostas para este caso.	

Fonte: Elaborada pelo autor.

Tabela 22 – Caso de teste TC002 - Realizar cadastro de empresas.

Caso de teste	TC002	
Nome:	Realizar cadastro de empresas	
Descrição:	Tem como principal objetivo inserir novas empresas no sistema.	
Pré-condições:	O usuário que irá realizar o novo cadastro deve estar logado no sistema.	
Fluxo Básico:	Procedimento	Resultado esperado
	<ol style="list-style-type: none"> <li>1. Acessar o sistema;</li> <li>2. Abrir o menu e acessar a tela de empresas;</li> <li>3. Clicar em cadastrar nova;</li> <li>4. Informar os dados obrigatórios;</li> <li>5. Clicar em cadastrar.</li> </ol>	<ol style="list-style-type: none"> <li>1. Empresa incluída no sistema.</li> </ol>
Resultado encontrado:	Os cadastros foram concluídos com sucesso.	
Resposta do usuário:	O sistema deve fornecer uma máscara para o campo telefone	

Fonte: Elaborada pelo autor.

Tabela 23 – Caso de teste TC003 - Realizar cadastro de localizações.

Caso de teste	TC003	
Nome:	Realizar cadastro de localizações	
Descrição:	Tem como principal objetivo inserir novas localizações no sistema.	
Pré-condições:	O usuário que irá realizar o novo cadastro deve estar logado no sistema.	
Fluxo Básico:	Procedimento	Resultado esperado
	<ol style="list-style-type: none"> <li>1. Acessar o sistema;</li> <li>2. Abrir o menu e acessar a tela de empresas;</li> <li>3. Clicar em cadastrar nova localização;</li> <li>4. Informar os dados obrigatórios;</li> <li>5. Clicar em cadastrar.</li> </ol>	<ol style="list-style-type: none"> <li>1. Localização incluída no sistema.</li> </ol>
Resultado encontrado:	Os cadastros foram concluídos com sucesso.	
Resposta do usuário:	O sistema deve fornecer os estados pré-cadastrados e não permitir a inclusão de cidades duplicadas.	

Fonte: Elaborada pelo autor.

Tabela 24 – Caso de teste TC004 - Realizar cadastro de OS.

Caso de teste	TC004	
Nome:	Realizar cadastro de OS	
Descrição:	Tem como principal objetivo inserir novas ordens de serviço no sistema.	
Pré-condições:	O usuário que irá realizar o novo cadastro deve estar logado no sistema.	
Fluxo Básico:	Procedimento	Resultado esperado
	<ol style="list-style-type: none"> <li>1. Acessar o sistema;</li> <li>2. Abrir o menu e acessar a tela de ordens de serviço;</li> <li>3. Clicar em cadastrar nova;</li> <li>4. Informar os dados obrigatórios;</li> <li>5. Clicar em cadastrar.</li> </ol>	<ol style="list-style-type: none"> <li>1. OS incluída no sistema.</li> </ol>
Resultado encontrado:	Os cadastros foram concluídos com sucesso.	
Resposta do usuário:	Não foram deixadas respostas para este teste.	

Fonte: Elaborada pelo autor.

### 5.2.2 Teste de regressão

O teste de regressão tem a pretensão de garantir que as alterações feitas em parte do sistema não afetem as partes já testadas. Conforme relata [Wazlawick \(2013\)](#), os testes de regressão sempre devem ser executados quando um sistema em operação sofre manutenção, visando garantir que nenhuma das atividades foi afetada com falhas devido a essas mudanças.

Com base nisso, após os testes de aceitação realizados na Seção 5.2.1, foram executados testes de regressão manualmente utilizando os mesmos parâmetros e princípios, com propósito de garantir as funcionalidades existentes no *software* desenvolvido e a correta manutenção de *bugs* apresentados. Para isso, a cada alteração feita no *software*, foi solicitado aos usuários que realizassem novamente os casos de teste anteriormente apresentados. Por fim, após a correção de falhas, verificou-se que todos os requisitos foram atendidos.

### 5.2.3 Teste de usabilidade

Os testes de usabilidade são feitos com o principal objetivo de avaliar a praticidade que um serviço, ferramenta, *software* oferece ao usuário que a está utilizando para realizar uma tarefa específica.

Tabela 25 – Caso de teste TC005 - Realizar cadastro de item OS.

Caso de teste	TC005	
Nome:	Realizar cadastro de item OS	
Descrição:	Tem como principal objetivo inserir novos itens as ordens de serviço do sistema.	
Pré-condições:	Necessário ter realizado o cadastro da OS anteriormente.	
Fluxo Básico:	Procedimento	Resultado esperado
	<ol style="list-style-type: none"> <li>1. Acessar o sistema;</li> <li>2. Abrir o menu e acessar a tela de ordens de serviço;</li> <li>3. Clicar em ver mais;</li> <li>4. Rolar a página e clicar em cadastrar novo item.</li> <li>5. Informar os dados obrigatórios;</li> <li>6. Clicar em cadastrar.</li> </ol>	<ol style="list-style-type: none"> <li>1. Item incluído na OS.</li> </ol>
Resultado encontrado:	Os cadastros foram concluídos com sucesso.	
Resposta do usuário:	Não foram deixadas respostas para este teste.	

Fonte: Elaborada pelo autor.

Tabela 26 – Caso de teste TC006 - Realizar cadastro de contas.

Caso de teste	TC006	
Nome:	Realizar cadastro de contas	
Descrição:	Tem como principal objetivo inserir novas contas no sistema	
Pré-condições:	O usuário que irá realizar o cadastro deve estar logado.	
Fluxo Básico:	Procedimento	Resultado esperado
	<ol style="list-style-type: none"> <li>1. Acessar o sistema;</li> <li>2. Abrir o menu e acessar a tela de contas;</li> <li>3. Clicar em cadastrar nova;</li> <li>4. Informar os dados obrigatórios;</li> <li>5. Clicar em cadastrar.</li> </ol>	<ol style="list-style-type: none"> <li>1. Conta incluída com sucesso.</li> </ol>
Resultado encontrado:	Os cadastros foram concluídos com sucesso.	
Resposta do usuário:	Não foram deixadas respostas para este teste.	

Fonte: Elaborada pelo autor.

Conforme relata [Teixeira \(2015\)](#), “a usabilidade é algo difícil de medir - ou pelo menos quantitativamente”. De toda forma, também reforça que os testes de usabilidade podem apresentar rapidamente as tarefas que as pessoas têm mais dificuldade. Apoiado a isso, aplicou-se um questionário aos usuários do sistema utilizando um dos métodos mais conhecidos de usabilidade de um sistema, o SUS e foi apurado o resultado em relação aos atributos de usabilidades e dos requisitos não funcionais especificados no projeto.

### 5.2.3.1 SUS

Conforme destaca [Teixeira \(2015\)](#), o método SUS foi criado em 1986 por John Brooke, podendo ser utilizado para avaliar a efetividade, eficiência e satisfação de produtos, serviços, softwares, entre outros. O SUS faz a utilização de um questionário que consiste em 10 questões sugeridas por John, as quais foram respondidas via formulário digital por 4 usuários. Logo após a aplicação, as respostas foram avaliadas aplicando as fórmulas do método e foram gerados os gráficos correspondentes conforme pode ser visto na sequência.

Antes da aplicação do SUS, é necessário entender seu método de apuração, que basicamente tem o seguinte funcionamento:

Tabela 27 – Caso de teste TC007 - Realizar alteração de status da OS.

Caso de teste	TC007	
Nome:	Realizar alteração de status da OS	
Descrição:	Tem como principal objetivo alterar o status de uma ordem de serviço no sistema	
Pré-condições:	O usuário deve possuir cadastro como Administrador.	
Fluxo Básico:	Procedimento	Resultado esperado
	<ol style="list-style-type: none"> <li>1. Acessar o sistema;</li> <li>2. Abrir o menu e acessar a tela de ordens de serviço;</li> <li>3. Clicar em ver mais;</li> <li>4. Clicar em alterar status os;</li> <li>5. Informar os dados obrigatórios;</li> <li>6. Clicar em cadastrar.</li> </ol>	1.Status alterado com sucesso.
Resultado encontrado:	Os Alterações foram concluídas com sucesso.	
Resposta do usuário:	Não foram deixadas respostas para este teste.	

Fonte: Elaborada pelo autor.

Tabela 28 – Caso de teste TC008 - Gerar relatórios.

Caso de teste	TC008	
Nome:	Gerar relatórios	
Descrição:	Tem como principal objetivo gerar relatórios via sistema.	
Pré-condições:	O usuário deve possuir cadastro como Administrador.	
Fluxo Básico:	Procedimento	Resultado esperado
	<ol style="list-style-type: none"> <li>1. Acessar o sistema;</li> <li>2. Abrir o menu e acessar a tela de relatórios;</li> <li>3. Selecionar o relatório que deseja baixar;</li> <li>4. Clicar em baixar.</li> </ol>	1.Relatório gerado com sucesso.
Resultado encontrado:	Os relatórios foram gerados com sucesso.	
Resposta do usuário:	O sistema deve permitir filtros de data para limitação de registros.	

Fonte: Elaborada pelo autor.

Tabela 29 – Caso de teste TC009 - Realizar pesquisas de contas a receber/pagar.

Caso de teste	TC009	
Nome:	Realizar pesquisas de contas a receber/pagar	
Descrição:	Tem como principal objetivo realizar a busca por contas a pagar/receber no sistema	
Pré-condições:	O usuário deve estar logado no sistema.	
Fluxo Básico:	Procedimento	Resultado esperado
	<ol style="list-style-type: none"> <li>1. Acessar o sistema;</li> <li>2. Abrir o menu e acessar a tela de contas;</li> <li>3. Utilizar os filtros para pesquisa desejada;</li> <li>4. Clicar em buscar.</li> </ol>	1.Retornar dados da pesquisa solicitada com sucesso.
Resultado encontrado:	Os dados das pesquisas foram retornados com sucesso.	
Resposta do usuário:	O sistema deve permitir mais filtros como data e status de pagamento.	

Fonte: Elaborada pelo autor.



Tabela 30 – Caso de teste TC010 - Realizar acesso ao Fluxo de caixa.

Caso de teste	TC010	
Nome:	Realizar acesso ao Fluxo de caixa	
Descrição:	Tem como principal objetivo realizar o acesso ao fluxo de caixa da empresa.	
Pré-condições:	O usuário deve possuir cadastro como Administrador.	
Fluxo Básico:	Procedimento	Resultado esperado
	1. Acessar o sistema; 2. Abrir o menu e acessar a tela de Fluxo de caixa; 3. Filtrar a data desejada; 4. Clicar em buscar.	1.Retornar dados financeiros da empresa no período.
Resultado encontrado:	Os dados das pesquisas foram retornados com sucesso.	
Resposta do usuário:	O sistema deve conter gráficos informativos.	

Fonte: Elaborada pelo autor.

Tabela 31 – Caso de teste TC011 - Realizar login com usuário e senha inválidos.

Caso de teste	TC011	
Nome:	Realizar login com usuário e senha inválidos	
Descrição:	Tem como principal objetivo tentar acessar o sistema com um usuário e uma senha inválida, com intuito de realizar o Teste de Segurança.	
Pré-condições:	O usuário não pode estar logado no sistema.	
Fluxo Básico:	Procedimento	Resultado esperado
	1. Acessar o website do sistema. 2. Inserir um usuário e senha inválidos; 3. Clicar em acessar o sistema	1. Retornar uma mensagem de erro e não permitir o acesso.
Resultado encontrado:	Após a aplicação, o usuário teve o acesso negado.	
Resposta do usuário:	Não foram deixadas respostas para este caso.	

Fonte: Elaborada pelo autor.

Tabela 32 – Caso de teste TC012 - Realizar login com usuário e senha válidos.

Caso de teste	TC012	
Nome:	Realizar login com usuário e senha válidos.	
Descrição:	Tem como principal objetivo tentar acessar o sistema com um usuário e uma senha válida, com intuito de realizar o Teste de Segurança.	
Pré-condições:	O usuário não deve estar logado.	
Fluxo Básico:	Procedimento	Resultado esperado
	1. Acessar o sistema; 2. Inserir um usuário e senha válidos; 2. Clicar no botão acessar.	1.Login realizado com sucesso.
Resultado encontrado:	Após a aplicação, o usuário foi logado no sistema.	
Resposta do usuário:	Não foram deixadas respostas para este caso.	

Fonte: Elaborada pelo autor.

1. As perguntas ímpares, referem-se a perguntas com aspectos positivos, dessa forma, quanto maior for a quantidade de respostas com índice 4 e 5, melhor será a pontuação obtida.
2. As perguntas pares, referem-se a perguntas com aspectos negativos, sendo assim, a pontuação obtida será maior caso as respostas com índice 1 e 2 sejam maiores.
3. Para consideração do resultado final, utilizou-se uma escala de 0 a 100, sendo que quanto mais

Tabela 33 – Caso de teste TC013 - Realizar logout do sistema.

Caso de teste	TC013	
Nome:	Realizar logout do sistema	
Descrição:	Tem como principal objetivo realizar o logout do usuário no sistema	
Pré-condições:	O usuário deve estar logado.	
Fluxo Básico:	Procedimento	Resultado esperado
	1. Acessar o sistema; 2. Clicar no botão logout.	1. Logout realizado com sucesso.
Resultado encontrado:	Após a aplicação, o usuário foi deslogado do sistema.	
Resposta do usuário:	Não foram deixadas respostas para este caso.	

Fonte: Elaborada pelo autor.

próximo de 100, melhor o resultado.

As 10 perguntas abaixo, fazem parte do questionário proposto pelo método SUS e antes de serem aplicadas, foram adaptadas para se encaixarem melhor no contexto do *software* e da organização:

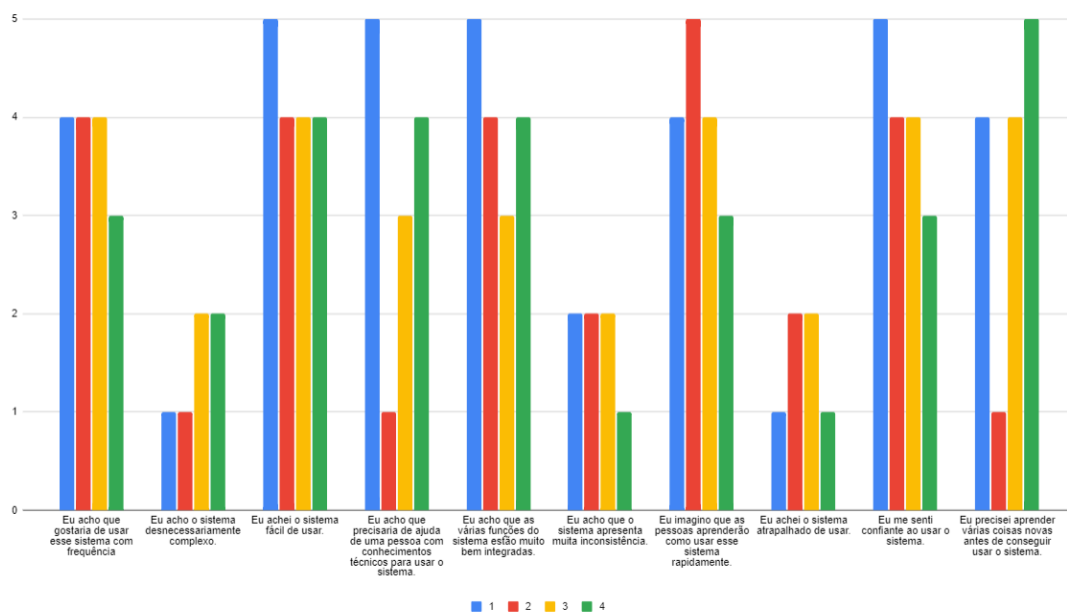
1. Eu acho que gostaria de usar esse sistema com frequência.
2. Eu acho o sistema desnecessariamente complexo.
3. Eu achei o sistema fácil de usar.
4. Eu acho que precisaria de ajuda de uma pessoa com conhecimentos técnicos para usar o sistema.
5. Eu acho que as várias funções do sistema estão muito bem integradas.
6. Eu acho que o sistema apresenta muita inconsistência.
7. Eu imagino que as pessoas aprenderão como usar esse sistema rapidamente.
8. Eu achei o sistema atrapalhado de usar.
9. Eu me senti confiante ao usar o sistema.
10. Eu precisei aprender várias coisas novas antes de conseguir usar o sistema.

A Figura 24 mostra as respostas do questionário aplicado aos usuários. Nele, é possível identificar que nas perguntas ímpares não houve ocorrência de respostas com índice 1 (discordo totalmente) . Já nas perguntas pares, apenas 2 ocorrências de respostas com índice 5 (concordo totalmente). Dessa forma, podemos considerar que ambos os casos tiveram um bom desempenho, considerando os métodos de apuração elaborados por Brooke.

Para realização da soma de pontuação dos testes, foram utilizadas as regras apontadas por Brooke no método SUS, conforme visto em [Teixeira \(2015\)](#), entre as regras estão:

- Para as respostas das questões ímpares, subtraímos 1 da pontuação que o usuário respondeu;
- Para as respostas das questões pares, a resposta é subtraída de 5, dessa forma, se o usuário respondeu 2, contabilizamos 3;
- Por fim, são somados todos os valores das perguntas e multiplicado por 2.5, obtendo o resultado final.

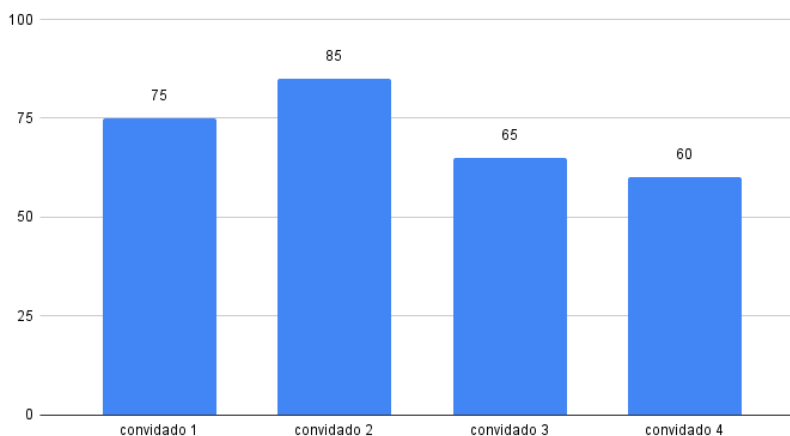
Figura 24 – Análise SUS - Quantidade de resposta por pergunta.



Fonte: Elaborada pelo autor, 2022.

A Figura 25 mostra o índice SUS calculado para o *software* desenvolvido neste projeto, no qual o maior índice apresentado foi 85 e o menor índice foi de 60, considerando a escala de 0 a 100, resultando em uma média de 71,25 pontos no geral.

Figura 25 – Análise SUS - Escala SUS de cada resposta.



Fonte: Elaborada pelo autor, 2022.

Sendo assim, levando em consideração que a média geral do SUS é de aproximadamente 68 pontos, chega-se à conclusão de que o nível de usabilidade apresentado pelo protótipo é boa, demonstrando que ele cumpriu praticamente com todos os requisitos, não afetando de maneira significativa o desempenho apresentado pelo usuário ao utilizar o *software*.

## 6 CONCLUSÕES

Este trabalho teve como principal objetivo realizar o desenvolvimento de um *software* que servisse como ferramenta para o gerenciamento financeiro de uma empresa no ramo de costura. Com base nas pesquisas, informações e requisitos levantados, foi possível dar continuidade ao desenvolvimento do projeto e realizar os testes junto a empresa de forma satisfatória, conseguindo validar a eficiência e usabilidade do projeto, através dos métodos dispostos no Capítulo 4.

Em relação aos objetivos específicos deste projeto, chegou-se a conclusão que os mesmos foram alcançados com sucesso e que o ERP é o sistema mais adequado para o gerenciamento financeiro dentro da organização, onde foi realizado o mapeamento dos processos junto a empresa, identificado o tipo de sistema de informação utilizado para solução dos problemas, feito o levantamento de requisitos conforme as necessidades da organização, modelado e desenvolvido o software, além de aplicados testes de aceitação e usabilidade. Ainda que o mesmo não tenha sido completamente concluído, pretende-se adicionar mais recursos ao mesmo, envolvendo outros processos da empresa, além de aperfeiçoar os já existentes.

Durante o período de desenvolvimento e testes do projeto, foram identificadas algumas possíveis melhorias que devem ser realizadas. Assim sendo, em função da indisponibilidade de tempo para a conclusão desta implantação, recomenda-se para trabalhos futuros a inclusão de uma visão da tela de fluxo de caixa, onde seja possível visualizar um comparativo semanal/mensal de receitas e despesas previstas x concretizadas, gerando a possibilidade uma rápida análise de atingimento de metas, também, a inclusão de um gráfico de acompanhamento das atividades, na tela de listagem de OS, onde será possível acompanhar de maneira mais ágil, como se encontra o status de andamento de cada ordem de serviço.

Por fim, sugere-se também aperfeiçoar os processos existentes, considerando as condições atuais da organização e as possíveis demandas que surgirão. Além disso, sugere-se que sempre após a criação ou alteração de funcionalidades do sistema que alterem as anteriormente criadas, sejam executados os testes de regressão.

## REFERÊNCIAS

- BATISTA, E. d. O. *Sistemas de informação*. [S.l.]: Saraiva Educação SA, 2013. Citado 4 vezes nas páginas 13, 16, 17 e 18.
- CAVICHIOLO, A. et al. "Os sistemas de informação como fator competitivo: Um estudo de caso em uma prestadora de serviços de costura". 2016. Disponível em: <<http://www.revistaespacios.com/a16v37n27/16372708.html>>. Acesso em: 17 de maio 2022. Citado na página 27.
- DOUGLAS. "O que é um diagrama de classe UML?". 2018. Disponível em: <<https://www.lucidchart.com/pages/pt/o-que-e-diagrama-de-classe-uml>>. Acesso em: 22 de abril 2022. Citado na página 23.
- FIGUEIRA, I. F. R.; ORIKASA, L. Y. S. I. "Implantação de software de eficiência produtiva em microempresa de confecção". 2021. Disponível em: <<https://www.revistasuninter.com/revistaorganizaosistemica/index.php/organizacaoSistemica/article/view/493>>. Acesso em: 17 de maio 2022. Citado na página 27.
- FOUNDATION, D. S. "Django". 2013. Disponível em: <<https://www.djangoproject.com/>>. Acesso em: 26 de agosto 2022. Citado na página 24.
- LAUDON, K. C.; LAUDON, J. P. *Sistemas de Informação Gerenciais*. [S.l.]: Pearson Education do Brasil Ltda, 2015. Citado 2 vezes nas páginas 15 e 16.
- LEANDRO. "O que é UML e Diagramas de Caso de Uso: Introdução Prática à UML". 2012. Disponível em: <<https://www.devmedia.com.br/o-que-e-uml-e-diagramas-de-caso-de-uso-introducao-pratica-a-uml/23408>>. Acesso em: 22 de abril 2022. Citado 2 vezes nas páginas 23 e 50.
- MARQUEZ, G. "O que é ERP financeiro? 10 vantagens do planejamento de recursos empresariais para uma gestão eficaz". 2019. Disponível em: <<https://nfe.io/blog/financeiro/o-que-e-erp-financeiro/>>. Citado na página 21.
- MEIRELES, M. *SISTEMAS DE INFORMAÇÃO*. [S.l.]: Arte e Ciência, 2001. Citado na página 19.
- MONITORA, E. "Entenda como a tecnologia nas empresas é fundamental para alcançar sucesso". 2021. Disponível em: <<https://www.monitoretec.com.br/blog/tecnologia-nas-empresas/>>. Acesso em: 17 de abril 2022. Citado na página 13.
- NOLETO, C. "UML: o que é, para que serve e quando usar essa linguagem de notação?". 2020. Disponível em: <<https://blog.betrybe.com/tecnologia/uml/>>. Acesso em: 22 de abril 2022. Citado 2 vezes nas páginas 22 e 35.
- PINHEIRO, F. "Entendendo o MTV do Django". 2019. Disponível em: <<https://www.treinaweb.com.br/blog/entendendo-o-mtv-do-django>>. Citado na página 24.
- PIRES, R. "Aprenda a usar a técnica MoSCoW nos projetos da sua agência!". 2019. Disponível em: <<https://rockcontent.com/br/blog/metodo-moscow/>>. Citado na página 32.
- PRADO, E. P. V.; SOUZA, C. A. de. *Fundamentos de Sistemas de Informação*. [S.l.]: Elsevier Editora Ltda, 2014. Citado 5 vezes nas páginas 16, 18, 19, 20 e 21.
- PRATA, G. "PERT: saiba o que é o Program Evaluation and Review Technique". 2022. Disponível em: <<https://www.sience.com.br/blog/pert-program-evaluation-and-review-technique/>>. Citado na página 42.
- PRATES, G. A.; OSPINA, M. T. "Tecnologia da informação em pequenas empresas: fatores de êxito, restrições e benefícios.". 2009. Disponível em: <<https://doi.org/10.1590/S1415-65552004000200002>>. Acesso em: 12 de maio 2022. Citado na página 15.
- SILVA, F. A. R.; TARRENTO, G. E. "APLICAÇÃO DE FERRAMENTAS DE CONTROLE DE QUALIDADE EM UMA EMPRESA DE CONFECÇÃO DO INTERIOR DO ESTADO DE SÃO PAULO". 2020. Disponível em: <<http://revista.fatecbt.edu.br/index.php/tl/article/view/702/428>>. Acesso em: 17 de maio 2022. Citado na página 27.

- SOMMERVILLE, I. *engenharia de SOFTWARE*. [S.l.]: Pearson Education do Brasil, 2011. Citado 5 vezes nas páginas 22, 23, 25, 32 e 34.
- STAIR, R. M.; REYNOLDS, G. W. *Princípios de Sistemas de Informação*. [S.l.]: Cengage Learning, 2016. Citado 5 vezes nas páginas 13, 16, 18, 19 e 25.
- TEIXEIRA, F. "O que é o SUS (System Usability Scale) e como usá-lo em seu site". 2015. Disponível em: <<https://brasil.uxdesign.cc/o-que-%C3%A9-o-sus-system-usability-scale-e-como-us%C3%A1-lo-em-seu-site-6d63224481c8>>. Citado 2 vezes nas páginas 53 e 56.
- TOTVS, E. "ERP Financeiro: o que é, como funciona e por que implementar". 2021. Disponível em: <<https://www.totvs.com/blog/negocios/erp-financeiro/>>. Citado 2 vezes nas páginas 21 e 22.
- WAZLAWICK, R. S. *Engenharia de software: conceitos e práticas*. [S.l.]: Elsevier, 2013. Citado na página 52.

# APÊNDICE A – PERGUNTAS DA ENTREVISTA INICIAL

## Perguntas da entrevista inicial

Como é feito o armazenamento das informações da empresa?

Sua resposta \_\_\_\_\_

Como é realizado o gerenciamento financeiro na organização?

Sua resposta \_\_\_\_\_

Quais as principais funções que o sistema deve possuir para auxiliar o gerenciamento financeiro da empresa?

Sua resposta \_\_\_\_\_

Quais seriam as principais formas de acesso ao sistema?

Sua resposta \_\_\_\_\_

# APÊNDICE B – SUS – QUESTIONÁRIO

## SUS – Questionário

Eu acho que gostaria de usar esse sistema com frequência *						
1	2	3	4	5		
Discordo Completamente	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Concordo Completamente

Eu acho o sistema desnecessariamente complexo. *						
1	2	3	4	5		
Discordo Completamente	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Concordo Completamente

Eu achei o sistema fácil de usar. *						
1	2	3	4	5		
Discordo Completamente	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Concordo Completamente

Eu acho que precisaria de ajuda de uma pessoa com conhecimentos técnicos para usar o sistema. *						
1	2	3	4	5		
Discordo Completamente	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Concordo Completamente

Eu acho que as várias funções do sistema estão muito bem integradas. *						
1	2	3	4	5		
Discordo Completamente	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Concordo Completamente

Eu acho que o sistema apresenta muita inconsistência. *						
1	2	3	4	5		
Discordo Completamente	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Concordo Completamente



## continuação apêndice B

Eu imagino que as pessoas aprenderão como usar esse sistema rapidamente. \*

1 2 3 4 5

Discordo Completamente      Concordo Completamente

Eu achei o sistema atrapalhado de usar. \*

1 2 3 4 5

Discordo Completamente      Concordo Completamente

Eu me senti confiante ao usar o sistema. \*

1 2 3 4 5

Discordo Completamente      Concordo Completamente

Eu precisei aprender várias coisas novas antes de conseguir usar o sistema. \*

1 2 3 4 5

Discordo Completamente      Concordo Completamente