

Desenvolvimento de um *software* para o estudo de Estatística Descritiva

Adriel G. Abranjo¹, Douglas A. Vieira¹, Vilma G. Karsburg¹, José D. P. da Silva¹

¹ Instituto Federal de Santa Catarina – Campus Lages (IFSC)
Rua Heitor Villa Lobos, 222 – 88.506-400 – Lages – SC – Brasil

adrieltgoulartdeabranjo@gmail.com, douglasalissonvieira@gmail.com

vilma.karsburg@ifsc.edu.br, jose.silva@ifsc.edu.br

Abstract. *The low rates of success in exams such as ENEM (National High School Exam), in mathematical content, show the students difficulty in mastering this area. Then is necessary to use new approaches, as allied to the teaching and learning of mathematics. This article presents the creation of a software that aims to assist to learn and to teach statistics, which was integrated to IFMath as one of its modules. The developed system allows interaction with graphs, tables and step-by-step explanations of how to carry out its construction. A survey carried out with teachers and students showed that 60% or more of the participants considered the system built to be good or excellent.*

Resumo. *Os baixos índices de aproveitamento em exames como o ENEM (Exame Nacional de Ensino Médio), em conteúdos matemáticos evidenciam a dificuldade dos alunos perante o domínio desta área. Assim, se faz necessário utilizar novas abordagens, como aliadas ao ensino e aprendizagem de matemática. Este artigo apresenta a criação de um software que visa auxiliar no estudo e ensino de estatística descritiva, que foi integrado ao IFMath como um de seus módulos. O sistema desenvolvido permite a interação com gráficos, tabelas e explicações passo a passo de como realizar a sua construção. Uma pesquisa realizada com professores e alunos mostrou que 60% ou mais dos participantes consideraram bom ou excelente o sistema construído.*

1. Introdução

A estatística possui um índice de aproveitamento relativamente baixo no quesito aprendizagem por parte dos alunos, o que pode demonstrar uma falta de interesse dos mesmos aliada a uma carência de ferramentas que auxiliem no seu ensino. Verificando tal dificuldade, novas abordagens de ensino pelos professores se fazem necessárias com o intuito de estimular e otimizar o rendimento dos alunos.

Os conteúdos de estatística descritiva conforme documento oficial do Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (INEP) estão presentes no eixo Matemática e suas Tecnologias do Exame Nacional do Ensino Médio (ENEM). No ano de 2018 a nota média nesse eixo, foi de 535,5 pontos, porém neste mesmo ano 43,5% dos estudantes não atingiram 500 pontos, o que corresponde a 50% da nota máxima do eixo INEP (2018), e isso demonstra a necessidade de atenção com as competências que precisam ser desenvolvidas em relação a essa área.

Conforme Borba et al. (2018), em meados de 2004 ocorreu o advento da Internet rápida, com isso o número de dados que são produzidos e disseminados se elevaram consideravelmente. Como consequência disso, o número de pesquisas voltadas à educação estatística no Brasil vem crescendo significativamente.

Ignácio (2010), afirma que o grande volume de informações produzido pelo mundo moderno precisa ser analisado adequadamente e esse suporte ocorre por meio da estatística a todas as áreas do conhecimento humano. Tendo em vista que o uso de tecnologias pode ser uma grande aliada à análise de dados, neste trabalho é proposto o desenvolvimento de um *software* para o estudo de estatística descritiva.

Diante do problema com aprendizado e a importância de *softwares* para auxiliar no ensino, mencionados acima, o objetivo geral deste trabalho se concentra na criação de um *software* que realiza cálculos de estatística descritiva, que apresente uma breve explicação sobre o conteúdo, bem como as operações que são realizadas e a opção de visualizar gráficos e tabelas geradas a partir de tais operações.

Como objetivos específicos, tem-se:

- Estudar *softwares* e artigos sobre o ensino de estatística descritiva;
- Levantar tópicos trabalhados em estatística descritiva e estabelecer os que estarão presentes no *software*;
- Definir as funcionalidades e o *layout* do módulo do *software* desenvolvido;
- Implementar o *software* desenvolvido e integrar como um módulo do *IFMath*;
- Validar as construções realizadas no *software* com professores e alunos da área.

Para atingir os objetivos deste trabalho, o processo foi organizado em três etapas. A primeira corresponde à seleção dos tópicos trabalhados em estatística descritiva junto com todo o seu referencial teórico. Nessa etapa foram realizadas leituras à respeito dos tópicos utilizando livros, *softwares*, artigos e *sites*. A segunda etapa se deu pela escrita do documento de requisitos e a aplicação do conhecimento obtido na primeira etapa no desenvolvimento da aplicação proposta. A terceira etapa consistiu em realizar testes no *software* construído com os usuários da área de ensino de estatística.

Zanella (2013), em seu livro classifica as pesquisas quanto à metodologia de pesquisa, à abordagem, aos objetivos e aos procedimentos adotados. Nesse sentido, em relação a metodologia esta pesquisa é classificada como pesquisa científica aplicada, pois tem-se como finalidade gerar soluções direcionadas ao baixo aproveitamento dos discentes e entender como abordar esse problema, neste caso propõe-se a criação de um *software* voltado à solucionar esta adversidade quanto ao aprendizado da estatística descritiva.

Quanto à abordagem a pesquisa é qualitativa, pois, conforme Dalfovo et al. (2008) tem foco na interpretação e não na quantificação, já que neste caso o problema em si requer a incorporação do elemento humano, no sentido de como o *software* pode auxiliar no ensino e aprendizagem de estatística descritiva, além de que tal abordagem vai auxiliar nas etapas finais do projeto quando este for avaliado por docentes e discentes da área, visando obter uma perspectiva mais ampla em relação às deficiências do *software*.

Em relação aos seus objetivos se classifica como exploratória, já que se pretende, por meio deste trabalho, ampliar o conhecimento retido através das metodologias de ensino habituais, para que seja possível compreender e aplicar de forma coesa este aprendizado, sugerindo aos docentes novas práticas de ensino utilizando o *software* desenvolvido.

E quanto aos procedimentos adotados para a coleta de dados é experimental, pois a pesquisa envolve ao final, um experimento em um ambiente destinado à avaliação e coleta de dados pertinentes a qualidade do *software* com profissionais da área e discentes.

O restante deste artigo está organizado da seguinte maneira. Na Seção 2 serão apresentados os conceitos básicos para entendimento deste trabalho, escritos através de bases bibliográficas, artigos referentes ao ensino de estatística descritiva, *softwares* educacionais e trabalhos com ideias semelhantes à esse. Na Seção 3 é descrito o desenvolvimento do trabalho; na Seção 4 são apresentados os resultados e discussões. E na Seção 5 são apresentadas as considerações finais e os trabalhos futuros.

2. Fundamentação Teórica

Esta seção compõe-se do embasamento teórico, o qual fornece os conceitos técnicos que permitem a assimilação da área de aplicação desejada deste trabalho. A estrutura foi organizada em cinco subseções, onde a Subseção 2.1 é caracterizada pela busca por trabalhos semelhantes, relatando como se deu a sua formulação, na Subseção 2.2 é esclarecida a motivação do ensino estatístico descritivo por meio de tecnologias educacionais e o levantamento dos *softwares* usualmente adotados para suprir essa carência. Na Subseção 2.3 a definição do *IFMath*, sistema no qual se pretende incluir o *software* desenvolvido neste trabalho. Já nas Subseções 2.4 e 2.5 apresentam-se os conceitos relacionados a sistemas *web* e tecnologias, respectivamente.

2.1. Trabalhos semelhantes

A busca por diferentes trabalhos relacionados ao tema da pesquisa percorreu os Trabalhos de Conclusão de Curso (TCC) do curso de ciência da Computação do Instituto Federal de Santa Catarina (IFSC) e da Universidade Federal de Santa Catarina (UFSC), com o objetivo de levantar o que a comunidade científica havia produzido sobre desenvolvimento de *softwares* educacionais e o uso de tecnologias no ensino de Estatística Descritiva.

No IFSC, foram elaborados dois Trabalhos de Conclusão de Curso voltados ao desenvolvimento de *softwares* educacionais para o ensino e aprendizagem de Matemática, são eles: Desenvolvimento de uma ferramenta para o estudo de geometria plana e Sistema especialista para o ensino-aprendizagem de expressões algébricas, ambos os trabalhos resultaram em *softwares* que foram incorporados ao *software IFMath*¹.

Com o intuito de auxiliar no ensino de conteúdos matemáticos, Madruga et al. (2018) desenvolveram uma proposta de trabalho que destinou-se a criar um *software*, cujas funcionalidades e objetivos estariam voltados a auxiliar no estudo de geometria plana. O *software* possui uma interface amigável contendo uma base teórica que apresenta as propriedades do que pode ser construído e o passo a passo dos cálculos relacionados a essas construções, permitindo assim a interação do aluno com o objeto de estudo.

As construções foram viabilizadas usando uma biblioteca chamada *Three.js* a qual possibilitou a construção das figuras geométricas 2D do *software*, além disso o *framework Angular* facilitou no desenvolvimento e construção das páginas *web*.

Em relação ao passo a passo dos cálculos envolvendo os tipos de geometrias, foi construído envolvendo a entrada e resolução dinâmicas, ou seja, o usuário pode alterar os

¹O *software* será descrito na Subseção 2.3

valores da entrada que a resolução muda conforme o novo valor. O usuário pode também alterar a forma da geometria manualmente que os valores são atualizados dinamicamente, isso permite que o aluno veja quais e como os valores devem ser usados nos cálculos.

Avila et al. (2018) desenvolveram um *software* com a intenção de auxiliar no ensino da matemática, mais especificamente focado no ensino da álgebra. Neste *software* é desenvolvido todo o passo a passo de como as expressões algébricas são desenvolvidas, além de conter uma explicação de como os cálculos são desenvolvidos e uma base do conteúdo para auxiliar o estudante no seu aprendizado.

Os autores, afirmam que o *software* foi construído como um sistema especialista, o que significa dizer que o *software* toma decisões da mesma forma que um especialista humano em determinadas situações, para assim ser possível desenvolver uma solução para o problema algébrico presente. Foi necessário a criação de um compilador que fosse capaz de ler e converter uma expressão convencional para código de três endereços, para que assim o sistema especialista conseguisse realizar todas as operações necessárias.

Na UFSC, houve um Trabalho de Conclusão de Curso voltado ao desenvolvimento de um *software* educacional para o ensino e aprendizagem de Estatística denominado *SEstat.Net* - Ensino de estatística mediado por computador. Aliando-se à retórica do estudo viabilizado por meio de ferramentas tecnológicas, Nakazawa e Marafon (2003) se dedicaram à construção e implementação de um *software* estatístico abordando o conceito de ensino a distância. O sistema em questão foi denominado como *SEstat.Net*, já que seria a adaptação de outro sistema conhecido como *SEstat* usado até então pela Universidade Federal de Santa Catarina (UFSC) no ensino presencial de estatística. O *SEstat.Net* se baseou na arquitetura de seu antecessor adotando um sistema especialista o qual é uma forma de sistema baseado no conhecimento especialmente projetado para emular a especialização humana de algum domínio, nesse caso o conhecimento sobre conteúdos estatísticos.

O *SEstat.Net* possui uma maior abrangência na forma de acesso do que o *SEstat*, já que ele foi construído para ser utilizado via rede resolvendo um dos problemas recorrentes de apenas conseguir ter e utilizar o *software* localmente. A construção do *software* foi realizada utilizando a plataforma *Java 2* mais especificamente utilizando *Java 2 Platform, Enterprise Edition (J2EE)* a qual suporta serviços *web* e controla a infra-estrutura da aplicação, além de utilizar *Java Server Page (JSP)* na construção das páginas *Hyper-Text Markup Language (HTML)* da aplicação. A interface possibilita análise estatística de dados usando base de dados flexível, que disponibiliza aos alunos a realização de inferências estatísticas ou análises descritivas com uma ou duas variáveis, oferecendo os resultados estatísticos da análise, os possíveis caminhos de raciocínio e a linha de aprendizagem efetuada pelo usuário, garantindo assim que o aluno se torne o próprio agente de aprendizado.

2.2. Softwares Estatísticos e o Ensino de Estatística Descritiva

Segundo Campos et al. (2013) nos diversos campos de formação acadêmica, a estatística é comumente categorizada como uma disciplina obrigatória e de vital importância para as diferentes áreas, apesar de ser conhecida geralmente em cursos das Ciências Exatas. Sua relevância tanto para as Ciências Sociais, Humanas e Biomédicas deve ser ressaltadas, visto que essas podem usufruir da estatística como uma ferramenta para o estudo e

análise dos fenômenos de interesse geral e específicos inerentes a cada área. Com isso a disseminação dessa disciplina pelas mais variadas áreas de formação ganhou força e vemos a estatística presente em diferentes cenários.

Contudo, em relação a sua significância para a formação e desenvolvimento do estudante, o ensino de estatística vem sofrendo sérios problemas em todos os níveis de ensino, o que pode explicar a aversão que os alunos têm a esta disciplina e a sua dificuldade em assimilar os conteúdos estatísticos apresentados. A estatística descritiva por ser usualmente um dos primeiros conteúdos a serem mostrados aos alunos em estatística, tem um papel importante na introdução da disciplina e dos conceitos centrais dela, caso esta não seja bem elaborada e conduzida pelos professores o aluno possuirá uma curva de aprendizado lenta e portanto dificuldade desde o princípio na disciplina.

Borba et al. (2018) citam as principais ideias relacionadas à noção de “seres humanos-com-mídias” e trabalham com as fases das tecnologias educacionais. Os autores afirmam que o surgimento de uma nova tecnologia permite que novos tipos de problemas matemáticos possam ser explorados. Assim, se usássemos a tecnologia como meio de alavancar o ensino de estatística descritiva, além de utilizar a tecnologia como meio de aproximar o aluno à disciplina estaríamos dando a ele uma ferramenta de auxílio que propicia um ambiente de aprendizagem, onde os alunos podem realizar suas inferências e tirar suas próprias conclusões a respeito dos conteúdos abordados.

Na mesma vertente, de acordo com Borba (1999, apud Borba, Silva e Gadaniadis, 2018), os ambientes de aprendizado proporcionados por aplicações informatizadas podem potencializar o processo de ensino e aprendizagem através da experimentação. Para a área de estatística descritiva o *software* viria a auxiliar a organização das etapas envolvendo cálculos e a visualização de gráficos ou tabelas de forma rápida, além de permitir ao aluno interagir com as informações obtidas, podendo assim gerar novas conclusões sobre o objeto de estudo ou somente ter um alicerce de entendimento mais amplo.

No restante desta Subseção 2.2 apresentam-se *softwares* e/ou linguagens de programação frequentemente utilizados para o ensino de estatística e suas funcionalidades.

2.2.1. Geogebra Planilhas

GeoGebra (2020) é uma aplicação que possui foco no aprendizado e ensino da matemática em todos os níveis da educação de forma dinâmica e que abrange diversas áreas como estatística, geometria, álgebra, entre outras. O desenvolvimento foi iniciado por Markus Hohenwarter na Universidade de Salzburg em 2001, o projeto é mantido pela Universidade de Linz com o apoio de desenvolvedores.

É um *software* livre desenvolvido *a priori* utilizando a linguagem *Java* em sua versão para computador, mas está disponível em diversas plataformas, possuindo um aplicativo para dispositivos móveis e, também, uma versão *web* onde não é necessário nenhuma instalação para a utilização.

A planilha de estatística acrescenta alguns recursos ao *GeoGebra*, nela estão presentes algumas funções estatísticas, onde o usuário precisa escolher o intervalo da planilha com que se vai trabalhar para que possam ser realizadas as operações. Com essa ferra-

menta também é possível realizar análise univariada, bivariada, multivariada e calcular probabilidades.

2.2.2. Scilab

Scilab (2020) é uma linguagem de programação orientada a análise numérica de alto nível, que é acompanhada de um interpretador de mesmo nome. Com ela é possível de realizar diversas operações de maneira muito mais simples e eficiente comparado com outras linguagens de programação.

O *Scilab* apresenta inúmeras funcionalidades contendo entre elas, a possibilidade de mostrar gráficos 2D ou 3D, interpolação, regressões, polinômios, funções racionais, grafos e redes, também dando suporte a diversas análises da estatística descritiva. Com isso o *Scilab* torna-se uma alternativa ao *MATLAB*, já que é uma linguagem semelhante à este e é um *software* livre. A interface do *software* possui uma apresentação bem simples e intuitiva para o usuário, onde apresenta alguns ícones na parte superior e logo abaixo a área de desenvolvimento, local onde é realizado a escrita dos algoritmos.

O *Scilab* foi desenvolvido para ser usado em diversas áreas, podendo ser aplicado e utilizado tanto na educação quanto na indústria, mas como se trata de uma linguagem para o desenvolvimento de algoritmos acaba tendo a sua utilização na educação mais voltada para o ensino superior.

2.2.3. Minitab

Minitab (2020) é um *software* estatístico voltado para o âmbito comercial, com foco em mineração de dados. Por se tratar de um *software* voltado para meios profissionais e empresariais possui uma licença paga, o que praticamente inviabiliza sua utilização no ensino. Teve seu desenvolvimento iniciado em 1972 por pesquisadores da Universidade da Pensilvânia, com o foco inicial de incorporar computadores ao ensino nos cursos de estatística.

O *software* possui uma interface muito parecida com outros editores de planilhas eletrônicas, porém com o diferencial de que na mesma tela possui um campo para inserção de dados no formato de texto ou em planilha, sem a necessidade de trocar sua tela de trabalho dentro do *software*. Com o *Minitab* é possível realizar diversas operações que podem incluir: a criação de gráficos, estatística descritiva, correlação, covariância, regressão linear, regressão não linear, regressão de *Poisson*.

2.2.4. R-Commander

Rcommander (2020) é um interface gráfica desenvolvida para a utilização da linguagem R, desenvolvida e utilizada para a realização de operações estatísticas. É uma ferramenta bastante utilizada na mineração de dados e no ensino superior, por se tratar de uma linguagem poderosa na criação de *softwares* estatísticos. O R é mantido no formato código aberto. Por se tratar de uma linguagem de programação, possui algumas características como ser uma linguagem fracamente tipada e dinâmica.

O *software* possui uma interface bem simples, com um campo onde é possível a criação do *script* em R e logo abaixo uma área onde o resultado é apresentado, bem semelhante a outras interfaces gráficas para o desenvolvimento. Sua utilização na educação básica torna-se complexa pelo fato de exigir noções de programação dos discentes.

2.3. IFMath

Segundo Avila et al. (2018), o *IFMath* é um *software web* gratuito criado com a finalidade de auxiliar na resolução e ensino de exercícios matemáticos, contando para isso com um forte embasamento teórico dos conteúdos abordados e a resolução detalhada dos exercícios presentes ou aplicados sobre ele, bem como a visualização gráfica do problema se necessário. Dessa forma o *IFMath* se propõe a auxiliar tanto no ensino quanto na aprendizagem dos alunos.

O *software IFMath* foi desenvolvido no Instituto Federal de Santa Catarina (IFSC) - câmpus Lages. Até o momento ele conta com os módulos de: Números e Operações, Geometria, Álgebra e Funções e Estatística. Abaixo estão listados os projetos de pesquisa que já foram desenvolvidos no câmpus com foco em seu desenvolvimento.

- Desenvolvimento de um aplicativo para a apresentações de dados por meio de gráficos e tabelas;
- Desenvolvimento de um aplicativo para o ensino e aprendizagem e produtos notáveis e polinômios;
- Desenvolvimento de um aplicativo para o estudo da estatística descritiva;
- Produção de um aplicativo para o estudo da geometria espacial;
- Um Software web para o estudo de Números e Operações.

Além disso, foram desenvolvidos dois Trabalhos de Conclusão de Curso (além deste) visando o seu desenvolvimento, os quais estão listados abaixo.

- Desenvolvimento de uma ferramenta para o estudo de geometria plana;
- Sistema especialista para o ensino-aprendizagem de expressões algébricas.

O *IFMath* possui compatibilidade com diversos tipos de dispositivos eletrônicos. Para ter acesso à ferramenta basta possuir acesso a Internet e um navegador. O sistema permite ao usuário trabalhar com dados dinâmicos, ou seja, o usuário é capaz de informar os dados de entrada da questão e as respectivas respostas são apresentadas conforme eles são alterados.

Além disso, o sistema contém o passo a passo da resolução do problema em questão, auxiliando o usuário a compreender de onde e como aqueles resultados foram obtidos. O *software* permite ainda visualizações e interações gráficas em conteúdos que abordam temas como a geometria, garantindo uma experiência de aprendizado diferenciada, através do contato do aluno com o objeto de estudo.

2.4. Tecnologias

Nesta subseção encontram-se as descrições das ferramentas utilizadas para a implementação do referido módulo.

2.4.1. Sistema Web

Sistemas *web* são, segundo Messenlehner e Coleman (2014), aplicações projetadas para serem executadas através do uso de navegadores *web* como o Google Chrome, Mozilla Firefox, Opera etc. O desenvolvimento *web* é possibilitado pelo manuseio de três diferentes linguagens essenciais na sua concepção, que são o *HTML*, *CSS* e o *JavaScript*.

O *HTML* (*Hypertext Markup Language*) é uma Linguagem de Marcação de Hipertexto, corresponde ao esqueleto da página, ficando ao seu cargo indicar ao *browser* por meio dos elementos chamados *tags* a disposição das seções na página.

O *CSS* (*Cascading Style Sheets*) ou Folhas de Estilo em Cascata permite estilizar as páginas, mais precisamente controla a apresentação dos elementos que estão dentro do *HTML* (McGrath, 2010). Pode-se aplicar sobre esses elementos estilos dos mais diferentes teores, criando para tal o que chamam de regras de estilo em seu documento, ou seja, ele seleciona elementos específicos do *HTML* e aplica um novo *layout* sobre eles em concordância com sua regra.

Por fim, o *JavaScript* é a linguagem de programação cujo interpretador é embutido nos navegadores, isso permite que os *scripts* contidos nas páginas sejam interpretados quando a página é carregada no mesmo, para fornecer funcionalidade (McGrath, 2010). Sendo assim, a principal função do *JavaScript* é a criação de códigos para a interação com o *DOM* (*Document Object Model* ou Modelo de Objeto de Documento) que é a interface que representa a página e permite modificá-la.

Estas três linguagens são tecnologias básicas que todos os navegadores devem compreender (Scheidt, 2015). Da mesma forma, o autor complementa a necessidade da compreensão das linguagens por parte dos programadores que querem criar aplicações *web*, ainda mais com o aumento constante da complexidade neste desenvolvimento.

Em decorrência dos fatos salientados, cabe ressaltar as características que tornam esses sistemas vantajosos, pois como declara Loundon (2010), esse tipo de aplicação assegura longevidade de código, disponibilidade contínua, acesso a partir de diferentes ambientes e redimensionamento das telas de acordo com os tipos de dispositivos.

Igualmente o *IFMath* e seu novo módulo de estatística descritiva proposto neste trabalho, se enquadram ao ramo de aplicabilidade aqui constatado, consistindo em uma aplicação *web* para o auxílio no estudo e ensino de conteúdos matemáticos, possuindo sua base formada empregando *frameworks* de construção destinados a esse ramo, sendo o *Angular* a ferramenta principal utilizada na formulação do *software*, descrita na Subseção 2.4.3.

Assim sendo, como se trata de uma ferramenta *web* cujas funcionalidades são disponibilizadas e supridas aos usuários de maneira local nos navegadores, sem restrições ou níveis de acesso, este sistema não necessita guardar qualquer tipo de dado da aplicação em bancos de dados, ou outros meios de armazenamento.

2.4.2. Visual Studio Code

Segundo o seu *site* oficial o Visual Studio Visual Studio Code (2020) mais popularmente conhecido como *VSCode* é um editor de código open source criado pela *Microsoft* destinado em especial ao desenvolvimento de aplicações *web*, possui uma ampla capacidade de suporte às linguagens *web* se tornando dessa forma compatível com uma grande quantidade de projetos, além de estar disponível em muitas plataformas como *Windows*, *Mac* e *Linux*.

Da mesma forma, o editor conta com ferramentas de desenvolvimento que auxiliam desde a organização do projeto até a inclusão de extensões, as quais alavancam e interferem de maneira positiva no tempo de finalização e qualidade dos mesmos, sem mencionar que o editor conta com suporte interno para operações como depuração e controle de versão.

Devido ao já mencionado objetivo deste trabalho, juntamente com as características estruturais inerentes ao *IFMath*, o *Visual Studio Code* se mostrou uma ferramenta poderosa e completa que garante uma maior flexibilidade e compatibilidade com as demais tecnologias necessárias para a elaboração do *software* proposto.

2.4.3. Angular

Segundo Singh (2018) o *Angular* é um *framework open source* (código aberto) desenvolvido e mantido pela Google para a construção de aplicações *web*. O *framework* une as tecnologias primordiais e essenciais no desenvolvimento *web* já que trabalha com *HTML*, *CSS* e *Typescript* sendo este último, de acordo com Alves (2014) responsável por possibilitar a criação de um código *JavaScript* mais claro e bem escrito.

Booth (2017) afirma que os mecanismos trazidos pelo *Angular* viabilizam o desenvolvimento orientado a componentes, garantindo a criação de componentes desacoplados que ao longo do tempo possam ser reutilizados em projetos futuros ou facilmente substituíveis se necessário. Ao mesmo tempo oferece padrões de estruturação do código organizando a aplicação em módulos, permitindo dessa forma tornar a aplicação segregada em recursos e partes reutilizáveis, além de separar as partes do sistema de acordo com a sua respectiva classificação e garantir uma maior legibilidade estrutural.

De acordo com a documentação oficial, cabe ressaltar a presença de elementos responsáveis por proporcionar facilidades na interação em tempo real entre o *HTML* e o *Typescript*, como é o caso do *data-binding* cuja funcionalidade permite que ao modificar uma variável no *HTML* o valor também seja modificado no *Typescript*, da mesma forma acontece se o valor no *Typescript* for modificado, o valor no *HTML* irá mudar também (Angular, 2020). Essa característica é de objetividade excepcional em páginas que exigem interações mais dinâmicas para com o usuário, como é o caso e o grande diferencial do *IFMath* na resolução passo a passo dos problemas matemáticos.

Levando em consideração esses aspectos e a necessidade de garantir a produção do módulo de estatística do *IFMath* usando as mesmas tecnologias básicas para evitar o desmembramento do *software* em diferentes sistemas, o *Angular* permanece como a plataforma principal de desenvolvimento.

2.4.4. Bootstrap

O *Bootstrap* é um *framework front-end* de código aberto utilizado em projetos *web* na produção de *layouts* responsivos, inclui modelos de design com base no *HTML* e *CSS* como botões, formulários e tabelas, além de oferecer recursos responsivos que se ajustam a tela desde telefones a *desktops* (Zabot e Matos, 2020).

De acordo com Jakobus e Marah (2016) o *Bootstrap* foi criado por Mark Otto e Jacob Thornton como uma solução interna do *Twitter* para resolver as inconsistências de código dentro de sua equipe de desenvolvimento e em 2011 foi lançado como um produto de código aberto e conforme o *site* oficial da ferramenta ele é o *framework* mais popular do mundo para a construção de aplicações *web* responsivas (Bootstrap, 2020).

Sua responsividade se deve ao fato da presença de um sistema denominado *Grid System*, cuja infraestrutura permite a presença de 12 colunas ao longo da página e, conforme Snege (2018), por ser um sistema responsivo é capaz de reorganizar essas colunas automaticamente de acordo com o tamanho da tela.

A referida ferramenta foi selecionada por garantir meios facilitados na elaboração visual do *software*, demandando menos esforço produtivo e criativo na determinação do *layout*, já que disponibiliza modelos predefinidos, ou seja, concede códigos fonte prontos para utilizar nos projetos, oportunizando ao desenvolvedor uma maior rapidez, facilidade e reusabilidade no processo de estilização do sistema, sem mencionar a sua compatibilidade com todos os navegadores modernos.

2.4.5. KaTeX

De modo a garantir uma representação precisa de fórmulas e resultados matemáticos que por conseguinte apresentam teor essencial na elaboração da interação visual com os usuários pelo modelo de aplicação proposto, a biblioteca *JavaScript* denominada *KaTeX* será utilizada na renderização *web* de notações matemáticas, bem como na padronização dos resultados obtidos essenciais a composição do módulo de estatística descritiva do *IFMath*.

Essa biblioteca possui compatibilidade com as principais tecnologias utilizadas, como é o caso do *Angular*, onde ela pode ser instalada via *npm* que é um gerenciador de pacotes do *Node.js*. Após a sua instalação, o *ng-katex* se apresenta no *Angular* como um módulo e pode ser utilizado através da sua importação no projeto. A partir dessa importação seu uso se torna possível por meio da inclusão da *tag ng-katex* dentro do *HTML* de um componente, onde é possível escrever equações *LaTeX* completas sobre a mesma (npm, 2020).

Latex pode ser definido como uma ferramenta de auxílio que inclui recursos no processo de produção de documentação científica e técnica, estando presente na facilitação do leitor sobre o conteúdo, em especial nesse caso a formatação, visualização e entendimento dos resultados matemáticos obtidos na resolução dos problemas (LaTeX, 2020).

2.4.6. ApexCharts

ApexCharts é uma biblioteca *open-source* destinada a prover a criação de gráficos em páginas *web*. Conta com uma documentação muito bem organizada, com mais de 100 exemplos para serem utilizados como base, contemplando integrações com diversas tecnologias de desenvolvimento *front-end* como o *Angular*, *Vue*, *React* e o próprio *JavaScript* (ApexCharts, 2020).

A integração com *Angular* se dá via instalação *npm* que, como já mencionado no tópico anterior, é o gerenciador de pacotes do *Node.js*. Após a sua instalação basta adicionar o seu módulo ao projeto e seguir as recomendações para a produção de um exemplo na seção de gráficos com *Angular* do seu *site* oficial.

A fim de viabilizar as construções gráficas, optou-se por usar essa biblioteca devido a sua ampla capacidade de oferecer gráficos de diferentes segmentos com a sua devida documentação, contendo exemplos de utilização e a possibilidade de adicionar novos incrementos funcionais de acordo com as características desejadas.

3. Desenvolvimento

O desenvolvimento deste trabalho se sucedeu através de uma pesquisa bibliográfica que de acordo com Macedo (1995) é a busca por informações, seleção dos documentos que se relacionam com o problema da pesquisa (livros, artigos, teses, etc) e o fichamento das referências para que sejam utilizadas, concebendo a primeira etapa de qualquer tipo de pesquisa científica.

Assim sendo, a pesquisa consistiu na obtenção de referencial teórico a fim de aumentar o panorama e compreensão sobre o campo de trabalho associado à pesquisa. Englobando o estudo de trabalhos semelhantes voltados à concepção de *softwares* educacionais, à comprovação da importância do ensino de estatística descritiva e como ferramentas informatizadas podem fortificar no seu aprendizado. O levantamento do conteúdo foi realizado após a análise dos conteúdos estudados em estatística descritiva.

Esta seção está subdividida em quatro Subseções que descrevem o desenvolvimento da pesquisa. Na Subseção 3.1 encontra-se o levantamento de conteúdos que estarão presentes no Módulo de Estatística do *IFMath*, na Subseção 3.2 há o planejamento do módulo de estatística e na Subseção 3.3 encontra-se o processo implementação.

3.1. Levantamento de conteúdos do Módulo de Estatística Descritiva do *IFMath*

Diante do já mencionado e do fato que o *IFMath* se propõe a auxiliar e compreender em termos de conteúdos matemáticos, o referido trabalho se propõe a desenvolver o módulo de estatística descritiva desse *software*, visto a sua importância em diversas áreas de estudo e a dificuldade obtida no seu aprendizado nos mais diversos níveis de ensino.

O módulo possui definições com forte embasamento teórico, gráficos e tabelas inerentes e necessárias em estatística descritiva, seguindo o modelo padrão do *IFMath* quanto a interação com o usuário. A parte visual se dará de maneira semelhante as demais telas presentes no *software*, respeitando o padrão de *layout* adotado até então.

Além disso, propõem-se que o *software* realize os cálculos relacionados à estatística descritiva, como: medidas de tendência central, dispersão, assimetria e curtose a

partir da inserção de dados pelo usuário. Abaixo encontram-se as Subseções de 3.1.1 a 3.1.2, referentes aos tópicos que estarão presentes no módulo a ser desenvolvido.

3.1.1. Apresentação de dados

A apresentação de dados, conforme Spiegel e Stephens (2009) consiste na forma como os dados são dispostos para visualização. Na estatística, conforme os autores pode-se trabalhar com duas formas distintas:

- Tabelas: onde os dados são dispostos em linhas e colunas de forma a facilitar sua leitura. Existem regras para a criação de tabelas, como elementos chaves que devem estar contidos dentro da mesma, que são: o título, onde contém a informação sobre o que a tabela trata; o corpo, onde estão presentes os dados que serão apresentados; e, a fonte, que informa de onde os dados presentes foram coletados.
- Gráficos: são maneiras geométricas de apresentar os dados, podendo assumir diversos tipos de formas e composições. Alguns exemplos de gráficos mais utilizados são os gráficos de barras, de colunas e de setores.

Para melhor apresentar dados, deve-se levar em consideração a maneira como os dados são classificados, quanto aos seus tipos, tratando-se de variáveis qualitativas ou variáveis quantitativas.

Variáveis qualitativas, conforme Spiegel e Stephens (2009), são aquelas que não podem ser quantificadas, sendo assim definidas por categorias onde se é possível representar a classificação dos indivíduos. Essas variáveis podem ser divididas em outros dois grupos: as nominais, onde não existe uma ordenação entre as categorias; e as ordinais, onde a partir das categorias criadas é possível haver uma ordenação das mesmas.

Spiegel e Stephens (2009) definem as variáveis quantitativas como aquelas variáveis que podem ser apresentadas de maneira numérica. Por sua vez, podem ser divididas em: contínuas, onde podem estar presentes valores fracionários durante a contagem; e discretas onde apenas podem assumir valores inteiros durante a contagem ou, ao menos se consegue uma correspondência biunívoca entre as variáveis e o conjunto dos números naturais.

3.1.2. Medidas de tendência central, dispersão, assimetria e curtose

Medidas de tendência central, dispersão, assimetria e curtose são parâmetros estatísticos calculados a partir de variáveis quantitativas, que como registrado na Subseção 3.1.1, são dados numéricos utilizados para mensurar algo.

Quando se realiza uma coleta de dados, conforme Barbetta et al. (2018), eles podem ser extraídos de duas maneiras: em forma amostral, onde apenas uma parcela de toda a população é escolhida para realizar os cálculos, mas essa parcela deve ser escolhida de maneira aleatória para que ela possa representar a população; e como censo, utilizando os dados de todos os elementos da população para realizar as operações.

Percebe-se que há a necessidade de trabalhar os dados de maneira distinta, caso trabalhe-se com uma amostra ou uma população. Ainda, é possível manipular dados

conforme a forma que se tem acesso a estes, podendo ser na forma de dados isolados, dados agrupados ou dados em classes.

Medidas de tendência central ou medidas de posição, conforme Spiegel e Stephens (2009), são os valores que se encontram centralmente em um conjunto de dados ordenados, destacam-se os seguintes parâmetros:

- Média: representa o valor médio do conjunto;
- Moda: o valor que ocorre com maior frequência dentro do conjunto;
- Mediana: o valor central que divide o grupo em duas partes, considerando que o conjunto esteja ordenado;
- Quartis: ideia semelhante à mediana, mas dividindo o grupo em quatro partes, sendo possível, calcular os valores de Q_1 , Q_2 e Q_3 (três quartis), sendo Q_2 igual a mediana;
- Decis: dividem o conjunto de dados em dez partes com a mesma quantidade de elementos; e,
- Percentis: dividem o conjunto em 100 partes com a mesma quantidade de elementos.

As medidas de dispersão, por sua vez, representam o grau de dispersão dos dados em torno do valor médio do conjunto (Spiegel e Stephens, 2009). As medidas de dispersão mais utilizadas são:

- Amplitude total: que calcula a diferença entre o maior e o menor valor do conjunto;
- Desvio médio: mostra a dispersão dos dados em relação à média aritmética;
- Desvio padrão: que representa o grau de dispersão de um conjunto de dados; e,
- Variância: que representa o quão distante cada valor do conjunto está do valor médio.

Por fim, serão trabalhados os coeficientes de assimetria de Pearson e curtose. Segundo Spiegel e Stephens (2009), pode-se dizer que uma distribuição é simétrica quando possui os mesmos valores para moda, média e mediana, e será assimétrica à esquerda ou negativa, quando a curva de frequência da distribuição está mais longa e para a esquerda com a média da distribuição, quando o inverso ocorre é chamado de assimetria positiva ou assimetria à direita. Recomenda-se que seja realizada a análise gráfica para verificar as assimetrias, através de um histograma ou uma curva de distribuição por exemplo. Na Figura 1 estão explícitos os três tipos de distribuição possíveis: assimétrica à esquerda, simétrica e assimétrica à direita.

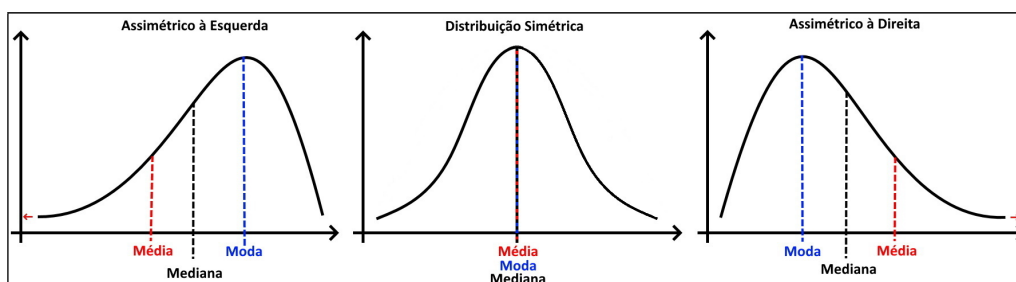


Figura 1. Exemplo dos tipos de assimetria.

Os coeficientes de assimetria auxiliam na comparação do grau de assimetria entre diferentes conjuntos de dados. Existem diferentes coeficientes de assimetria, este trabalho aborda os dois coeficientes de assimetria de Pearson, que foram desenvolvidos a partir do comportamento esperado das medidas de tendência central, em particular, da moda e da mediana, e são dados por:

$$A_1 = \frac{(\bar{x} - Mo)}{s}$$

Onde:

- A_1 = Primeiro coeficiente de assimetria
- $\bar{x}$ = Média da amostra
- Mo = Moda da amostra
- s = desvio padrão da amostra

e,

$$A_2 = \frac{3 * (\bar{x} - Md)}{s}$$

Onde:

- A_2 = Segundo coeficiente de assimetria
- $\bar{x}$ = Média da amostra
- Md = Mediana da amostra
- s = desvio padrão da amostra

Ainda, segundo Spiegel e Stephens (2009), a curtose mede o grau de achatamento da curva da distribuição, o seu cálculo faz uso dos quartis e percentis, conforme a fórmula:

$$C = \frac{Q_3 - Q_1}{2 * (P_{90} - P_{10})}$$

Onde:

- C = Coeficiente Percentílico de Curtose
- Q_1 = Primeiro Quartil
- Q_3 = Terceiro Quartil
- P_{10} = Décimo Percentil
- Q_{90} = Nonagésimo Percentil

A partir do resultado deste cálculo, pode-se dizer se essa distribuição possui ou não um achatamento normal de sua curva. Por exemplo, se para uma amostra de valores numéricos o resultado da curtose for igual a 0,263 classifica-se como normal recebendo o nome de mesocúrtica. Caso resultado for menor que 0,263 classifica-se como leptocúrtica e possui um formato mais alongado. Por fim se o resultado for maior que 0,263, trata-se de uma distribuição platicúrtica, possuindo um formato mais achatado. Na Figura 2 há um exemplo de achatamento das curvas de distribuição.

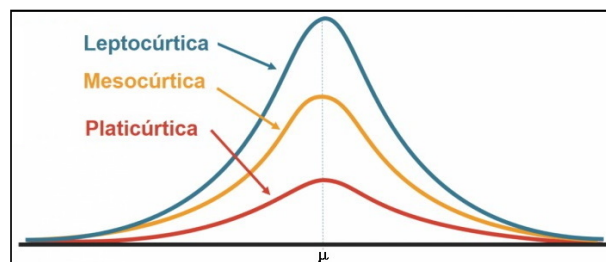


Figura 2. Exemplo dos achatamentos.

3.2. Planejamento do módulo de estatística

Esta subseção contempla as atividades relacionadas ao processo de desenvolvimento na produção e implementação do *software* de estatística descritiva proposto no presente trabalho. Sua objetividade se baseia no propósito de garantir eficiência produtiva e qualidade na organização das etapas envolvendo o projeto.

As etapas de produção da aplicação foram escolhidas com o auxílio de conceitos direcionados aos princípios de planejamento, que como afirma Pressman (2011) devem servir de guia para a equipe. Tais princípios relatados pelo autor colaboraram para o levantamento dos itens que devem constituir o processo de planejamento. A partir destes, a equipe de trabalho definiu as prioridades e as restrições do projeto.

Após adquirir o conhecimento sobre o escopo do projeto, a fundamentação dos seus aspectos centrais e principais se deu pela divisão do processo de produção em três etapas. A primeira voltada à modelagem do módulo de estatística descritiva contendo alguns casos de uso existentes, para estabelecer o entendimento de como as funcionalidades do sistema interagem umas com as outras e como serão utilizadas pelo usuário. A segunda etapa do planejamento compreende a construção dos modelos de *layout* desejáveis das telas do módulo. Por conseguinte, na última etapa estão as definições conceituais a respeito das características do sistema a ser desenvolvido, contendo as devidas explicações sobre sistemas *web*.

3.2.1. Modelagem do módulo

A modelagem do sistema *web* é apresentada nesta seção, contemplando os requisitos de formação da aplicação. Estes requisitos se referem a criação de estruturas denominadas casos de uso, que proporcionam uma melhor representação visual das funcionalidades requeridas ao *software*, nesse mesmo sentido Pressman (2011) afirma que um conjunto coerente de casos de uso poderia servir como alicerce na descoberta de um ou mais padrões de análise.

Na elaboração dos casos de uso conforme afirma Pressman (2011), o primeiro passo a ser seguido na sua escrita é a definição do conjunto dos “atores”, isto é, os envolvidos na interação com o produto, sejam as diferentes pessoas ou dispositivos que fazem uso dele, uma vez que os atores tenham sido identificados os casos de uso podem ser construídos.

Após realizar a análise das peculiaridades envolvendo o escopo funcional arquitetural necessário, os casos de uso do sistema em questão foram produzidos e podem ser

analisados ao observar a Figura 3.

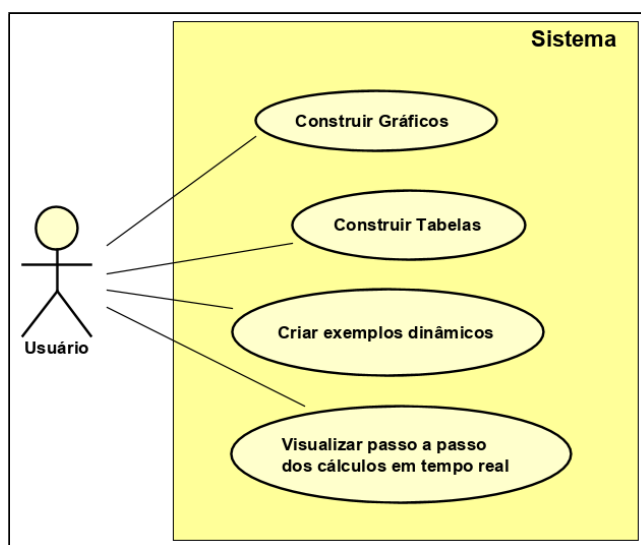


Figura 3. Diagrama de caso de uso do sistema.

Em conformidade com as funcionalidades desejáveis levantadas, caberá ao *software* realizar operações de construções de gráficos e tabelas, ao mesmo tempo caberá também a ele exibir os resultados de maneira dinâmica com detalhamentos passo a passo na resolução dos cálculos.

O processo de construção de gráficos e tabelas bem como o passo a passo das resoluções os envolvendo se inicia conforme o diagrama de atividades da Figura 4. Após os usuários inserirem os dados nos campos de *input*, o sistema realiza uma verificação nos dados, de modo a garantir a compatibilidade entre campos com variáveis associadas. Essa análise está presente na parte dos gráficos por exemplo, pois alguns campos dependem entre si de quantidades de elementos iguais, como é o caso de variáveis qualitativas, onde para cada *input* qualitativo (nome) deve haver um valor quantitativo (número) correspondente àquela variável.

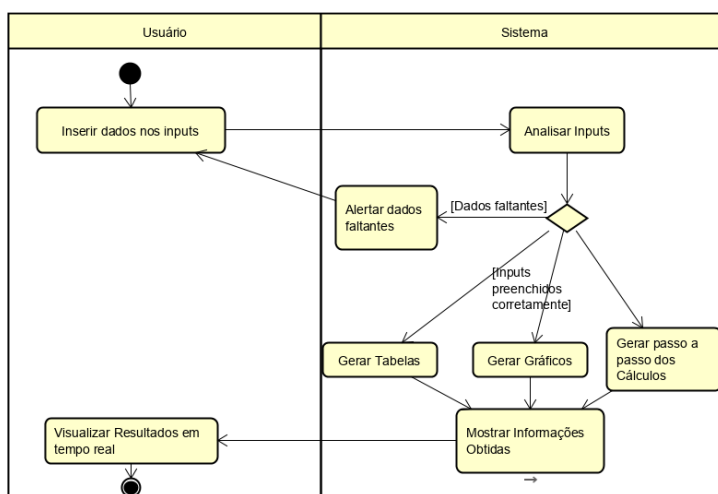


Figura 4. Diagrama de Atividades do sistema.

Em decorrência da verificação, o sistema realiza a construção dos gráficos e tabelas apenas se nenhuma incompatibilidade nos *inputs* for encontrada, caso contrário emite uma mensagem de erro no respectivo campo. Por fim, se todas as inserções de dados forem corretas o usuário será capaz de visualizar as construções feitas a partir das suas entradas de dados.

Tendo em vista os aspectos que foram observados, a parte cuja implementação vai ser incluída no módulo de estatística do *IFMath* seguindo as especificidades discutidas nesta seção, será a parte de estatística descritiva. Que compreende os conteúdos subdivididos em Apresentação de Dados, Medidas de Tendência Central, Dispersão, Assimetria e Curtose.

3.2.2. Interface gráfica

A fim de garantir compatibilidade aos *layouts* das telas já existentes no *IFMath*, os modelos para as páginas do módulo estatístico em questão vão seguir esses mesmos padrões de interface estabelecidos previamente, garantindo compatibilidade visual com o restante do sistema. Para isso, protótipos foram esboçados visando demonstrar o aspecto visual inicial com a disposição dos conteúdos sobre a tela ao usar a padronização existente no *software*.

Consequentemente ao desejo de seguir os modelos de tela do *IFMath*, surgiu a necessidade de analisar a interface da aplicação para então escolher a melhor forma de organização, que fosse compatível e atendesse as novas telas. Ao examinar as diferentes páginas, a combinação visual que garante excelente correspondência às características primordiais buscadas, tanto na organização de gráficos e tabelas quanto na de medidas estatísticas é o padrão apresentado na Figura 5.

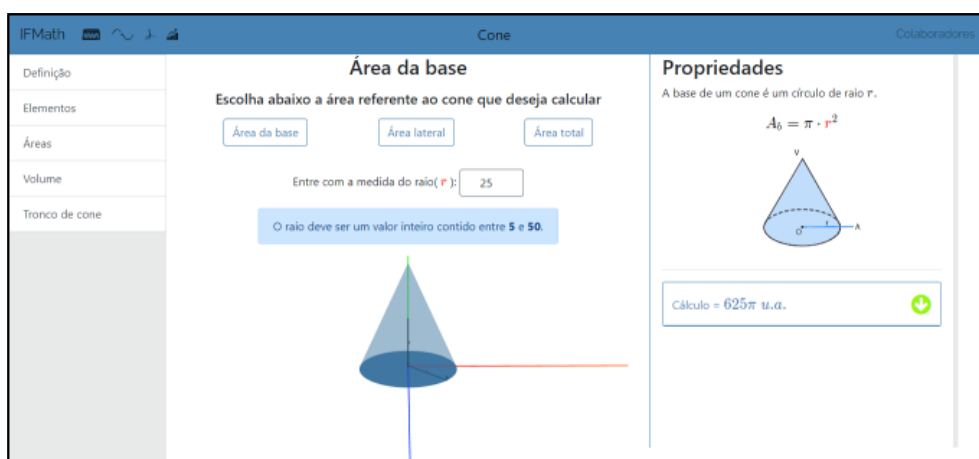


Figura 5. Tela da área da base do Cone presente no *IFMath*.

Na Figura 5, tem-se que o *layout* é apresentado dividindo o seu espaço em 3 áreas principais, que são o menu onde se localizam as áreas de acesso aos conteúdos contemplados, a região onde se distribuem os *inputs*, botões e a tecnologia principal de estudo que nesse caso é figura geométrica, e finalmente a parte equivalente às propriedades do objeto de estudo junto com suas definições conceituais e o passo a passo dos cálculos.

Todavia, dispor das ideias de divisão mencionadas, propiciou a etapa de prototipação envolvendo as interfaces de Gráficos e Medidas de Tendência que podem ser visualizadas nas Figuras 6 e 7. Os protótipos foram feitos utilizando o Vectr uma ferramenta *web* própria para a construção de designs. As telas a seguir foram obtidas a partir do *Vectr* e simbolizam a aparência que a área de interação vai possuir.

Figura 6. Protótipo da tela de Gráficos.

Com base no modelo de tela observado na área da base do cone, a disposição estrutural presente no protótipo dos gráficos se deu de maneira semelhante. No lado esquerdo o menu contendo as áreas de estudo trabalhadas, no centro a localização dos componentes principais de visualização e interação para com o usuário, por fim na terceira divisão como já constatado se localiza toda a parte conceitual e explicativa envolvendo o conteúdo.

Figura 7. Protótipo da tela de Medidas de Tendência Central.

Em suma, o aspecto interativo com o usuário permanece semelhante aos demais módulos do *IFMath*, permitindo ao mesmo tempo gerar exemplos dinâmicos com resultados em tempo real e todas as demais peculiaridades conceituais inerentes de cada conteúdo, se diferenciando apenas nas funcionalidades novas propostas.

3.3. Implementação

A criação do módulo de estatística descritiva no *IFMath* foi a primeira atividade de implementação realizada. O *IFMath* por utilizar o *framework Angular* possui sua base arquitetural de desenvolvimento consolidada pela divisão das suas áreas de estudo em estruturas denominadas de módulos, ou seja, cada tema de estudo dentro do sistema irá possuir um módulo associado. Para realizar a criação do módulo, foi necessário analisar a estrutura do sistema como um todo, de modo a evitar uma construção equivocada e fora dos padrões pré estabelecidos.

Conforme a análise da estrutura do projeto, foram encontradas algumas particularidades referentes à hierarquia e criação de pastas essenciais. Em cada módulo para garantir uma melhor organização, uma pasta chamada *components* foi criada para armazenar todas as telas presentes no módulo e uma pasta *shared* para conter todos os componentes e estilos que podem ser reaproveitados em outras telas, como botões, modelos de telas base entre outros. Outro aspecto importante é que além da necessidade de um arquivo de módulo para declarar os componentes, um arquivo de *Routing* (rotas) deve existir também, pois é nele que definem-se as *urls* de acesso dos componentes criados.

Da mesma forma, estilos globais para a formatação das interfaces foram produzidos, totalizando em três arquivos *CSS* na pasta *styles* dentro da pasta *shared*, todos eles definidos para um bloco específico da tela, conforme delimitada na concepção dos protótipos das interfaces gráficas, conteúdos centrais (*inputs*, botões, gráficos e tabelas) e os conteúdos da área de propriedades (passo a passo dos cálculos e definições).

Qualquer página, sendo ela pertencente a gráficos, ou medidas, deve usar dois desses estilos obrigatoriamente, o *est-conteudo* e o *est-rightbar* importando o diretório dos mesmos no arquivo *CSS* do componente em desenvolvimento. Os arquivos estão relacionados à formatação da tela no seu centro e região lateral, provendo a elas as características visuais essenciais. Quanto ao arquivo opcional *est-center-items*, deve ser utilizado no momento que for preciso existir uma barra de rolagem no centro da página, caso houver muitos campos declarados.

Quanto ao aspecto visual das telas, de modo a otimizar o tempo gasto no desenvolvimento, um componente genérico de base para as telas do módulo foi elaborado, e adicionado à pasta global de componentes (*shared*) mencionada acima. Nesse arquivo são delimitadas as áreas e formatos que as telas podem e devem assumir, por meio de uma estrutura *HTML* chamada *section*, a qual define a estrutura da tela de forma genérica, contendo todos os itens que ela pode possuir, e invocá-los, garantindo que apenas os itens desejados sejam utilizados. Na Figura 8 segue a figura do *HTML* referente a esse elemento de uso global, denominado de *est-base-view*:

```

<section class="ifmath-view">

  <div class="row mt-2">

    <!-- Botões e conteúdo -->
    <div class="col-lg-7 col-12">
      <!-- Botões -->
      <div class="row justify-content-around">
        <ng-content select=".btn-container"></ng-content>
      </div>
      <!-- Coloca os componentes centrais da tela sobre uma barra de rolagem -->
      <div>
        <ng-content select=".scroll-center"></ng-content>
      </div>
      <!-- Inputs -->
      <div class="form-group row">
        <ng-content select=".input-container"></ng-content>
      </div>
      <!-- Conteúdo abaixo dos botões -->
      <div class="row justify-content-center">
        <ng-content select=".content-container"></ng-content>
      </div>
    </div>

    <!-- Propriedades -->
    <div class="col-lg-5 col-12">
      <ng-content select=".rightbar"></ng-content>
    </div>
  </div>

</section>

```

Figura 8. *Section HTML* do *est-base-view*, que contém todos os elementos e características que a tela pode assumir.

Por conseguinte o processo de desenvolvimento das interfaces foi iniciado, tomando como base os *layouts* apresentados na Subseção 3.2.2. A ordem inicial de implementação das telas seguiu pelo desenvolvimento dos gráficos e tabelas estatísticas e por fim as medidas de tendência.

3.3.1. Gráficos e Tabelas

A concepção inicial do módulo de estatística descritiva, iniciou por essa etapa. Tomando como base a numerosa quantidade de componentes que seriam desenvolvidos para gráficos e tabelas estatísticas, uma alteração no padrão normal de pastas desse módulo no *IFMath* se fez necessária, ao invés de alocar todos os componentes desenvolvidos em uma única pasta raiz, a pasta *components*, foram criadas pastas adicionais para a melhor organização dos componentes desenvolvidos. Essa nova organização adicionou uma pasta denominada “Gráficos” dentro do diretório raiz do módulo (*components*), a qual possui outros dois diretórios, um para os gráficos qualitativos e o outro para os quantitativos, assim os componentes criados são alocados de forma mais organizada e de acordo com a sua área de estudo.

Com relação a nomenclatura dos componentes, um padrão também foi estabelecido, onde no seu nome deve constar a sigla do módulo seguido do tipo de componente (gráfico ou medida), sigla da área de estudo (caso houver alguma específica) e por fim o nome do item a ser produzido, ou seja temos uma estrutura equivalente a *siglaModulo-tipoComponente-areaEstudo-nomeItem*. Lembrando que na criação dos componentes com o *Angular* são gerados 4 arquivos e uma pasta, onde todos eles possuem esse mesmo nome estabelecido.

O próximo passo, consistiu na integração da biblioteca *ApexCharts* ao *IFMath*, sua instalação se deu pelo *npm* o gerenciador de dependências do *Angular*, após isso bastou seguir os exemplos fornecidos pelo *site* da biblioteca para gerar os gráficos de acordo com o estabelecido. Apesar de os gráficos serem compatíveis com *Angular*, precisaram ser ajustados para funcionar no projeto, pois a versão do *Angular* utilizada no *IFMath* é a 5.2, portanto incompatível com as demos de integração disponibilizadas pela biblioteca. A solução para esse problema foi adaptar os exemplos existentes de *JavaScript* do *site* para o *Typescript* dos componentes. Na Figura 9 segue um trecho de código mostrando o processo de construção dos gráficos.

```
this.chartOptions = {
  series: this.quantidade,
  chart: {
    width: 400,
    type: "pie"
  },
  labels: this.varQual,
  colors: this.chartColors,
  responsive: [
    {
      breakpoint: 360,
      options: {
        chart: {
          width: 200,
        },
        legend: {
          position: "bottom"
        },
      },
    },
  ],
  plotOptions: {
  },
  title: {
    text: this.title,
    align: "center",
  },
};
this.chart = new ApexCharts(document.querySelector("#chart"), this.chartOptions);
this.chart.render();
```

Figura 9. Figura da estrutura utilizada pela biblioteca para a construção do gráfico de setores no *Typescript*.

Na Figura 9 são mostrados os objetos de construção característicos do *ApexCharts*,

onde a partir da sua documentação é possível incluir itens e customizar os gráficos se assim desejar, além de possuir as especificações técnicas de suas funcionalidades. Na construção dos gráficos um objeto do tipo *ApexOptions* é criado, na Figura 9 está denominado como *chartOptions*.

Na renderização do gráfico criado internamente, é preciso ter um objeto instanciado do tipo *ApexCharts*, o qual tem a capacidade de chamar a função *render* responsável por construir graficamente o objeto elaborado previamente em *chartOptions*. Esse objeto ao ser instanciado, como na Figura 9, possui dois parâmetros, o id do elemento *HTML* onde será renderizado, e o próprio elemento criado internamente. No *HTML*, após passar por todos esses procedimentos o gráfico é gerado na *tag* com o id especificado, por padrão e convenção o nome será o mesmo em todos os gráficos.

Quanto ao aspecto visual das telas, um modelo de tela base foi criado. Esse arquivo base denomina-se *est-base-view*, que por meio de uma propriedade conhecida como *selector*, segundo a documentação do *Angular*, permite que qualquer tipo de componente criado e declarado em seu respectivo módulo possa ser reaproveitado em outras telas, bastando apenas declarar seu nome no arquivo *HTML* desejado.

<pre><est-base-view> </est-base-view></pre>	<pre>@Component({ selector: 'est-base-view', templateUrl: './est-base-view.component.html', styleUrls: ['./est-base-view.component.scss'] })</pre>
---	--

Figura 10. Modelo da tela base.

Na Figura 10, à esquerda pode-se ver como ficaria a declaração nos arquivos *HTMLS* das interfaces dos gráficos, e à direita nota-se a presença da estrutura de chamada padrão dos componentes, onde o *selector* define o nome com o qual o componente deve ser chamado, já o *templateUrl* por sua vez, é responsável por invocar o *template HTML* definido para esse item (nesse caso, *section*), por fim o *styleUrls* define os estilos associados ao *template HTML* do componente em questão.

Após realizar a declaração do componente base no *HTML* do gráfico, a próxima etapa é adicionar os elementos necessários na tela a ser desenvolvida, de acordo com o que o *est-base-view* aceita, pois dentro dessa *tag* é que são adicionadas as partes definidas previamente em sua estrutura, conforme a implementação do gráfico.

Para produzir a tela do primeiro gráfico, o gráfico de setores, as estruturas que são as responsáveis por conter os títulos, *inputs*, botões e o passo a passo foram inseridas dentro da declaração *est-base-view* no *HTML* do gráfico, por meio da chamada em *divs* dos seus *selects* correspondentes definidos na *section* como já observado na Figura 8. Com a inserção dessas estruturas finalizada, bastaram adicionar dentro delas os seus tipos de campos associados. Como seria necessário a utilização em alguns casos de muitos campos na região central, um elemento de barra de rolagem foi definida na confecção do arquivo de base *est-base-view*, para facilitar na organização visual dos campos de interações.

Por conseguinte, entrando no quesito das tabelas, as mesmas consistiram em utilizar tabelas do *Bootstrap*, pois estas propiciam reaproveitamento e capacidade criativa amplificada dos mesmos, sem contar no apoio que sua ampla documentação fornece. Na Figura 11 há um trecho de código *HTML* que simboliza os modelos de partida de todas as tabelas utilizadas no módulo.

```
<table class="table">
  <caption class="titleTable"><b>{{title}}</b></caption>
  <thead class="table-info" style="text-align: center;">
    <tr>
      <th scope="col">{{nomeVariavel}}</th>
      <th scope="col">Frequência (fi)</th>
    </tr>
  </thead>
  <tbody>
    <tr class="table-light" *ngFor="let element of varQual; let i = index" style="text-align: center;">
      <td>{{element}}</td>
      <td>{{quantidade[i]}}</td>
    </tr>
    <tr class="table-info" style="text-align: center;">
      <th scope="row">Total</th>
      <td>{{qtdTotal}}</td>
    </tr>
  </tbody>
  <caption>Fonte: {{fonteDados}}</caption>
</table>
```

Figura 11. Modelo de partida para criação de todas as tabelas.

O código dentro da *table HTML* se divide em dois principais setores, a *thead*, onde serão colocados e ajustados as colunas da tabela, ou seja quantas e quais serão os nomes dessas colunas, podendo ser nomes dinâmicos fornecidos pelo usuário ou estáticos. Por fim, o segundo setor, o *tbody*, contém todos os dados brutos fornecidos pelo usuário e transformados pelo código do *Typescript*, associados a uma das colunas descritas na *thead*. A organização das colunas e seus dados associados variam de acordo com o tipo de tabela necessária, ajustando os dois setores para tal.

O acesso às tabelas construídas na tela, ocorre através de um botão no canto inferior direito da região central logo abaixo do gráfico, intitulado “ver a tabela equivalente”, no momento do clique uma tela *modal* se abre no centro da tela com a tabela construída. O *modal* é uma espécie de janela que aparece como um elemento secundário (filho) no sistema para enfatizar algo contido ou requerido pelo seu elemento primário (pai), em outras palavras é uma subtela que aparece sempre no centro da página quando, nesse caso o botão é clicado (Bootstrap, 2020).

Na Figura 12, é ilustrado como foram estabelecidos no *HTML* os elementos de concepção da tabela, onde um botão genérico para chamada do *modal* foi criado para ser reaproveitado nas demais interfaces, ele se encontra na pasta de componentes compartilhados, já citada anteriormente. Para utilizá-lo basta chamar pelo seu *selector est-btn-grafico* e atribuir a ele dois parâmetros, o nome para colocar no botão, e o nome do *modal* a ser invocado na tela por ele.

```

<!-- Botão Tabela-->
<div class="btn-container mt-2 row align-items-center justify-content-center">
  <est-btn-grafico [name]='Ver Tabela Equivalente' [modalTarget]='#myModal'></est-btn-grafico>
</div>
<!-- Modal janela que vai aparecer-->
<div id="myModal" class="modal fade" role="dialog">
  <div class="modal-dialog" role="document">
    <!-- Modal content-->
    <div class="modal-content">
      <div class="modal-header justify-content-center">
      </div>
      <div class="modal-body">
        <table class="table">
          <!--Aqui vai o Conteúdo da Tabela-->
        </table>
      </div>
      <div class="modal-footer">
        <button type="button" class="btn btn-default" data-dismiss="modal">Fechar</button>
      </div>
    </div>
  </div>
</div>

```

Figura 12. Representa a estrutura completa que mostra as tabelas.

Após o término do documento *HTML*, a etapa final do desenvolvimento se volta ao código *Typescript*, arquivo responsável por aplicar toda a lógica por trás das interfaces, levando-a de estática para dinâmica e funcional. O aspecto que denota maior característica positiva do *IFMath*, as interações dinâmicas com os usuários pertencem a essa etapa, pois aqui são criadas as variáveis que trocam informações com o página *HTML* e realizam operações sobre eles até que se obtenha o resultado final desejado.

As interações com dados dinâmicos, se tornam possíveis graças à funcionalidade *Data Binding* fornecida pelo *Angular*, que propicia a comunicação entre *HTML* e *Typescript* usando uma variável em comum, ou seja quando uma variável é modificada no *Typescript* o valor no *HTML* muda, da mesma forma também acontece se o valor no *HTML* for alterado, o valor no *Typescript* será atualizado.

Utilizando essa propriedade, temos que para viabilizar a comunicação bidirecional comentada, a função *ngModel* é aplicada nos *inputs*, associando uma variável do *Typescript* a ele, como o mostrado acima, onde a variável *title* é declarada nessa função, e se houver algum conteúdo previamente inserido na variável pelo *Typescript*, o *input* é preenchido com esses valores.

Desta maneira, para chamar e mostrar os valores dos campos, em tabelas ou no passo a passo, as variáveis podem ser chamadas pela interpolação do *Angular*, bastando inserir no código *HTML* o seguinte trecho de código nomeDaVariável a ser utilizada adicionando duas chaves a esquerda e duas na direita. A Figura 13 representa a estrutura para a interação com dados dinâmicos em *inputs*. Mediante o exposto, depois de toda a página construída e seus devidos campos de variáveis e *inputs* associados, tanto na região central quanto no passo a passo, o desenvolvimento da tela do gráfico termina e o mesmo processo é aplicado nas demais restantes.

```

<div class="input-group mb-3">
  <div class="input-group-prepend">
    <label class="label">Título dos Dados: </label>
  </div>
  <input type="text" class="form-control" id="inputEmail3" placeholder="Título" [(ngModel)]="title"
    (ngModelChange)="updateChart()">
</div>

```

Figura 13. Representa estrutura para a interação com dados dinâmicos.

3.3.2. Medidas de Tendência

Em conformidade com a sequência de desenvolvimento estipulada, a última parte produzida do módulo de estatística descritiva correspondeu às Medidas de Tendência. Estando ciente da nova estrutura de pastas por conteúdo discutida e colocada em prática na subseção dos gráficos e tabelas, uma pasta para as medidas de tendência foi adicionada da mesma forma no diretório raiz na pasta *components* do módulo. Igualmente o nome dos componentes seguiu por definições semelhantes, onde no seu nome deve constar a sigla do módulo seguido do tipo de componente (medida), sigla da área de estudo (caso houver alguma específica) e por fim o nome do item a ser produzido, ou seja temos a mesma estrutura utilizada nos componentes dos gráficos, *siglaModulo-tipoComponente-areaEstudo-nomeItem*.

Partindo para a implementação das telas, os procedimentos tomados se assemelham quase inteiramente aos usados na criação dos gráficos, já que as telas partem de um modelo base já estabelecido (*est-base-view*), e que todos os componentes devem usufruir. As diferenças estão ligadas ao *layout* na presença de três botões centralizados, que são os responsáveis por direcionar o usuário às telas de medidas, para dados isolados, agrupados e em classes, ou seja, são botões de navegação.

```

<!-- Botões de navegação -->
<div class="btn-container mt-2 row align-items-center justify-content-center">
  <div class="col-12 text-center">
    <h3 class="text-center">Medidas</h3>
  </div>
  <h5 class="mt-2 col-12 text-center">Escolha abaixo a maneira de entrada de dados:</h5>
  <est-btn *ngFor="let button of buttons; let i = index" [name]="button.title" [route]="button.route"></est-btn>
  <br><br><br>
</div>

```

Figura 14. Representa a div que contém título, e os botões de navegação.

Observando a Figura 14, é possível notar a chamada do componente compartilhado o *est-btn*, que como vimos se dá a partir do seu *selector*, nesse caso o botão foi feito especialmente para os casos em que haja a necessidade de chamar rotas, e portanto levar o usuário a outras telas, definindo como parâmetros o nome do botão e a rota de acesso correspondente ao componente desejado, declarada no arquivo de *routing* do módulo.

Nas páginas presentes em medidas de tendência, vale ressaltar que em ambas, o item mostrado na Figura 14, deve estar presente, pois em todas o cabeçalho de navegação é o mesmo. Quanto aos campos de interação com o usuário, se diferenciam de acordo com os dados trabalhados: isolados, agrupados ou em classes. Da mesma forma funcionam as tabelas e o passo a passo, no caso das tabelas, ao invés de serem chamadas em um *modal*, aqui elas ficam fixas na tela, dentro das estruturas normais da *section*, variando

sua estrutura também de acordo com os conteúdos trabalhados.

Conforme pode ser visualizado na Figura 15 vale ressaltar a presença de dois *checks*, caixas de seleção contendo opções para a escolha, que irão influenciar nos tipos de cálculos mostrados e exemplificados no passo a passo. A partir da escolha de um deles, elementos do *HTML* são escondidos, pois ao clicar em uma dessas opções, uma função no *JavaScript* é invocada alterando o valor de uma variável responsável por determinar o tipo de passo a passo a ser mostrado, seja para dados envolvendo uma amostra ou população.

```
<div class="form-check">
  <input class="form-check-input" type="radio" name="exampleRadios" id="amostra" value="option1" checked
    (click)="setRadio(true)">
  <label class="form-check-label" for="amostra">
    Amostra
  </label>
</div>
<div class="form-check">
  <input class="form-check-input" type="radio" name="exampleRadios" id="populacao" value="option2"
    (click)="setRadio(false)">
  <label class="form-check-label" for="populacao">
    População
  </label>
</div>
```

Figura 15. Elementos de Check responsáveis por mudar a visualização do passo a passo.

Após a aplicação dos campos centrais e o passo a passo lateral usando o *est-base-view*, o desenvolvimento se encaminhou para a criação da lógica sobre a tela, onde são trabalhados os cálculos em médias, modas, medianas, desvio padrão, etc. Com a lógica da tela finalizada, basta adicionar as variáveis e os resultados obtidos no *Typescript* aos campos no *HTML*, especificamente nas tabelas e explicações envolvendo os cálculos, adotando os conceitos de *Data Binding*, interpolação para as tabelas e passo a passo, e utilização do *ngModel* em campos de interação real do usuário com o sistema.

4. Resultados

Como resultado desenvolveu-se o módulo do *IFMath* que trabalha com estatística descritiva, onde o usuário tem as opções de visualizar gráficos ou trabalhar com medidas de tendência, possibilitando uma nova ferramenta para professores e alunos com o intuito de auxiliar no ensino deste conteúdo.

Nesta seção serão apresentados os resultados do presente trabalho. Na Subseção 4.1 estão apresentados os resultados da implementação dos gráficos e tabelas do software e na Subseção 4.2 os resultados referentes às medidas de tendência. Já na Subseção 4.3 apresenta-se a validação do módulo de Estatística Descritiva desenvolvido.

4.1. Gráficos e Tabelas

A organização do *software* se dá através de abas. Sua navegação também pode ser realizada através de um menu, no lado esquerdo da tela. Como primeira parte do sistema, temos uma tela onde estão presentes algumas definições básicas sobre estatística e estatística descritiva, para servir como um embasamento para o aluno seguir no seu estudo, conforme podemos visualizar na Figura 16.

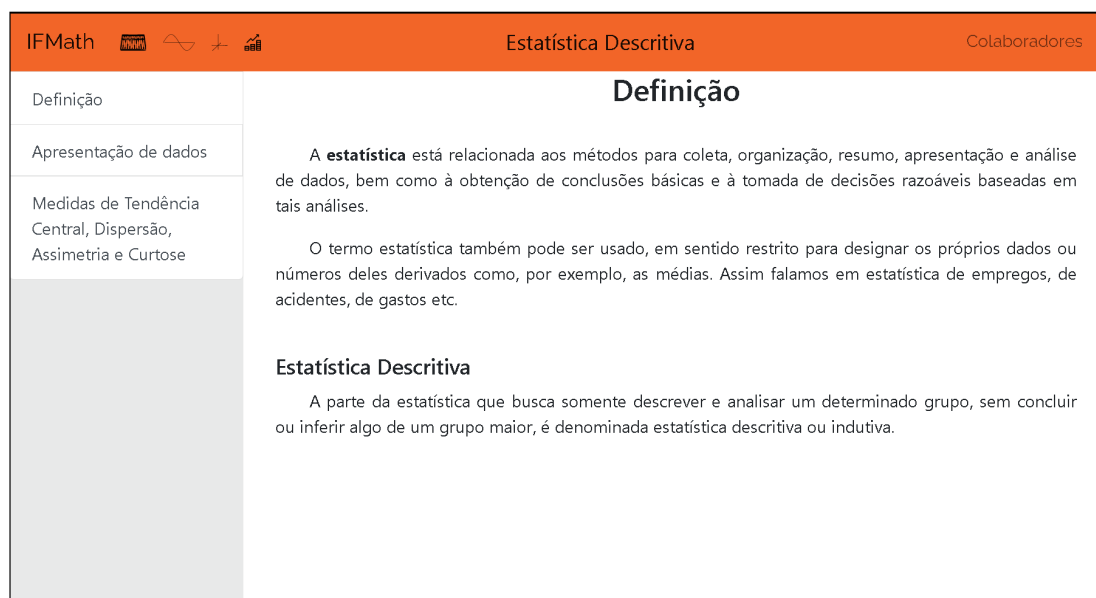


Figura 16. Tela contendo as definições básicas.

Na próxima aba denominada de “Apresentação de dados”, estão presentes conceitos teóricos referentes às maneiras em que os dados podem ser classificados e distribuídos, variáveis qualitativas e quantitativas, as quais já foram descritas neste artigo na Subseção 3.1. A partir desta tela o usuário pode ir para opção de visualização de gráficos.

Seguindo o fluxo para a parte de gráficos o usuário é redirecionado para uma tela onde estão presentes todos os tipos de gráficos, como pode ser visualizado na Figura 17, que podem ser construídos com a ferramenta, divididos nas seções de variáveis qualitativas e quantitativas.



Figura 17. Tela para a seleção dos gráficos.

Após escolher qual gráfico deseja construir o usuário é direcionado para uma tela

que contém os seguintes elementos: em sua parte central superior temos as entradas de dados, que podem variar para cada tipo de gráfico e de dados que o usuário inserir. Abaixo temos a construção do gráfico com todas as informações disponibilizadas e à direita temos uma explicação sobre o gráfico em questão, juntamente com o passo a passo de como este gráfico é construído, sendo possível expandir cada um dos elementos que são apresentados para ver o passo a passo da construção do mesmo.

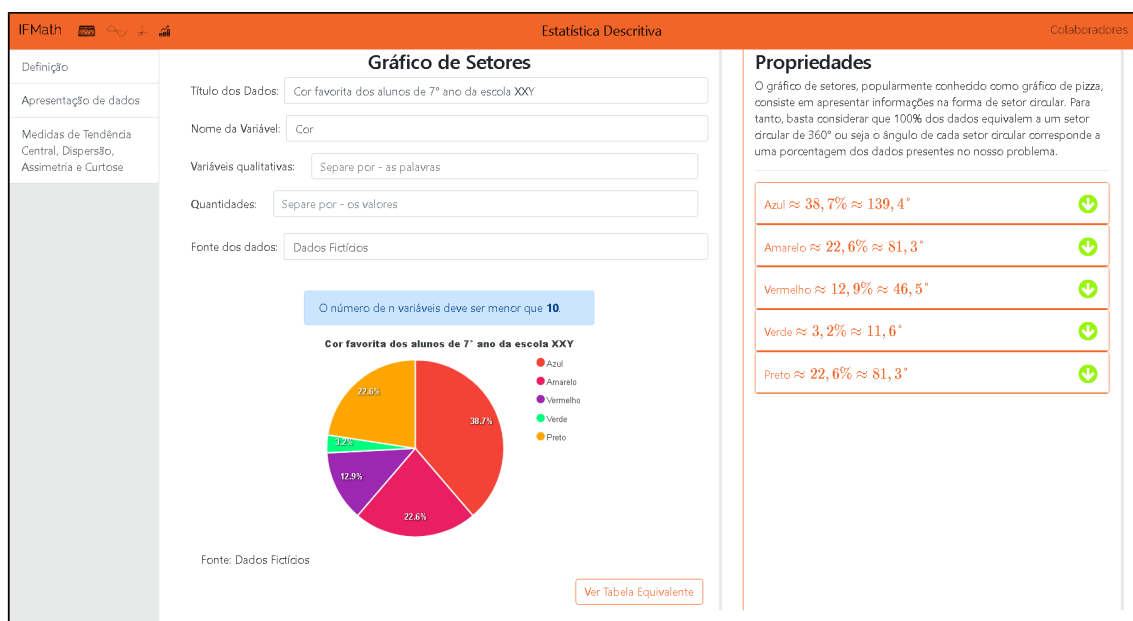


Figura 18. Exemplo de gráfico de setores com resultados.

Na Figura 18 é apresentado um gráfico de setores, popularmente conhecido como gráfico de pizza, onde é possível visualizar a inserção de dados na área central e o gráfico construído, também podemos visualizar na lateral direita a porcentagem que determina cada setor do gráfico. Após alterar os dados, o sistema atualiza o gráfico e de maneira iterativa nos retorna a resolução passo a passo tendo como base as regras para sua criação. Como pode ser observado, em cada operação envolvida na criação dos gráficos é possível visualizar uma explicação sobre como a operação em questão foi realizada.

Juntamente com a criação dos gráficos, pode-se visualizar uma tabela que apresenta os dados referentes ao gráfico criado. Esta tabela possui os seguintes elementos: título, corpo e fonte. Tal tabela é gerada a partir das informações fornecidas pelo usuário, na parte superior da tabela pode ser visualizado seu título. No corpo da tabela estão presentes as variáveis, as quantidades ou a frequência de cada variável e logo abaixo do corpo temos a fonte dos dados. Na Figura 19 pode-se visualizar um exemplo de tabela gerada pelo *software*.

Cor favorita dos alunos de 7º ano da escola XXY	
Cor	Frequência (fi)
Azul	12
Amarelo	7
Vermelho	4
Verde	1
Preto	7
Total	31

Fonte: Dados Fictícios

Fechar

Figura 19. Exemplo de tabela gerada.

Na Figura 20 pode-se visualizar os gráficos que foram abordados no tema de variáveis qualitativas neste trabalho. Lembrando que cada um deles possui as suas peculiaridades para construção e que todos seguiram o padrão descrito acima.

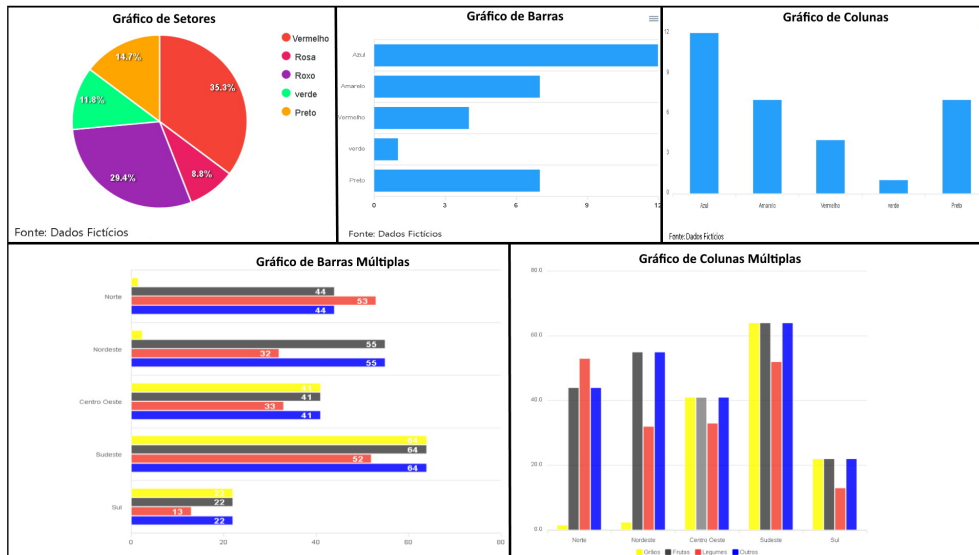


Figura 20. Todos os gráficos que podem ser gerados pelo sistema a partir de variáveis qualitativas.

Na Figura 21 pode-se visualizar os gráficos que podem ser trabalhados quando o usuário se refere à variáveis quantitativas.



Figura 21. Todos os gráficos que podem ser gerados pelo sistema a partir de variáveis quantitativas.

4.2. Medidas de Tendência

Após os gráficos temos o bloco voltado para os cálculos referentes às medidas de tendência, onde foi seguido o mesmo modelo de apresentação de resultados mostrado anteriormente na subseção dos gráficos. Neste bloco sua navegação se dá pelos botões presentes na parte superior da tela, logo após os botões na área central temos a entrada dos dados e o *display* da tabela gerada a partir dos dados de entrada. No centro da tela, temos uma tabela que representa os dados inseridos variando em cada um dos modos de entrada sendo entre eles: dados isolados, dados agrupados e dados agrupados em classes. E também em seu corpo está presente algumas informações pertinentes a cada modelo de entrada.

Na primeira tela, apresentada na Figura 22, temos os cálculos dos dados isolados, sendo este o mais simples dos três modelos, contendo a entrada do título, dos dados a serem trabalhados e a fonte de onde esses dados foram retirados. A tabela referente a esses dados apresenta título, em seu corpo temos os dados e sua frequência, caso esta tabela apresente mais de dez entradas ela passa a ser uma tabela distribuída em frequências, e após o corpo temos a informação sobre a fonte dos dados.

Dados Isolados

Escolha abaixo a maneira de entrada de dados:

☒ Dados Isolados ☐ Dados Agrupados ☐ Dados em Classes

Título dos Dados: Idade dos estudantes da turma XXy

Entre com os Dados: 7 - 8 - 7 - 9 - 7 - 6

Fonte dos dados: Dados Fictícios

Xi	fi
6	1
7	3
8	1
9	1
Soma	6

Fonte: Dados Fictícios

Medidas de Tendência Central

As medidas de tendência central são medidas que caracterizam um conjunto. Tais parâmetros estatísticos indicam valores representativos de um conjunto de dados.

Média: $\bar{x} \approx 7,333$

Moda: $Mo = 7$

Mediana: $Md = 7$

Medidas de Dispersão

As medidas de dispersão são medidas que caracterizam um conjunto de dados e são usadas para determinar a variedade de um conjunto de dados.

Amplitude Total: $A_t = 3$

Desvio Padrão: $s \approx 1,033$

Figura 22. Tela de entrada e saída dos Dados Isolados.

A segunda tela é referente ao cálculo dos dados distribuídos por frequências, nesta está presente um campo para a entrada das informações relacionadas sobre a frequência de cada dado, assim gerando uma tabela de frequências. Como pode ser visualizado na Figura 23.

Dados Agrupados

Escolha abaixo a maneira de entrada de dados:

☐ Dados Isolados ☒ Dados Agrupados ☐ Dados em Classes

Título dos Dados: Idade dos estudantes da turma XXy

Entre com os valores de xi: 6 - 7 - 8 - 9

Entre com os valores de fi: 1 - 3 - 1 - 1

Fonte dos dados: Dados Fictícios

O número de elementos em xi deve ser igual ao número de elementos em fi.

Xi	fi
6	1
7	3
8	1
9	1
Soma	6

Medidas de Tendência Central

As medidas de tendência central são medidas que caracterizam um conjunto. Tais parâmetros estatísticos indicam valores representativos de um conjunto de dados.

Média: $\bar{x} \approx 7,333$

Moda: $Mo = 7$

Mediana: $Md = 7$

Medidas de Dispersão

As medidas de dispersão são medidas que caracterizam um conjunto de dados e são usadas para determinar a variedade de um conjunto de dados.

Amplitude Total: $A_t = 3$

Desvio Padrão: $s \approx 1,033$

Variação: $s^2 \approx 1,067$

Figura 23. Tela de entrada e saída dos Dados Agrupados com frequência.

Por fim, temos a tela referente aos dados agrupados em classes, sendo esta a mais complexa pois sua entrada de dados é dada de uma maneira mais extensa, e apresenta informações extras em sua lateral direita relacionada à medidas de assimetria e curtose, além de cálculos como quartis e percentis que não estão presentes nas telas anteriores.

Em relação a entrada de dados, estão presentes os campos de título, frequência e fonte de dados. A grande diferença é que tem-se o *input* das informações referentes às classes, que são os limites inferiores e limites superiores.

A tabela também apresenta mudanças significativas, conforme pode-se visualizar na Figura 24, pois precisa acomodar mais informações, no seu corpo temos cinco colunas.

A coluna limites representa os limites inferiores e superiores de cada classe, X_i a média entre os limites, f_i a frequência de cada classe, F_i os dados acumulados em cada classe e as últimas duas $X_i * f_i$ e $X_i^2 * f_i$ são informações necessárias para os cálculos das estatísticas.

Dados em Classes

Escolha abaixo a maneira de entrada de dados:

☐ Dados Isolados ☐ Dados Agrupados ☐ Dados em Classes

Título dos Dados: Idade dos estudantes da turma XXy

Entre com os limites inferiores: 6 - 7 - 8 - 9

Entre com os limites superiores: 7 - 8 - 9 - 10

Entre com os valores de fi: 1 - 3 - 1 - 1

Fonte dos dados: Dados Fictícios

Limites	X_i	f_i	F_i	$x_i \cdot f_i$	$x_i^2 \cdot f_i$
6 - 7	6,5	1	1	6,5	42,25
7 - 8	7,5	3	4	22,5	168,75
8 - 9	8,5	1	5	8,5	72,25
9 - 10	9,5	1	6	9,5	90,25
Soma	-	6	-	47	373,5

Fonte: Dados Fictícios

Dados em Classes

☒ Amostra ☐ População

Medidas de Tendência Central

As medidas de tendência central são medidas que caracterizam um conjunto. Tais parâmetros estatísticos indicam valores representativos de um conjunto de dados.

Média: $\bar{x} \approx 7,833$

Moda: $Mo = 7,5$

Mediana: $Md = 7,667$

Quartis: $Q_1 = 7,167$ $Q_2 = 7,667$ $Q_3 = 8,5$

Percentis: $P_{10} = 6,6$ $P_{90} = 9,4$

Medidas de Dispersão

As medidas de dispersão são medidas que caracterizam um conjunto de dados e são usadas para determinar a variedade de um conjunto de dados.

Amplitude Total: $A_t = 4$

Figura 24. Tela de entrada e saída dos Dados em Classes.

Todos os cálculos de estatísticas são apresentados passo a passo, caso o usuário queira visualizá-los basta clicar na seta à direita da respectiva medida de tendência. Na Figura 25 há um exemplo de passo a passo, é importante observar que conforme a entrada de dados é alterada, esses passos são ajustados de maneira dinâmica.

Medidas de Tendência Central

As medidas de tendência central são medidas que caracterizam um conjunto. Tais parâmetros estatísticos indicam valores representativos de um conjunto de dados.

Média: $\bar{x} \approx 7,333$

$\sum x_i \cdot f_i = 44$

Primeiramente calcula-se o somatório de $\sum x_i \cdot f_i$ que é dado por $6,1 + 7,3 + 8,1 + 9,1$

$\bar{x} = \frac{44}{6}$

Substituindo os dados na fórmula: $\bar{x} = \frac{\sum x_i \cdot f_i}{\sum f_i}$

$\bar{x} = 7,333$

Resultado final após a divisão.

Figura 25. Exemplo de resolução passo a passo de uma operação de média.

4.3. Validação do Software

Após a conclusão do desenvolvimento do *software*, foram realizados testes sobre o mesmo com docentes da disciplina de Matemática e Estatística de diferentes redes de ensino e com discentes da disciplina de Estatística nos cursos técnicos do Instituto Federal de Santa Catarina - Câmpus Lages. Os testes foram aplicados em todos os níveis de ensino como já mencionado anteriormente. Este teste acontece de maneira prática onde o usuário realiza interações com o sistema de maneira que seja possível verificar os resultados apresentados pelo *software*.

Após a realização dos testes práticos foi disponibilizado um formulário *online*, que consta no apêndice deste trabalho, onde foi feita a captação de maneira anônima das respostas e opiniões dos usuários, solicitando o consentimento dos usuários para a participação nesta pesquisa, a qual contou com 15 participantes. Neste foi avaliada a qualidade do *software* como uma ferramenta para auxiliar no processo de ensino-aprendizagem, a qualidade dos conteúdos presentes, a qualidade visual adotada e a facilidade de utilização. A partir dos resultados coletados é possível fazer alterações no *software* para que ocorra uma melhor adesão por parte dos usuários (alunos e professores).

Na Figura 26 são apresentados resultados referentes à qualidade da interface do *software* e se está adequada para estudos dos conteúdos abordados neste trabalho. O gráfico mostra que cerca de 66,7% dos usuários acharam os aspectos visuais excelentes, enquanto 33,3% os classificaram como bom. Em sua totalidade as respostas obtidas indicam uma percepção positiva por parte dos usuários quanto ao *layout* criado.



Figura 26. Gráfico de qualidade da interface.

A Figura 27, trata das respostas quanto ao nível de qualidade que as interações entre usuário e *software* proporcionaram, obtendo um percentual de 60% para excelente e outros 40% para bom.

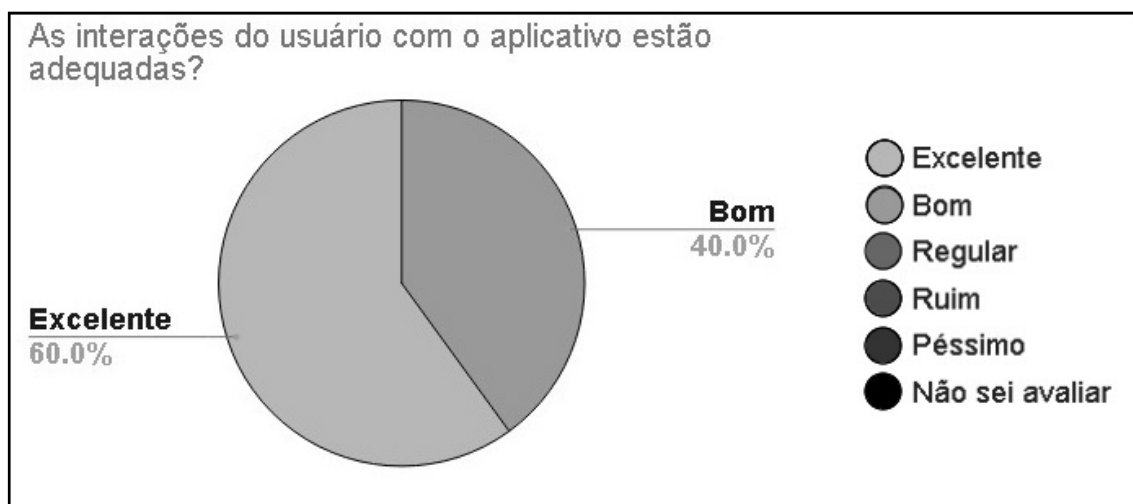


Figura 27. Gráfico sobre a qualidade das interações com o usuário.

Na Figura 28, cerca de 86,7% dos participantes indicaram ser excelentes ou boas e de fácil entendimento as informações para a criação de gráficos utilizando o *software*, ocupando apenas 13,3% os que elencaram o nível de informações como regular, assinalando a presença de informações relevantes e de fácil compreensão pela maioria dos docentes e alunos.



Figura 28. Gráfico referente ao nível de informações para a criação dos gráficos.

O gráfico abaixo, representado pela Figura 29, mede a satisfação que os gráficos ao serem gerados deram ao usuários, onde 93,3% dos indivíduos concordaram como sendo excelentes e 6,7% como bom. Sendo assim, temos que os gráficos construídos sobre a tela, agradaram a todos os indivíduos da pesquisa.



Figura 29. Resultado da satisfação dos gráficos gerados.

Quanto ao nível de satisfação aos usuários que a geração das tabelas obtiveram, podem ser representadas pela Figura 30 logo abaixo, totalizando 80% em excelente e 20% como bom.

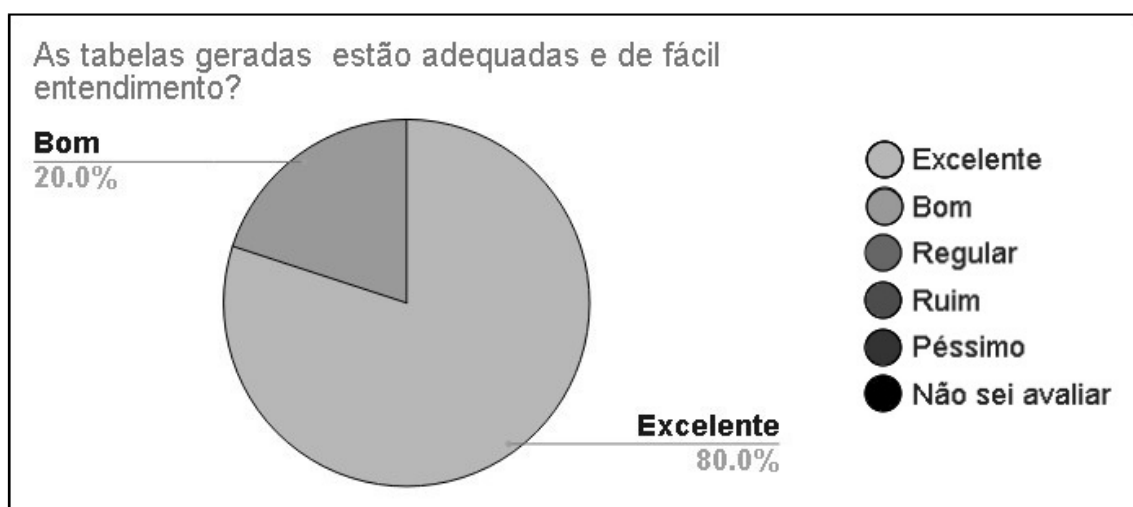


Figura 30. Resultado da satisfação das tabelas geradas.

Em relação a profundidade e qualidade dos temas apresentados no *software*, os usuários foram quase unânimes na escolha classificando como adequado totalizando 93,3%, restando um percentual de 6,7% indicando que deveria se aprofundar mais nos conteúdos abordados, como mostra a Figura 31.



Figura 31. Profundidade dos temas abordados.

Por fim, a última pergunta a respeito do que foi desenvolvido se refere a importância dos conteúdos abordados para o ensino de estatística descritiva e se eles realmente são relevantes na aprendizagem, como pode ser verificado na Figura 32, relatando que 86,7% das pessoas consideraram excelente os conteúdos utilizados, o que mostra o acerto na escolha das matérias abordadas.



Figura 32. Nível de importância dos conteúdos abordados.

5. Considerações Finais

O referido trabalho apresentou a construção de um *software* destinado ao auxílio no ensino e aprendizagem de estatística descritiva, por meio da aplicação de conteúdos teóricos e interativos com os usuários através do computador. Ao empregar a tecnologia em áreas educacionais, essa passa a se tornar forte aliada no processo de ensino-aprendizagem, ampliando as possíveis abordagens nos conteúdos.

A estatística descritiva compreende uma área importante de introdução aos aspectos centrais da disciplina de Estatística, sendo assim a partir da criação e desenvolvimento

do módulo de estatística descritiva no *IFMath* os usuários, sejam eles alunos ou professores, tem em mãos uma ferramenta que possibilita a interação com dados dinâmicos. Isso quer dizer que eles podem inserir seus próprios dados e criar novos gráficos e tabelas estatísticas a partir deles, além de possuir as devidas explicações técnicas de como chegar nesses valores apresentados pelo sistema.

Ao longo deste trabalho foram realizados levantamentos dos conteúdos a serem incorporados no *software* para garantir a inclusão de aspectos técnicos mais incisivos e simples a respeito de estatística descritiva. Para isso, pesquisas a respeito dos temas a serem trabalhados foram feitas, buscando em artigos, livros e *sites* que serviram como alicerce na extração do conhecimento necessário para viabilizar a construção das páginas do sistema, que seguiram os modelos de *layout* anteriormente definidos nos demais módulos do *IFMath*.

As avaliações de usabilidade sobre o *software* foram realizadas com alunos, professores de matemática e estatística, resultando em um *feedback* positivo a respeito da ferramenta de ensino e aprendizagem desenvolvida, pois em todas as perguntas da pesquisa a classificação que mais prevaleceu foi a de excelente, correspondendo a 60% ou mais das respostas. Com base na boa avaliação obtida foi possível perceber o nível de qualidade que os usuários sentiram ao interagir com as funcionalidades e conteúdos propostos. Com o término da validação pode-se constatar que o trabalho atingiu com êxito todos os objetivos propostos inicialmente.

Em relação aos trabalhos futuros, vale ressaltar a possibilidade de adicionar correções que melhorem a experiência do usuário com o módulo de estatística descritiva, podendo utilizar o *feedback* deixado pelos docentes durante a fase de avaliação, onde sugeriram possíveis melhorias no *software*. Cabe mencionar também a possível publicação do trabalho em revistas e eventos educacionais de modo a disseminar os resultados obtidos. Além de oferecer apoio na continuidade do desenvolvimento do módulo de estatística do *IFMath* em especial na criação do módulo de estatística inferencial, área da estatística que se preocupa em realizar a estimação de parâmetros e também em testar hipóteses estatísticas.

Referências

- Alves, W. P. (2014). *Projetos de Sistemas Web: Conceitos, Estruturas, Criação de Banco de Dados e Ferramentas de Desenvolvimento*. Editora Érica, 1. edition.
- Angular (2020). Introduction to angular concepts. <https://angular.io/guide/architecture/>. [Acessado: 10-08-2020].
- ApexCharts (2020). What is apexcharts. <https://apexcharts.com/>. [Acessado: 28-07-2020].
- Avila, A. J., de Oliveira Junior, E. G., Neto, W. C. B., e Durigon, A. (2018). *Sistema especialista para o ensino-aprendizagem de expressões algébricas. TCCs do Curso de Ciência da Computação, IFSC, Câmpus Lages*.
- Barbetta, P. A., Reis, M. M., e Bornia, A. C. (2018). *Estatística para cursos de engenharia e informática*. Atlas, 3. edition.
- Booth, J. D. (2017). *Angular 2 Succinctly*. Sync-fusion.
- Bootstrap (2020). Introduction. <https://getbootstrap.com/docs/4.5/getting-started/introduction/>. [Acessado: 26-07-2020].

- Borba, M., Silva, R., e Gadanidis, G. (2018). *Fases das tecnologias em Educação Matemática: Sala de aula e internet em movimento*. Autêntica, 2. edition.
- Campos, C. R., Wodewotzki, M. L. L., e Jacobini, O. R. (2013). *Educação Estatística: teoria e prática em ambientes de modelagem matemática*. Autêntica Editora, 2. edition.
- Dalfovo, M. S., Adilson, R., e Silveira, L. A. (2008). Métodos quantitativos e qualitativos: um resgate teórico. <https://docplayer.com.br/5035727-Metodos-quantitativos-e-qualitativos-um-resgate-teorico.html>. [Acessado: 13-08-2020].
- GeoGebra (2020). Materiais didáticos. <https://www.geogebra.org/materiais>. [Acessado: 01-06-2020].
- Ignácio, S. A. (2010). *Importância da estatística para o processo de conhecimento e tomada de decisão*. Revista Paranaense de Desenvolvimento.
- INEP (2018). Resultados do ENEM 2018 são divulgados. http://portal.inep.gov.br/artigo/-/asset_publisher/B4AQV9zFY7Bv/content/resultados-do-enem-2018-sao-divulgados/21206. [Acessado: 13-03-2020].
- Jakobus, B. e Marah, J. (2016). *Mastering Bootstrap 4*. Packt.
- LaTeX (2020). A document preparation system. <https://www.latex-project.org>. [Acessado: 02-08-2020].
- Loundon, K. (2010). *Desenvolvimento de Grandes Aplicações Web*. Novatec Editora Ltda.
- Macedo, N. D. (1995). *Iniciação a Pesquisa Bibliográfica*. Unimarco, 2. edition.
- Madruga, I. C. T., Godoi, F. A., Durigon, A., e Junior, V. H. (2018). *Desenvolvimento de uma ferramenta para o estudo de geometria plana. TCCs do Curso de Ciência da Computação, IFSC, Câmpus Lages*.
- McGrath, M. (2010). *HTML, CSS JavaScript*. In Easy Steps Limited.
- Messenlehner, B. e Coleman, J. (2014). *Criando Aplicações Web com WordPress: WordPress como um Framework de Aplicações*. Novatec Editora Ltda.
- Minitab (2020). Powerful statistical software everyone can use. <https://www.minitab.com/en-us/products/minitab/>. [Acessado: 02-06-2020].
- Nakazawa, C. A. e Marafon, M. J. (2003). Sesstatnet - ensino de estatística mediado por computador. <https://repositorio.ufsc.br/handle/123456789/183907>. [Acessado: 05-05-2020].
- npm (2020). ng-katex. <https://www.npmjs.com/package/ng-katex/>. [Acessado: 01-08-2020].
- Pressman, R. S. (2011). *Engenharia de Software Uma Abordagem Profissional*. AMGH editora Ltda, 7. edition.
- Rcommander (2020). What is R? <http://www.R-project.org/>. [Acessado: 20-06-2020].
- Scheidt, F. A. (2015). *Fundamentos de CSS: criando design para sistemas web*. Outbox Interativa.
- Scilab (2020). Tutorials. <https://www.scilab.org/tutorials>. [Acessado: 05-06-2020].
- Singh, A. (2018). *Angular 2 Interview Questions and Answers: Including Angular 6,5,4 and 2*. BPB Publications, 1. edition.
- Snege, D. (2018). *Bootstrap: Guia para iniciantes (Programação Livro 1)*. Amazon Servicos de Varejo do Brasil Ltda, 1. edition.
- Spiegel, M. R. e Stephens, L. J. (2009). *Estatística*. Bookman, 4. edition.

- Visual Studio Code (2020). Introduction. <https://code.visualstudio.com/docs/>. [Acessado: 25-07-2020].
- Zabot, D. e Matos, E. (2020). *Aplicativos com Bootstrap e Angular: Como Desenvolver Apps Responsivos*. Editora Érica, 1. edition.
- Zanella, L. C. H. (2013). *Metodologia de Pesquisa* . Universidade Federal de Santa Catarina, 2. edition. [Acessado: 18-05-2020].