



Universidade de Brasília

Instituto de Ciências Exatas  
Departamento de Ciência da Computação

**Observatório de Dados Educacionais: Um sistema  
para as disciplinas de Programação do Departamento  
de Ciência da Computação da Universidade de  
Brasília**

Aécio F. G. Magalhães

Monografia apresentada como requisito parcial  
para conclusão do Curso de Engenharia da Computação

Orientadora  
Prof.a Dr.a Maristela Terto de Holanda

Brasília  
2021



# Dedicatória

Dedico este trabalho a minha família e aos meus amigos, que estiveram do meu lado durante todo o percurso da graduação, nos momentos fáceis e nos difíceis. Sem eles, eu não haveria chegado até aqui.

# Agradecimentos

A minha família, que me deu o suporte e encorajamento necessários para o meu desempenho.

À Prof.<sup>a</sup> Dr.<sup>a</sup> Maristela Terto de Holanda, pela oportunidade deste trabalho e pelo incentivo e apoio durante a orientação.

À Luiza Aguiar Hansen, pela colaboração com uma parte de seu trabalho desenvolvido no mestrado e por todo apoio em algumas dificuldades.

A todos os demais professores que fizeram parte da minha graduação, contribuindo para a minha formação.

# Resumo

O ensino da primeira linguagem de programação tem sido estudado ao longo do tempo e é um desafio para alguns alunos ao entrar na Universidade. As disciplinas introdutórias de programação da Universidade de Brasília apresentam altos índices de reprovação em cursos como Licenciatura em Computação (51,4% em Algoritmos e Programação de Computadores) e Matemática (46,1% em Introdução à Ciência da Computação para o período do primeiro semestre de 2016 ao segundo semestre de 2019). Em alguns cenários, é perceptível uma maior dificuldade para alunos pertencentes ao sistema de cotas, evidenciado pelo aumento na taxa de reprovação para o curso de Matemática, que torna-se 47,6%. Este contexto pode ser melhor investigado por meio de técnicas de visualização de dados.

Este trabalho apresenta o Observatório de Dados Educacionais, um sistema desenvolvido para exibir gráficos baseados em diferentes algoritmos de visualização do histórico das turmas de disciplinas introdutórias de programação do Departamento de Ciência da Computação da Universidade de Brasília. O sistema pode ser utilizado por professores e coordenadores dos cursos para reconhecer o perfil dos alunos, visando dimensionar o problema das reprovações e auxiliar na tomada de decisão.

**Palavras-chave:** Visualização de Dados Educacionais, Ciência da Computação, Programação

# Abstract

Teaching the first programming language has been studied over time and is a challenge for some students when entering University. The introductory programming courses at the University of Brasília have high failure rates in courses such as Bachelor's Degree in Education in Computer Science (51.4% in Algorithms and Computer Programming) and Mathematics (46.1% in Introduction to Computer Science for the period from the first semester of 2016 to the second semester of 2019). In some scenarios, it is noticeable a greater difficulty for students belonging to the quota system, evidenced by the increase in the failure rate for the Mathematics course, which becomes 47.6%. This context can be investigated through data visualization techniques.

This work presents the "Observatório de Dados Educacionais", a system developed to display graphs based on different algorithms for visualizing the history of classes in introductory programming courses at the Department of Computer Science at the University of Brasília. The system can be used by professors and course coordinators to recognize the profile of students, aiming to scale the problem of failure and assist in decision making.

**Keywords:** Educational Data Visualization, Computer Science, Programming

# Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Objetivos . . . . .	2
1.1.1	Objetivo Geral . . . . .	2
1.1.2	Objetivos Específicos . . . . .	2
1.2	Estrutura do Trabalho . . . . .	2
<b>2</b>	<b>Referencial Teórico</b>	<b>4</b>
2.1	Ensino da Primeira Linguagem de Programação . . . . .	4
2.2	Visualização de Dados Educacionais . . . . .	5
2.2.1	Algoritmos de Visualização de Dados . . . . .	6
2.3	Tecnologias utilizadas . . . . .	10
<b>3</b>	<b>Observatório de Dados Educacionais</b>	<b>13</b>
3.1	Processo de Desenvolvimento do Observatório . . . . .	13
3.1.1	Gráficos . . . . .	13
3.1.2	Protótipo e Requisitos . . . . .	14
3.1.3	Gráficos Definitivos . . . . .	19
3.2	Arquitetura do Sistema . . . . .	21
3.2.1	Django e Arquitetura <i>Model-View-Template</i> . . . . .	21
3.2.2	Sistema de Filtros . . . . .	21
3.2.3	Autenticação . . . . .	22
3.2.4	Problemas Identificados . . . . .	25
<b>4</b>	<b>Análise de Resultados</b>	<b>26</b>
4.1	Interface do Observatório . . . . .	26
4.1.1	Página inicial . . . . .	26
4.1.2	Navegação . . . . .	27
4.1.3	Autenticação . . . . .	27
4.1.4	Página de Informações Gerais . . . . .	28
4.1.5	Página de Informações Acadêmicas . . . . .	30

4.1.6	Página de Informações de Percepção . . . . .	33
4.1.7	Página de Gráficos Interativos . . . . .	33
<b>5</b>	<b>Conclusão</b>	<b>38</b>
	<b>Referências</b>	<b>40</b>
	<b>Apêndice</b>	<b>41</b>
<b>A</b>	<b>Figuras Complementares de Gráficos Interativos</b>	<b>42</b>



# Lista de Figuras

2.1	Exemplo de Gráfico de pizza. . . . .	7
2.2	Exemplo de Gráfico de barras. . . . .	7
2.3	Exemplo de Gráfico de radar. . . . .	8
2.4	Exemplo de Gráfico de linhas. . . . .	8
2.5	Exemplo de Gráfico de área. . . . .	8
2.6	Exemplo de Gráfico de caixa. . . . .	8
2.7	Exemplo de Gráfico de dispersão. . . . .	8
2.8	Exemplo de Gráfico de violino. . . . .	8
2.9	Exemplo de Gráfico de coordenadas paralelas. . . . .	9
2.10	Exemplo de Diagrama de sankey. . . . .	9
2.11	Exemplo de Gráfico treemap. . . . .	9
2.12	Estrutura de arquivos do projeto no Atom. . . . .	11
3.1	Protótipo da interface da página de Informações Gerais. . . . .	16
3.2	Protótipo da interface da página de Informações Acadêmicas. . . . .	16
3.3	Protótipo da interface da página de Informações de Percepção. . . . .	17
3.4	Modelo do processo de operação de um usuário comum do sistema. . . . .	17
3.5	Modelo do processo de operação de um usuário administrador do sistema. . . . .	17
3.6	Modelo do banco de dados relacional utilizando informações do SIGRA. Fonte: [1]. . . . .	18
3.7	Perspectiva MVT do Django (adaptada) [2]. . . . .	22
3.8	Resultado dos filtros na <i>model</i> de usuário. . . . .	23
3.9	Novo perfil cadastrado pelo usuário, destacado na cor verde. . . . .	24
3.10	Gerenciamento de grupos de usuários disponíveis no Observatório. . . . .	24
4.1	Página Inicial do Observatório. . . . .	26
4.2	Menu Lateral para Navegação no Sistema. . . . .	27
4.3	Página de Cadastro de Usuário. . . . .	28
4.4	Gráficos em Informações Gerais para a disciplina de APC. . . . .	29
4.5	Gráficos em Informações Gerais para a disciplina de APC. . . . .	29

4.6	Gráficos Filtrados em Informações Gerais para a disciplina de APC. . . . .	30
4.7	Gráficos em Informações Acadêmicas para a disciplina de ICC. . . . .	31
4.8	Gráficos em Informações Acadêmicas para a disciplina de ICC. . . . .	31
4.9	Gráficos Filtrados em Informações Acadêmicas para a disciplina de ICC. . . . .	32
4.10	Gráficos em Informações Acadêmicas para a disciplina de APC. . . . .	32
4.11	Gráficos em Informações de Percepção para a Disciplina de APC. . . . .	33
4.12	Gráficos Interativos para a Disciplina de APC - Departamento de Ciência da Computação e Engenharia Mecatrônica. . . . .	35
4.13	Gráficos Interativos para a Disciplina de APC - Departamento de Ciência da Computação e Engenharia Mecatrônica. . . . .	35
4.14	Gráficos Interativos para a Disciplina de APC - Departamento de Ciência da Computação e Engenharia Mecatrônica. . . . .	36
4.15	Gráficos Interativos para a Disciplina de APC - Departamento de Ciência da Computação e Engenharia Mecatrônica. . . . .	36
4.16	Gráficos Interativos para a Disciplina de APC - Departamento de Ciência da Computação e Engenharia Mecatrônica. . . . .	37
4.17	Gráficos Interativos para a Disciplina de APC - Departamento de Ciência da Computação e Engenharia Mecatrônica. . . . .	37
A.1	Gráficos Filtrados em Informações Acadêmicas para a disciplina de APC. . . . .	42
A.2	Gráficos Filtrados em Informações Acadêmicas para a disciplina de APC. . . . .	43
A.3	Gráficos Filtrados em Informações Acadêmicas para a disciplina de APC. . . . .	43
A.4	Gráficos Interativos para a Disciplina de APC - Curso de Ciência da Computação. . . . .	44
A.5	Gráficos Interativos para a Disciplina de APC - Curso de Ciência da Computação. . . . .	44
A.6	Gráficos Interativos para a Disciplina de APC - Curso de Ciência da Computação. . . . .	45
A.7	Gráficos Interativos para a Disciplina de APC - Curso de Ciência da Computação. . . . .	45
A.8	Gráficos Interativos para a Disciplina de APC - Curso de Ciência da Computação. . . . .	46
A.9	Gráficos Interativos para a Disciplina de APC - Curso de Ciência da Computação. . . . .	46
A.10	Gráficos Interativos para a Disciplina de APC - Curso de Computação (Licenciatura). . . . .	47
A.11	Gráficos Interativos para a Disciplina de APC - Curso de Computação (Licenciatura). . . . .	47

A.12 Gráficos Interativos para a Disciplina de APC - Curso de Computação (Licenciatura). . . . .	48
A.13 Gráficos Interativos para a Disciplina de APC - Curso de Computação (Licenciatura). . . . .	48
A.14 Gráficos Interativos para a Disciplina de APC - Curso de Computação (Licenciatura). . . . .	49
A.15 Gráficos Interativos para a Disciplina de APC - Curso de Computação (Licenciatura). . . . .	49
A.16 Gráficos Interativos para a Disciplina de APC - Curso de Engenharia da Computação. . . . .	50
A.17 Gráficos Interativos para a Disciplina de APC - Curso de Engenharia da Computação. . . . .	50
A.18 Gráficos Interativos para a Disciplina de APC - Curso de Engenharia da Computação. . . . .	51
A.19 Gráficos Interativos para a Disciplina de APC - Curso de Engenharia da Computação. . . . .	51
A.20 Gráficos Interativos para a Disciplina de APC - Curso de Engenharia da Computação. . . . .	52
A.21 Gráficos Interativos para a Disciplina de APC - Curso de Engenharia da Computação. . . . .	52
A.22 Gráficos Interativos para a Disciplina de APC - Curso de Engenharia Mecatrônica. . . . .	53
A.23 Gráficos Interativos para a Disciplina de APC - Curso de Engenharia Mecatrônica. . . . .	53
A.24 Gráficos Interativos para a Disciplina de APC - Curso de Engenharia Mecatrônica. . . . .	54
A.25 Gráficos Interativos para a Disciplina de APC - Curso de Engenharia Mecatrônica. . . . .	54
A.26 Gráficos Interativos para a Disciplina de APC - Curso de Engenharia Mecatrônica. . . . .	55
A.27 Gráficos Interativos para a Disciplina de APC - Curso de Engenharia Mecatrônica. . . . .	55

# Lista de Tabelas

2.1 Análise de cenário dos algoritmos de visualização estudados. Fonte: adaptado de Hansen, L. (p. 32) [1]. . . . .	10
4.1 Mapeamento de Notas para Menções. . . . .	31
4.2 Detalhamento das Colunas dos Gráficos. . . . .	34

# Capítulo 1

## Introdução

Os cursos do Departamento de Ciência da Computação da Universidade de Brasília apresentam uma alta taxa de desistência [3, 4]. Este problema está relacionado com diversos fatores, entre eles, as reprovações em disciplinas. Entre os anos de 2000 e 2013, destacou-se a disciplina de Computação Básica (CB), disciplina introdutória de programação, como uma das disciplinas com maior índice de reprovação nos cursos de Computação da UnB [4].

Segundo Beaubouef e Mason [5], diversos fatores podem influenciar nas taxas de evasão dos estudantes do curso de Ciência da Computação, dentre elas:

- Pouca habilidade matemática e pouca habilidade de resolver problemas: segundo os autores, essa é uma habilidade que pode ser ensinada;
- Pouca prática / pouco *Feedback*: alguns professores escolhem abordagens diferentes no que se refere às atividades e ao *feedback* para o aluno. Retornos automáticos para as atividades dos alunos muitas vezes não costumam gerar informações significativas - por isso, alguns professores acabam optando por elaborar seu próprio *feedback*, permitindo a identificação de áreas em que o estudante precisa de mais atenção;
- Laboratórios mal projetados: é citado pelos autores o trabalho de Walker [6], alegando que muitas instituições não utilizam seus laboratórios de programação de uma apropriada e que, muitas vezes, as interações são mal distribuídas quando trata-se da atenção do professor para a turma.

A implementação de ferramentas para visualização de dados já é objeto de estudo e seus resultados, dada a priorização de dados estratégicos dos alunos em questão, mostram a presença de indicadores que podem ser utilizados no auxílio para detectar o risco de evasão [7].

## 1.1 Objetivos

### 1.1.1 Objetivo Geral

Este trabalho apresenta um sistema *web* - Observatório de Dados Educacionais -, com o objetivo de disponibilizar para professores diversos gráficos que refletem o desempenho das turmas das matérias introdutórias de programação do Departamento de Ciência da Computação da Universidade de Brasília - Algoritmos e Programação de Computadores, Computação Básica e Introdução à Ciência da Computação. Sua elaboração foi realizada de maneira integrada com o trabalho de mestrado da aluna Luiza Aguiar Hansen [1], responsável pela escolha dos algoritmos a serem utilizados no sistema desenvolvido no presente trabalho.

### 1.1.2 Objetivos Específicos

Para atingir o objetivo geral do trabalho, foi necessário o cumprimento dos seguintes objetivos específicos:

- Identificar um *framework* que permita executar algoritmos de gráficos desenvolvidos na linguagem Python;
- Desenvolver um sistema *web* com o *framework* proposto;
- Implementar o banco de dados desenvolvido em [1] a ser utilizado pelo sistema;
- Implementar os algoritmos de visualização de dados educacionais desenvolvidos em [1] no sistema;
- Desenvolver uma interface para o sistema que permita filtrar os dados por determinada disciplina, curso, gênero, presença ou não de cotas e/ou período.

## 1.2 Estrutura do Trabalho

Este trabalho é composto pelos seguintes capítulos:

- Capítulo 2 - Referencial Teórico: apresentação dos conceitos envolvidos no desenvolvimento deste trabalho. Neste capítulo são abordados: Ensino da Primeira Linguagem de Programação, Visualização de Dados Educacionais e Tecnologias utilizadas.
- Capítulo 3 - Observatório de Dados Educacionais: apresentação do processo de desenvolvimento do Observatório, bem como sua arquitetura.

- Capítulo 4 - Análise de resultados: apresentação de telas do sistema e discussão dos gráficos.
- Capítulo 5 - Conclusão: apresentação das conclusões.

# Capítulo 2

## Referencial Teórico

Neste capítulo são apresentados os conceitos envolvidos no desenvolvimento deste trabalho, estruturados nas seguintes seções: Ensino da Primeira Linguagem de Programação (Seção 2.1), Visualização de Dados Educacionais (Seção 2.2) e Tecnologias utilizadas (Seção 2.3).

### 2.1 Ensino da Primeira Linguagem de Programação

A primeira disciplina de ensino de programação é objeto de estudo há décadas. Luxton-Reilly et al. [8] classificou um total de 1666 artigos neste contexto de introdução à programação em grupos e subgrupos a partir dos dados extraídos de cada um. Dentre estes grupos, 489 artigos faziam parte do grupo "O estudante", onde, dentre os subgrupos que o compunham, destaca-se "grupos sub-representados".

A representação de estudantes do gênero feminino em cursos de graduação na área da computação tem se mantido baixa nas últimas décadas - no Brasil, somente 17% dos concluintes de cursos relacionados à computação entre 2000 e 2013 foram do sexo feminino [9]. Com a aplicação de diversos estudos, é percebido que um fator bastante influente no desempenho de grupos que fazem parte de uma minoria é a autoconfiança, característica que encontra-se baixa diversos estudantes do gênero feminino [8, 10].

Becker [11] analisa 481 artigos a respeito da matéria de programação introdutória - Computer Science 1 -, apresentados durante os primeiros 50 anos do Simpósio Técnico SIGCSE (Special Interest Group Computer Science Education). Estes artigos foram separados em diversas categorias, explicitando os assuntos relacionados (forma de ensino, perfil dos alunos, primeiras linguagens, entre outros) no estudo da matéria introdutória. Dentre os 481 artigos, 12 aprofundavam o estudo da programação introdutória com viés em diversidade de gênero, acessibilidade e/ou inclusão, sendo 5 artigos da década de 2000 e 7 artigos da década de 2010 [11].



No contexto do Brasil, foi implementada em agosto de 2012 a Lei Nº 12.711 [12] (Lei de Cotas) que, para as 59 universidades federais, reserva 50% das vagas dos concursos seletivos para estudantes que cursaram o ensino médio integralmente em escolas públicas e, destas, 50% são destinadas a estudantes de famílias com renda igual ou inferior a 1,5 salário-mínimo per capita. Essas vagas reservadas devem ser preenchidas por alunos autodeclarados pretos, pardos e indígenas e por pessoas com deficiência. Em 2016, a implementação da Lei atingiu a reserva de 50% das vagas em cada concurso seletivo para ingresso nos cursos de graduação, por curso e turno.

Ainda acerca do perfil dos alunos nas disciplinas introdutórias de programação, foi aplicado um questionário nas turmas de Algoritmos e Programação de Computadores e Introdução à Ciência da Computação do segundo semestre de 2019, da Universidade de Brasília, com o intuito de analisar diferenças quanto à percepção de eficácia pessoal de alunos que ingressaram na universidade através do sistema de cotas em relação à alunos que não utilizaram desse sistema. Apesar de ser avaliada em uma amostra pequena, é perceptível uma reação mais negativa por parte dos estudantes com cotas à questões como "Você acha que seus colegas de turma pensam que você não é suficientemente inteligente para esta graduação?", onde, de 22 alunos de APC, para uma parcela de 16 estudantes com cotas foram obtidas 6 respostas "Sim", onde para os 6 alunos sem cotas, todos responderam que não compartilham desta visão [13].

Dentro do estudo do ensino da primeira linguagem de programação, alguns artigos se comprometem a entender a questão do desempenho nestas disciplinas. Segundo Watson e Li, as taxas de aprovação na disciplina introdutória não têm apresentado melhora ao longo dos anos de 2000 à 2013, dada uma amostra de 54 artigos, distribuídos em 37 conferências, 11 periódicos, 3 teses, 2 não publicadas e 1 capítulo de livro. Esta amostra foi realizada com 161 cursos de CS1 (Computer Science 1, disciplina introdutória de programação), abrangendo 51 instituições de ensino distribuídas em 15 países diferentes, incluindo o Brasil [14].

## 2.2 Visualização de Dados Educacionais

O crescimento da área de ciência de dados permite cada vez mais sua utilização em diversos setores. Técnicas de visualização de dados têm sido utilizadas para avaliar aspectos sociais e históricos de alunos de computação para prever questões do seu desempenho em semestres futuros [15].

## 2.2.1 Algoritmos de Visualização de Dados

A visualização de dados tem por premissa facilitar o entendimento de um conjunto de informações para um usuário através da manipulação da amostra, permitindo a criação de imagens representativas com focos específicos, a depender do objetivo [16].

A exploração deste campo faz surgir diversas técnicas, como os algoritmos de visualização de dados, capazes de gerar gráficos através da programação. Por meio de seu trabalho de mestrado, a aluna Luiza Hansen realizou um estudo destas técnicas no contexto educacional com foco nas primeiras disciplinas de programação do Departamento de Ciência da Computação da UnB [1]. Dentre as técnicas estudadas, compõem o Observatório de Dados Educacionais:

- Gráfico de pizza: esta forma de visualização tem por objetivo representar a frequência de ocorrência da variável em questão através do tamanho da fatia correspondente. É bem utilizada em análises de escopo limitado e em conjunto de informações adicionais, visando o fácil entendimento comparativo das fatias [17]. O exemplo pode ser observado na Figura 2.1;
- Gráfico de barras: esta forma de visualização é comumente utilizada para representar um comparativo de frequências em diferentes categorias, sendo melhor utilizada em análises cujos valores para cada uma delas possui uma diferença notável pelo tamanho de cada barra. O exemplo pode ser observado na Figura 2.2;
- Gráfico de radar: este gráfico é utilizado para a visualização de valores em múltiplas variáveis quantitativas. Contudo, o layout circular pode prejudicar um pouco a leitura e interpretação por um usuário ao comparar-se com um gráfico de barras [18]. O exemplo pode ser observado na Figura 2.3;
- Gráfico de linhas: esta forma de visualização tem por objetivo representar a relação entre duas variáveis distintas, comumente utilizada para visualizações que envolvem variações ao longo do tempo, descrito no eixo horizontal. O exemplo pode ser observado na Figura 2.4;
- Gráfico de área: esta forma de visualização, semelhante ao gráfico de linha, representa a relação entre duas variáveis distintas, contudo possui um preenchimento na área entre a linha e o eixo X do gráfico, a fim de se destacar a grandeza dos valores. O exemplo pode ser observado na Figura 2.5;
- Gráfico de caixa: esta forma de visualização representa um conjunto de cinco características estatísticas - mínimo, primeiro quartil, mediana, terceiro quartil e máximo - para um conjunto de dados. O exemplo pode ser observado na Figura 2.6;

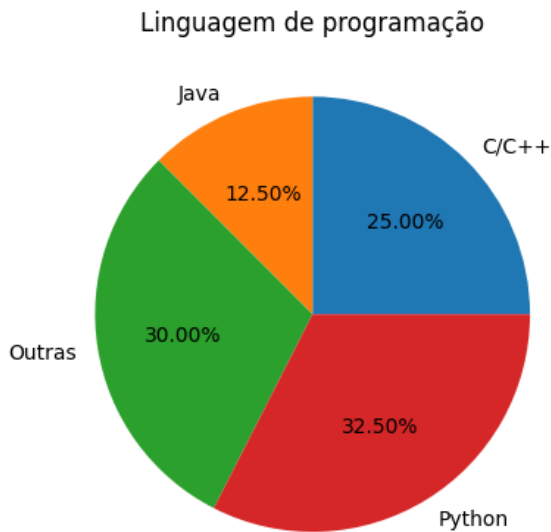


Figura 2.1: Exemplo de Gráfico de pizza.

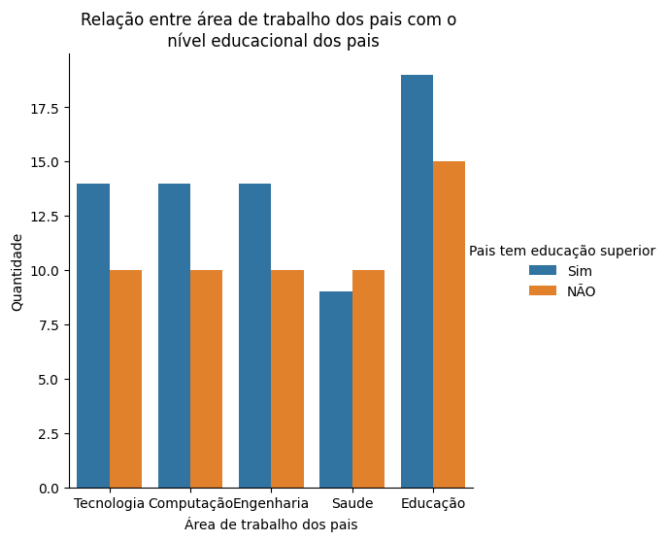


Figura 2.2: Exemplo de Gráfico de barras.

- Gráfico de dispersão: este gráfico tem por objetivo representar uma visualização em duas dimensões da ocorrência de um dado. É comumente utilizado para verificar o nível de relacionamento entre as duas dimensões elencadas. O exemplo pode ser observado na Figura 2.7;
- Gráfico de violino: esta forma de visualização representa a distribuição de dados quantitativos sobre uma variável, particularizando as mesmas características estatísticas do gráfico de caixa, porém apresentando a distribuição dos dados por completo. O exemplo pode ser observado na Figura 2.8;
- Coordenadas Paralelas: esta forma de visualização permite exibir grupos de dados absolutamente diferentes entre si, comparando-os e verificando suas relações. Cada variável do conjunto de dados é exibida como uma coluna vertical no gráfico, posicionadas paralelamente. O exemplo pode ser observado na Figura 2.9.
- Diagrama de sankey: esta forma de visualização exibe um fluxo entre uma variável e outra. A largura das conexões exibidas é proporcional ao tamanho deste fluxo. O exemplo pode ser observado na Figura 2.10.
- Gráfico Treemap: este gráfico auxilia a visualização dos dados de maneira hierárquica, utilizando retângulos dimensionados de acordo com a quantidade representada por cada um. O exemplo pode ser observado na Figura 2.11;

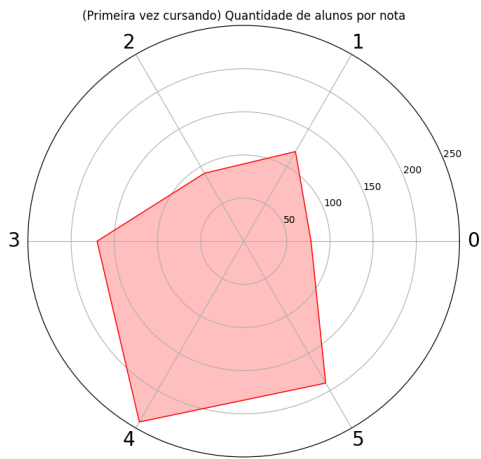


Figura 2.3: Exemplo de Gráfico de radar.

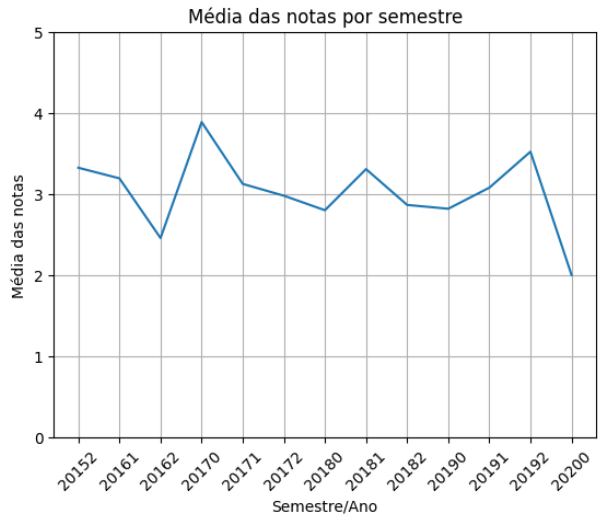


Figura 2.4: Exemplo de Gráfico de linhas.

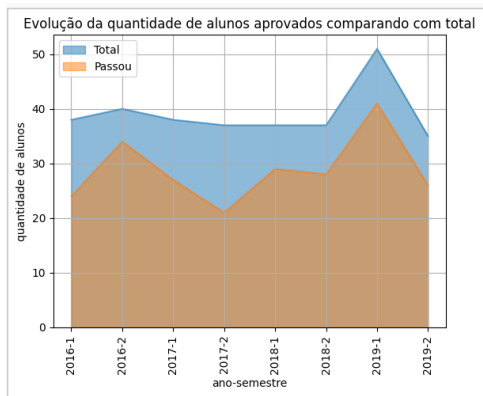


Figura 2.5: Exemplo de Gráfico de área.

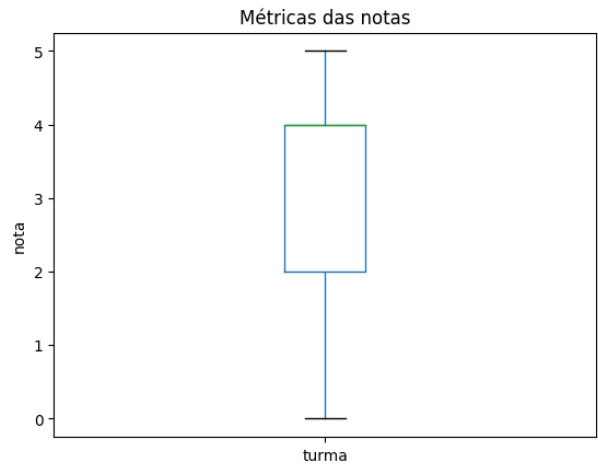


Figura 2.6: Exemplo de Gráfico de caixa.

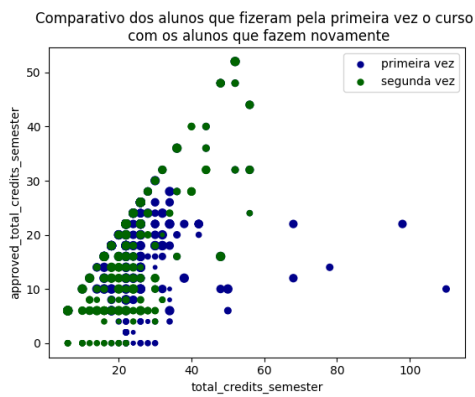


Figura 2.7: Exemplo de Gráfico de dispersão.

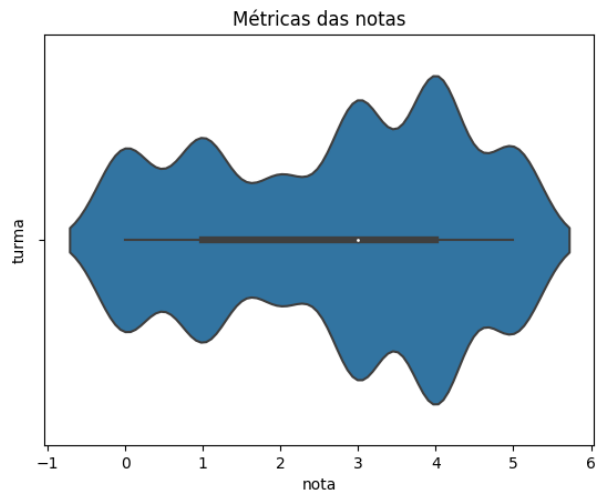


Figura 2.8: Exemplo de Gráfico de violino.

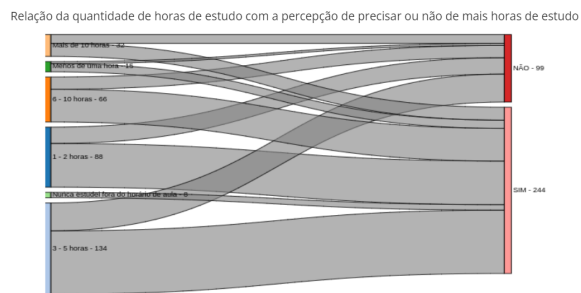
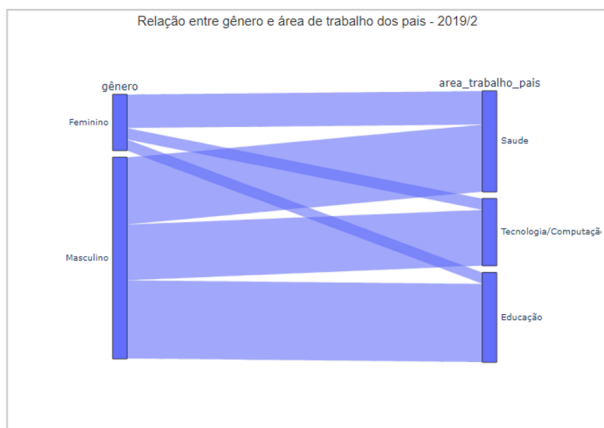


Figura 2.10: Exemplo de Diagrama de sankey.

Figura 2.9: Exemplo de Gráfico de coordenadas paralelas.



Figura 2.11: Exemplo de Gráfico treemap.

Tabela 2.1: Análise de cenário dos algoritmos de visualização estudados. Fonte: adaptado de Hansen, L. (p. 32) [1].

<b>Algoritmo</b>	<b>Análise de cenário</b>
<b>Gráfico de pizza</b>	- visualizar relação entre os valores - visualizar relação de um valor único com o total
<b>Gráfico de linhas</b>	- visualizar tendências ou mudanças ao longo do tempo
<b>Gráfico de barras</b>	- comparar diversos valores
<b>Gráfico de área</b>	- comparar as tendências de volume em série de tempo - enfatizar a magnitude da alteração ao longo do tempo - chamar a atenção para o valor total entre uma tendência
<b>Gráfico de dispersão</b>	- verificar se existe uma relação entre causa e efeito entre duas variáveis
<b>Gráfico de caixa</b>	- comparar o intervalo e distribuição de grupos de dados numéricos
<b>Gráfico de radar</b>	- comparar membros de uma dimensão em uma função de várias métricas
<b>Coordenadas paralelas</b>	- mostrar a comparação de elementos de dados em um agrupamento
<b>Diagrama de sankey</b>	- mostrar quantidades específicas - localizar as contribuições mais significativas para um fluxo geral
<b>Treemap</b>	- trabalhar com grandes quantidades de dados estruturados hierarquicamente
<b>Gráfico de violino</b>	- comparar a distribuição de uma determinada variável

A escolha das visualizações utilizadas está relacionada com o cenário a ser analisado. A Tabela 2.1 apresenta o conjunto de gráficos que, dentre os estudados pela aluna, está presente neste trabalho, junto do cenário de aplicação.

## 2.3 Tecnologias utilizadas

Nesta seção, são apresentadas as tecnologias utilizadas no desenvolvimento do Observatório de Dados Educacionais.

### Django

O sistema foi desenvolvido utilizando o *framework* Django 3.1.6, dada a necessidade de embarcar códigos de visualizações de dados desenvolvidos em Python. Seguindo a lógica de implementação de um sistema em Django, sua estrutura pode ser observada na Figura 2.12, que exhibe a pasta do projeto do Observatório de Dados Educacionais (tcc\_django-main).

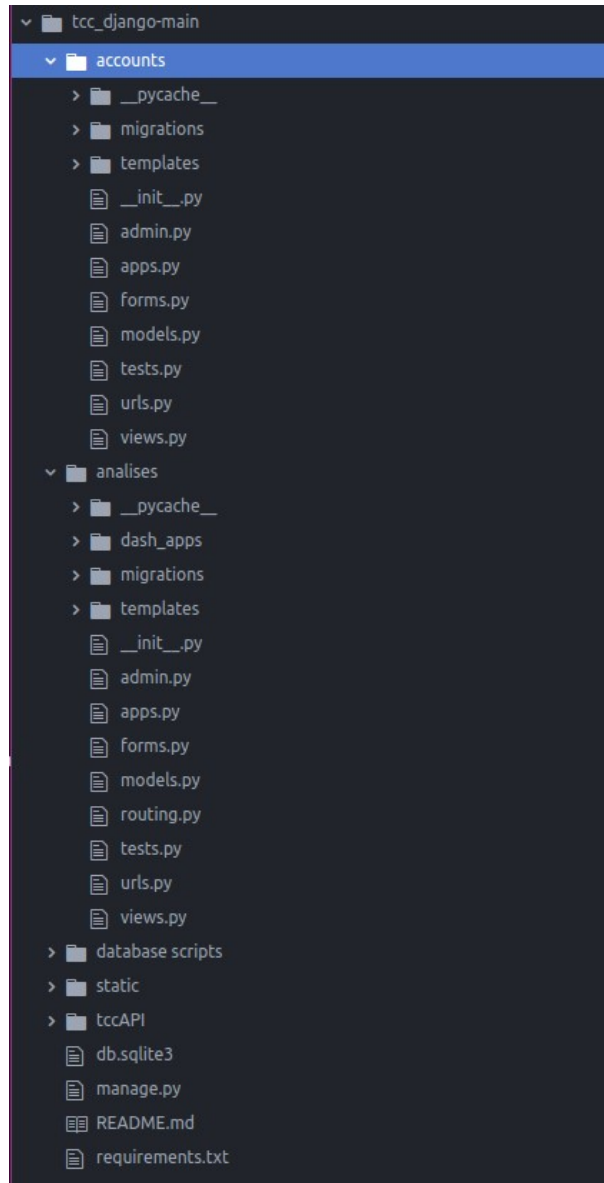


Figura 2.12: Estrutura de arquivos do projeto no Atom.

## Python 3.8

Python é uma linguagem de programação interpretada de alto nível, que suporta diversos tipos de paradigmas de programação, como a programação orientada a objetos. A linguagem possui uma extensa documentação<sup>1</sup> e suporte a diversas bibliotecas, com um conteúdo vasto para a área de ciência de dados.

---

<sup>1</sup><https://docs.python.org/pt-br/3.8/>

## HTML/CSS

HTML (Linguagem de Marcação de HiperTexto) é responsável por definir a estrutura das páginas dos *templates* utilizados. A marcação HTML pode ser estruturada com diferentes elementos, como `<head>`, `<body>`, `<div>`, entre outros [19].

CSS (Folhas de Estilo em Cascata) é uma linguagem utilizada para descrever o estilo de elementos do documento HTML, como cores, fontes de texto, margens, entre outros [20].

## Banco de Dados

Todas as bases utilizadas (dados provenientes do SIGRA, formulário aplicado aos alunos e tabelas auxiliares para formulários e *models*) foram criadas em MySQL, um sistema de gerenciamento de bancos de dados.

## Editor de texto e Versionamento

Junto do Python 3.8, o projeto foi desenvolvido utilizando o editor de texto Atom, que permite a navegação pela estrutura dos arquivos do projeto a partir de uma interface bem organizada.

O controle de versões do sistema foi realizado pelo Github<sup>2</sup>, através de um repositório **privado**, a fim de proteger-se as informações dos dados de alunos.

## Pythonanywhere

Para facilitar a utilização do sistema por parte da professora orientadora e da aluna colaboradora, o sistema foi hospedado no Pythonanywhere<sup>3</sup>, um serviço de hospedagem que possui suporte a projetos desenvolvidos em Django com planos que variam de acordo com a necessidade do desenvolvedor.

Neste capítulo foram expostos os conceitos utilizados na elaboração deste trabalho. O próximo capítulo exhibe o processo utilizado no desenvolvimento do Observatório de Dados Educacionais, bem como características da arquitetura do sistema.

---

<sup>2</sup><https://github.com/>

<sup>3</sup><https://www.pythonanywhere.com/>



# Capítulo 3

## Observatório de Dados Educacionais

Neste capítulo é apresentado o Observatório de Dados Educacionais, com o objetivo de embarcar um conjunto de gráficos referentes às disciplinas introdutórias de programação do Departamento de Ciência da Computação da Universidade de Brasília em uma interface simples, facilitando a visualização para professores, coordenadores e chefe de departamento. Este capítulo está dividido nas seções: Processo de Desenvolvimento do Sistema (Seção 3.1) e Arquitetura do Sistema (Seção 3.2).

### 3.1 Processo de Desenvolvimento do Observatório

Nesta seção, são descritas as atividades que compuseram o processo de criação do Observatório, desde o protótipo até o fluxo de navegação do usuário comum e do usuário administrador.

#### 3.1.1 Gráficos

Os gráficos disponibilizados no sistema compõem o resultado do trabalho de mestrado da aluna Luiza Aguiar Hansen, onde foram desenvolvidos algoritmos para gerar as visualizações apresentadas no Capítulo 2, como colaboração a este trabalho de graduação.

Os gráficos foram construídos utilizando dados obtidos pelo SIGRA (Sistemas Acadêmicos da Universidade de Brasília), com registros das disciplinas de Introdução à Ciência da Computação, Computação Básica e Algoritmos e Programação de Computadores, descritas para os cursos:

- Ciência da Computação;
- Computação;
- Engenharia Civil;

- Engenharia de Computação;
- Engenharia Florestal;
- Engenharia Mecânica;
- Engenharia Mecatrônica;
- Estatística;
- Matemática.

O conjunto de dados utilizado é referente a semestres a partir de 1984/2 até o semestre de verão de 2020. O acesso a esses dados foi aprovado pelo comitê de ética com o parecer número 4.283.719, de 17/09/2020.

Além destes dados, são utilizadas informações de um formulário de percepção aplicado nas turmas de APC (Algoritmos e Programação de Computadores) nos semestres de 2019/2 e 2020/1, com o intuito de construir análises sobre experiências prévias dos alunos em programação e informações sociais.

### 3.1.2 Protótipo e Requisitos

O primeiro protótipo do sistema foi cedido pela aluna colaboradora Luiza Hansen, junto de uma elaboração inicial de funcionalidades que o sistema deveria atender. A proposta inicial consistia em três páginas distintas:

- Página de **Informações Gerais**, exibindo visualizações acerca do desempenho geral nas disciplinas;
- Página de **Informações Acadêmicas**, exibindo o desempenho das turmas ao longo do tempo;
- Página de **Informações de Percepção**, exibindo visualizações que correspondem ao formulário aplicado nas turmas de Algoritmos e Programação de Computadores.

Complementando a interface, sugeriu-se um menu lateral para a navegação entre as três páginas e, adicionalmente, filtros destacados no menu superior para a visualização de informações mais específicas, envolvendo a consulta por registros de alunos que possuem ou não cotas, consultas por cursos específicos e/ou consultas por gênero específico para as análises envolvendo Informações Gerais e Informações Acadêmicas e consulta por período (semestre) específico e/ou consulta por turma específica para as Informações de Percepção.

As primeiras propostas de gráficos a serem disponibilizados no sistema, com base nos cenários descritos na Tabela 2.1 presente no capítulo anterior, são elencadas a seguir:

### **Gráficos da página de Informações Gerais**

- Quantidade de alunos por semestre que cursaram a matéria (Treemap);
- Quantidade de alunos por quantidade de vezes que cursaram a matéria (Gráfico de barras);
- Quantidade de alunos por Aprovação/Reprovação (Gráfico de pizza);
- Quantidade de alunos Aprovados/Reprovados na primeira vez que fizeram a matéria (Gráfico de pizza);
- Quantidade de alunos por nota (Gráfico radar).

### **Gráficos da página de Informações Acadêmicas**

- Valor médio das notas dos alunos;
- Quantidade de alunos fazendo a matéria e quantidade de alunos aprovados no decorrer dos semestres (Gráfico de área);
- Evolução das notas durante o semestre na matéria (Gráfico de linha);
- Média das notas na matéria (Gráfico de caixa);
- Média das notas na matéria (Gráfico violino).

### **Gráficos da página de Informações de Percepção**

- Você já tinha experiência em programação? (Gráfico de pizza);
- Se você tinha experiência em programação antes dessa disciplina, informe em quais linguagens (Gráfico de barras);
- Qual o seu curso de graduação? (Gráfico de barras);
- Em que área os seus pais e irmãos trabalham? Seu pai e/ou sua mãe terminaram algum curso superior? (Diagrama de sankey);
- Durante o semestre, quantas horas você passou fazendo os deveres de casa e estudando fora da sala de aula por semana para essa disciplina? Você acha que precisa de mais horas de estudos nessa disciplina do que seus amigos de sala? (Diagrama de sankey);
- Com que frequência você ouve "Engenharia/Computação é para pessoas super inteligentes"? (Gráfico de pizza);

- Com que frequência você ouve "Engenharia/Computação é para homens"? (Gráfico de pizza).



Figura 3.1: Protótipo da interface da página de Informações Gerais.



Figura 3.2: Protótipo da interface da página de Informações Acadêmicas.

A partir destas escolhas, foi elaborado o modelo inicial da interface do sistema, mostrado nas Figuras 3.1 a 3.3.

Após a apresentação da interface, foi elaborada uma modelagem do fluxo de operação de um usuário comum do sistema, descrita da seguinte forma:

- Inicialmente, o usuário acessa a *url* do sistema. A fim de se proteger as informações disponíveis nos gráficos, estes só podem ser acessados caso o usuário esteja homologado no sistema;
- Caso o usuário não possua um cadastro no sistema, este deve ser realizado. Após a confirmação das informações pelo usuário administrador do sistema, o cadastro do usuário é confirmado;

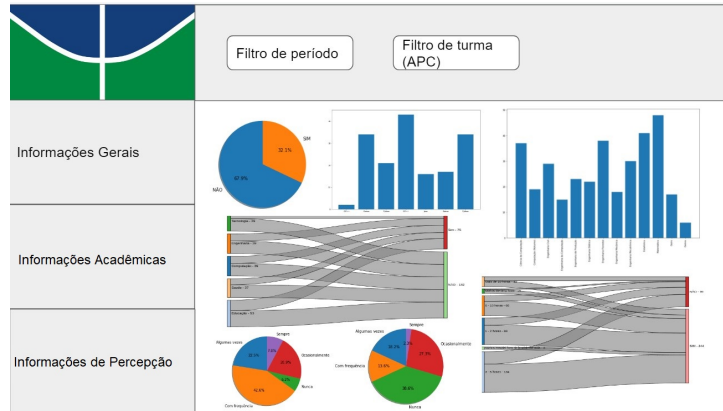


Figura 3.3: Protótipo da interface da página de Informações de Percepção.

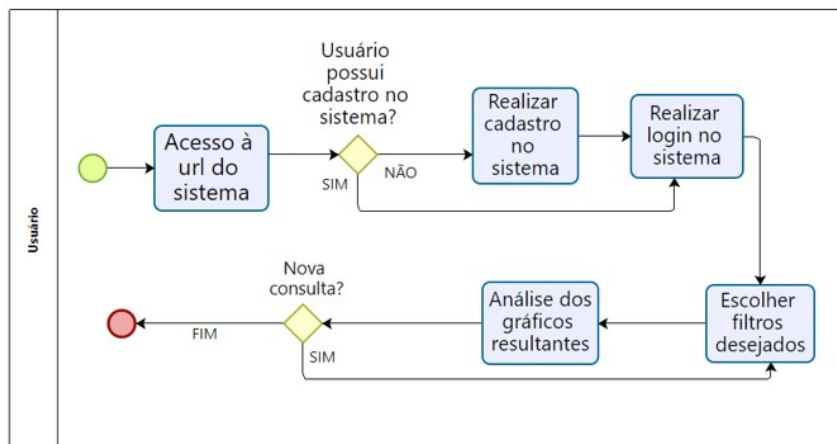


Figura 3.4: Modelo do processo de operação de um usuário comum do sistema.

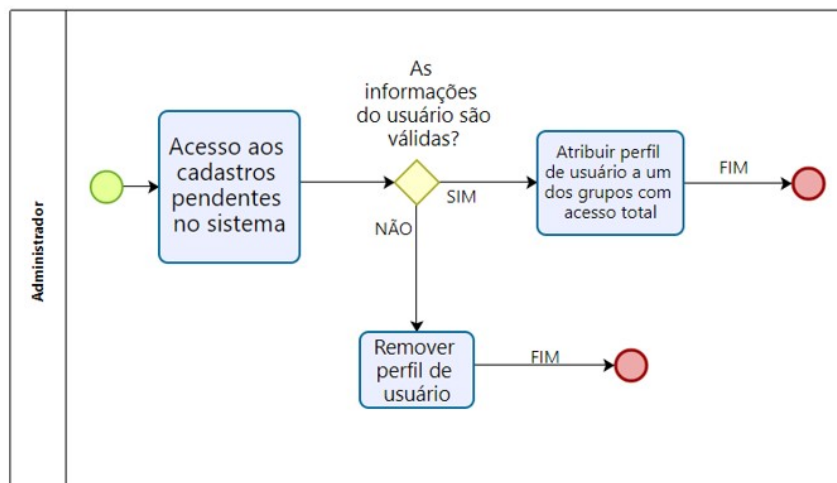


Figura 3.5: Modelo do processo de operação de um usuário administrador do sistema.

- É efetuado o *login* no sistema;
- O usuário, agora, possui acesso às páginas de gráficos disponibilizados no sistema, podendo efetuar consultas através da aplicação dos filtros no menu superior e navegação nas páginas no menu lateral.

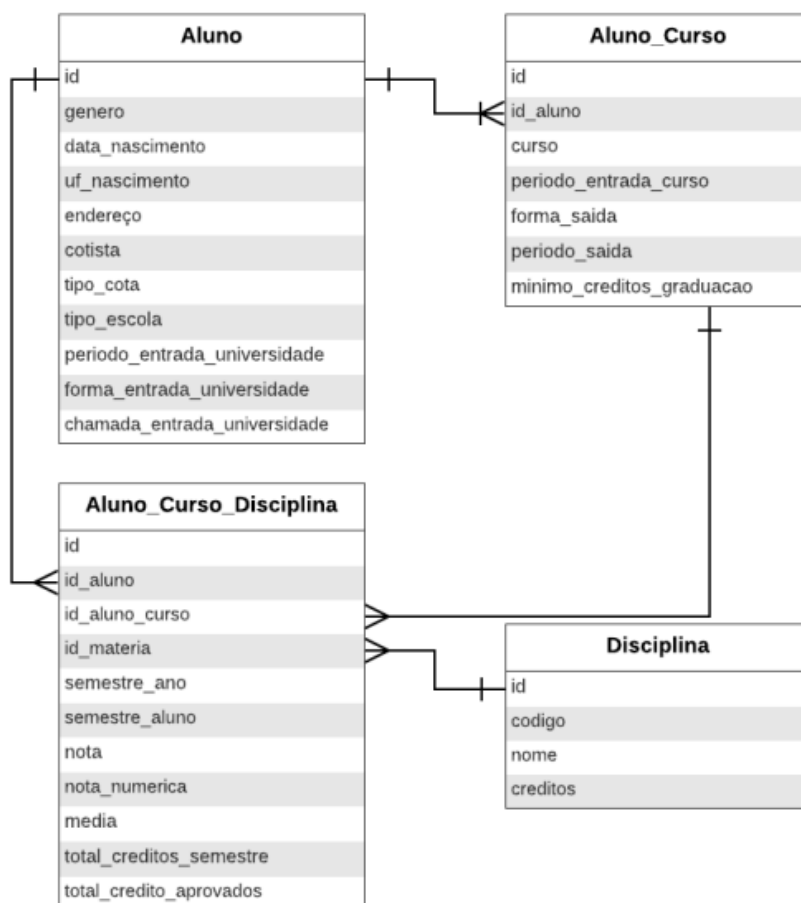


Figura 3.6: Modelo do banco de dados relacional utilizando informações do SIGRA. Fonte: [1].

O modelo de operação de usuário comum descrito acima pode ser observado na Figura 3.4

De forma complementar, como pode ser observado na Figura 3.5, o fluxo de operação de um usuário administrador é descrito por:

- Acessar os cadastros pendentes no sistema, na página de administrador do sistema;

- Verificar as informações dos cadastrantes, de forma a confirmar se o usuário solicitante faz parte de um dos grupos que podem obter acesso ao sistema (professor, coordenador ou chefe de departamento);
- Em caso afirmativo, atribuir o perfil de usuário a um destes grupos, já existentes no sistema;
- Em caso negativo, remover o perfil de usuário do sistema.

A Figura 3.6 mostra o Modelo Relacional proposto para o banco responsável por abarcar os dados provenientes do SIGRA, desenvolvido em [1] e implementado neste trabalho. O banco apresenta quatro tabelas:

- Aluno: apresenta informações gerais acerca dos estudante;
- Disciplina: apresenta código, nome e quantidade de créditos presentes nas disciplinas;
- Aluno\_Curso: representa a relação entre todos os cursos de graduação que cada aluno presente na base de dados já foi matriculado;
- Aluno\_Curso\_Disciplina: representa a relação entre as disciplinas que já foram cursadas por um aluno enquanto pertencente a um curso.

### 3.1.3 Gráficos Definitivos

Após a conclusão do trabalho de mestrado da aluna Luiza, foram alterados os gráficos que fazem parte do sistema, dado o resultado de questionários aplicados a professores e gestores educacionais, visando extrair as melhores visualizações para cada cenário.

Para a página de Informações Gerais, foram definidos os gráficos:

- Quantidade geral de aprovações/reprovações na matéria (gráfico de pizza);
- Primeira vez cursando a disciplina (Alunos/Reprovação) (gráfico de pizza);
- Em qual semestre os alunos cursaram a matéria (gráfico de barras);
- Cursos dos alunos que fizeram a disciplina pela primeira vez em comparação com os que fazem novamente (gráfico de barras com linhas);
- Quantidade de alunos por nota (Primeira vez cursando a disciplina) (gráfico de radar).

Para a página de Informações Acadêmicas, foram definidos os gráficos:

- Média das notas por semestre (gráfico de linhas);
- Evolução da quantidade de alunos aprovados comparado com o total (gráfico de área);
- Métrica das notas (frequência por nota) (gráfico de barras);
- Quantidade de créditos comparando com o número de aprovações (gráfico de barras com linhas);
- Comparativo dos alunos que fizeram pela primeira vez o curso com os que fazem novamente (gráfico de dispersão);
- Comparativo dos alunos que fizeram pela primeira vez o curso com os que fazem novamente (gráfico de linhas);
- Métrica das notas (gráfico de caixa);
- Métrica das notas (gráfico de violino).

Para a página de Informações de Percepção, foram definidos os gráficos:

- Você já tinha experiência em programação? (Gráfico de barras);
- Se você tinha experiência em programação antes dessa disciplina, informe em quais linguagens (gráfico de pizza);
- Relação entre área de trabalho dos pais com o nível educacional dos pais (gráfico de barras);
- Frequência com que os alunos costumam escutar que Engenharia/Computação é para pessoas super inteligentes (gráfico de barras);
- Com que frequência os alunos que já escutaram a frase "Engenharia/Computação é para homens"? (Gráfico de pizza).

Foi adicionada uma página denominada **Gráficos Interativos**, para a exibição de um conjunto de visualizações adicionais ainda acerca do formulário aplicado nas turmas de Algoritmos e Programação de Computadores. Todos os gráficos desta página são exibidos como coordenadas paralelas, e a biblioteca utilizada para a criação destas visualizações permite ao usuário a interação com os gráficos, facilitando a quantificação dos dados e organização dos atributos nas colunas conforme o desejado.

Para a página de Gráficos Interativos, foram definidos os gráficos:

- Relação entre gênero e área de trabalho dos pais (gráfico de coordenadas paralelas);



- Relação entre nível educacional dos pais e cotas (gráfico de coordenadas paralelas);
- Relação da quantidade de horas de estudo com a percepção de precisar ou não de mais horas de estudo (gráfico de coordenadas paralelas).

## 3.2 Arquitetura do Sistema

Nesta seção, são descritas as características da arquitetura do Observatório, englobando *framework* utilizado, bancos de dados e hospedagem do sistema.

### 3.2.1 Django e Arquitetura *Model-View-Template*

Django é um *framework* para desenvolvimento web, baseado na linguagem de programação Python. Para que o sistema pudesse embarcar análises gráficas desenvolvidas também em Python, esta solução apresentou o suporte necessário. A abstração da arquitetura utilizada pelo Django é classificada como *Model-View-Template*<sup>1</sup>, onde:

- *Model*: é a camada responsável pela caracterização dos dados que estão sendo armazenados, sendo mapeada para as tabelas no seu banco de dados - cada atributo da *model* representa uma coluna na tabela equivalente;
- *View*: é a camada responsável pelo processamento de um requerimento do usuário e o comportamento e resposta do sistema;
- *Template*: é a camada responsável por apresentar o conteúdo do sistema para o usuário como uma interface.

Como pode ser observado na Figura 3.7, que detalha as relações entre as camadas de *model*, *view*, *template* e o banco de dados de um sistema desenvolvido com o Django, os componentes lógicos do projeto são implementados na camada da *view*. Desta forma, os algoritmos de visualização puderam ser utilizados com o mínimo de adaptações diretamente como funções individuais da *view*.

### 3.2.2 Sistema de Filtros

A elaboração do sistema de filtros para os gráficos foi baseada na criação de uma variável utilizada nas cláusulas "*where*", presente em todas as etapas de consultas à base de dados para a elaboração dos gráficos e adaptada para a interface do usuário através de um formulário<sup>2</sup>. A ferramenta é capaz de alterar de maneira dinâmica a condição, conforme o seguinte exemplo:

<sup>1</sup><https://docs.djangoproject.com/en/3.2/>

<sup>2</sup><https://docs.djangoproject.com/en/3.2/topics/forms/>

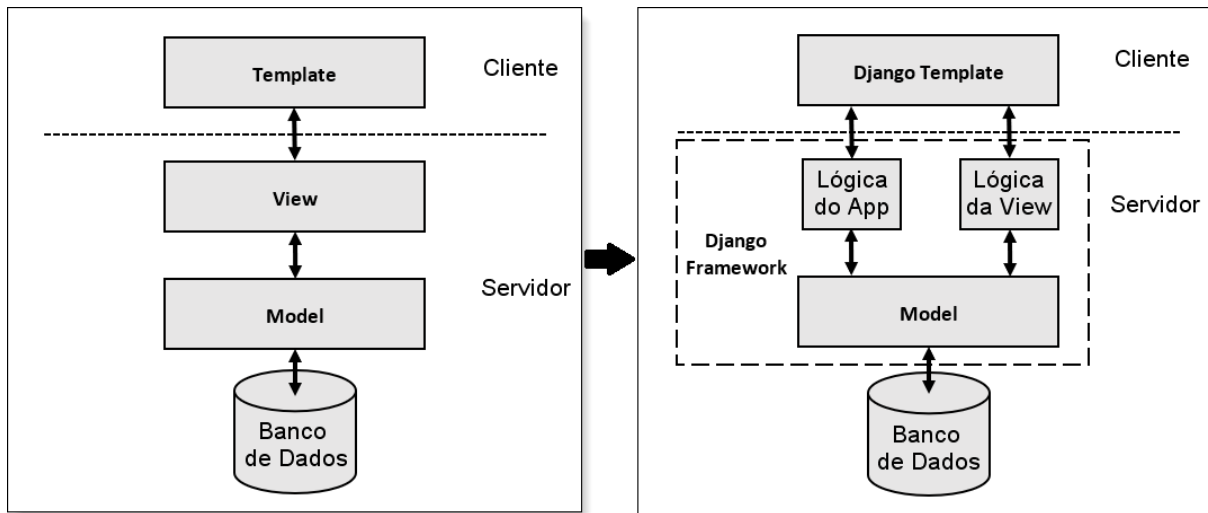


Figura 3.7: Perspectiva MVT do Django (adaptada) [2].

```
query = "select * from students "+ clausula
```

onde a variável **clausula** é obtida anteriormente através de uma consulta à tabela de usuários do sistema, cujas colunas são mapeadas pela *model* do usuário utilizado para a autenticação, fazendo com que cada campo escolhido no menu de filtros preencha com um trecho do código SQL na tabela de usuário a ser consultada, gerando a cláusula completa.

A utilização de formulários do Django para essa tarefa permite que o usuário realize a atualização da cláusula na tabela através da interface do menu superior e, desta forma, o código gerador do gráfico atualize de forma dinâmica a condição da consulta para atender à solicitação do usuário.

O intuito de utilizar a *model* de usuário para armazenar os valores correspondentes dos filtros é a individualização, fazendo com que cada usuário tenha suas próprias consultas e que mantenha-se o estado anterior ao sair do sistema. Na Figura 3.8, é possível ver os atributos utilizados para o menu dos filtros (sublinhado em verde, filtros para Informações Gerais e Informações Acadêmicas e, sublinhado em azul, filtros para Informações de Percepção).

### 3.2.3 Autenticação

#### *Model* de Usuário

O modelo inicial do perfil de usuário a ser utilizado no sistema sugere a existência de 6 atributos para o cadastro: username, senha, email, nome, matrícula e função (professor, coordenador ou chefe de departamento).

## Change user profile



User:	professor  
Nome:	pedro
Matricula:	123456
Funcao:	Professor
<u>Disciplina:</u>	Introducao a Ciencia da Computacao
<u>Curso:</u>	Matematica
<u>Genero:</u>	-
<u>Possui cotas:</u>	Possui cotas
<u>Periodo:</u>	Pós 2016
<u>Periodo form:</u>	-
<u>Turma form:</u>	-

Figura 3.8: Resultado dos filtros na *model* de usuário.

O *framework* Django oferece seu próprio sistema de autenticação<sup>3</sup> com algumas limitações. Os atributos existentes para a *model* de usuário padrão do Django englobam:

- username;
- senha;
- email;
- primeiro\_nome;
- ultimo\_nome.

Assim, foi necessária a criação de uma nova *model*, denominada UserProfile, para a composição dos atributos faltantes, incluindo as colunas referentes aos filtros. O perfil de um usuário, então, é cadastrado nesta nova *model* e armazenado na tabela de usuário.

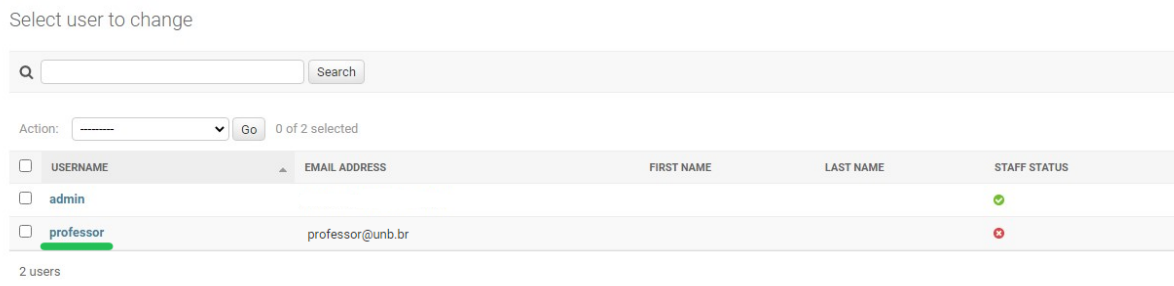


Figura 3.9: Novo perfil cadastrado pelo usuário, destacado na cor verde.

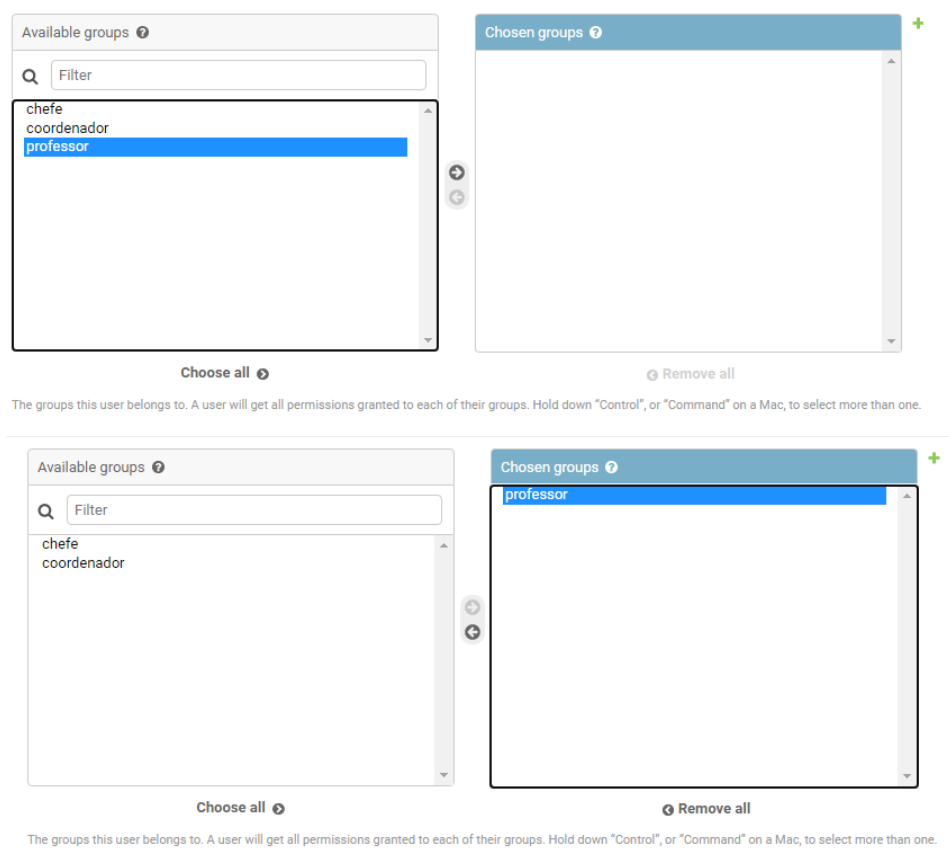


Figura 3.10: Gerenciamento de grupos de usuários disponíveis no Observatório.

## Validação por Administrador

Como medida de proteção às informações apresentadas no sistema, um perfil de usuário só possui acesso às análises gráficas após ser validado por um usuário administrador e atribuído a um dos grupos (professor, coordenador ou chefe de departamento).

<sup>3</sup><https://docs.djangoproject.com/en/3.2/topics/auth/>

Os cadastros de novos usuários aparecem na administração do site<sup>4</sup>, conforme a Figura 3.9, que exibe os usuários presentes no sistema. Sublinhado em verde está um perfil de novo usuário, criado por um professor.

Após a verificação das informações enviadas pelo usuário no cadastro, o administrador do sistema deve atribuir o perfil a um grupo, conforme a Figura 3.10, que exibe os grupos disponíveis e os grupos que o usuário selecionado faz parte.

### 3.2.4 Problemas Identificados

Durante o desenvolvimento do sistema, foram identificados problemas referentes a limitações das bibliotecas gráficas e também ao desempenho do serviço de hospedagem utilizado.

Com relação ao primeiro problema, alguns algoritmos de visualização não geraram resultados como imagem em funções da *view* diretamente, tanto por falta de suporte ao tipo de gráfico quanto falta de suporte à algumas bibliotecas. Diante disso, foi necessário utilizar um gráfico embutido gerado através do *framework* Dash para as análises que envolviam **Gráficos de coordenadas paralelas**. Essa alternativa gerou incompatibilidade com a ferramenta de filtros adotada, impossibilitando a consulta dinâmica para estes casos. Assim, tais gráficos foram disponibilizados em uma página separada - Gráficos Interativos -, sem a utilização dos filtros.

Acerca do segundo problema, em geral o serviço comporta-se como esperado; contudo, há situações sem motivo específico nas quais as imagens dos gráficos não carregam em um primeiro momento, exigindo que recarregue-se a página.

Neste capítulo foi explicada a ideia inicial para o Observatório de Dados Educacionais, bem como a atualização das visualizações correspondendo ao resultado dos questionários aplicados em [1]. Também foi desenvolvida a questão da arquitetura do sistema, exibindo conceitos aplicados neste trabalho e dificuldades identificadas. O capítulo seguinte exibe a interface do sistema, apresentando as páginas e gráficos contidos nele.

---

<sup>4</sup><https://docs.djangoproject.com/en/3.2/ref/contrib/admin/>

# Capítulo 4

## Análise de Resultados

Neste capítulo é apresentada a interface do Observatório de Dados Educacionais. São detalhadas as páginas desenvolvidas para o sistema e gráficos presentes.

### 4.1 Interface do Observatório

#### 4.1.1 Página inicial

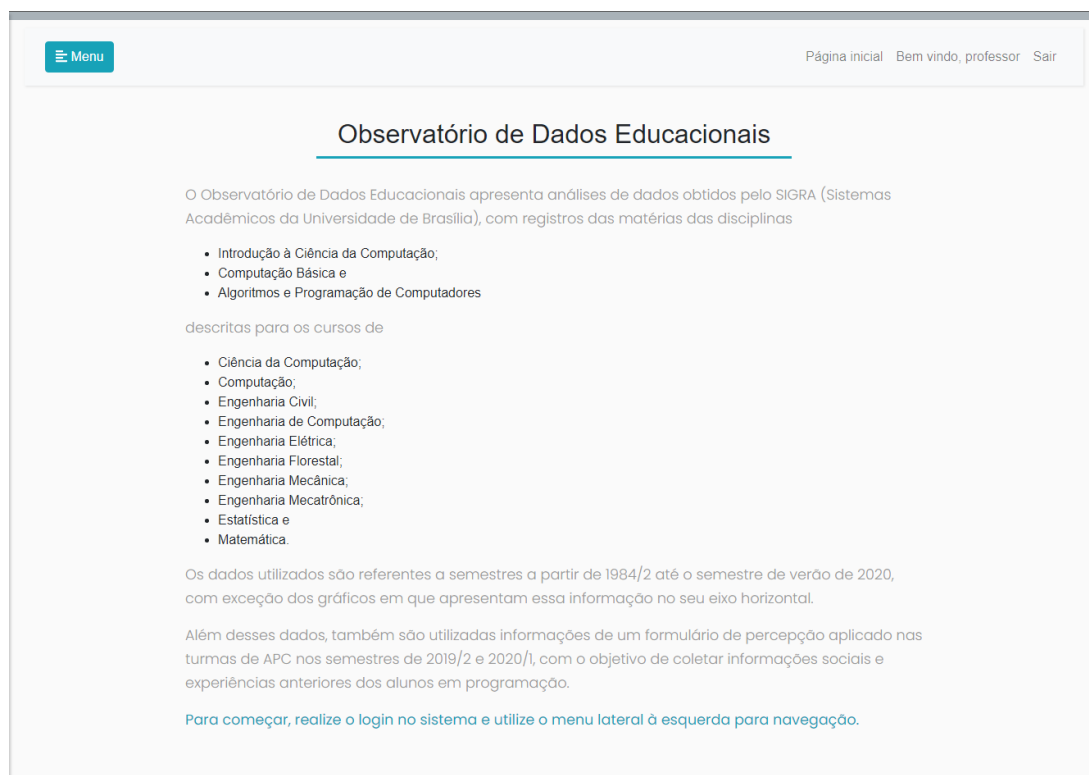


Figura 4.1: Página Inicial do Observatório.

A página inicial do sistema apresenta informações acerca dos dados utilizados nas análises gráficas, explicitando as disciplinas presentes, os cursos descritos e o período analisado, como pode ser observado na Figura 4.1.

### 4.1.2 Navegação

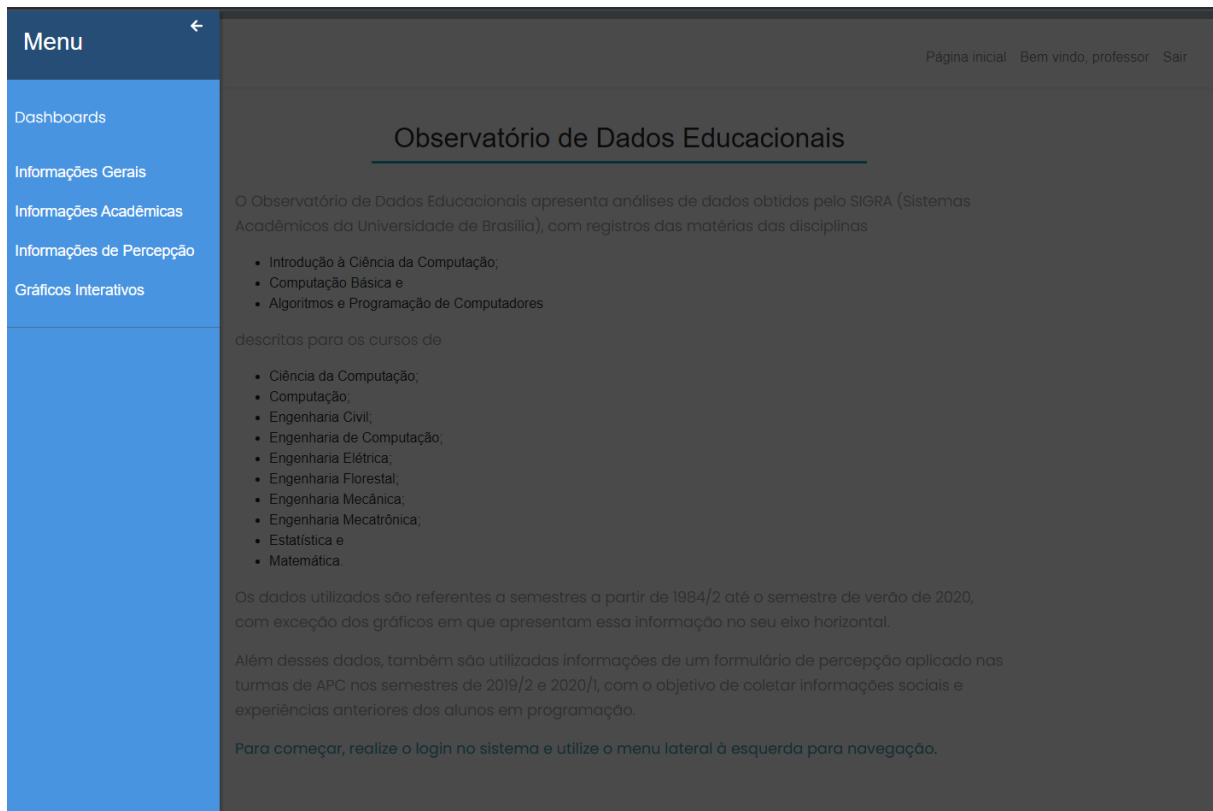


Figura 4.2: Menu Lateral para Navegação no Sistema.

A navegação entre as páginas de visualização do Observatório é realizada pelo menu lateral à esquerda, como é apresentado na Figura 4.2, que exibe os itens disponíveis para as demais páginas.

### 4.1.3 Autenticação

Para o acesso total ao sistema, o usuário deve realizar o *login*, acessível através do menu superior, ao clicar-se em "Entrar". Caso não possua um cadastro, este deve realizá-lo na página de cadastro (Figura 4.3, acessível tanto pelo menu superior ("Cadastrar") quanto pela tela de *login* ("Não possuo cadastro").

The image shows a web page for user registration. At the top left, there is a 'Menu' button. At the top right, there are links for 'Página inicial', 'Entrar', and 'Cadastrar'. The main content area is a form titled 'Informe seus dados'. The form contains the following fields: 'Username' (text input), 'E-mail' (text input), 'Senha' (text input), 'Confirme sua senha' (text input), 'Nome' (text input), 'Matrícula' (text input), and 'Função' (dropdown menu with 'Professor' selected). Below the form is a blue button labeled 'Enviar cadastro' and a link labeled 'Já possui cadastro'.

Figura 4.3: Página de Cadastro de Usuário.

#### 4.1.4 Página de Informações Gerais

Após efetuar o *login*, o usuário passa a ter acesso às três páginas de análises gráficas do sistema. A primeira, referente a Informações Gerais, exibe gráficos relacionados a:

- Percentuais de aprovação no geral, descritos nos gráficos de pizza;
- Semestre do currículo no qual o aluno cursou a matéria - que, apesar de serem matérias introdutórias de programação, podem ocorrer não apenas no primeiro, mas nos próximos semestres, devido à reprovações -, descritos no gráfico de barras;
- Comparativo dos cursos de alunos que fazem a disciplina mais de uma vez, descrito no gráfico de barras com linhas;
- Análise da distribuição das notas dos alunos na primeira vez cursando a disciplina, descrito no gráfico de radar.



Tratam-se de visualizações com foco no desempenho geral das disciplinas, permitindo a especialização das consultas através da utilização dos filtros presentes no menu superior, como pode ser observado nas Figuras 4.4 e 4.5.

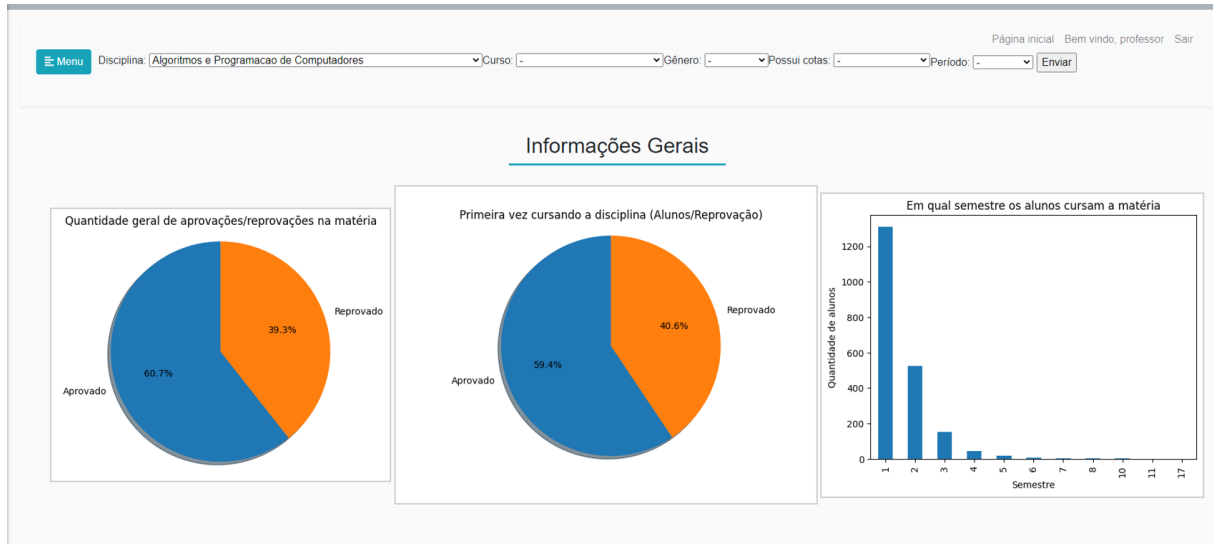


Figura 4.4: Gráficos em Informações Gerais para a disciplina de APC.

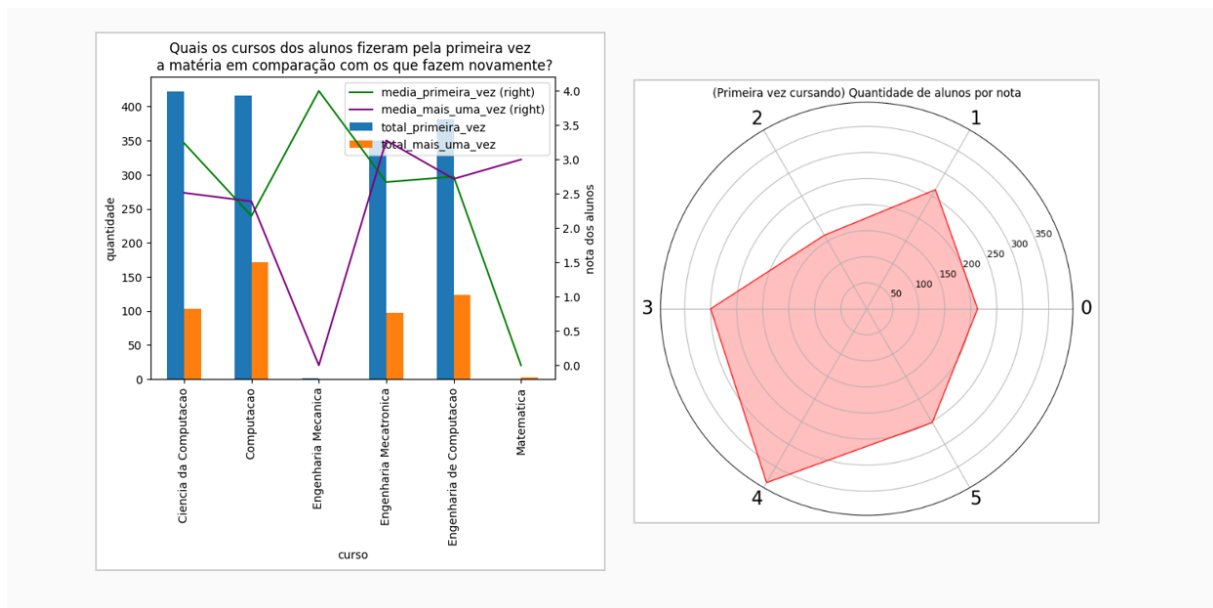


Figura 4.5: Gráficos em Informações Gerais para a disciplina de APC.

É perceptível a queda no quadro de desempenho da turma de Algoritmos e Programação de Computadores quando aplicado-se o filtro para apenas alunos com cotas (Figura 4.6).

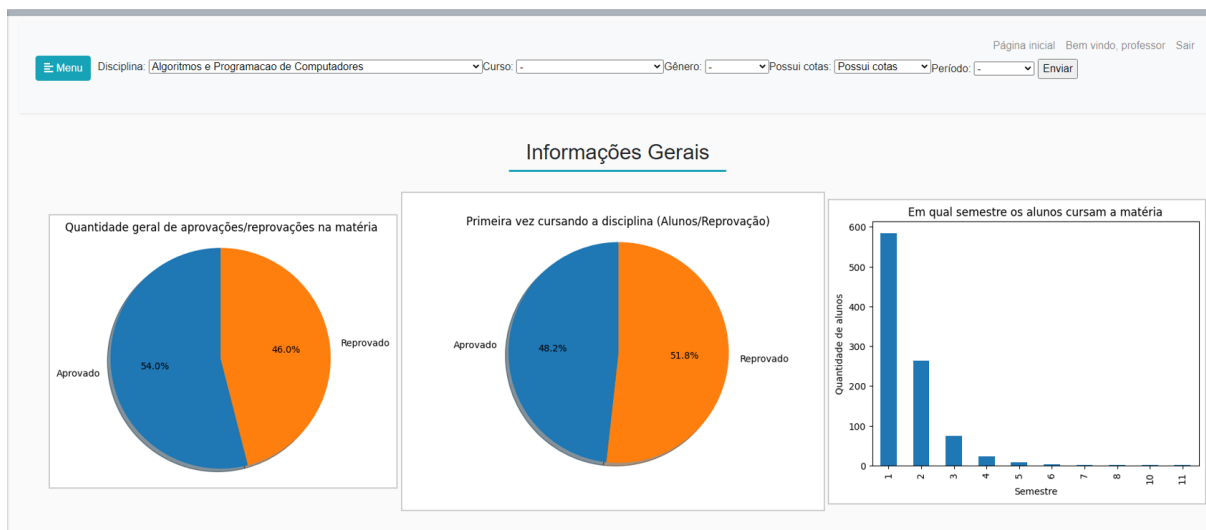


Figura 4.6: Gráficos Filtrados em Informações Gerais para a disciplina de APC.

O gráfico para "Quais os cursos dos alunos que fizeram pela primeira vez a matéria em comparação com os que fazem novamente" deve ser analisado apenas quando não há seleção no filtro de curso, visto que o objetivo deste é a comparação entre os diversos cursos que possuem a matéria filtrada em seu currículo.

#### 4.1.5 Página de Informações Acadêmicas

A página de Informações Acadêmicas apresenta análises que demonstram o desempenho ao longo do tempo, a partir do segundo semestre de 2013, bem como análises das métricas das notas. Estão presentes algoritmos para as visualizações relacionadas a:

- Média das notas por semestre, descrita no gráfico de linha;
- Evolução da quantidade de alunos aprovados em comparação com o total de alunos que cursaram a disciplina, descrita no gráfico de área;
- Distribuição das notas classificadas no formato de menção (segundo a tabela Tabela 4.1), descrita nos gráficos de barras, gráfico de caixa e gráfico de violino;
- Comparativo da quantidade de créditos cursados com o desempenho na disciplina, descrito no gráfico de barras com linhas;
- Comparativo da quantidade de créditos cursados com total de créditos aprovados, descrito no gráfico de dispersão;

Tabela 4.1: Mapeamento de Notas para Menções.

Nota	Menção
0	SR
1	II
2	MI
3	MM
4	MS
5	SS

- Comparativo da média das notas de alunos cursando a disciplina na primeira vez com a média das notas de alunos cursando a disciplina mais de uma vez, descrito no gráfico de linhas.

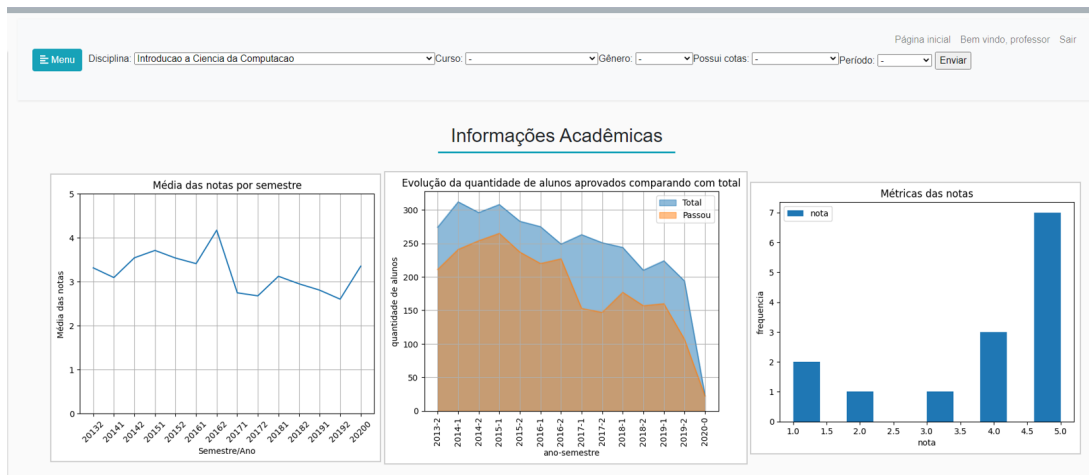


Figura 4.7: Gráficos em Informações Acadêmicas para a disciplina de ICC.

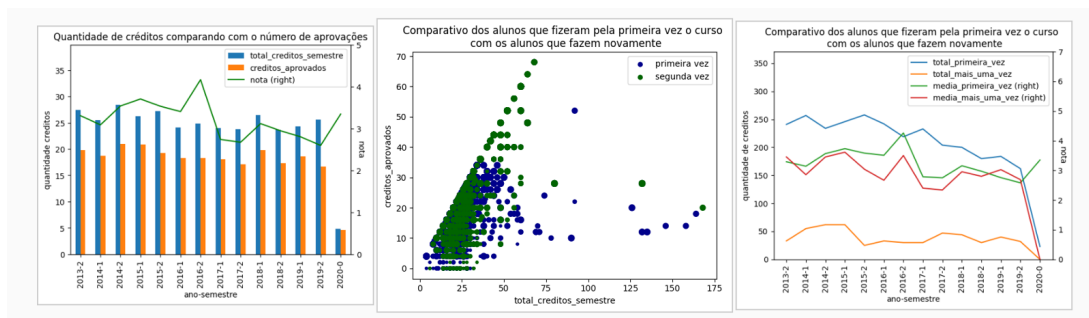


Figura 4.8: Gráficos em Informações Acadêmicas para a disciplina de ICC.

As Figuras 4.7 e 4.8 exibem exemplos de gráficos presentes nesta página.

De maneira análoga, é possível observar uma queda no desempenho ao analisar-se a média das notas, bem como a relação entre alunos aprovados por total de alunos quando se aplicados os filtros de cota para o curso de Matemática em Introdução a Ciência da Computação, segundo a Figura 4.9.

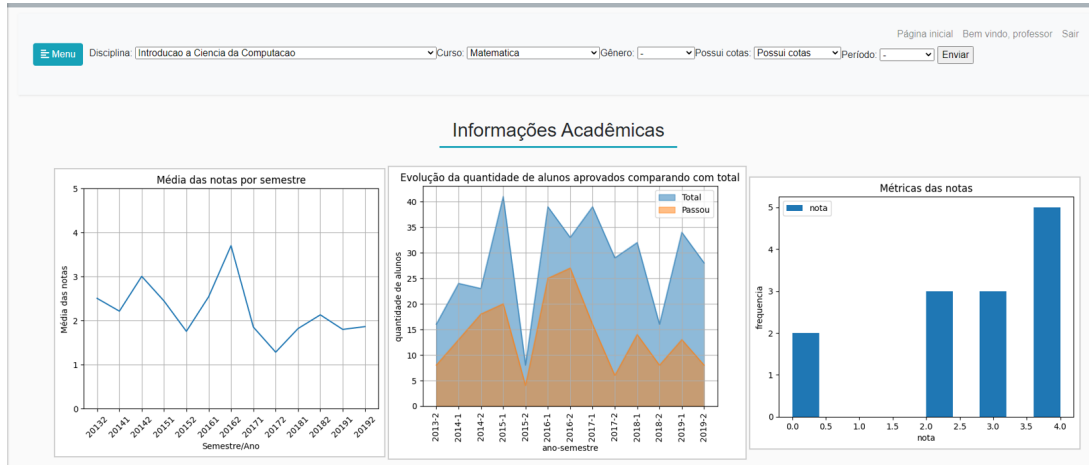


Figura 4.9: Gráficos Filtrados em Informações Acadêmicas para a disciplina de ICC.

Outra questão que pode ser observada nesta página é a relação que os semestres de verão (2017-0, 2018-0, 2019-0 e 2020-0, no eixo horizontal do segundo gráfico da página) tem com o índice de aprovações na disciplina de Algoritmos e Programação de Computadores, como pode ser observado na Figura 4.10, que exhibe uma alta taxa de alunos aprovados nestes semestres em comparação com os demais.

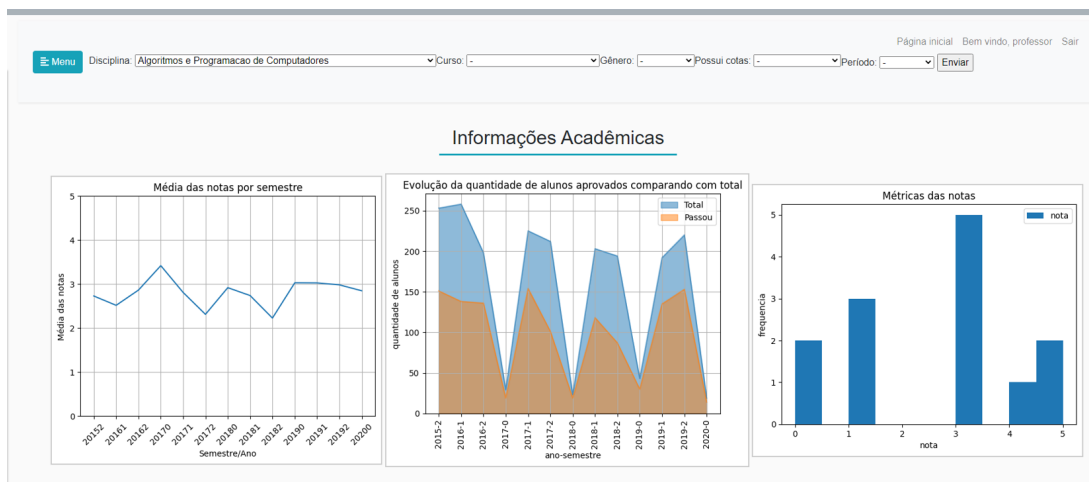


Figura 4.10: Gráficos em Informações Acadêmicas para a disciplina de APC.

As demais visualizações presentes nesta página estão presentes no Apêndice A, nas Figuras A.1 a A.3, filtradas para a disciplina de Algoritmos e Programação de Computadores e para alunos com cotas.

#### 4.1.6 Página de Informações de Percepção

Na página de Informações de Percepção estão os gráficos correspondentes aos dados resultantes dos formulários de percepção aplicados nas turmas de Algoritmos e Programação de Computadores nos semestres de 2019/2 e 2020/1.

Estão presentes visualizações acerca de:

- Contato prévio com programação, descrito no gráfico de barras;
- Linguagem de programação utilizada anteriormente, descrita no gráfico de pizza;
- Relação entre área de trabalho dos pais com o nível educacional dos pais, descrita no gráfico de barras;
- Frequência da associação entre Engenharia/Computação "ser para pessoas muito inteligentes", descrita no gráfico de barras;
- Frequência da associação entre Engenharia/Computação "ser para homens", descrita no gráfico de pizza.

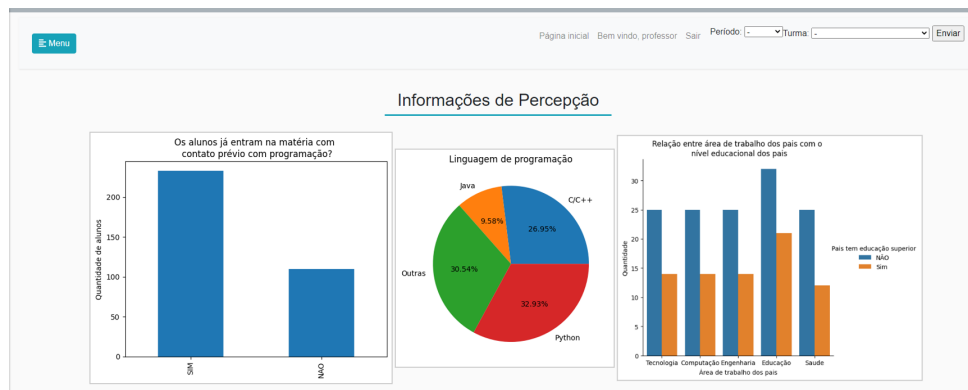


Figura 4.11: Gráficos em Informações de Percepção para a Disciplina de APC.

#### 4.1.7 Página de Gráficos Interativos

Na página de Gráficos Interativos, são disponibilizadas visualizações realizadas através de outra biblioteca (Dash), com o objetivo de responder algumas questões mais específicas, descritas como:

Tabela 4.2: Detalhamento das Colunas dos Gráficos.

<b>Nome</b>	<b>Detalhamento</b>
gênero	Gênero do aluno
area_trabalho_pais	Área de trabalho dos pais
pais_tem_ensino_superior	Resposta à "Seu pai e/ou sua mãe possui curso superior?"
eh_cotista	Aluno possui ou não cotas
horas_estudo	Quantidade de horas que o aluno dedicou semanalmente à disciplina de APC
precisa_mais_horas	Percepção do próprio aluno sobre precisar ou não de mais horas de estudo que seus amigos de sala

- Relação entre gênero do aluno e área de trabalho dos pais;
- Relação entre nível educacional dos pais e cotas de alunos;
- Relação entre a quantidade de horas de estudo do aluno com a sua percepção de precisar ou não de mais horas de estudo.

Estas visualizações são todas apresentadas no formato de gráficos de coordenadas paralelas, descritas para o Departamento de Ciência da Computação junto do curso de Engenharia Mecatrônica e, separadamente, para cada um dos cursos que compõem o departamento (Engenharia da Computação, Ciência da Computação, Licenciatura em Computação) e Engenharia Mecatrônica, para os semestres de 2019/2 e 2020/1 da disciplina de Algoritmos e Programação de Computadores. As colunas utilizadas nos gráficos estão detalhadas na Tabela 4.2. As Figuras 4.12 a 4.17 exibem os gráficos no contexto geral (Departamento de Ciência da Computação e Engenharia Mecatrônica) e os demais são apresentados no Apêndice A.

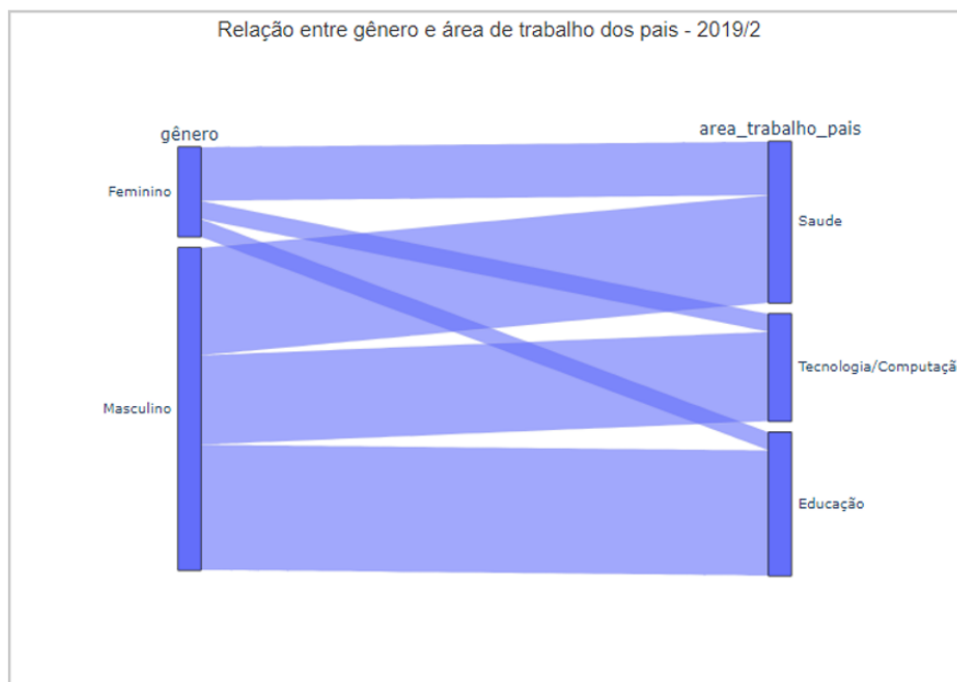


Figura 4.12: Gráficos Interativos para a Disciplina de APC - Departamento de Ciência da Computação e Engenharia Mecatrônica.



Figura 4.13: Gráficos Interativos para a Disciplina de APC - Departamento de Ciência da Computação e Engenharia Mecatrônica.

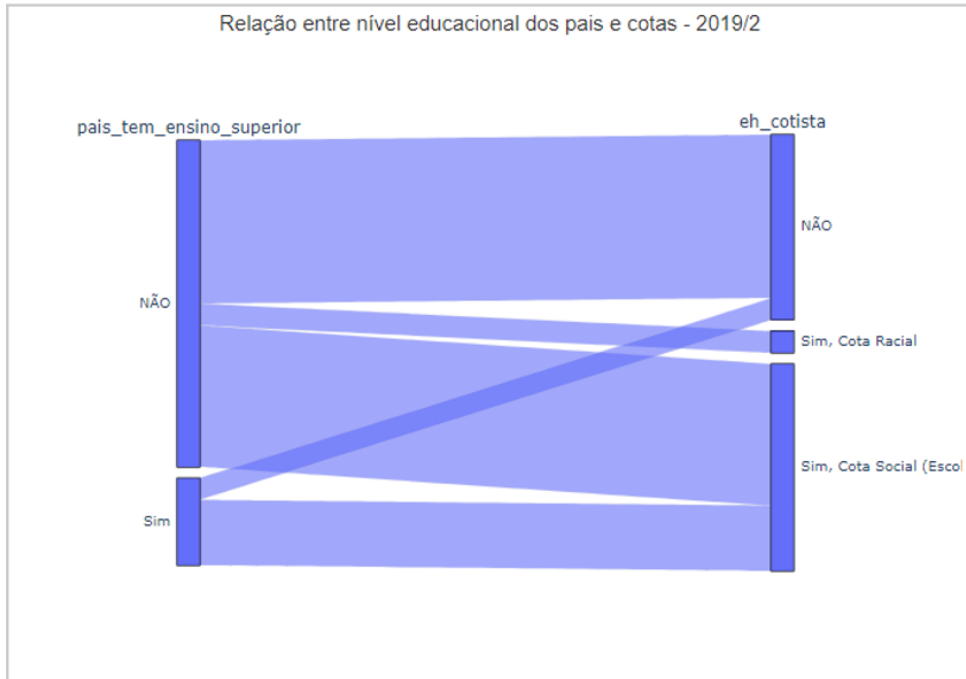


Figura 4.14: Gráficos Interativos para a Disciplina de APC - Departamento de Ciência da Computação e Engenharia Mecatrônica.

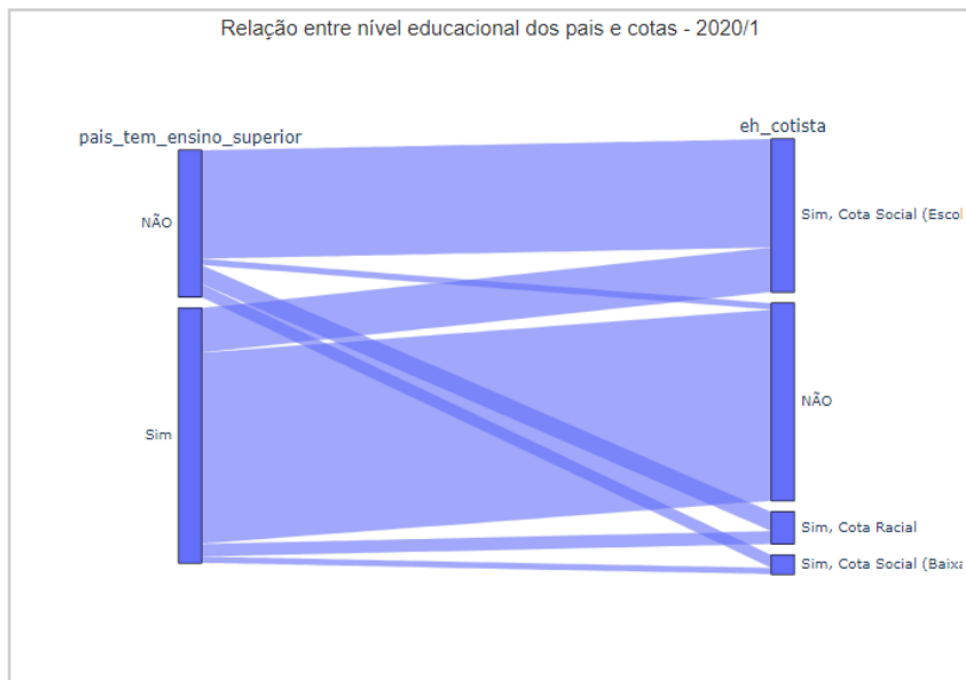


Figura 4.15: Gráficos Interativos para a Disciplina de APC - Departamento de Ciência da Computação e Engenharia Mecatrônica.



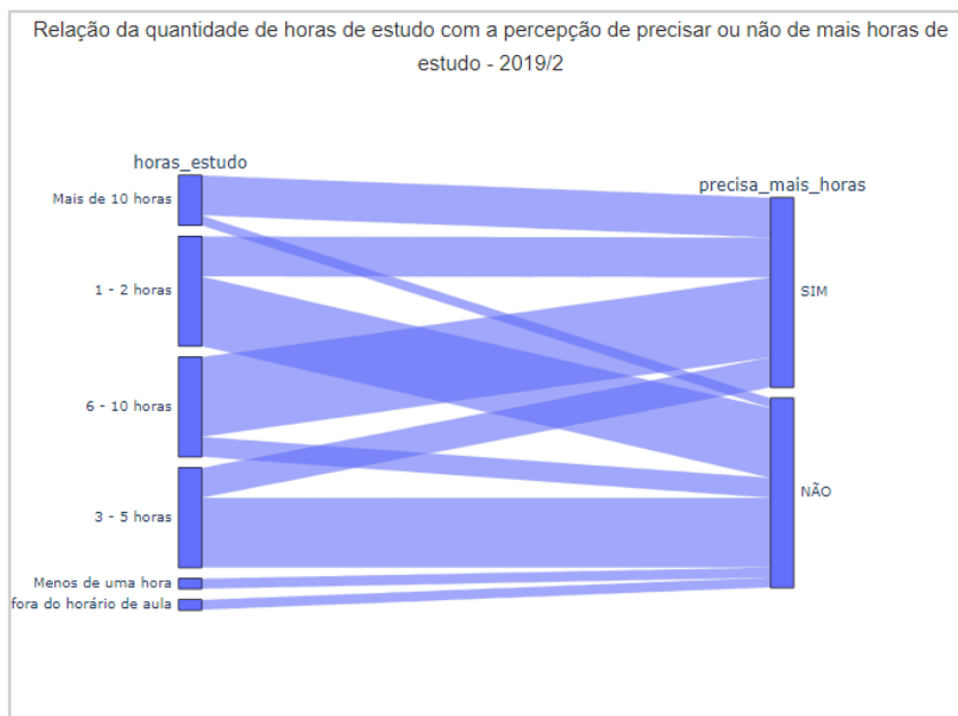


Figura 4.16: Gráficos Interativos para a Disciplina de APC - Departamento de Ciência da Computação e Engenharia Mecatrônica.

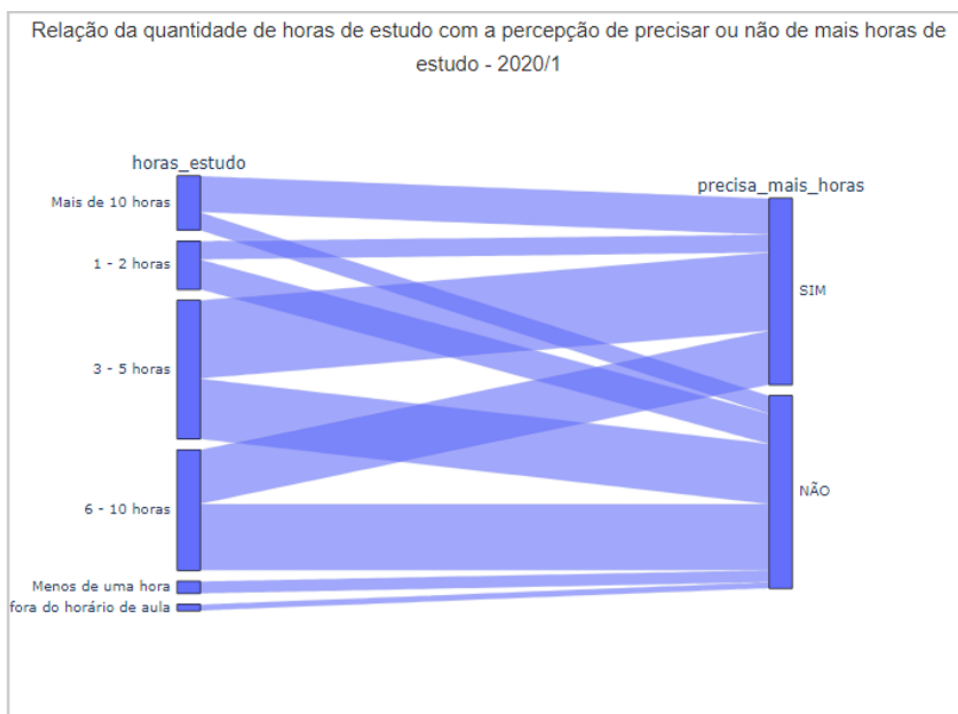


Figura 4.17: Gráficos Interativos para a Disciplina de APC - Departamento de Ciência da Computação e Engenharia Mecatrônica.

# Capítulo 5

## Conclusão

Este trabalho descreve a criação do Observatório de Dados Educacionais, um sistema para embarcar gráficos baseados em algoritmos de visualização para dados educacionais dos cursos introdutórios de programação do Departamento de Ciência da Computação da Universidade de Brasília, integrado com o trabalho de mestrado da aluna Luiza Hansen. A partir da transformação e carga dos dados provenientes do SIGRA e do formulário de percepção aplicado, foram criados os bancos de dados e as visualizações no sistema, permitindo verificar os indicadores do histórico das turmas de programação junto da utilização de filtros para diferentes cenários.

O ensino da primeira linguagem de programação é estudado há décadas, onde diversos fatores sociais são analisados associadamente às questões de altas taxas de reprovação e desistência, como a recorrência deste indicador em grupos que fazem parte de minorias. A utilização de ferramentas para visualização auxilia na compreensão dos dados e na transmissão de informação.

A partir do protótipo do sistema e dos algoritmos de visualização escolhidos, foi iniciado o processo de desenvolvimento, utilizando-se das tecnologias Python, Django, HTML, CSS, MySQL, Atom, Github e o serviço de hospedagem Pythonanywhere. Foi elaborada a lógica para a criação da ferramenta de filtros, utilizando-se uma tabela de usuários no banco de dados para a atualização dinâmica das *queries* dos algoritmos dos gráficos.

Além disso, o sistema possui uma etapa de autenticação, visando proteger os dados, permitindo a utilização do sistema apenas por contas aprovadas por um usuário administrador, limitando o acesso para professores, coordenadores e chefes de departamento.

O Observatório foi capaz de exibir as análises gráficas de maneira dinâmica com relação aos filtros, separados nas seções de Informações Gerais, Informações Acadêmicas, Informações de Percepção e Gráficos Interativos, facilitando a visualização das informações por parte dos docentes.

Como continuação deste trabalho, é sugerida a implementação de uma ferramenta que permita a atualização dos bancos de dados utilizados através do *upload* de arquivos, visando o progresso conforme novos semestres forem acontecendo e, deste modo, novas informações forem surgindo, tanto com relação aos dados de desempenho quanto com relação aos formulários de percepção aplicados semestralmente nas turmas de Algoritmos e Programação de Computadores. Em um primeiro momento, algumas dificuldades foram encontradas na implementação desta ferramenta, envolvendo a questão da necessidade recorrente de alteração dos nomes dos arquivos (a cada semestre, um novo arquivo com um novo nome deve ser adicionado) no algoritmo responsáveis por popular o banco de dados já existente.

Além disso, outra ideia consiste em incrementar as possibilidades de análises. É sugerida a utilização de outras variáveis presentes no *dataset* como, por exemplo, a forma de ingresso na Universidade ou o tipo de escola em que o aluno estudou, bem como a extensão do sistema para outras disciplinas dos currículos dos cursos do Departamento de Ciência da Computação. Ademais, é sugerida a possibilidade de correlacionar os resultados de algumas visualizações no próprio sistema, permitindo uma análise mais específica, por exemplo, na questão da "Relação da quantidade de horas de estudo com a percepção de precisar ou não de mais horas de estudo", respondida por alunos, com o desempenho observado nos gráficos de Informações Acadêmicas.

# Referências

- [1] Hansen, Luiza A.: *Análise visual de dados educacionais: Um estudo de caso das disciplinas introdutórias de programação da unb*. Tese de Mestrado, Programa de Pós-graduação em Informática, Universidade de Brasília, 2021. ix, xii, 2, 6, 10, 18, 19, 25
- [2] George, N: *Mastering Django*. GNW Independent Publishing, 2020. ix, 22
- [3] *Cpa/unb - comissão própria de avaliação da unb - relatório do perfil dos estudantes*. [http://cpa.unb.br/index.php?option=com\\_content&view=article&id=456&Itemid=305](http://cpa.unb.br/index.php?option=com_content&view=article&id=456&Itemid=305). Acesso em: Maio, 2021. 1
- [4] Holanda, Maristela, Marília Dantas, Gustavo Couto, Jan M. Correa, Aleteia P. F. Araújo e Maria E. T. Walter: *Perfil das alunas no departamento de computação da universidade de Brasília*. XXXVII Congresso da Sociedade Brasileira de Computação. 11º WIT - Women in Information Technology1209, 2017. 1
- [5] Beaubouef, Theresa e John Mason: *Why the high attrition rate for computer science students: Some thoughts and observations*. The SIGCSE Bulletin, 2005. 1
- [6] Walker, G: *Experimentation in the computer programming lab*. Inroads, 2004. 1
- [7] Brito, Maria T S, Francisco P A Medeiros, Ed P Bezerra e Alex S R Barbosa: *Contribuições de um plugin do tipo report para a identificação do risco de evasão no ava moodle com base em visualização de dados*. Revista Brasileira de Informática na Educação – RBIE, 2019. 1
- [8] Luxton-Reilly, A., B. A. Becker, L. OTT, Simon, M. Giannakos, J. Paterson, I. Albluwi, A. N. Kumar, M. J. Scott e J. Sheard: *Introductory programming: A systematic literature review*. in Proceedings of the 2014 conference on Innovation technology in computer science education (ITiCSE '14), 2014. 4
- [9] Maia, M. M.: *Limites de gênero e presença feminina nos cursos superiores brasileiros do campo da computação*. 2016. 4
- [10] Lishinski, Alex, Aman Yadav, Jon Good e Richard Enbody: *Learning to program: Gender differences and interactive effects of students' motivation, goals, and self-efficacy on performance*. 2016. <https://doi.org/10.1145/2960310.2960329>. 4
- [11] Becker, Brett A e Keith Quille: *50 years of cs1 at sigcse: A review of the evolution of introductory programming education research*. SIGCSE, 2019. 4

- [12] *Lei nº 12.711, de 29 de agosto de 2012.* <https://www.gov.br/planalto/pt-br>. Acesso em: Novembro, 2021. 5
- [13] Holanda, Maristela, George von Borries, Dilma da Silva, Camilo Dorea, Roberta B. Oliveira e Edison Ishikawa: *The intellectual sense of belonging and self-efficacy in the introduction to computer science courses at university of brasilia in brazil.* páginas 1–8, 2020. 5
- [14] Watson, Christopher e Frederick W B Li: *Failure rates in introductory programming revisited.* in Proceedings of the 2014 conference on Innovation technology in computer science education (ITiCSE '14), 2014. 5
- [15] Saa, Amjad A: *Educational data mining students' performance prediction.* International Journal of Advanced Computer Science and Applications, 2016. 5
- [16] Bajaj, C.: *Data visualization techniques.* John Wiley Sons Ltd, 1998. 6
- [17] SAS: *Data visualization techniques: From basics to big data.* SAS® Visual Analytics, 2014. 6
- [18] *The radar chart and its caveats.* <https://www.data-to-viz.com/caveat/spider.html>. Acesso em: Setembro, 2021. 6
- [19] *Mozilla developer network web docs.* <https://developer.mozilla.org/pt-BR/docs/Web/HTML>. Acesso em: Setembro, 2021. 12
- [20] *Mozilla developer network web docs.* <https://developer.mozilla.org/pt-BR/docs/Web/CSS>. Acesso em: Setembro, 2021. 12

# Apêndice A

## Figuras Complementares de Gráficos Interativos

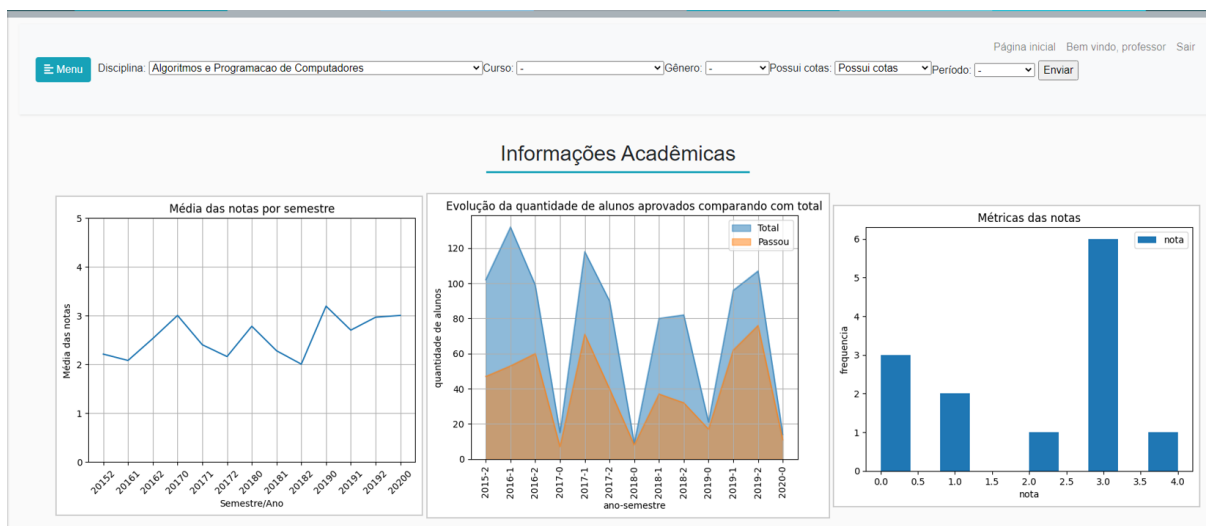


Figura A.1: Gráficos Filtrados em Informações Acadêmicas para a disciplina de APC.

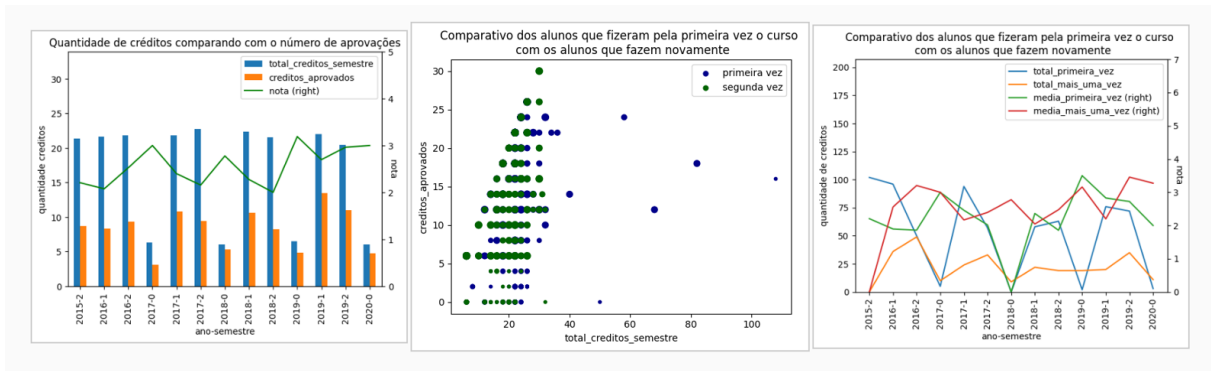


Figura A.2: Gráficos Filtrados em Informações Acadêmicas para a disciplina de APC.

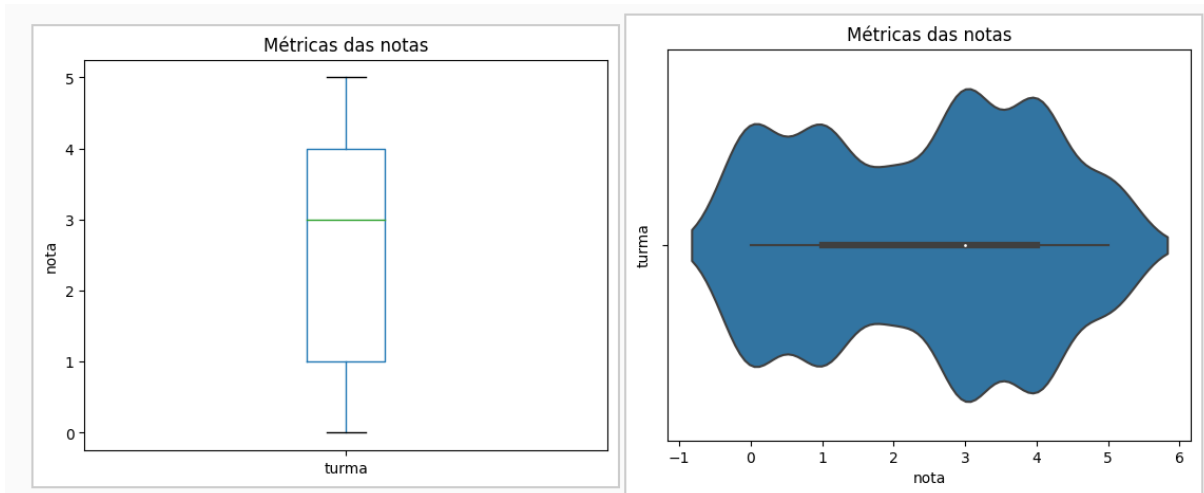


Figura A.3: Gráficos Filtrados em Informações Acadêmicas para a disciplina de APC.

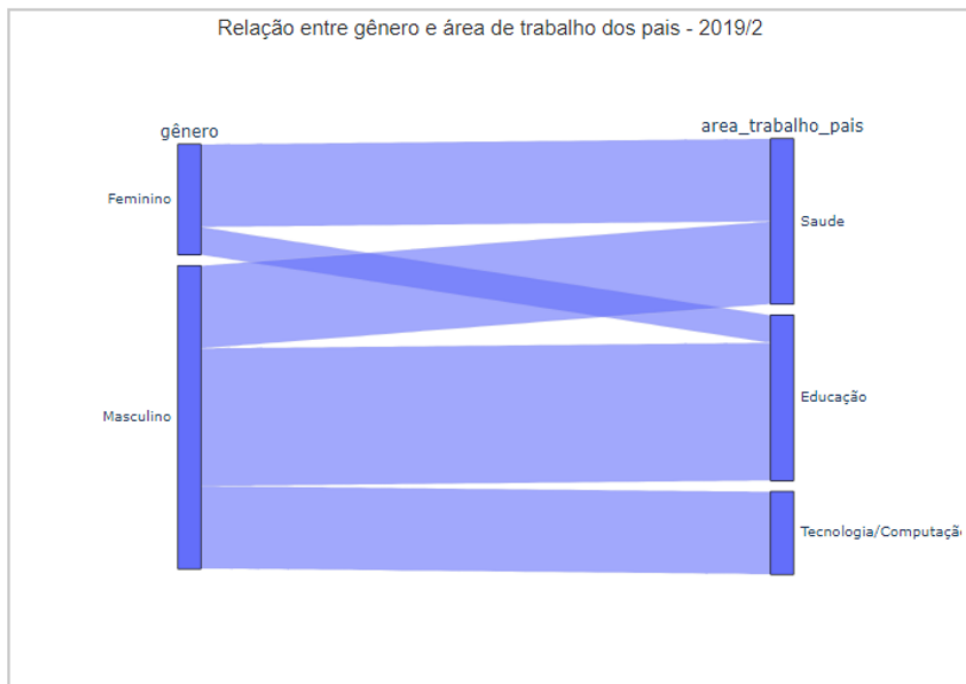


Figura A.4: Gráficos Interativos para a Disciplina de APC - Curso de Ciência da Computação.

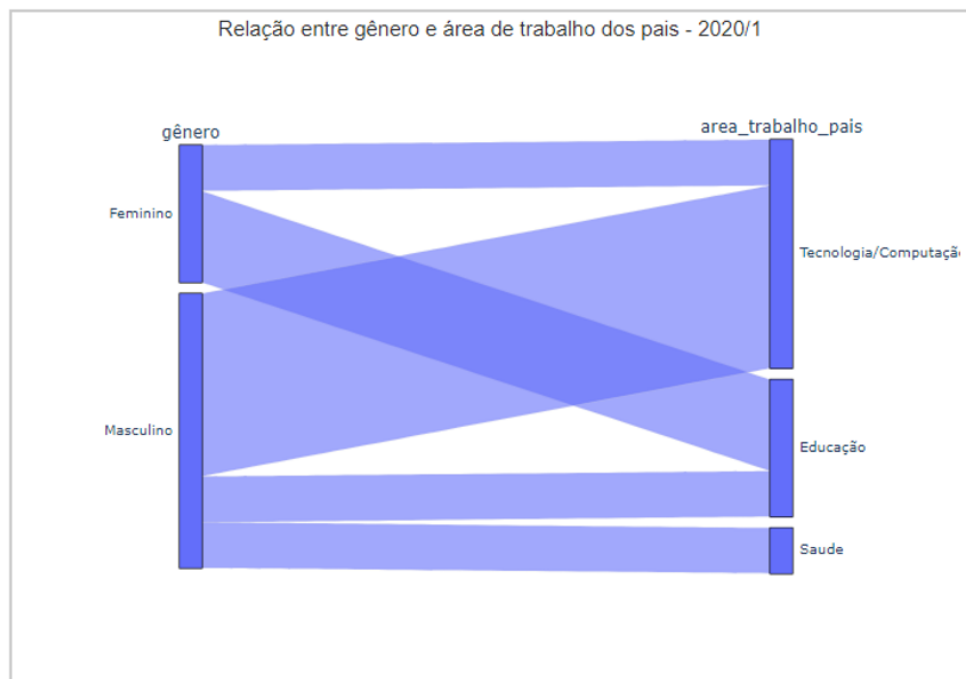


Figura A.5: Gráficos Interativos para a Disciplina de APC - Curso de Ciência da Computação.



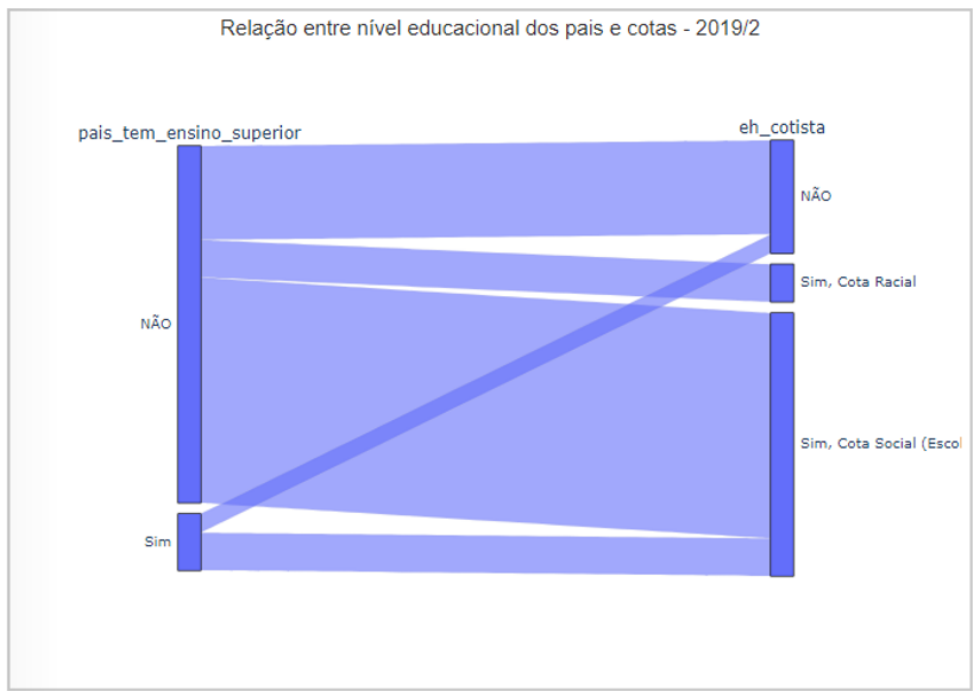


Figura A.6: Gráficos Interativos para a Disciplina de APC - Curso de Ciência da Computação.

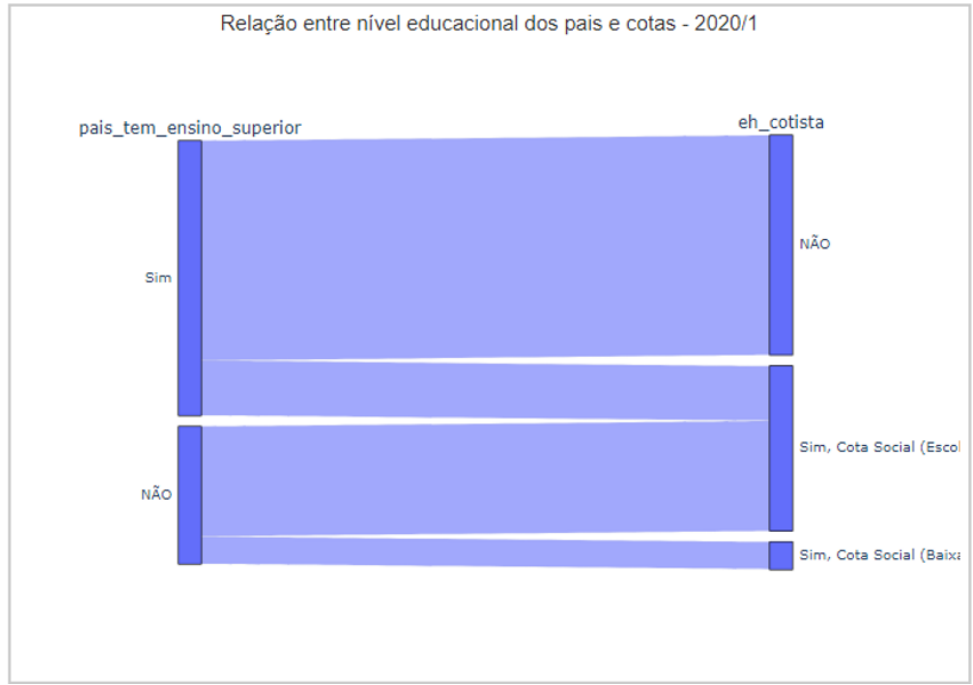


Figura A.7: Gráficos Interativos para a Disciplina de APC - Curso de Ciência da Computação.

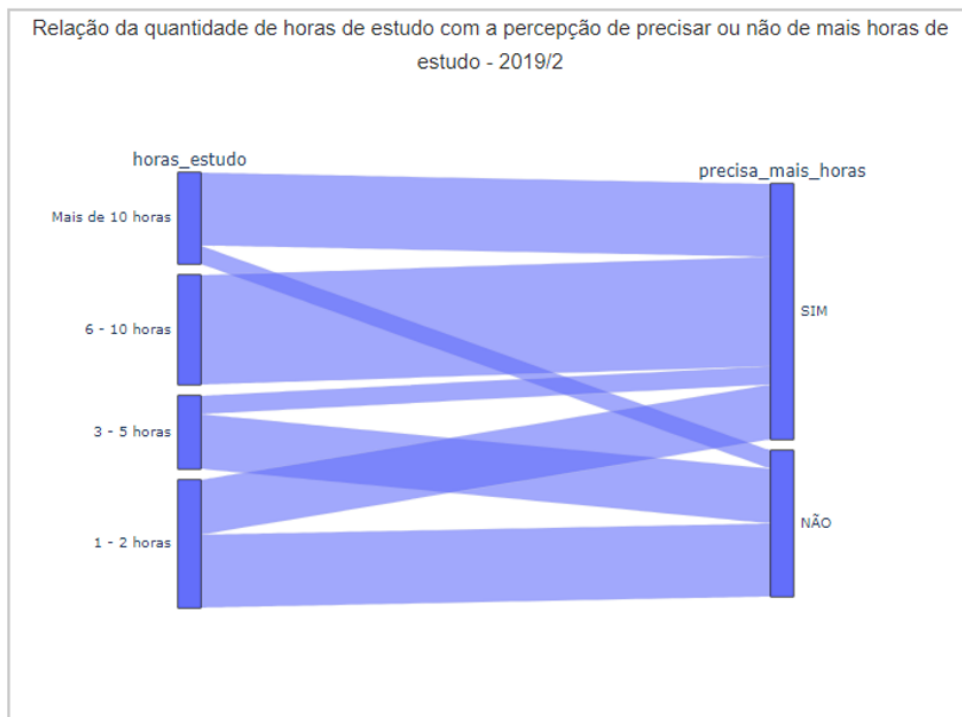


Figura A.8: Gráficos Interativos para a Disciplina de APC - Curso de Ciência da Computação.



Figura A.9: Gráficos Interativos para a Disciplina de APC - Curso de Ciência da Computação.

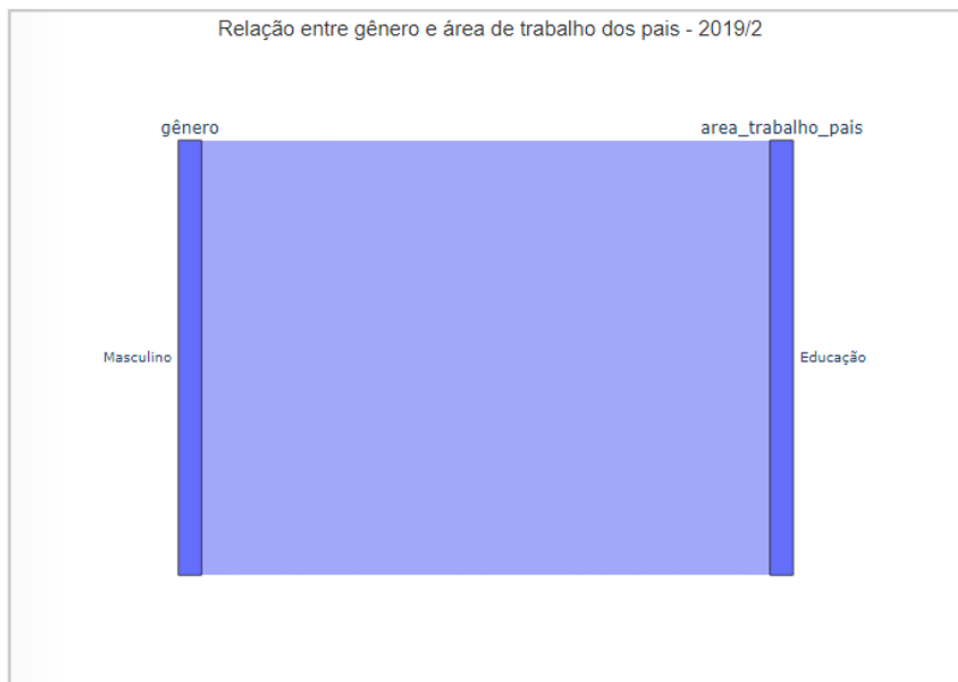


Figura A.10: Gráficos Interativos para a Disciplina de APC - Curso de Computação (Licenciatura).

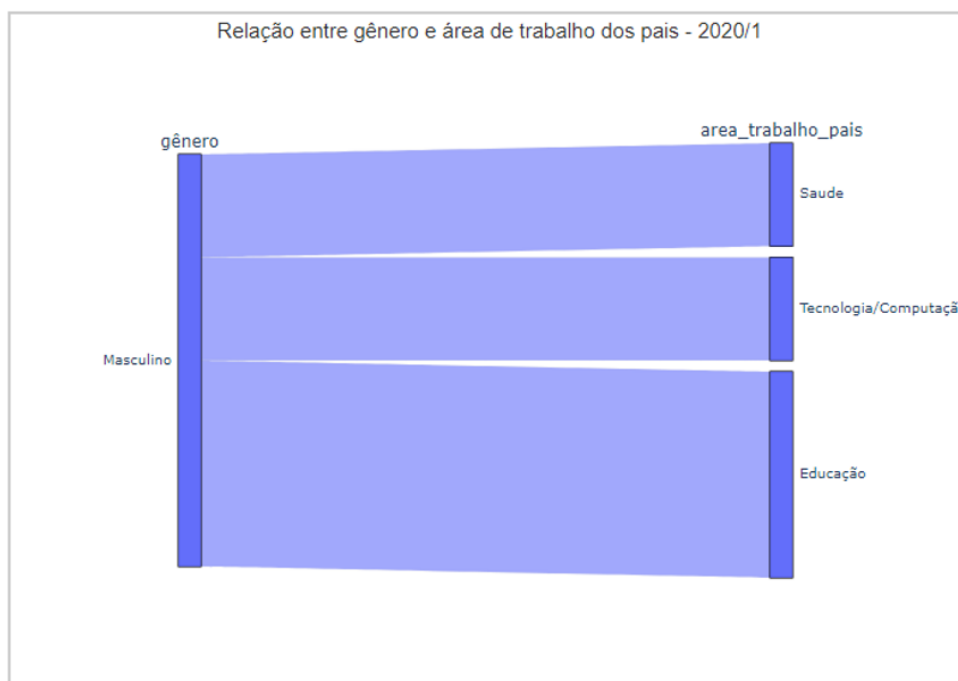


Figura A.11: Gráficos Interativos para a Disciplina de APC - Curso de Computação (Licenciatura).

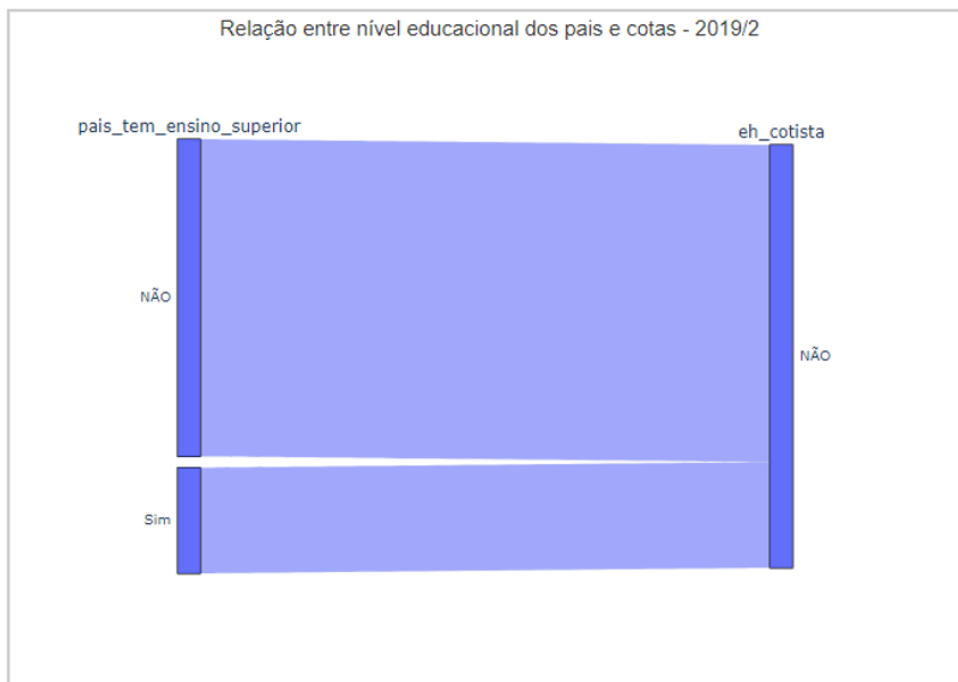


Figura A.12: Gráficos Interativos para a Disciplina de APC - Curso de Computação (Licenciatura).

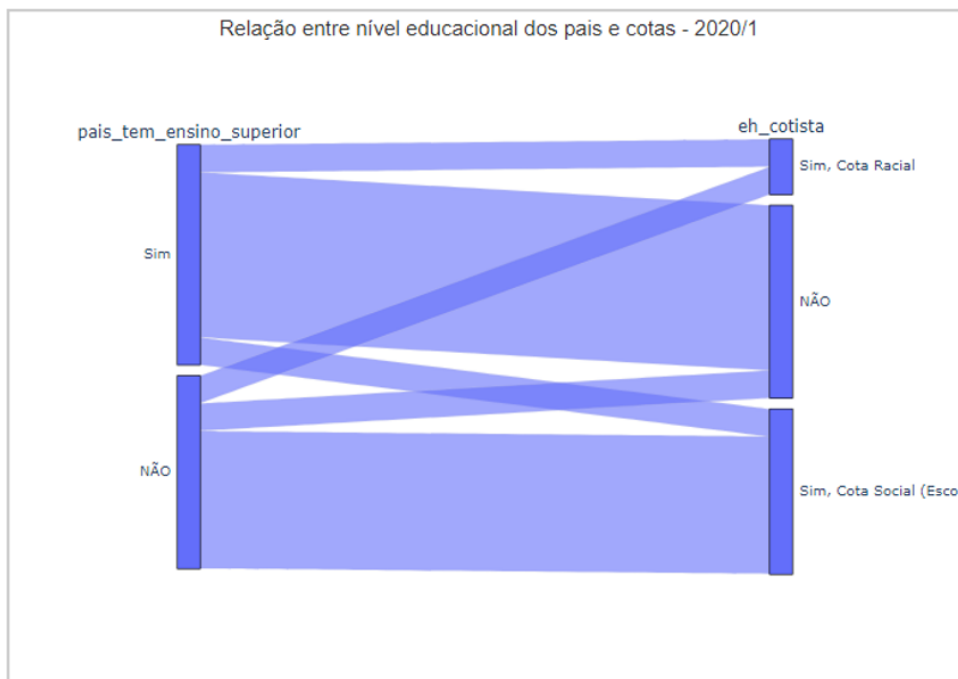


Figura A.13: Gráficos Interativos para a Disciplina de APC - Curso de Computação (Licenciatura).

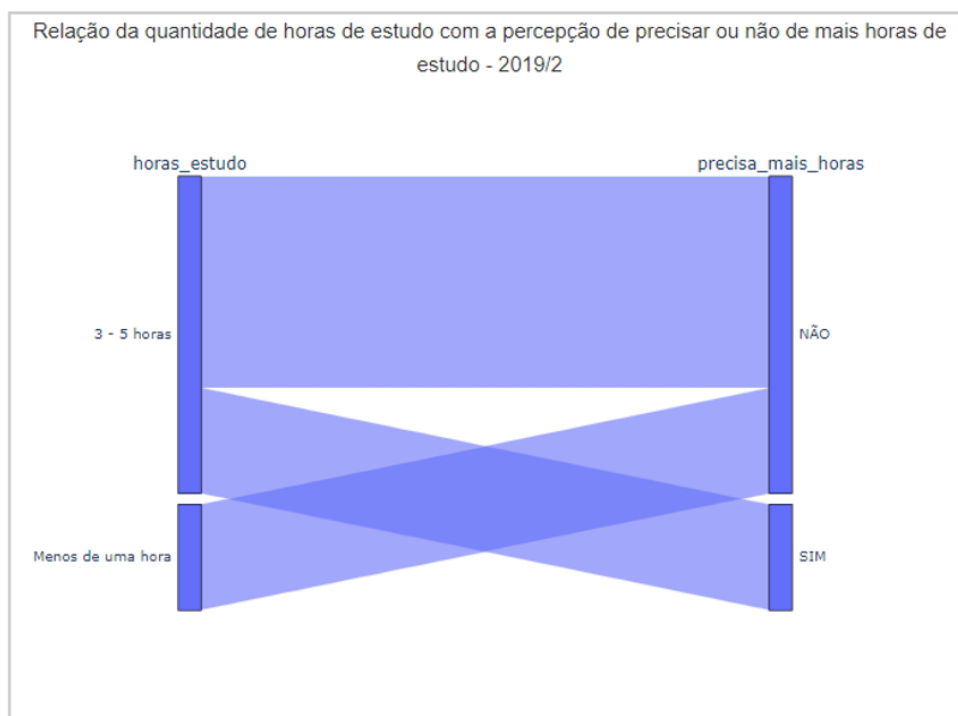


Figura A.14: Gráficos Interativos para a Disciplina de APC - Curso de Computação (Licenciatura).

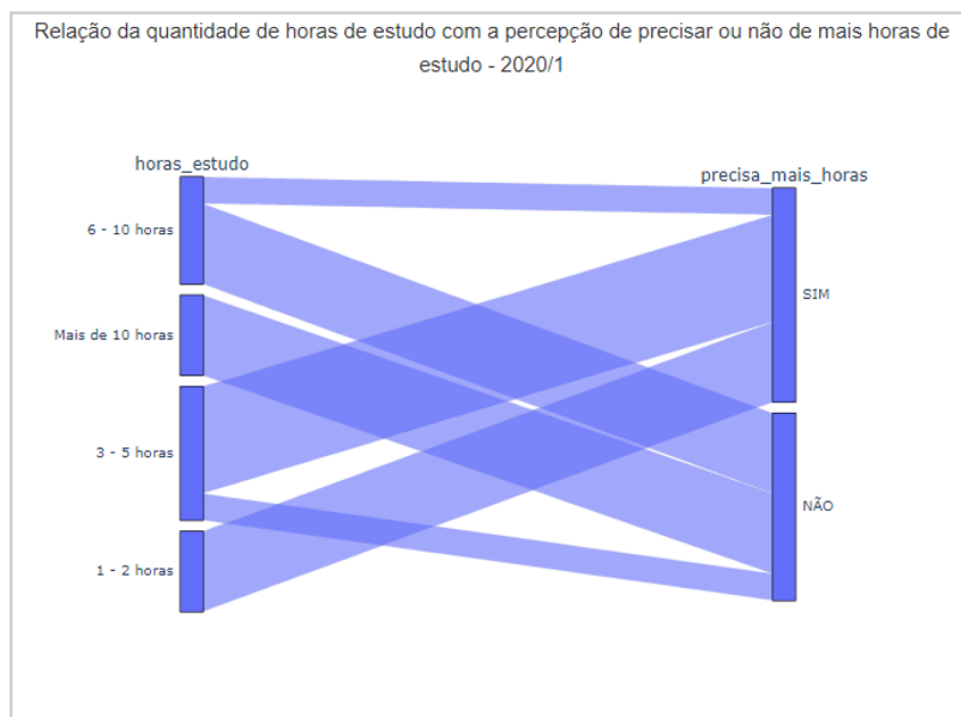


Figura A.15: Gráficos Interativos para a Disciplina de APC - Curso de Computação (Licenciatura).

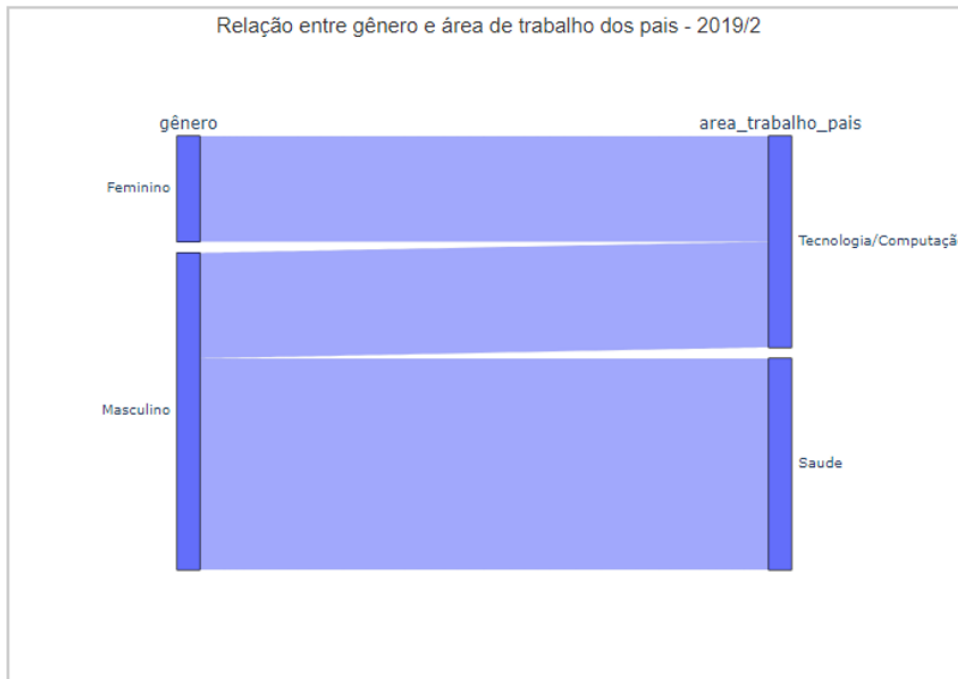


Figura A.16: Gráficos Interativos para a Disciplina de APC - Curso de Engenharia da Computação.

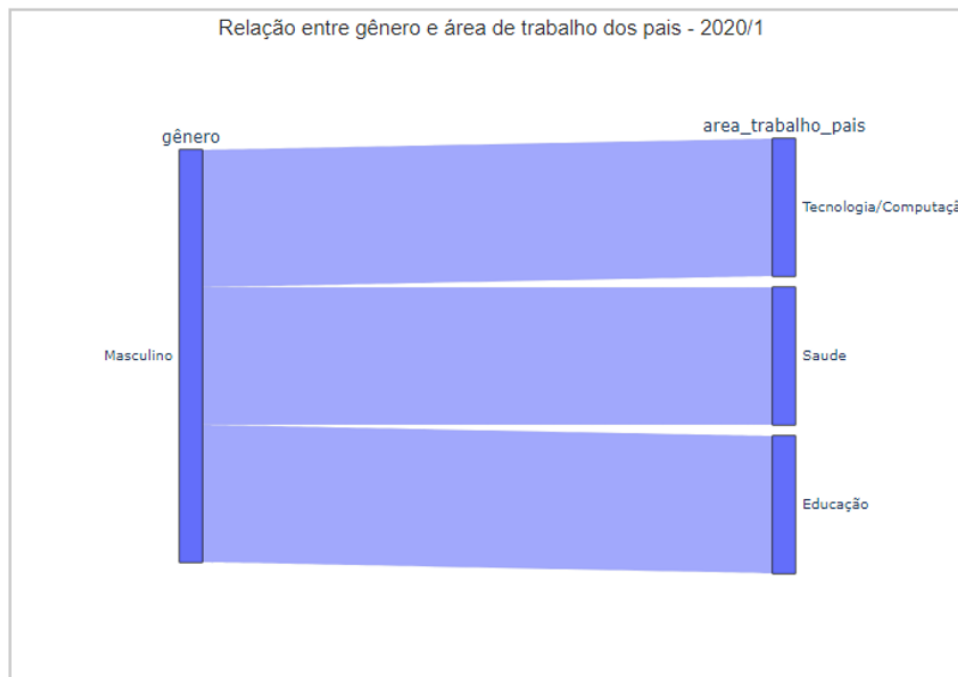


Figura A.17: Gráficos Interativos para a Disciplina de APC - Curso de Engenharia da Computação.

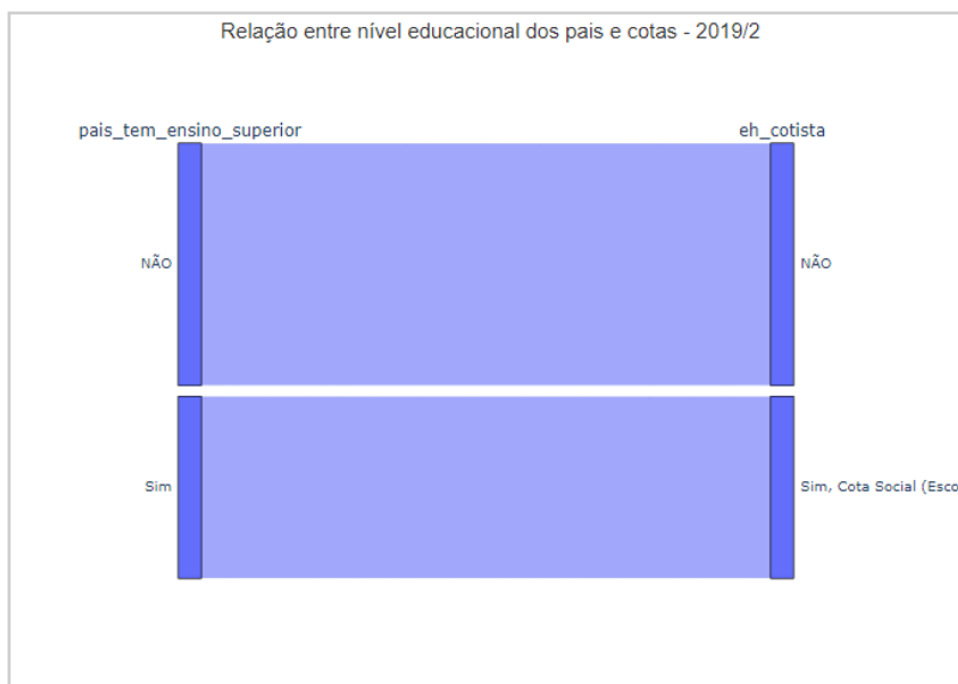


Figura A.18: Gráficos Interativos para a Disciplina de APC - Curso de Engenharia da Computação.

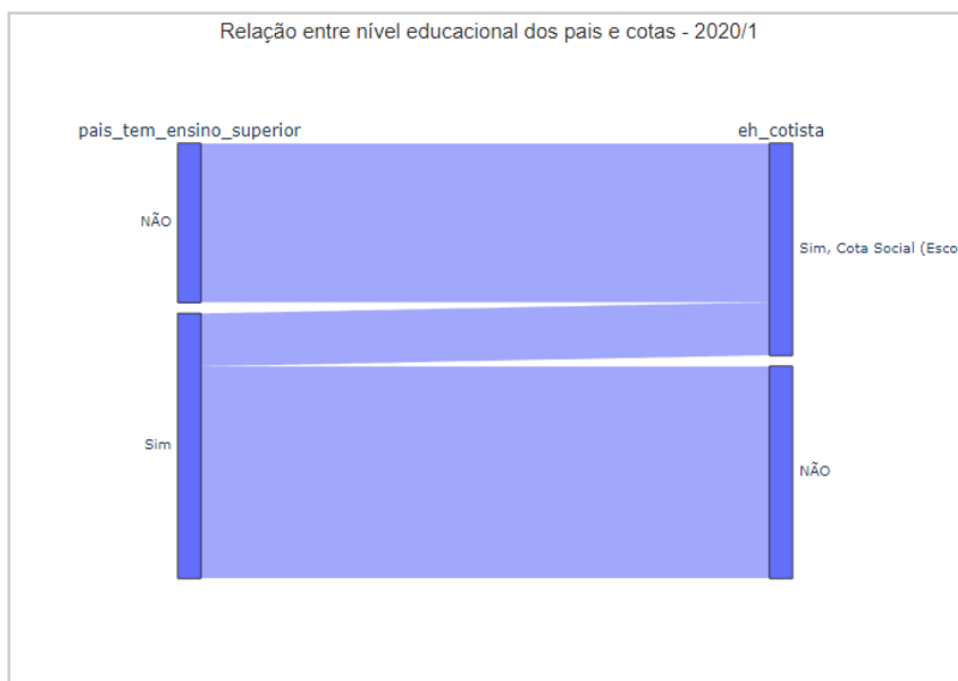


Figura A.19: Gráficos Interativos para a Disciplina de APC - Curso de Engenharia da Computação.

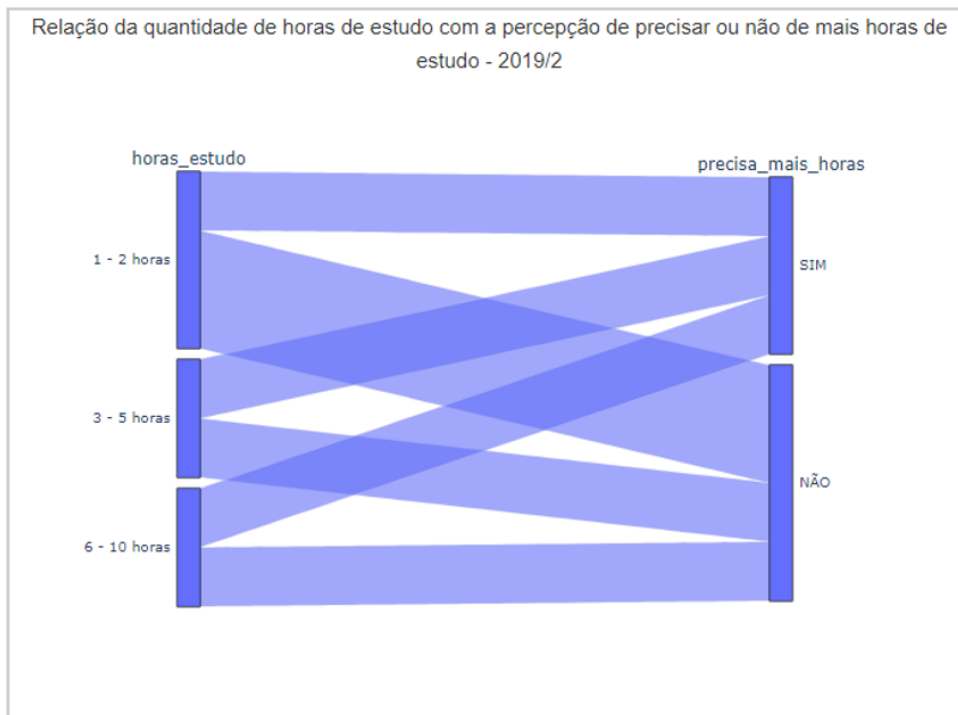


Figura A.20: Gráficos Interativos para a Disciplina de APC - Curso de Engenharia da Computação.

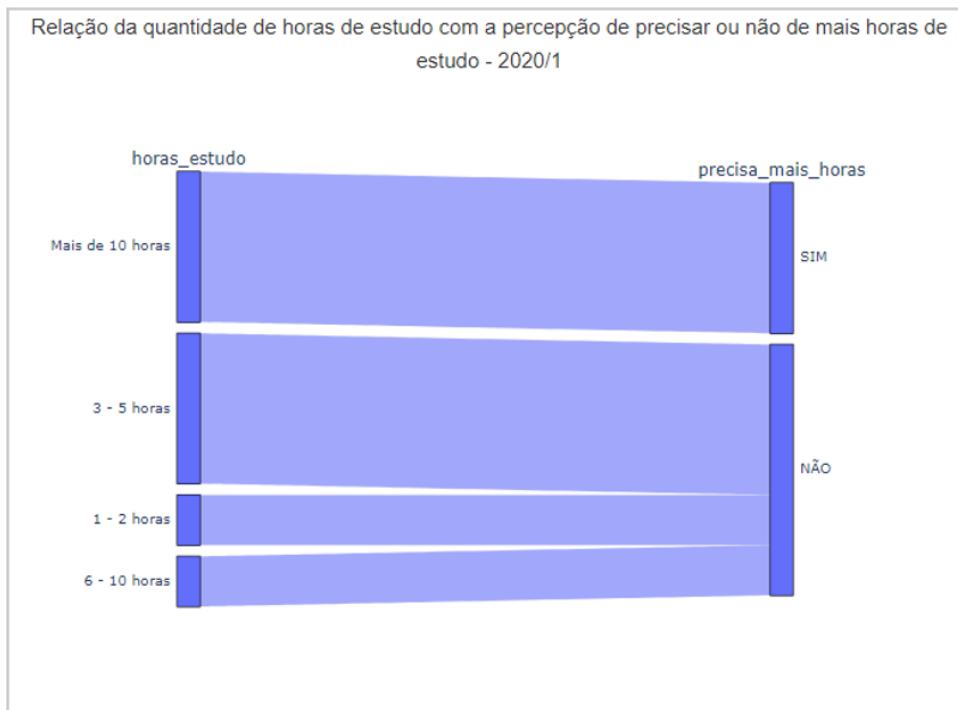


Figura A.21: Gráficos Interativos para a Disciplina de APC - Curso de Engenharia da Computação.



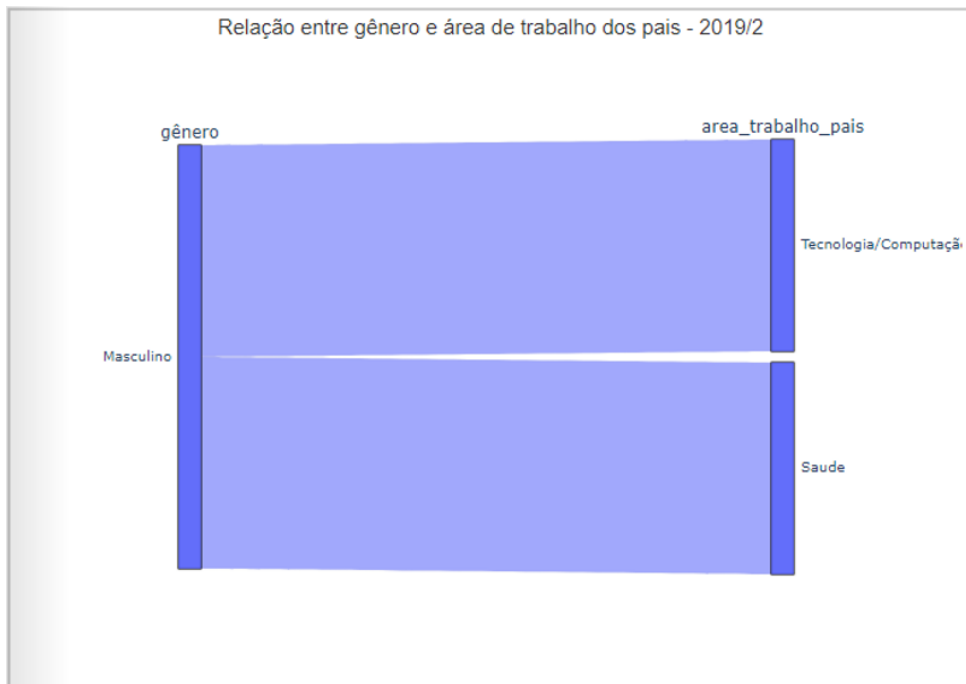


Figura A.22: Gráficos Interativos para a Disciplina de APC - Curso de Engenharia Me-  
catrônica.

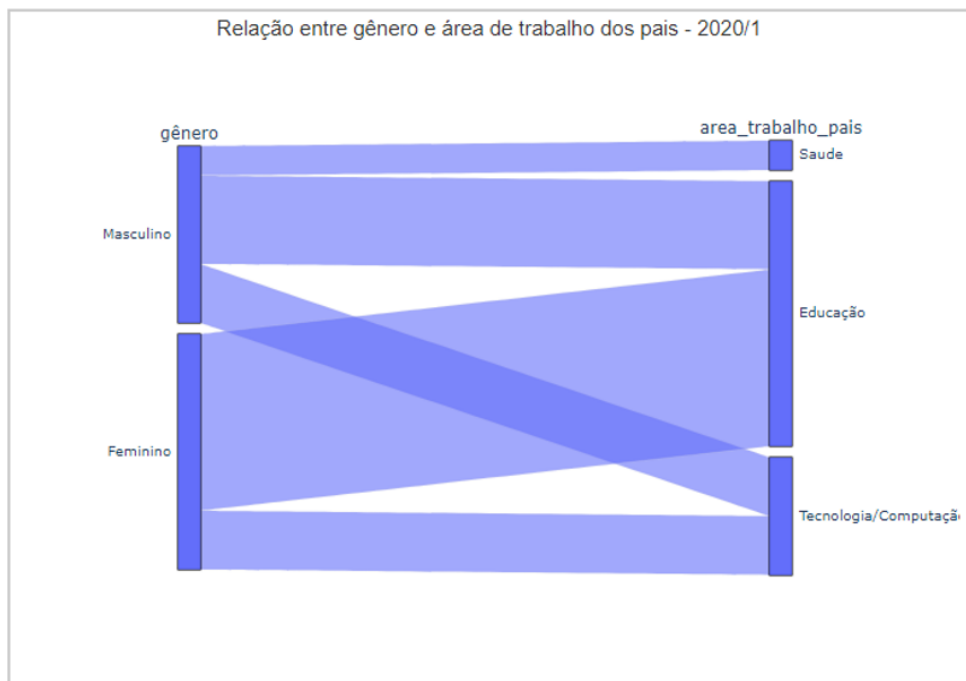


Figura A.23: Gráficos Interativos para a Disciplina de APC - Curso de Engenharia Me-  
catrônica.

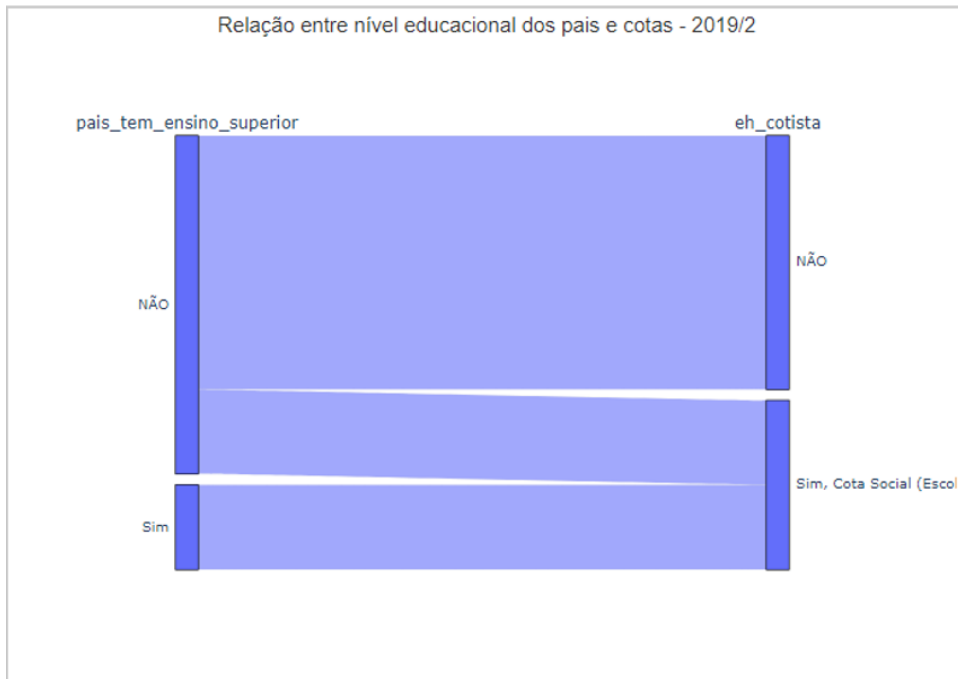


Figura A.24: Gráficos Interativos para a Disciplina de APC - Curso de Engenharia Me-  
catrônica.

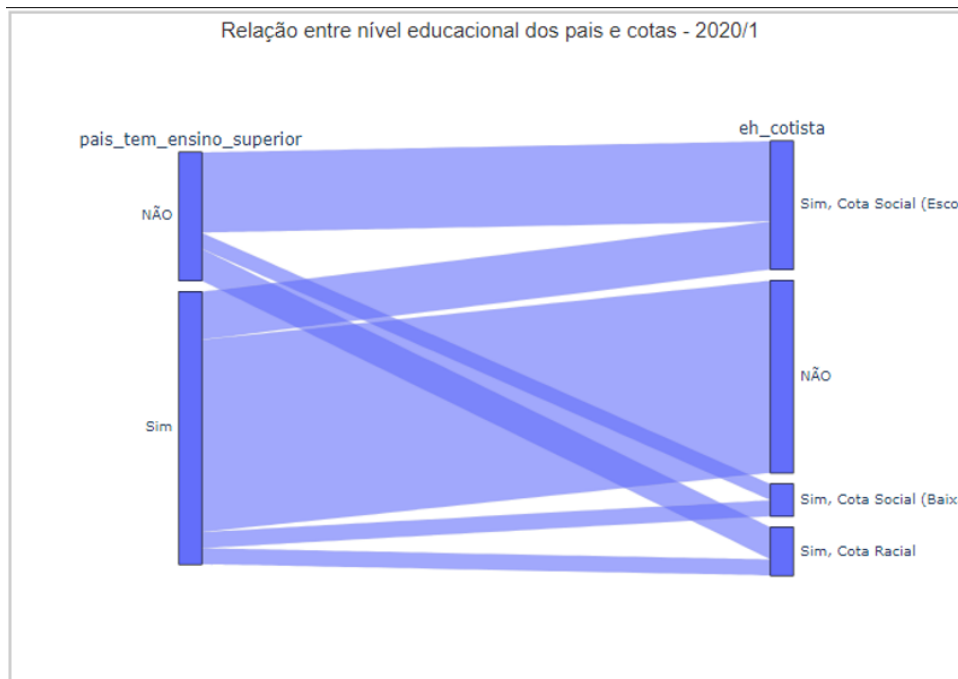


Figura A.25: Gráficos Interativos para a Disciplina de APC - Curso de Engenharia Me-  
catrônica.

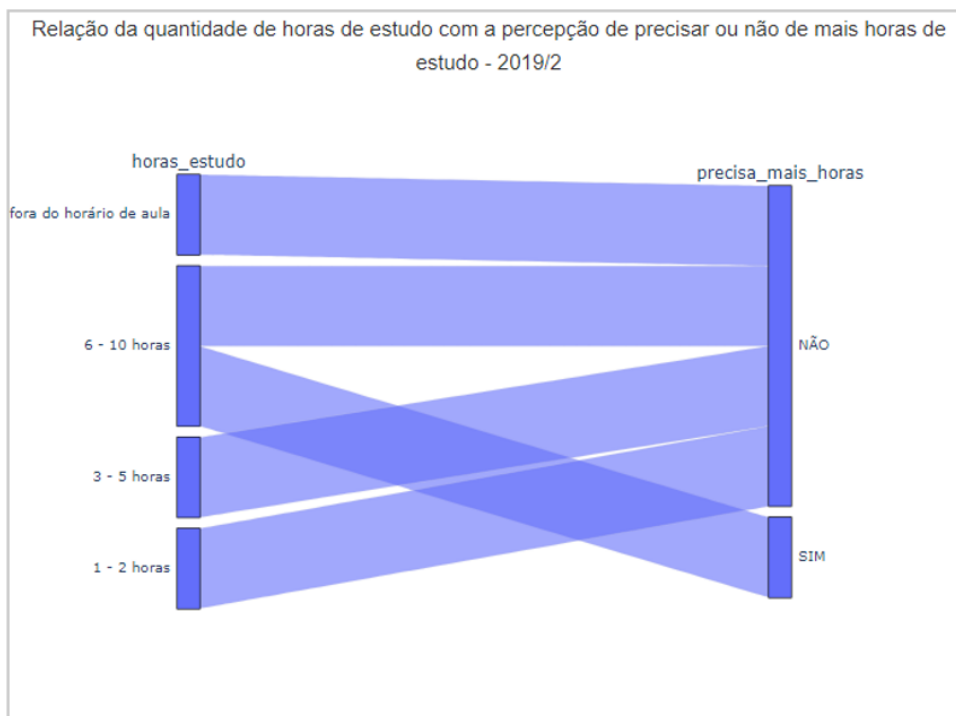


Figura A.26: Gráficos Interativos para a Disciplina de APC - Curso de Engenharia Me-  
catrônica.

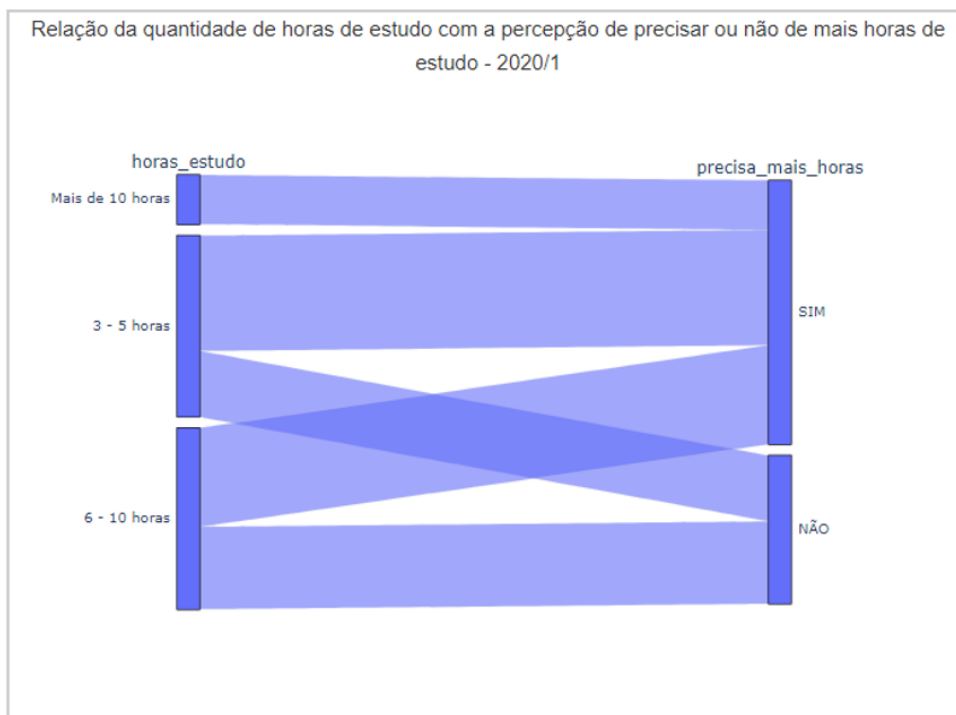


Figura A.27: Gráficos Interativos para a Disciplina de APC - Curso de Engenharia Me-  
catrônica.