

IFSP - Instituto Federal de Educação, Ciência e Tecnologia de  
São Paulo

Campus São Paulo

## **Projeto Integrado I - PI1A5**

MVP - Mínimo Produto Viável

Beatriz Andrade - SP3098991

Isadora Vieira Câmara - SP3094383

Letícia Gonçalves Baião - SP3098818

Suanne Barbosa - SP3099067

Tamiris Delfino de Faria Jesus - SP3095339

Professor: Johnata Souza Santicioli

IFSP - Instituto Federal de Educação, Ciência e Tecnologia de São Paulo

Campus São Paulo

São Paulo, SP

Julho de 2024

Beatriz Andrade - SP3098991  
Isadora Vieira Câmara - SP3094383  
Letícia Gonçalves Baião - SP3098818  
Suanne Barbosa - SP3099067  
Tamiris Delfino de Faria Jesus - SP3095339

## **Projeto Integrado I - PI1A5**

MVP - Mínimo Produto Viável

Julho de 2024

# Histórico de Revisões

Revisão	Data	Autores	Descrição
1.0	06-04-2024	Tamiris, Beatriz, Suanne, Letícia, Isadora	Primeira versão, entrega do tema
2.0	08-05-2024	Tamiris, Beatriz, Suanne, Letícia, Isadora	Segunda versão, entrega da prova de conceito
3.0	23-07-2024	Tamiris, Beatriz, Suanne, Letícia, Isadora	Terceira versão, entrega do MVP

# RESUMO

A escolha do curso superior é um momento crucial na vida de muitos jovens, representando um marco significativo na trajetória educacional e profissional em suas vidas. Porém, o momento de transição da escola para a faculdade se torna um desafio para muitos estudantes da rede pública de ensino no Brasil, que frequentemente enfrentam barreiras socioeconômicas e não possuem o mesmo nível de acesso à informação quanto um estudante da rede particular, dificultando assim a tomada de decisões com relação ao seu futuro, e possivelmente gerando uma falta de perspectiva com relação ao acesso ao ensino superior de qualidade. Segundo pesquisa do [Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira \(INEP\)](#), em 2020, apenas 31,7% dos jovens concluintes do ensino médio em escolas públicas ingressaram no ensino superior. Esse dado revela a necessidade de medidas com o objetivo de facilitar o acesso à informação e apoiar esses estudantes no momento da escolha de qual caminho seguir academicamente, com o intuito de abrir as portas das universidades públicas para essa parcela da população. Nesse contexto surge a Plataforma Vocco, que se propõe a ser uma ferramenta de apoio a esses jovens, fornecendo um guia a respeito de qual o curso mais indicado para o respectivo perfil de cada estudante, utilizando-se de um teste vocacional baseado na teoria do cientista John Holland, que mapeia as seis principais dimensões de interesses vocacionais, para designar o estudante à carreira mais adequada. Ademais, a Vocco se propõe a auxiliar o estudante com informações de quais faculdades públicas do país oferecem tais cursos, e como o estudante irá se manter no ensino superior, com informações de políticas de cotas e políticas de permanência (auxílios, moradia) das respectivas universidades. A metodologia adotada para o desenvolvimento da Vocco envolveu a análise da literatura existente para fundamentar as funcionalidades da plataforma e uma abordagem ágil para a criação e teste do protótipo, assegurando que a aplicação atenda às necessidades identificadas e seja eficaz na melhoria do acesso ao ensino superior. A base de dados principal utilizada foi o [Sistema eletrônico do Ministério da Educação \(E-MEC\)](#), além da classificação de cursos definida pelo [Conselho Nacional de Desenvolvimento Científico e Tecnológico \(CNPQ\)](#), garantindo a precisão e a relevância das informações fornecidas.

**Palavras-chave:** *curso superior; trajetória educacional; estudantes da rede pública; barreiras socioeconômicas*

# ABSTRACT

The choice of higher education is a crucial moment in the lives of many young people, representing a significant milestone in their educational and professional trajectories. However, the transition from school to college becomes a challenge for many students in Brazil's public education system, who often face socioeconomic barriers and lack the same level of access to information as private school students. This disparity makes it difficult to make informed decisions about their future and can result in a lack of perspective on accessing quality higher education. According to research from [INEP](#), in 2020, only 31.7% of high school graduates from public schools enrolled in higher education. This data highlights the need for measures to facilitate access to information and support these students in choosing their academic path, aiming to open the doors of public universities to this segment of the population. In this context, the Vocco Platform emerges as a tool to support these young people by providing guidance on the most suitable course for each student's profile, using a vocational test based on John Holland's theory, which maps the six main dimensions of vocational interests to match students with the most appropriate careers. Additionally, Vocco aims to assist students with information about which public colleges in the country offer these courses and how students can sustain themselves in higher education, including details on quota policies and support policies (scholarships, housing) at the respective universities. The methodology adopted for the development of Vocco involved analyzing existing literature to underpin the platform's features and using an agile approach to create and test the prototype, ensuring that the application meets identified needs and is effective in improving access to higher education. The primary database used was [E-MEC](#), along with the course classification defined by [CNPQ](#), ensuring the accuracy and relevance of the provided information.

**Keywords:** *higher education; educational trajectory; public school students; socioeconomic barriers*

# Lista de ilustrações

Figura 1 – Gráfico de Gantt do Projeto . . . . .	23
Figura 2 – Qrcode URL do Blog . . . . .	25
Figura 3 – Qrcode URL do Canal no YouTube . . . . .	26
Figura 4 – Caso de Uso: Funcionamento do Sistema . . . . .	30
Figura 5 – Arquitetura da aplicação . . . . .	35
Figura 6 – Qrcode URL da Aplicação . . . . .	37
Figura 7 – Modelo Entidade Relacionamento (MER) do projeto . . . . .	39
Figura 8 – Diagrama Entidade Relacionamento (DER) do projeto . . . . .	40
Figura 9 – Qrcode Repositório do GitHub . . . . .	43
Figura 10 – Tabela de métricas do projeto primeiro semestre 2024 . . . . .	51
Figura 11 – Dados Gerais sobre o repositório <i>back-end</i> do GitHub . . . . .	52
Figura 12 – Dados Gerais sobre o repositório <i>back-end</i> do GitHub . . . . .	53
Figura 13 – Número de commits em cada mês do ano de 2024 . . . . .	53
Figura 14 – Número de commits por autor . . . . .	53
Figura 15 – Número de linhas de código por autor . . . . .	54
Figura 16 – Gráfico número de commits por autor . . . . .	54
Figura 17 – Gráfico número de commits por autor . . . . .	55
Figura 18 – Tabela autor com o maior número de commits em cada mês do ano de 2024 . . . . .	55
Figura 19 – Gráfico com a relação número de arquivos adicionados e linhas de código	56
Figura 20 – Tipos de arquivos presentes no repositório do projeto <i>back-end</i> do GitHub	56
Figura 21 – Dados de arquivos presentes no repositório do projeto <i>back-end</i> . . . . .	57
Figura 22 – Dados Gerais sobre o repositório <i>front-end</i> do GitHub . . . . .	57
Figura 23 – Dados Gerais sobre o repositório <i>front-end</i> do GitHub . . . . .	58
Figura 24 – Número de commits em cada mês do ano de 2024 . . . . .	58
Figura 25 – Número de commits por autor . . . . .	58
Figura 26 – Número de linhas de código por autor . . . . .	59
Figura 27 – Gráfico número de commits por autor . . . . .	59
Figura 28 – Gráfico número de commits por autor . . . . .	60
Figura 29 – Tabela autor com o maior número de commits em cada mês do ano de 2024 . . . . .	60
Figura 30 – Gráfico com a relação número de arquivos adicionados e linhas de código	61
Figura 31 – Tipos de arquivos presentes no repositório do projeto <i>front-end</i> do GitHub	61
Figura 32 – Dados de arquivos presentes no repositório do projeto <i>front-end</i> . . . . .	62
Figura 33 – Cenário ideal cadastro de instituição . . . . .	66

Figura 34 – Cenário de exceção de cadastro de instituição . . . . .	67
Figura 35 – Cenário ideal de Listagem de instituição . . . . .	68
Figura 36 – Cenário de exceção de listagem de instituição . . . . .	68
Figura 37 – Cenário ideal de detalhamento de instituição . . . . .	69
Figura 38 – Cenário de exceção de detalhamento de instituição . . . . .	70
Figura 39 – Cenário ideal de associação de instituição com curso . . . . .	71
Figura 40 – Cenário de exceção associação de instituição com curso . . . . .	72
Figura 41 – Cenário ideal de exclusão de instituição . . . . .	73
Figura 42 – Cenário de exceção de exclusão de instituição . . . . .	74
Figura 43 – Tela inicial do administrador . . . . .	77
Figura 44 – Cadastro de dados gerais da instituição . . . . .	78
Figura 45 – Gerenciamento de instituições . . . . .	78
Figura 46 – Cadastro de dados gerais da instituição . . . . .	79
Figura 47 – Cadastro de dados gerais da instituição . . . . .	79
Figura 48 – Cadastro de dados gerais da instituição . . . . .	80
Figura 49 – Inserção dos cursos na instituição . . . . .	80
Figura 50 – Inserção das políticas na instituição . . . . .	81
Figura 51 – Primeira Publicação do Blog . . . . .	94
Figura 52 – Segunda Publicação do Blog . . . . .	95
Figura 53 – Terceira Publicação do Blog . . . . .	96
Figura 54 – Quarta Publicação do Blog . . . . .	97
Figura 55 – Quinta Publicação do Blog . . . . .	98
Figura 56 – Sexta Publicação do Blog . . . . .	99
Figura 57 – Sétima Publicação do Blog . . . . .	99
Figura 58 – Oitava Publicação do Blog . . . . .	100
Figura 59 – Nona Publicação do Blog . . . . .	101
Figura 60 – Décima Publicação do Blog . . . . .	102
Figura 61 – Décima Primeira Publicação do Blog . . . . .	103

# Lista de tabelas

Tabela 2 – Fluxo de Cadastro de Usuário (Administrador) . . . . .	30
Tabela 3 – Fluxo de Cadastro de Instituição Pública . . . . .	31
Tabela 4 – Fluxo de Cadastro de Usuário (Estudante) . . . . .	31
Tabela 5 – Fluxo de Manter Perguntas do Teste Vocacional . . . . .	32
Tabela 6 – Fluxo de Manter Informações sobre Cursos . . . . .	32
Tabela 7 – Fluxo de Responder o Teste Vocacional . . . . .	33
Tabela 8 – Fluxo de Apresentar Recomendação de Curso . . . . .	33
Tabela 9 – Estimativa dos Custos Operacionais . . . . .	47
Tabela 10 – Detalhamento dos Custos Operacionais . . . . .	48
Tabela 11 – Estimativa de Receitas Operacionais . . . . .	48



# Lista de abreviaturas e siglas

API	Interface de Programação de Aplicação - Citado em <a href="#">41</a> , <a href="#">44</a> , <a href="#">46</a> , <a href="#">64</a>
CD	Distribuição Contínua - Citado em <a href="#">42</a>
CEP	Código de Endereçamento Postal - Citado em <a href="#">41</a>
CI	Integração Contínua - Citado em <a href="#">42</a>
CNCT	Catálogo Nacional de Cursos Técnicos - Citado em <a href="#">15</a>
CNPQ	Conselho Nacional de Desenvolvimento Científico e Tecnológico - Citado em <a href="#">4</a> , <a href="#">5</a>
CRUD	Create, Read, Update, Delete - Citado em <a href="#">36</a>
DER	Diagrama Entidade-Relacionamento - Citado em <a href="#">40</a>
DTO	Data Transfer Object - Citado em <a href="#">45</a>
E-MEC	Sistema eletrônico do Ministério da Educação - Citado em <a href="#">4</a> , <a href="#">5</a>
ENEM	Exame Nacional do Ensino Médio - Citado em <a href="#">21</a> , <a href="#">64</a>
HTTP	<i>Hypertext Transfer Protocol</i> - Protocolo de Transferência de Hipertexto - Citado em <a href="#">36</a> , <a href="#">46</a>
HTTPS	<i>Hypertext Transfer Protocol Secure</i> - Protocolo de Transferência de Hipertexto Seguro - Citado em <a href="#">42</a>
IES	Instituições de Ensino Superior - Citado em <a href="#">13</a> , <a href="#">28</a> , <a href="#">29</a> , <a href="#">49</a>
INEP	Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira - Citado em <a href="#">4</a> , <a href="#">5</a>
LGPD	Lei Geral de Proteção de Dados - Citado em <a href="#">46</a> , <a href="#">47</a>
MER	Modelo Entidade Relacionamento - Citado em <a href="#">38</a>
MSC	Model-Service-Controller - Citado em <a href="#">35</a> , <a href="#">36</a>
MVP	Produto Mínimo Viável - Citado em <a href="#">82</a> , <a href="#">83</a>
PoC	Prova de Conceito - Citado em <a href="#">17</a> , <a href="#">49</a>
PROUNI	Programa Universidade Para Todos - Citado em <a href="#">64</a>
SISU	Sistema de Seleção Unificada - Citado em <a href="#">64</a>

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>13</b>
<b>1.1</b>	<b>Justificativa</b>	<b>14</b>
<b>1.2</b>	<b>Objetivos Gerais</b>	<b>14</b>
<b>1.3</b>	<b>Objetivos Específicos</b>	<b>14</b>
<b>1.4</b>	<b>Análise da Concorrência</b>	<b>14</b>
1.4.1	Aplicativo Soutec	15
1.4.2	Kuau	15
1.4.3	Orientação Vocacional e Coaching	16
1.4.4	FutureMe	16
<b>2</b>	<b>METODOLOGIA</b>	<b>17</b>
<b>3</b>	<b>REVISÃO BIBLIOGRÁFICA</b>	<b>18</b>
<b>3.1</b>	<b>O processo de escolha profissional para jovens do Ensino Médio</b>	<b>18</b>
<b>3.2</b>	<b>Testes Vocacionais como Ferramenta de Orientação Profissional</b>	<b>19</b>
<b>3.3</b>	<b>Desigualdade no acesso ao ensino superior</b>	<b>20</b>
<b>4</b>	<b>GESTÃO DO PROJETO</b>	<b>22</b>
<b>4.1</b>	<b>Metodologia de gestão de projeto</b>	<b>22</b>
<b>4.2</b>	<b>Papéis e Atribuições</b>	<b>23</b>
4.2.1	Ferramentas de gestão utilizadas	24
4.2.2	Canais de comunicação	24
<b>5</b>	<b>DESENVOLVIMENTO DO PROJETO</b>	<b>27</b>
<b>5.1</b>	<b>Elicitação de Requisitos</b>	<b>27</b>
5.1.1	Requisitos funcionais	27
5.1.2	Requisitos não funcionais	28
5.1.3	Regras de negócio	29
<b>5.2</b>	<b>Escopo do projeto</b>	<b>29</b>
5.2.1	Casos de Uso	29
<b>5.3</b>	<b>Volumetria</b>	<b>34</b>
<b>5.4</b>	<b>Arquitetura</b>	<b>34</b>
5.4.1	Back-End	35
5.4.1.1	Gerenciamento de dependências	36
5.4.2	Front-End	36
5.4.3	Banco de dados	38

5.4.3.1	Modelo Entidade Relacionamento . . . . .	38
5.4.3.2	Diagrama Entidade Relacionamento . . . . .	40
5.4.4	Integrações . . . . .	41
5.4.5	Infraestrutura . . . . .	41
5.4.6	Front - End . . . . .	41
5.4.6.1	Back - End . . . . .	42
5.4.7	Escalabilidade . . . . .	42
5.4.8	Controle de versão . . . . .	42
<b>5.5</b>	<b>Manutenibilidade . . . . .</b>	<b>43</b>
5.5.1	Code Convention . . . . .	43
5.5.2	Ferramentas de testes . . . . .	44
5.5.3	Design Patterns . . . . .	45
<b>5.6</b>	<b>Segurança, Privacidade e Legislação . . . . .</b>	<b>46</b>
5.6.1	Lei Geral de Proteção de Dados (LGPD) . . . . .	46
5.6.2	Autenticação e Autorização . . . . .	46
5.6.3	Criptografia de Dados . . . . .	47
<b>5.7</b>	<b>Viabilidade Financeira . . . . .</b>	<b>47</b>
5.7.1	Monetização . . . . .	48
<b>5.8</b>	<b>Fases de Entrega . . . . .</b>	<b>49</b>
5.8.1	Prova de Conceito (POC) . . . . .	49
5.8.2	Produto Mínimo Viável (MVP) . . . . .	49
5.8.3	Produto Final . . . . .	50
<b>5.9</b>	<b>Métricas . . . . .</b>	<b>51</b>
5.9.1	Ferramentas Auxiliares . . . . .	52
5.9.2	Estatísticas do Projeto . . . . .	52
<b>6</b>	<b>DECISÕES . . . . .</b>	<b>63</b>
<b>6.1</b>	<b>Escolhas . . . . .</b>	<b>63</b>
6.1.1	Utilização do Java . . . . .	63
6.1.2	Utilização do Spring Boot . . . . .	63
6.1.3	Utilização do TypeScript . . . . .	63
6.1.4	Utilização do React . . . . .	64
<b>6.2</b>	<b>Descartes . . . . .</b>	<b>64</b>
<b>6.3</b>	<b>Implementações futuras . . . . .</b>	<b>64</b>
6.3.1	Integração com API do e-MEC . . . . .	64
6.3.2	Inclusão de Dados de Cursos Técnicos e Outras Instituições . . . . .	65
6.3.3	Funcionalidades de Busca Avançada . . . . .	65
<b>7</b>	<b>TESTES . . . . .</b>	<b>66</b>
<b>7.1</b>	<b>Plano de testes . . . . .</b>	<b>66</b>

7.1.1	Teste de funcionalidade de Instituição . . . . .	66
<b>REFERÊNCIAS . . . . .</b>		<b>75</b>
<b>APÊNDICES</b>		<b>76</b>
	<b>APÊNDICE A – PROTOTIPAÇÃO . . . . .</b>	<b>77</b>
A.1	Tela Inicial do administrador . . . . .	77
A.2	Tela de cadastro de dados gerais da instituição . . . . .	78
A.3	Tela de gerenciamento de instituições . . . . .	78
A.4	Tela de detalhamento de uma instituição . . . . .	79
A.5	Tela de edição de uma instituição . . . . .	79
A.6	Tela de exclusão de uma instituição . . . . .	80
A.7	Tela de associação de instituições com cursos . . . . .	80
A.8	Tela de associação de instituições com políticas . . . . .	81
	<b>APÊNDICE B – CRONOGRAMA . . . . .</b>	<b>82</b>
	<b>APÊNDICE C – ATAS DE REUNIÕES . . . . .</b>	<b>84</b>
	<b>APÊNDICE D – BLOG . . . . .</b>	<b>94</b>

# 1 Introdução

Ao longo dos anos, a tecnologia tem ampliado sua presença na vida de jovens e estudantes, participando também da sua vida educacional. Para ([ALVES, 2022](#)), ela faz parte das grandes transformações que estão ocorrendo na educação, pois incorpora no ambiente educacional o acesso diversificado de informações e ferramentas digitais que podem ser utilizadas para gerar conhecimento.

Essa realidade, abre caminho para a utilização de recursos tecnológicos como aliados no processo de escolha de carreira, apoiando na identificação e planejamento dos passos em direção ao futuro profissional, apresentando grande relevância para alunos de escola pública, inclusive no contexto de busca de cursos do ensino superior. Para esses alunos é importante ter conhecimento de opções que se apresentem mais viáveis e acessíveis em decorrência de sua condição socioeconômica, já que as famílias de menores faixas de renda têm na escola pública uma das poucas alternativas para a escolarização de seus filhos ([MATOS, 2012](#)).

Contudo, em contraponto a esse cenário, mesmo que os estudantes de escola pública representem a maioria dos estudantes do ensino médio, os alunos da rede pública ainda são uma minoria no ensino superior ([ALVARENGA, 2012](#)), sendo que os motivos variam entre as altas relações candidato/vaga das universidades públicas, falta de recursos financeiros para arcar com o ensino superior privado, dentre outros fatores como a dificuldade do acesso à informação à respeito de como entrar e se manter nas universidades públicas.

Além disso, apesar da quantidade de recursos digitais disponíveis, há uma carência de sites que consolidam informações vocacionais e de carreira, que atendam às especificidades dos alunos provenientes do sistema público de ensino. Essa falta de equilíbrio entre a orientação disponível e a realidade do indivíduo pode resultar em escolhas de carreira mal informadas e desconhecimento acerca de políticas afirmativas, fundamentais para assegurar o acesso ao ensino superior, perpetuando um ciclo de desinformação e privando esses alunos do acesso ao ensino superior.

Nesse cenário, a Vocco surge como uma plataforma dedicada a atender as necessidades dos alunos de escolas públicas que buscam orientações sobre o ensino superior. A plataforma disponibilizará diversas informações sobre as [Instituições de Ensino Superior \(IES\)](#) públicas brasileiras, incluindo formas de ingresso, meios de contato, endereços, políticas de cotas e programas de permanência estudantil. Esses auxílios são essenciais para que os alunos possam se manter durante a graduação. Além disso, a Vocco oferecerá um teste vocacional para ajudar os alunos do ensino médio a escolherem a carreira mais adequada para seguir. A concepção e desenvolvimento da Vocco foram orientados por uma

análise cuidadosa da literatura existente e por um processo iterativo, assegurando que a plataforma ofereça recursos relevantes e adequados às necessidades dos usuários.

## 1.1 Justificativa

A relevância deste sistema é evidenciada pela crescente demanda por serviços de orientação personalizados e acessíveis. Ao simplificar o acesso a informações pertinentes e oferecer orientação personalizada, o sistema não apenas preenche uma lacuna crítica nos recursos de auxílio à escolha profissional disponíveis, mas também contribui para a capacitação dos estudantes em tomar decisões mais alinhadas com suas habilidades e interesses.

## 1.2 Objetivos Gerais

Este projeto tem como objetivo geral auxiliar na democratização do acesso à informação com relação a carreiras e educação superior, além de ter o intuito de contribuir para diminuir a defasagem do acesso de jovens de classe baixa e as universidades públicas, e proporcionar aos jovens da rede pública informações necessárias para planejar seu futuro com maior segurança e confiança.

## 1.3 Objetivos Específicos

- Teste vocacional;
- Reconhecer as dificuldades enfrentadas pelos alunos no acesso às informações sobre o ensino superior público;
- Oferecer informações sobre políticas de permanência e de cotas auxiliando sobre como entrar e se manter no ensino superior em São Paulo;
- Análise do grau de empregabilidade das profissões relacionadas aos cursos recomendados.

## 1.4 Análise da Concorrência

Embora haja uma variedade de plataformas de serviços de orientação vocacional disponíveis, quatro delas se destacam como potenciais concorrentes do projeto, sendo elas: Soutec, Kuau, Orientação Vocacional e Coaching e FutureMe.

### 1.4.1 Aplicativo Soutec

O objetivo principal do aplicativo Soutec, é auxiliar os cidadãos, em especial os jovens, no planejamento de suas carreiras profissionais, por meio do suporte na identificação dos seus perfis, assim como na escolha do melhor curso técnico disponível para o desenvolvimento de suas competências. Para isso, o aplicativo Soutec disponibiliza testes especializados para identificação do perfil profissional, com detalhamento da profissão e recomendações de cursos técnicos baseados no [Catálogo Nacional de Cursos Técnicos \(CNCT\)](#) também considerando a localização de interesse do usuário. Mesmo com as similaridades com a Plataforma Vocco, os focos são distintos. O público alvo que a Soutec busca é predominantemente as pessoas que desejam informações sobre cursos técnicos, fazendo com que o jovem que opte por seguir pelo ensino superior não encontre orientações e auxílio para seguir por tal caminho, já o foco da Vocco são jovens de baixa renda de escolas públicas que desejam tanto informações de cursos a nível superior quanto cursos técnicos. Além disso, a Vocco apresenta maiores vantagens para o usuário ao indicar não somente os cursos adequados para o perfil do jovem, como também qual instituição pública de ensino no Brasil oferece aquele curso, e quais são as políticas de permanência presentes em cada instituição para que o aluno consiga se manter no decorrer do curso.

### 1.4.2 Kuau

O Kuau é um aplicativo de orientação profissional que oferece uma metodologia inovadora, apelidada de "Netflix das profissões". Ele utiliza vídeos de curta duração com depoimentos de universitários, recém-formados e profissionais para explorar diversas carreiras. Seus diferenciais incluem um "termômetro de afinidade", que ajuda os alunos a definirem suas preferências enquanto assistem aos vídeos, e um certificado de proficiência, que avalia o aprendizado sobre cada profissão. O aplicativo também se integra ao Projeto de Vida das escolas, fornecendo relatórios e indicadores de acompanhamento.

O conceito da abordagem da Kuau é muito distante da abordagem usada pela Vocco, não oferecendo um teste vocacional e focando somente na divulgação e detalhes das profissões que o usuário se interessa. Também não evidencia as informações sobre universidades públicas em que o usuário poderia ingressar para estudar tal profissão. Isso faz com que a Vocco se torne mais atrativa para os usuários que buscam um teste vocacional, já que nossa plataforma investe na centralização de informações de Instituições públicas de ensino no Brasil e suas políticas de permanência, além de uma análise da empregabilidade do curso indicado pela plataforma, fazendo com que os usuários não tenham somente a indicação de qual curso mais se adequa ao seu perfil como o termômetro de afinidade usado pela KUAU, mas também informa como o estudante conseguirá se manter financeiramente no curso, e quais as expectativas da empregabilidade no mercado.

### 1.4.3 Orientação Vocacional e Coaching

A plataforma oferece um método de orientação vocacional, projetado para ajudar os indivíduos a descobrir suas futuras profissões e definir seus percursos educacionais. Eles disponibilizam um *ebook* gratuito e um teste vocacional online adaptado para os sistemas educacionais de Portugal, Brasil, Angola e Moçambique, baseado no modelo hexagonal de John Holland, que também será a base para a recomendação de cursos da Vocco. O site também apresenta recursos de orientação vocacional abrangentes, incluindo informações úteis para orientadores e orientadores.

A Orientação Vocacional e Coaching não oferece orientações para os estudantes em relação às universidades que disponibilizam os cursos indicados pela plataforma, focando somente na orientação vocacional com testes adaptados e a disponibilização de *ebooks* gratuitos. Dessa forma a Vocco se destaca informando não somente um teste vocacional, mas sim informações da empregabilidade atual do curso indicado pelo teste, informações de qual instituição pública de ensino oferece tais cursos, e suas políticas de permanência, auxiliando o jovem na escolha de curso e em sua permanência.

### 1.4.4 FutureMe

A plataforma foca na orientação profissional autodirigida e gamificada para preparar estudantes para uma escolha profissional mais consciente. Eles atendem todo o Brasil e têm parcerias com mais de 150 escolas. A plataforma destaca a importância do autocohecimento e oferece uma trilha gamificada que os alunos podem percorrer durante 7 a 10 aulas do Projeto de Vida, culminando em um evento de compartilhamento de experiências e entrega de certificados.

Por ser predominantemente uma plataforma para orientação profissional, a FutureMe também não auxilia o usuário com informações de possíveis universidades de ingresso, além disso não disponibiliza um teste vocacional, mas sim uma orientação gamificada, sendo um foco diferente da plataforma Vocco. Essa plataforma se torna uma concorrente da Vocco por apresentar ao usuário um teste vocacional direcionando-o para uma potencial carreira adequada ao seu perfil, porém a Vocco se destaca ao informar não somente os cursos mais relevantes para o perfil do jovem, mas também qual a expectativa de empregabilidade daquele curso, quais instituições públicas do Brasil oferecem-no e quais as políticas públicas estão presentes naquela determinada instituição, auxiliando o jovem a encontrar uma potencial carreira e meios de se manter financeiramente no decorrer do curso.



## 2 METODOLOGIA

Para enfrentar o desafio do acesso ao ensino superior para jovens de baixa renda, este projeto adota uma metodologia integrada que combina revisão teórica, desenvolvimento prático e validação da aplicação proposta. Inicialmente, será conduzida a revisão bibliográfica da literatura existente para examinar estudos acadêmicos sobre o acesso ao ensino superior e os desafios enfrentados por estudantes de escolas públicas, além de explorar o modelo tipológico de John Holland e suas implicações para a orientação vocacional. Esse levantamento fornecerá a base teórica e os insumos necessários para a definição das funcionalidades da plataforma Vocco. Em seguida, uma análise das necessidades dos usuários será realizada através da avaliação de dados existentes, relatórios acadêmicos e estatísticas públicas sobre barreiras no acesso ao ensino superior. Com essas informações, será desenvolvido um protótipo da aplicação, utilizando uma abordagem ágil para permitir um desenvolvimento iterativo e flexível. Após a entrega do protótipo na fase da [Prova de Conceito \(PoC\)](#), serão implementadas as funcionalidades adicionais com base nas dificuldades identificadas nos estudos. Testes piloto com um grupo selecionado de usuários serão conduzidos para avaliar a eficácia da aplicação e sua aderência às necessidades dos usuários. O feedback obtido nesses testes permitirá ajustes e refinamentos, garantindo que a aplicação atenda de maneira eficaz às necessidades dos jovens de baixa renda e contribua significativamente para a melhoria do acesso ao ensino superior.

## 3 REVISÃO BIBLIOGRÁFICA

Para a elaboração da revisão bibliográfica, estudamos temas essenciais relacionados ao tema do nosso projeto, que são: à escolha profissional, orientação vocacional e acesso ao ensino superior. Sobre o primeiro, destacamos a importância crucial da escolha da carreira durante a transição do ensino médio para a vida adulta. A pesquisa de Lopes e Brito revela que fatores como situação financeira e influência familiar desempenham papéis significativos nesse processo, enquanto os desafios socioeconômicos podem limitar as opções dos jovens.

Em relação ao segundo tópico, exploramos os testes vocacionais como ferramentas valiosas na orientação profissional. O modelo proposto por Holland é discutido, destacando como os diferentes tipos de personalidade estão ligados a áreas específicas de interesse e trabalho. Nossa pesquisa ressalta a importância de considerar os estilos interpessoais ao orientar indivíduos em suas escolhas de carreira, reconhecendo que as tipologias de personalidade não são rígidas, mas fornecem uma estrutura útil para compreender a complexidade individual.

Por fim, abordamos a desigualdade no acesso ao ensino superior no contexto brasileiro. Apesar dos avanços na universalização do ensino fundamental e médio, ainda persistem barreiras socioeconômicas e raciais significativas. Destacamos a importância de políticas públicas para promover o acesso e a qualidade da educação, especialmente no ensino médio, como um meio crucial para democratizar o acesso ao ensino superior.

### 3.1 O processo de escolha profissional para jovens do Ensino Médio

A transição do ensino médio para a vida adulta é uma fase repleta de desafios, e a escolha da carreira profissional é uma das decisões mais impactantes que os jovens enfrentam durante esse período, uma escolha que muitas vezes não é estudada com a profundidade necessária. O artigo conduzido por Talisson de Sousa Lopes e Sônia Christo Aleixo A. Brito oferece análises valiosas sobre os fatores que influenciam essa escolha. A pesquisa, realizada com 78 alunos do terceiro ano do ensino médio, busca compreender os motivadores e influenciadores por trás das decisões de carreira desses jovens.

Inicialmente, o estudo revela que a situação financeira é um dos principais impulsores para os jovens buscarem uma colocação no mercado de trabalho ou ingressarem em cursos profissionalizantes ou superiores. Essa busca por melhores oportunidades é frequentemente encorajada pelos pais e familiares dos estudantes. Além disso, a pesquisa revela que a maioria dos participantes está na faixa etária de 16 a 19 anos, uma fase em

que a pressão para definir o futuro profissional é intensa. Um aspecto interessante levantado pelo estudo é que muitos alunos já têm uma ideia clara sobre seu futuro profissional após a conclusão do ensino médio, sendo que grande parte deles expressa o desejo de cursar uma faculdade para alcançar suas aspirações profissionais (LOPES, 2022).

No entanto, a pesquisa também destaca os desafios enfrentados pelos jovens nesse processo. A necessidade financeira surge como um fator limitante (LOPES, 2022), levando alguns estudantes a aceitarem empregos que não correspondem às suas aspirações, comprometendo suas oportunidades de crescimento profissional. Além disso, as condições socioeconômicas das famílias dos alunos impactam diretamente seu desempenho acadêmico, conforme evidenciado por estudos anteriores citados no artigo.

A influência familiar na escolha da profissão também é ressaltada (LOPES, 2022), com muitos alunos seguindo carreiras que se alinham aos interesses de seus pais. Isso evidencia a dificuldade dos jovens em definir uma carreira própria, especialmente quando enfrentam barreiras de acesso à informação durante essa fase de transição.

## 3.2 Testes Vocacionais como Ferramenta de Orientação Profissional

Os testes vocacionais têm sido amplamente utilizados como uma ferramenta de orientação profissional, auxiliando os jovens na identificação de suas habilidades, interesses e aptidões. Pesquisas evidenciam que a aplicação de testes vocacionais pode contribuir significativamente para a tomada de decisão dos estudantes em relação à escolha da carreira e do curso universitário. Além disso, esses testes podem ajudar a reduzir a evasão universitária, direcionando os alunos para áreas de maior afinidade e interesse.

O modelo tipológico de personalidades vocacionais proposto por (HOLLAND, 1997) tem sido fundamental na pesquisa e prática de avaliação de interesses ao longo das últimas décadas (MAGALHÃES, 2006). Holland postula que os interesses vocacionais são uma expressão da personalidade, e que pessoas dedicadas a uma mesma ocupação tendem a possuir personalidades similares. Seu modelo identifica seis tipos de personalidades: realista (R), investigativo (I), artístico (A), social (S), empreendedor (E) e convencional (C). Esses tipos de personalidades criam ambientes físicos e interpessoais distintos, que podem ser categorizados pela tipologia RIASEC. Cada tipo de ambiente tem estratégias preferidas de solução de problemas e estilos de interação interpessoal específicos (HOLLAND, 1997).

(MAGALHÃES, 2004) e (MARTINS, 1978) forneceram descrições recentes dos seis tipos de personalidades vocacionais, enquanto (MAGALHÃES, 2006) ofereceu uma revisão extensa do estado atual das pesquisas sobre o modelo. Essas pesquisas destacam

a importância do modelo de Holland na compreensão das relações entre personalidade, trabalho e desenvolvimento de carreira. Os resultados sugerem que os diferentes tipos vocacionais estão de fato associados a padrões distintos de comportamento interpessoal. Por exemplo, os tipos Artísticos tendem a exibir um estilo mais voltado para a originalidade e a expressão pessoal, enquanto os tipos Realistas são mais reservados e avessos à interação social. Essas descobertas são consistentes com as descrições feitas por Holland sobre as características de cada tipo.

Além disso, os estudos destacam a importância de considerar os estilos interpessoais ao orientar indivíduos em suas escolhas de carreira. Em vez de simplesmente vincular os interesses vocacionais a ocupações específicas, deve-se procurar entender como nossos interesses refletem metas e valores pessoais e como essas características são expressas no ambiente de trabalho. No entanto, estudos reconhecem que as tipologias de personalidade têm suas limitações e não devem ser vistas como rígidas ou definitivas. Em vez disso, elas fornecem uma estrutura útil para entender as semelhanças e diferenças entre as pessoas, enquanto ainda reconhecem a complexidade e a individualidade de cada caso.

### 3.3 Desigualdade no acesso ao ensino superior

O acesso à educação é um tema fundamental em qualquer sociedade, refletindo diretamente no desenvolvimento humano e socioeconômico de um país. No Brasil, país marcado por profundas desigualdades sociais e econômicas, a análise desse acesso, especialmente no que tange ao ensino superior, revela tanto avanços quanto desafios persistentes. Nos últimos anos, o debate sobre o acesso à educação no Brasil tem ganhado relevância tanto nas esferas acadêmicas quanto nas políticas públicas. Neste contexto, o estudo conduzido por Cibele Yahn oferece uma análise sobre essa questão, destacando os principais aspectos que influenciam o acesso à educação e apontando para a necessidade de políticas públicas eficazes para promover uma verdadeira democratização do ensino.

([ANDRADE, 2012](#)) começa por contextualizar o cenário educacional brasileiro, ressaltando a transformação significativa que ocorreu a partir dos anos 90, com a universalização do ensino fundamental e o aumento substancial das matrículas no ensino médio e superior. Entretanto, apesar desses avanços, a autora destaca que o acesso ao ensino superior ainda é limitado, especialmente quando comparado a países mais desenvolvidos.

Um dos aspectos mais importantes da análise é a diferenciação entre os jovens que acessam o ensino superior e aqueles que não conseguem, com base em variáveis como renda familiar e raça/cor autodeclarada. Os resultados obtidos evidenciam as desigualdades socioeconômicas e raciais persistentes no sistema educacional brasileiro, mostrando que os jovens mais pobres e os autodeclarados não brancos enfrentam maiores dificuldades de acesso ([ANDRADE, 2012](#)).

A autora também aborda o atraso escolar como uma barreira significativa para o acesso ao ensino superior, destacando que uma parcela considerável dos jovens brasileiros não possui os requisitos educacionais formais para ingressar nesse nível de ensino, ou seja, possuem o ensino fundamental e/ou médio incompleto. Além disso, a análise da evolução do acesso à educação ao longo dos anos revela tanto avanços quanto desafios, ressaltando a necessidade de ampliar o acesso e melhorar a qualidade da educação básica.

Outro aspecto analisado foi a relação entre renda familiar, raça/cor autodeclarada e acesso à educação. Os dados apresentados pela autora demonstram claramente que as desigualdades no acesso ao ensino superior são fortemente influenciadas pela renda familiar, mas também têm uma dimensão racial importante, com os jovens não brancos enfrentando maiores obstáculos de acesso.

Por fim, ([ANDRADE, 2012](#)) destaca a importância de políticas públicas voltadas para a promoção do acesso e da qualidade da educação, especialmente no ensino médio, como uma estratégia fundamental para aumentar o acesso ao ensino superior. A análise dos resultados do [Exame Nacional do Ensino Médio \(ENEM\)](#) é apresentada como um indicador crucial do nível de aprendizado dos jovens e da demanda qualificada para o ensino superior, destacando a necessidade de investimentos contínuos na melhoria do ensino básico para promover uma verdadeira democratização do acesso à educação no Brasil.

## 4 GESTÃO DO PROJETO

Na gestão de projetos de *software*, é fundamental estabelecer papéis e atribuições claras para cada membro da equipe, pois isso promove a clareza quanto às responsabilidades de cada um e evita conflitos e duplicação de esforços. Além disso, o uso de ferramentas de gestão adequadas, como sistemas de controle de versão, plataformas de gerenciamento de tarefas e cronogramas, ajuda a manter o projeto organizado e dentro do prazo. Outro aspecto crucial é estabelecer canais de comunicação confiáveis entre os membros da equipe, garantindo que as informações fluam de maneira rápida e precisa, facilitando a colaboração e a resolução de problemas.

### 4.1 Metodologia de gestão de projeto

A metodologia de gestão de projeto definida pela equipe foi a metodologia **Scrum**, pois proporciona um *framework* flexível e adaptativo que se alinha bem com as necessidades dinâmicas do desenvolvimento de *software*. O Scrum permite uma abordagem iterativa e incremental, o que possibilita a entrega rápida e frequente de funcionalidades, garantindo uma maior capacidade de resposta às mudanças nos requisitos do cliente e do mercado. Além disso, também promove uma cultura de transparência, colaboração e auto-organização dentro da equipe, incentivando a responsabilidade compartilhada e o foco na entrega de valor para o cliente. Outros benefícios incluem a redução de riscos, a melhoria da comunicação e a maximização da satisfação do cliente ao longo do ciclo de vida do projeto. Em resumo, a adoção do Scrum como metodologia de projeto oferece uma série de vantagens que podem contribuir significativamente para o sucesso e a qualidade do desenvolvimento de software.

É importante ressaltar que o Scrum é uma das muitas abordagens dentro do conjunto de **metodologias ágeis**, que surgiram como uma resposta à rigidez e à inadequação das metodologias tradicionais de desenvolvimento de software. As metodologias ágeis valorizam indivíduos e interações mais do que processos e ferramentas, adaptando-se a mudanças mais do que seguindo um plano rígido. No contexto do Scrum, essa agilidade é alcançada por meio de *Sprints*, reuniões diárias, retrospectivas e um foco constante na entrega incremental de valor ao cliente. O gráfico apresentado reflete a abordagem ágil adotada, com tarefas organizadas de forma a permitir ajustes e melhorias contínuas, alinhadas às práticas do Scrum.



Figura 1 – Gráfico de Gantt do Projeto

A coordenação das tarefas apresentadas na Figura 1 evidencia uma gestão bem estruturada, onde as atividades são planejadas de maneira a evitar sobreposições excessivas e gargalos. Essa organização não só melhora a eficiência da equipe, mas também minimiza o risco de atrasos, assegurando que o projeto progrida de forma fluida e dentro dos prazos estipulados.

## 4.2 Papéis e Atribuições

Dentro da metodologia Scrum, existem três papéis principais: o Product Owner, responsável por representar os interesses do cliente e definir as funcionalidades do produto; o Scrum Master, encarregado de garantir que a equipe siga os princípios e práticas do Scrum; e a equipe de desenvolvimento, responsável por realizar o trabalho necessário para entregar as funcionalidades do produto. Cada um desses papéis desempenha uma função específica no processo de desenvolvimento ágil, contribuindo para a eficiência e sucesso do projeto.

- **Scrum Master:** O Scrum Master, como líder servo da equipe, desempenha um papel crucial na remoção de obstáculos que possam impedir o progresso do projeto. Sua responsabilidade inclui facilitar reuniões, garantir a adesão aos princípios e práticas do Scrum, e promover um ambiente de trabalho colaborativo e auto-organizável.

Além disso, o Scrum Master atua como um mentor, capacitando a equipe a tomar decisões e resolver problemas de forma independente.

- Dentro do nosso projeto, esse papel foi desempenhado pela integrante Tamiris Delfino.
- **Product Owner:** O Product Owner é o representante dos *stakeholders* e tem a responsabilidade de maximizar o valor do produto. Isso envolve a definição e priorização do *backlog* do produto, garantindo que os requisitos do cliente sejam compreendidos e atendidos pela equipe de desenvolvimento. O Product Owner deve ter uma visão clara dos objetivos do negócio e trabalhar em estreita colaboração com a equipe para garantir que o produto seja desenvolvido de acordo com esses objetivos.
  - Dentro do nosso projeto, esse papel foi desempenhado pela integrante Letícia Baião.
- **Desenvolvedores:** Os Desenvolvedores são os membros da equipe responsáveis por transformar os itens do *backlog* do produto em incrementos potencialmente entregáveis do produto. Eles são auto-organizáveis e colaboram de perto com o Product Owner para entender os requisitos do produto. Os Desenvolvedores têm a responsabilidade de garantir que o trabalho seja concluído dentro do prazo estabelecido para cada Sprint, bem como de buscar continuamente maneiras de melhorar o processo de desenvolvimento.
  - Dentro do nosso projeto, esse papel foi desempenhado por todas as integrantes, dada a exigência de todas se envolverem com o desenvolvimento e o prazo para a entrega do projeto, que torna inviável a codificação ser feita por apenas três integrantes.

#### 4.2.1 Ferramentas de gestão utilizadas

No projeto, optamos por utilizar o **Jira** como nossa ferramenta de gerenciamento de projeto. O Jira foi escolhido devido à sua ampla gama de funcionalidades que atendem às necessidades específicas de gestão de projetos ágeis. Algumas das características chave do Jira incluem sua capacidade de criar e acompanhar histórias de usuário, gerenciar e priorizar o *backlog* do produto, atribuir tarefas aos membros da equipe, monitorar o progresso do projeto através de quadros Kanban e Scrum, e gerar relatórios sobre o desempenho da equipe e do projeto.

#### 4.2.2 Canais de comunicação

No projeto, optamos por utilizar o **WhatsApp** como meio de comunicação para coordenar e agendar nossas reuniões. Já para a realização das reuniões semanais, escolhe-



mos a plataforma **Meet**. Através dela, conseguimos conduzir nossas reuniões de forma remota, possibilitando a participação de todos os membros da equipe, mesmo em localidades diferentes.

Para facilitar o compartilhamento e acompanhamento do histórico de desenvolvimento do projeto, foi criado o blog ADA'S TECH IFSP. Este blog serve como um repositório de informações sobre a evolução do projeto, permitindo o acompanhamento das etapas. O acesso ao blog está disponível na Figura 2 e no link.



Figura 2 – QRcode URL do Blog

<https://adastech.wordpress.com/>

Além disso, foi criado o canal no YouTube @AdasTechIFSP, destinado à publicação de todos os vídeos produzidos ao longo do desenvolvimento do projeto, incluindo apresentações e vídeos do Gource. O acesso ao canal está disponível na Figura 3



Figura 3 – QRcode URL do Canal no YouTube

<https://www.youtube.com/@AdasTechIFSP>

Esses canais de comunicação garantem a eficiência na troca de informações e documentam o progresso do projeto de forma acessível e organizada. O blog e o canal no YouTube são ferramentas que mantêm todos os membros e interessados informados sobre as atividades e avanços do projeto.

## 5 DESENVOLVIMENTO DO PROJETO

No desenvolvimento de qualquer projeto, desde os mais simples até os mais complexos, uma fase crucial é a sua concepção e planejamento. Neste capítulo, exploraremos os diversos aspectos envolvidos nesse processo inicial, que nos guiarão durante o desenvolvimento do projeto. Desde a definição dos requisitos e escopo do projeto, passando pela elaboração de casos de uso e arquitetura, até a consideração da infraestrutura necessária para sua implementação, cada etapa desempenha um papel fundamental na construção do produto final. Além disso, discutiremos a importância da manutenibilidade do sistema, sua viabilidade financeira e as diferentes fases de entrega, visando garantir não apenas a entrega de um produto funcional, mas também sua sustentabilidade e sucesso a longo prazo.

### 5.1 Elicitação de Requisitos

Neste capítulo, vamos explorar o processo de elicitação de requisitos, fundamental para o desenvolvimento de *software*.

#### 5.1.1 Requisitos funcionais

Requisitos funcionais são especificações que descrevem as funções e operações que um sistema ou *software* deve ser capaz de realizar. Eles definem as funcionalidades específicas que o sistema deve fornecer, incluindo entradas, saídas e comportamentos esperados em diferentes cenários de uso. Para a elaboração do projeto, foram definidos oito requisitos funcionais.

- **Funcionalidades das Perguntas do Teste Vocacional:**

- **RF1** - Manter perguntas do teste vocacional (incluir, alterar, excluir, consultar).

- **Funcionalidades das respostas do Teste Vocacional:**

- **RF2** - Permitir ao estudante responder o teste.
- **RF3** - Manter respostas do teste vocacional (incluir, consultar).

- **Funcionalidades das Instituições Públicas:**

- **RF4** - Manter informações pertinentes sobre os cursos (incluir, alterar, excluir, consultar).

- **RF5** - Disponibilizar informações dos cursos e **IES** cadastradas, permitindo filtragem.
- **Funcionalidades dos Usuários (Administradores e Estudantes):**
  - **RF6** - Manter cadastro dos estudantes (incluir, alterar, excluir, consultar).
  - **RF7** - Manter cadastro dos administradores (incluir, alterar, excluir, consultar).
- **Apresentar Resultado de Indicação do Curso:**
  - **RF8** - Apresentar recomendação personalizada de cursos para o estudante após a conclusão do teste vocacional.

### 5.1.2 Requisitos não funcionais

Requisitos não funcionais referem-se a critérios que não estão diretamente relacionados às funcionalidades específicas do sistema, mas sim a atributos de qualidade, restrições ou características gerais que o sistema deve possuir. Isso inclui aspectos como desempenho, segurança, usabilidade, escalabilidade, confiabilidade e compatibilidade. Esses requisitos definem os padrões de qualidade e comportamento que o sistema deve atender, sem especificar diretamente as funções que ele deve realizar. Para a elaboração do projeto, foram definidos sete requisitos não funcionais

- **Logs:**
  - **RNF1** - Criação de testes e logs do sistema para reconhecimento e tratativa de erros
- **Usabilidade:**
  - **RNF2** - Interface amigável, fácil e intuitiva para todos os tipos de usuários.
- **Desempenho:**
  - **RNF3** - Respostas rápidas às solicitações do usuário, com tempo de carregamento eficiente para as funcionalidades do sistema.
- **Segurança:**
  - **RNF4** - Proteção de dados sensíveis dos usuários, incluindo autenticação e autorização para acessar determinadas funcionalidades.
  - **RNF5** - Os dados dos usuários devem ser coletados, armazenados e processados de acordo com a legislação local sobre proteção de dados.

- **Escalabilidade:**

- **RNF6** - O sistema deve ser capaz de lidar com um aumento no número de usuários e dados sem degradar a performance.

- **Compatibilidade:**

- **RNF7** - O sistema deve possuir suporte a diferentes navegadores (Chrome, Firefox e Opera GX) e dispositivos (responsividade).

### 5.1.3 Regras de negócio

Dentro de um *software*, as regras de negócio referem-se às diretrizes lógicas que governam o comportamento e as operações do sistema em relação aos processos de negócio da organização. Elas definem as condições, restrições e comportamentos que o *software* deve seguir para garantir que as operações empresariais sejam realizadas de acordo com as políticas e procedimentos estabelecidos.

As regras de negócio definidas para a aplicação Vocco são:

- As informações das universidades devem ser atualizadas semestralmente para refletir quaisquer mudanças nas políticas de admissão e nos cursos oferecidos.
- Será permitida a criação de contas para estudantes maiores de 14 anos.
- Só serão permitidos cadastros e recomendações de [IES](#) públicas.

## 5.2 Escopo do projeto

A definição do escopo do projeto foi embasada na utilização de casos de uso. Essa decisão foi tomada devido à necessidade de uma abordagem estruturada e detalhada para descrever as interações entre os usuários e o sistema.

Os casos de uso possibilitam uma especificação clara dos requisitos funcionais, identificando de forma precisa as principais funcionalidades do sistema e os comportamentos esperados.

Portanto, a escolha de utilizar apenas casos de uso foi considerada apropriada para garantir a compreensão precisa do escopo do projeto e assegurar o sucesso da implementação.

### 5.2.1 Casos de Uso

A seguir, apresenta-se o caso de uso desenvolvido para a plataforma Vocco, apresentado na Figura 4, acompanhado de suas descrições.

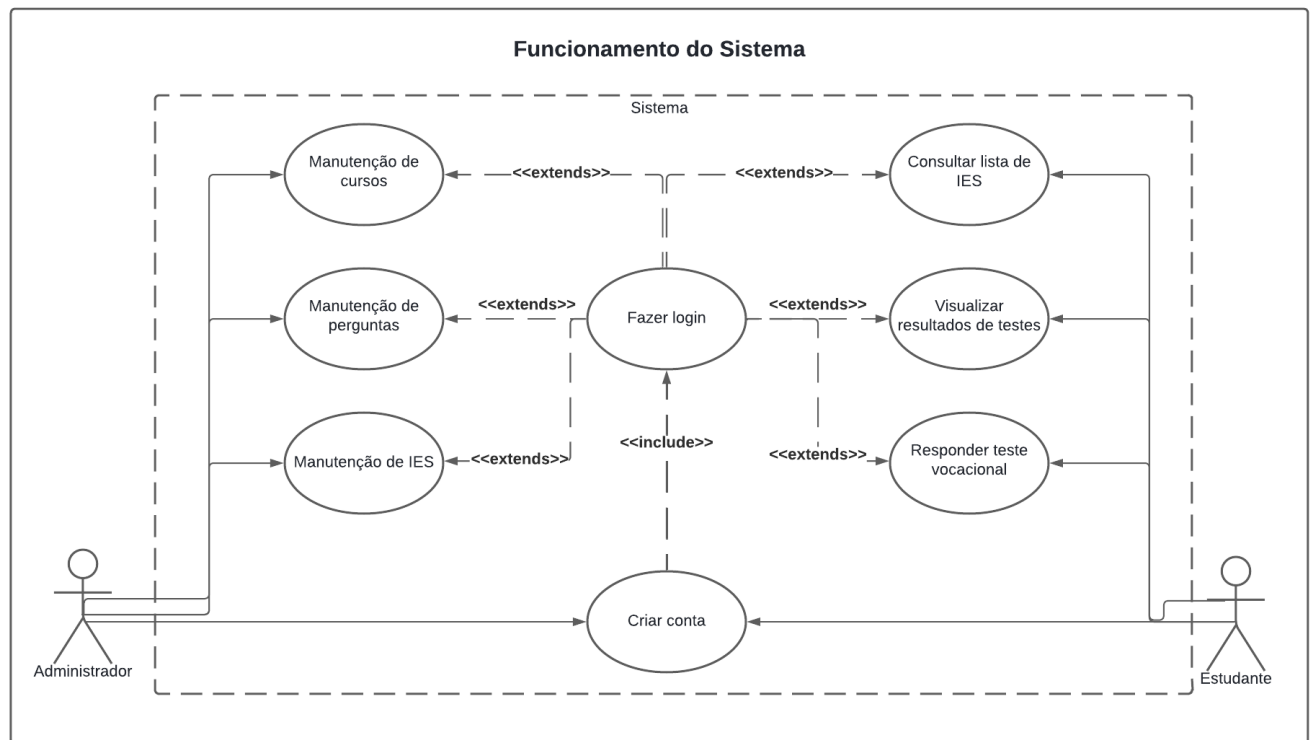


Figura 4 – Caso de Uso: Funcionamento do Sistema

Tabela 2 – Fluxo de Cadastro de Usuário (Administrador)

Caso de Uso	Cadastro de Usuário - Administrador
Ator Principal	Administrador
Interessados e Interesses	O administrador deseja criar e gerenciar sua conta para acessar as funcionalidades de manutenção das perguntas do teste vocacional.
Pré-Condições	O administrador possui permissão para ter um cadastro na plataforma.
Fluxo Básico	1. O administrador acessa a página de cadastro. 2. O sistema solicita dados pessoais. 3. O administrador insere os dados. 4. O sistema valida os dados. 5. O sistema verifica se o e-mail é único. 6. O sistema salva os dados. 7. O administrador recebe confirmação de sucesso.
Fluxos Alternativos	1. E-mail já em uso: mensagem de erro. 2. E-mail inválido: mensagem de erro. 3. CPF inválido: mensagem de erro. 4. Campos obrigatórios não preenchidos: mensagem de erro.
Fluxo de Exceção	Falha técnica: mensagem de erro, caso de uso encerrado.
Pós-Condições	Cadastro realizado com sucesso e administrador pode acessar a plataforma.

**Fonte:** Elaborado pelo autor.

Tabela 3 – Fluxo de Cadastro de Instituição Pública

<b>Caso de Uso</b>	<b>Cadastro de Instituição Pública</b>
Ator Principal	Administrador
Interessados e Interesses	O administrador deseja cadastrar uma instituição pública na plataforma para disponibilizar informações sobre seus cursos, formas de ingresso e políticas públicas de acesso e permanência.
Pré-Condições	1. O administrador possui cadastro na plataforma. 2. A instituição possui as informações necessárias para cadastro.
Fluxo Básico	1. O administrador acessa a funcionalidade de cadastro de instituição. 2. O sistema solicita os dados da instituição. 3. O administrador preenche os dados. 4. O sistema valida os dados. 5. O sistema valida e verifica a unicidade dos dados. 6. O sistema salva os dados. 7. O administrador recebe confirmação de sucesso.
Fluxos Alternativos	1. Nome da instituição já em uso: mensagem de erro. 2. Site inválido: mensagem de erro. 3. Nota do MEC inválida: mensagem de erro. 4. Endereço incompleto: mensagem de erro.
Fluxo de Exceção	Falha técnica: mensagem de erro, caso de uso encerrado.
Pós-Condições	Instituição pública cadastrada com sucesso na plataforma.

**Fonte:** Elaborado pelo autor.

Tabela 4 – Fluxo de Cadastro de Usuário (Estudante)

<b>Caso de Uso</b>	<b>Cadastro de Usuário - Estudante</b>
Ator Principal	Estudante
Interessados e Interesses	O estudante deseja se cadastrar na plataforma para ter acesso ao teste vocacional personalizado e às informações sobre as universidades públicas.
Pré-Condições	O estudante possui mais de 14 anos de idade.
Fluxo Básico	1. O estudante acessa a página de cadastro. 2. O sistema solicita dados pessoais. 3. O estudante insere os dados. 4. O sistema valida os dados. 5. O sistema cria a conta e salva os dados. 6. O estudante recebe confirmação de sucesso.
Fluxos Alternativos	1. E-mail inválido: mensagem de erro. 2. Campos obrigatórios não preenchidos: mensagem de erro. 4. Campos obrigatórios não preenchidos: mensagem de erro.
Fluxo de Exceção	Falha técnica: mensagem de erro, caso de uso encerrado.
Pós-Condições	Cadastro realizado com sucesso e estudante pode acessar a plataforma.

**Fonte:** Elaborado pelo autor.

Tabela 5 – Fluxo de Manter Perguntas do Teste Vocacional

<b>Caso de Uso</b>	<b>Manter Perguntas do Teste Vocacional</b>
Ator Principal	Administrador
Interessados e Interesses	O administrador deseja gerenciar as perguntas do teste vocacional para garantir a qualidade e relevância do teste para os estudantes.
Pré-Condições	1. O administrador possui acesso ao sistema. 2. Deve haver ao menos uma pergunta cadastrada para alterar/excluir.
Fluxo Básico	1. O administrador acessa a funcionalidade de manutenção de perguntas. 2. O sistema exibe a lista de perguntas. 3. O administrador escolhe incluir, alterar, excluir ou consultar uma pergunta. 4. O sistema executa a ação e atualiza a lista.
Fluxos Alternativos	1. Pergunta com texto vazio: mensagem de erro. 2. Pergunta inexistente: mensagem de erro na alteração/exclusão.
Fluxo de Exceção	Falha técnica: mensagem de erro, caso de uso encerrado.
Pós-Condições	Pergunta incluída, alterada, excluída ou consultada com sucesso, mantendo a qualidade do teste.

**Fonte:** Elaborado pelo autor.

Tabela 6 – Fluxo de Manter Informações sobre Cursos

<b>Caso de Uso</b>	<b>Manter Informações sobre Cursos</b>
Ator Principal	Administrador
Interessados e Interesses	O administrador deseja manter informações atualizadas sobre os cursos oferecidos pela instituição pública na plataforma.
Pré-Condições	1. O administrador possui acesso à plataforma. 2. A instituição já está cadastrada.
Fluxo Básico	1. O administrador acessa a funcionalidade de manutenção de cursos. 2. O sistema apresenta a lista de cursos. 3. O administrador escolhe incluir, alterar, excluir ou consultar um curso. 4. O sistema executa a ação e atualiza a lista.
Fluxos Alternativos	1. Informações incompletas/ inválidas: mensagem de erro na inclusão. 2. Curso inexistente: mensagem de erro na alteração/exclusão.
Fluxo de Exceção	Falha técnica: mensagem de erro, caso de uso encerrado.
Pós-Condições	Informações sobre os cursos atualizadas com sucesso, permitindo consulta pelos usuários.

**Fonte:** Elaborado pelo autor.



Tabela 7 – Fluxo de Responder o Teste Vocacional

Caso de Uso	Responder o Teste Vocacional
Ator Principal	Estudante
Interessados e Interesses	O estudante deseja responder ao teste vocacional disponibilizado pela plataforma.
Pré-Condições	O estudante deve estar autenticado no sistema.
Fluxo Básico	1. O estudante acessa a funcionalidade de responder ao teste. 2. O sistema apresenta as perguntas. 3. O estudante responde às perguntas. 4. O estudante finaliza o teste. 5. O sistema registra as respostas no banco de dados.
Fluxos Alternativos	1. Teste interrompido: sistema preserva respostas parciais, mas só armazena após a conclusão do teste.
Fluxo de Exceção	Falha técnica: mensagem de erro, estudante pode tentar novamente após resolução do problema.
Pós-Condições	Respostas registradas no sistema para gerar recomendações de cursos.

**Fonte:** Elaborado pelo autor.

Tabela 8 – Fluxo de Apresentar Recomendação de Curso

Caso de Uso	Apresentar Recomendação de Curso
Ator Principal	Sistema
Interessados e Interesses	O sistema precisa fornecer uma recomendação personalizada de cursos para o estudante com base nas respostas do teste vocacional, auxiliando-o na escolha de sua formação acadêmica.
Pré-Condições	1. O estudante realizou o teste vocacional. 2. O sistema possui algoritmos de análise e recomendação configurados.
Fluxo Básico	1. O sistema analisa as respostas do teste vocacional. 2. O sistema gera uma recomendação personalizada de cursos. 3. O sistema apresenta a recomendação ao estudante. 4. O sistema armazena a recomendação no banco de dados.
Fluxos Alternativos	1. Algoritmo de recomendação falha: sistema exibe uma mensagem de erro e sugere tentar novamente mais tarde.
Fluxo de Exceção	Falha técnica: mensagem de erro, estudante pode tentar novamente após resolução do problema.
Pós-Condições	Recomendação de curso gerada e apresentada ao estudante.

**Fonte:** Elaborado pelo autor.

## 5.3 Volumetria

Para a fase inicial do projeto, estimamos a seguinte volumetria de dados para a plataforma:

- Estudantes: A aplicação será lançada sem estudantes cadastrados.
- Administradores: Começaremos com 5 administradores cadastrados, que são os membros integrantes do projeto.
- Instituições: Inicialmente, todas as universidades federais do Brasil serão cadastradas, totalizando 69 instituições.
- Áreas: A aplicação iniciará com 12 áreas cadastradas.
- Cursos: A aplicação iniciará com 50 cursos cadastrados.
- Políticas Públicas: A aplicação será lançada com 20 políticas públicas cadastradas, relacionadas com as instituições que as possuem.
- Testes Vocacionais: Disponibilizaremos inicialmente uma única opção de teste vocacional para os usuários.
- Perguntas: O teste vocacional inicialmente terá 30 perguntas.
- Perfis: Para recomendar cursos aos usuários, utilizaremos os 6 perfis definidos pela Teoria de Escolhas Vocacionais de John Holland.

De acordo com o número de registros estimados, o espaço ocupado no nosso banco de dados no momento de lançamento da plataforma será aproximadamente 3 MB. Optamos por iniciar com esse volume de dados por ser adequado para a fase inicial do projeto. No entanto, é importante destacar que haverá uma expansão desses dados em futuras entregas da aplicação, conforme será detalhado no Capítulo 6, Seção 5.3.

## 5.4 Arquitetura

Nesse projeto, a **Clean Architecture** foi escolhida como a base para o desenvolvimento do *software*, refletindo a necessidade de criar um sistema que seja ao mesmo tempo flexível e de fácil manutenção. A adoção desta abordagem foi motivada pela sua capacidade de promover uma separação clara e eficiente entre diferentes camadas do sistema, como entidades de negócio, casos de uso, interfaces de usuário e infraestrutura. Essa estrutura não só garante uma organização modular do código, mas também facilita a realização de testes automatizados em cada camada, assegurando a qualidade e permitindo a detecção precoce de problemas.

A Clean Architecture permite que as regras de negócio sejam desacopladas das tecnologias específicas, o que confere ao sistema uma maior adaptabilidade às mudanças futuras. Isso é crucial para a evolução contínua e sustentável do software. No entanto, a implementação dessa arquitetura não foi isenta de desafios. A necessidade de definir claramente as responsabilidades de cada camada de forma a seguir o princípio de responsabilidade única e garantir uma comunicação eficaz entre elas exigiu planejamento e atenção aos detalhes. Além disso, a adoção de uma arquitetura modular impôs um esforço adicional na integração de componentes e na garantia de que as interfaces entre as camadas fossem bem projetadas e funcionais. Esses desafios foram superados por meio de uma abordagem iterativa e de refinamento contínuo, que assegurou que a arquitetura atendesse às necessidades do projeto e suportasse a evolução do software de maneira eficiente.

Na Figura 5 é possível observar a representação em diagrama da arquitetura escolhida para a aplicação.

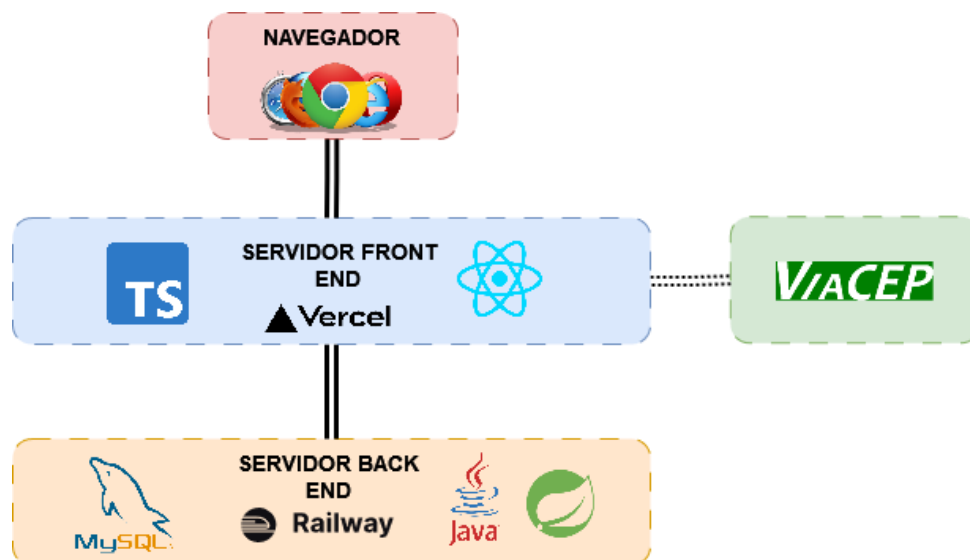


Figura 5 – Arquitetura da aplicação

A seguir detalharemos cada uma das tecnologias escolhidas e como será feita a comunicação entre elas.

#### 5.4.1 Back-End

No que diz respeito ao *back-end*, adotamos o *framework* **Spring Boot** em conjunto com a arquitetura **Model-Service-Controller (MSC)**. O Spring Boot é uma escolha popular para o desenvolvimento de aplicativos Java devido à sua facilidade de configuração e rápida inicialização, como discutido nas Subseções 5.1.1 e 5.1.2 do Capítulo 6. A arquitetura **MSC** divide a aplicação em três camadas distintas: **Model**, onde definimos as entidades e regras de negócio; **Service**, responsável pela lógica de negócio e manipulação dos dados;

e **Controller**, que recebe as requisições *Hypertext Transfer Protocol* (HTTP) e coordena a interação entre o cliente e o servidor.

Além das camadas Model, Service e Controller, também implementamos a camada **Repository** como parte da arquitetura *MSC*. A camada Repository é responsável pela interação com o banco de dados, encapsulando a lógica de acesso e manipulação dos dados. Utilizando interfaces e métodos específicos, o Repository abstrai as operações *Create, Read, Update, Delete* (CRUD), permitindo que as demais camadas da aplicação interajam com o banco de dados de forma desacoplada e eficiente.

A utilização da arquitetura em camadas proporcionou diversos benefícios para o projeto. Em primeiro lugar, a separação de responsabilidades entre as camadas Model, Service, Controller e Repository facilitou a organização e compreensão do código-fonte, tornando-o mais legível e manutenível. Cada camada possui um propósito claro e bem definido, o que simplifica o processo de desenvolvimento e permite que diferentes equipes ou desenvolvedores trabalhem de forma independente em áreas específicas da aplicação.

#### 5.4.1.1 Gerenciamento de dependências

Para o gerenciamento de dependências do projeto, optamos pelo **Maven**. Esta escolha se baseia nos inúmeros benefícios oferecidos por essa ferramenta. O Maven simplifica a gestão de dependências, permitindo que as bibliotecas necessárias sejam facilmente adicionadas ao projeto através de seu arquivo de configuração padrão (pom.xml). Além disso, o Maven automatiza tarefas como compilação, empacotamento e distribuição do projeto, tornando o processo de desenvolvimento mais eficiente e menos propenso a erros. Sua vasta gama de plugins oferece funcionalidades adicionais, como integração com ferramentas de teste e análise de código, promovendo uma abordagem mais completa para o desenvolvimento de *software*.

#### 5.4.2 Front-End

Nesse projeto, optamos por utilizar **React** em conjunto com **TypeScript** para o desenvolvimento do *front-end*. Essa escolha foi motivada pela necessidade de construir uma interface de usuário dinâmica e reativa, com uma base de código robusta e escalável, conforme discutido nas Subseções 5.1.3 e 5.1.4 do Capítulo 6, onde são abordadas a adoção do TypeScript e a utilização do React para garantir produtividade, qualidade do código. O React é uma biblioteca JavaScript amplamente adotada que nos permite criar componentes reutilizáveis e composicionais, facilitando a construção de interfaces complexas de forma modular. Além disso, a integração com TypeScript adiciona um sistema de tipos estáticos ao JavaScript, oferecendo benefícios como detecção de erros em tempo de compilação e melhorando a manutenibilidade do código. A arquitetura baseada

em componentes nos permitiu organizar a aplicação de forma hierárquica, facilitando a manutenção e o teste dos diferentes elementos da interface.

Na figura 6, é possível ver o QRcode da aplicação que foi publicada na internet:

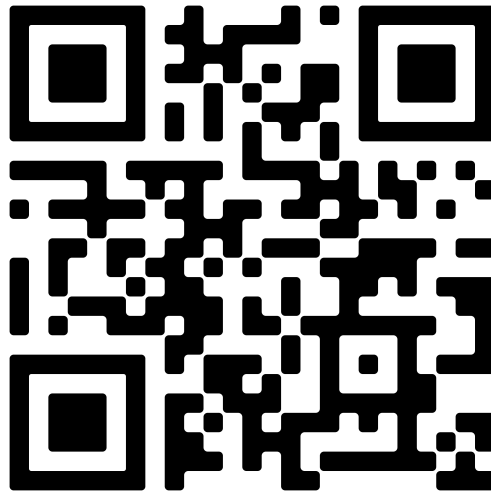


Figura 6 – QRcode URL da Aplicação

<https://vocco.vercel.app>

### 5.4.3 Banco de dados

Em relação ao banco de dados, optamos pelo **MySQL** devido à sua confiabilidade, desempenho e ampla adoção na indústria. O MySQL é um sistema de gerenciamento de banco de dados relacional que oferece recursos avançados de segurança e escalabilidade, sendo uma escolha sólida para aplicações de diferentes tamanhos e complexidades. Utilizamos o MySQL para armazenar e gerenciar os dados da aplicação, garantindo a integridade e disponibilidade das informações.

#### 5.4.3.1 Modelo Entidade Relacionamento

O **Modelo Entidade Relacionamento (MER)** é uma ferramenta fundamental no design de bancos de dados, pois permite visualizar de forma clara as relações entre as diversas entidades do domínio do sistema. O MER da plataforma Vocco pode ser observado na Figura 7.

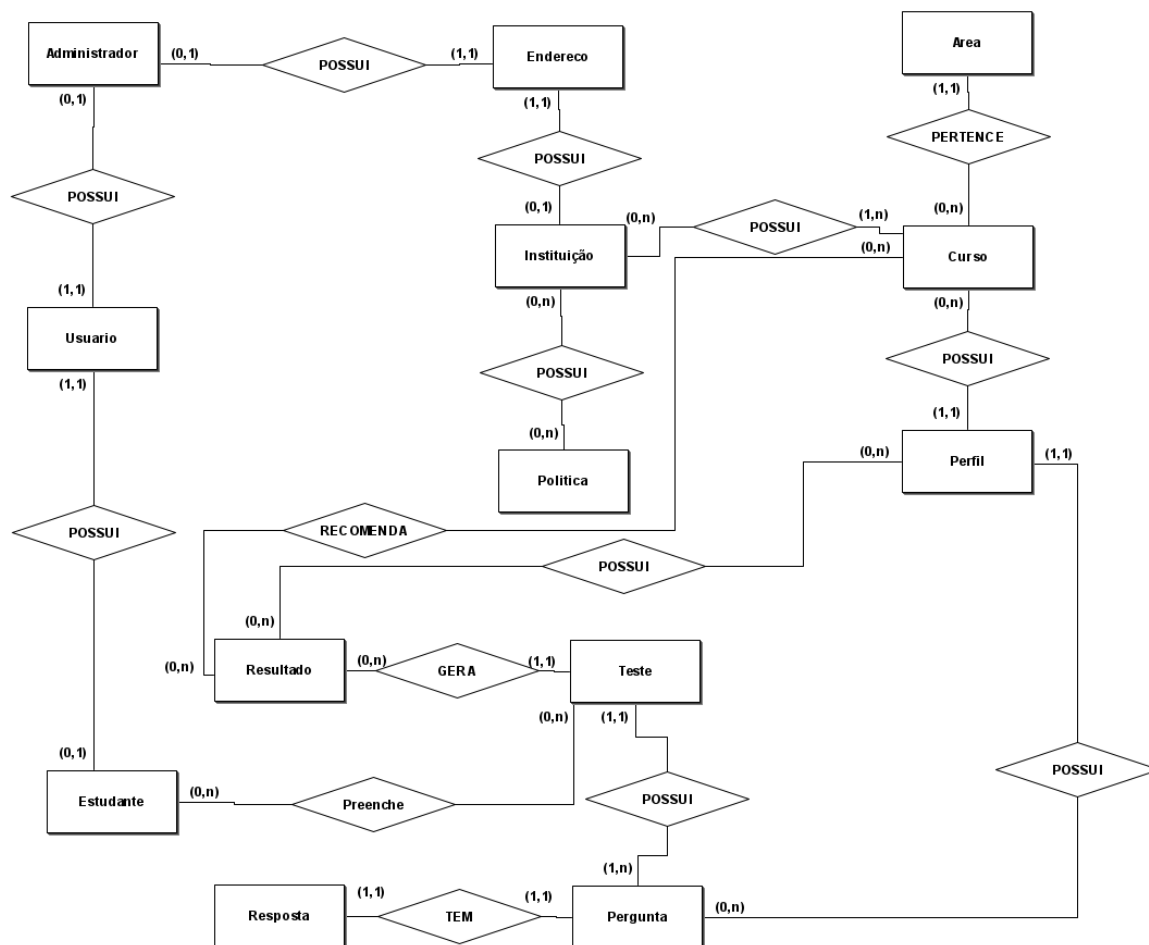


Figura 7 – Modelo Entidade Relacionamento (MER) do projeto

No processo de elaboração do diagrama, nosso foco primordial reside na visualização das relações entre as tabelas. Esse enfoque nos capacitou a estabelecer um sistema com informações confiáveis e interligadas, fundamentais para garantir a integridade e a eficácia dos dados.

### 5.4.3.2 Diagrama Entidade Relacionamento

O **Diagrama Entidade-Relacionamento (DER)** é uma ferramenta muito importante na modelagem de dados. Ele oferece uma visão clara das entidades, seus atributos e como elas se relacionam. Uma técnica central no DER é o uso de chaves estrangeiras para mapear as relações entre tabelas. Essas chaves garantem a integridade referencial entre os dados, assegurando que cada registro em uma tabela relacionada possa ser corretamente associado a outro em uma tabela diferente. O DER da plataforma Vocco pode ser observado na Figura 8

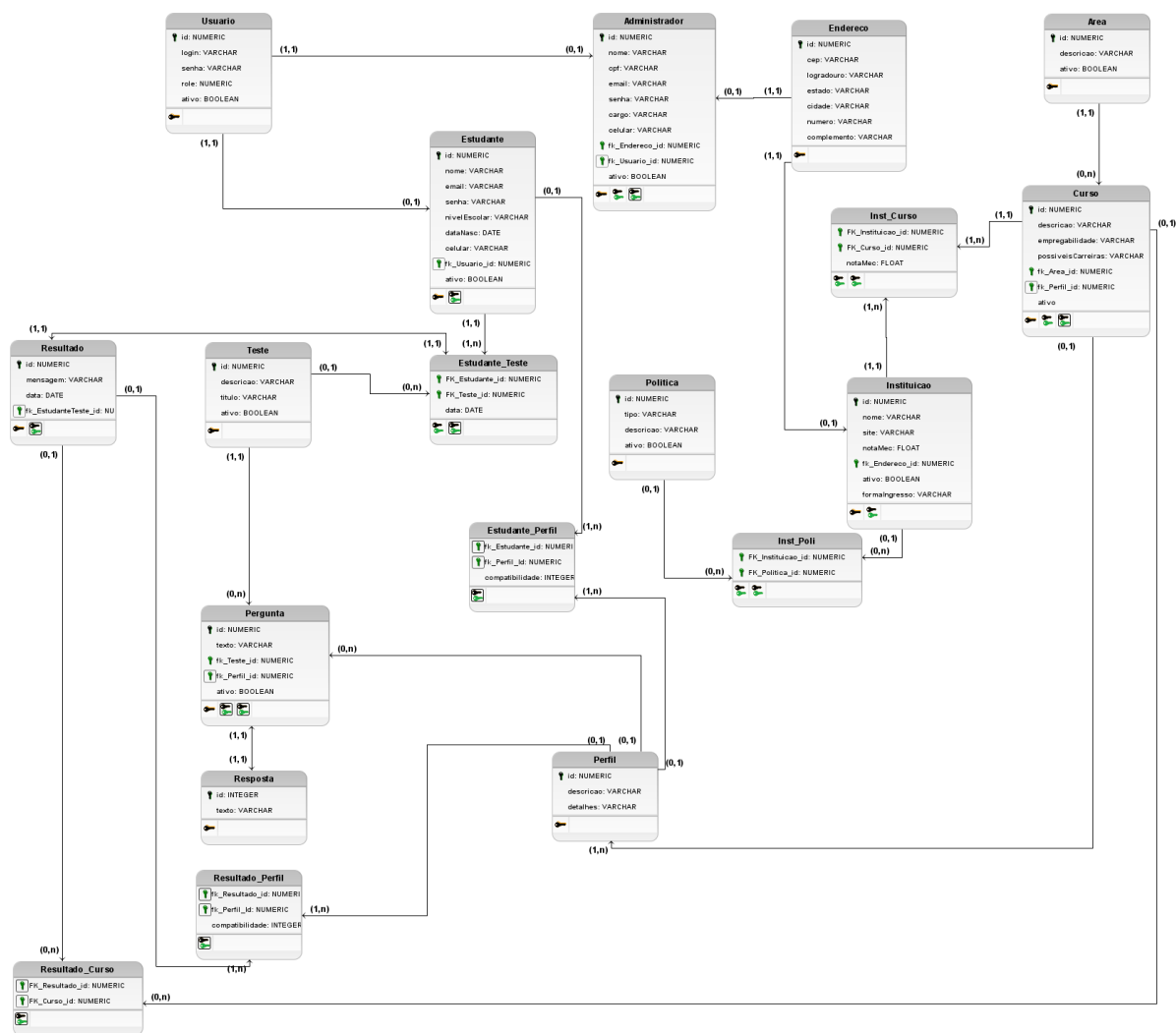


Figura 8 – Diagrama Entidade Relacionamento (DER) do projeto

Este diagrama oferece uma visão clara do processo de mapeamento dos relacionamentos entre as tabelas do banco de dados. Seguimos um padrão que envolve a criação de entidades associativas para representar os relacionamentos N-N. Para os relacionamentos 1-N, adicionamos a chave estrangeira na tabela que recebe o 'N' no mapeamento. Quanto



aos relacionamentos 1-1 e 0-1, a escolha da tabela que recebe a chave estrangeira foi feita com base no que a equipe considerou mais relevante, levando em conta os requisitos específicos da plataforma.

#### 5.4.4 Integrações

Para aprimorar a experiência do usuário e garantir a precisão dos dados de endereço nos cadastros, optamos por integrar a [Interface de Programação de Aplicação \(API\)](#) do **ViaCEP**. Essa integração permite que, ao preencher os campos de endereço em formulários de cadastro, o sistema automaticamente consulte a [API](#) do ViaCEP para obter informações detalhadas, como rua, bairro, cidade e estado, com base no [Código de Endereçamento Postal \(CEP\)](#) fornecido pelo usuário. Essa abordagem simplifica e agiliza o processo de preenchimento dos cadastros, além de garantir a consistência e atualização dos dados de endereço. Ao utilizar essa [API](#), estamos priorizando a precisão e a eficiência na coleta e manipulação das informações de endereço, proporcionando uma experiência mais fluida e intuitiva para os usuários.

#### 5.4.5 Infraestrutura

Por fim, para hospedar nosso projeto, optamos por utilizar serviços de hospedagem em nuvem para abrigar tanto o *front-end* quanto o *back-end*. Essa escolha foi motivada por diversos fatores que visam garantir uma maior eficiência e escalabilidade do sistema.

Primeiramente, a hospedagem em nuvem oferece uma infraestrutura altamente flexível, permitindo ajustes rápidos de recursos conforme a demanda do aplicativo. Além disso, ela proporciona uma maior disponibilidade e confiabilidade, com garantias de tempo de atividade elevado e redundância de dados. A facilidade de implementação e gerenciamento também foi considerada, já que a infraestrutura em nuvem permite uma rápida implantação e manutenção simplificada do sistema.

Por fim, os custos operacionais são otimizados, pois os serviços em nuvem geralmente seguem um modelo de pagamento conforme o uso, evitando gastos excessivos com infraestrutura desnecessária.

#### 5.4.6 Front - End

Para hospedar o *front-end*, escolhemos a plataforma **Vercel**, conhecida por sua facilidade de uso e escalabilidade. O *front-end* foi desenvolvido com React e TypeScript, aproveitando os benefícios de uma tipagem estática para garantir um código mais robusto e menos propenso a erros. O uso do Vite como *bundler* permitiu um processo de desenvolvimento mais rápido e eficiente, gerando builds otimizados e facilitando a implementação de novas funcionalidades.

Além disso, é importante destacar que o *front-end* se conecta ao *back-end* através de requisições *Hypertext Transfer Protocol Secure* (HTTPS), garantindo a segurança e integridade dos dados transmitidos entre as partes. Essa abordagem assegura uma comunicação confiável e protegida entre os diferentes componentes do sistema, contribuindo para uma experiência segura e estável para os usuários.

#### 5.4.6.1 Back - End

Para hospedar o *back-end*, foi escolhida a plataforma **Railway**, conhecida por sua eficiência e facilidade de integração. Para o gerenciamento do banco de dados, optamos pelo MySQL, um sistema robusto e amplamente utilizado. A integração entre o Spring Boot e o MySQL foi facilitada através de um container Docker, gerado automaticamente pela plataforma Railway, proporcionando uma configuração que garante a conexão eficiente entre o *core* da nossa aplicação e os dados persistidos, sendo que a conexão entre a aplicação e o banco foi realizada utilizando variáveis de ambiente definidas dentro do Railway, aumentando a segurança e confiabilidade.

#### 5.4.7 Escalabilidade

A escalabilidade é crucial para garantir que a Vocco possa se adaptar ao crescimento de forma eficaz, mantendo flexibilidade, confiabilidade, eficiência e manutenibilidade. Acreditamos que tais metas serão alcançadas por meio de nossas decisões de utilizar serviços em nuvem e um sistema gerenciador de banco de dados que facilite o processo de expansão do nosso sistema. Neste momento, nosso foco está em manter uma **escalabilidade vertical**, o que implica em aumentar a capacidade de armazenamento em vez de adicionar mais servidores. Essa abordagem nos permite otimizar recursos e simplificar o gerenciamento, garantindo que Vocco possa se adaptar ao crescimento de maneira ágil e econômica.

#### 5.4.8 Controle de versão

No desenvolvimento deste projeto, optamos por utilizar tanto o **Git** como o **GitHub** para o controle de versão do código-fonte. O Git foi escolhido como sistema de controle de versão devido à sua eficiência e robustez no gerenciamento de alterações de código, permitindo o acompanhamento do histórico de modificações, a colaboração entre os membros da equipe e a criação de branches para o desenvolvimento de novas funcionalidades de forma isolada. Além disso, o GitHub foi utilizado como plataforma de hospedagem remota dos repositórios Git, proporcionando uma maneira conveniente de compartilhar o código entre os membros da equipe, revisar alterações, gerenciar problemas e automatizar processos de *Integração Contínua* (CI) e *Distribuição Contínua* (CD).

Na figura 9, é possível acessar o nosso repositório do GitHub do projeto:

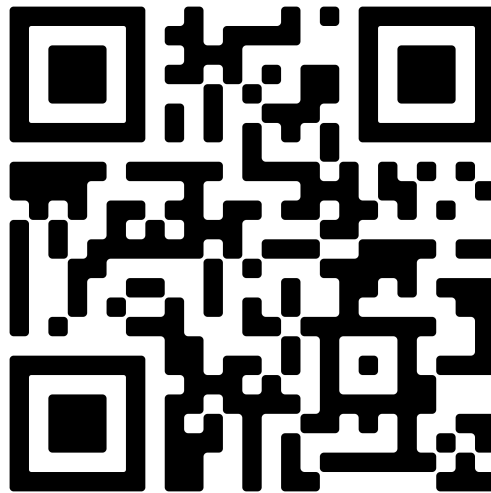


Figura 9 – QRcode Repositório do GitHub

A utilização do Git e do GitHub garante que os membros da equipe colaborem de maneira eficaz e organizada, mantendo o código atualizado e permitindo uma gestão eficiente das tarefas e do progresso do projeto.

## 5.5 Manutenibilidade

A manutenibilidade de um *software* refere-se à facilidade com que ele pode ser mantido e modificado ao longo do tempo. Em outras palavras, é a capacidade do sistema de ser compreendido, adaptado, corrigido e aprimorado de maneira eficiente e econômica. A manutenibilidade é uma característica fundamental de qualidade do *software*, pois afeta diretamente a capacidade da equipe de desenvolvimento de responder a novos requisitos, corrigir falhas e melhorar a funcionalidade do sistema ao longo do ciclo de vida do *software*.

### 5.5.1 Code Convention

A adoção de convenções de código padronizadas é fundamental para garantir a legibilidade, consistência e manutenção eficiente do software. Este capítulo aborda as práticas recomendadas e as diretrizes específicas que devem ser seguidas por todos os desenvolvedores do projeto ao escrever e revisar o código.

- **Back-End** Seguiremos as convenções de código para a linguagem Java conforme estabelecidas na documentação oficial da Oracle. Algumas das principais orientações incluem:

- **Nomenclatura de Classes, Interfaces e Tipos:** Utilizar PascalCase, começando com letra maiúscula;
  - **Nomenclatura de Métodos e Variáveis:** Utilizar camelCase, começando com letra minúscula;
  - **Constantes e variáveis:** Utilizar letras maiúsculas separadas por sublinhados;
  - **Indentação:** Utilizar quatro espaços para indentação e evitar o uso de tabulação;
  - **Comentários:** Explicar trechos de código complexos ou documentar [API](#);
  - **Linhas de Comprimento:** Limitar o comprimento a 80-100 caracteres para melhorar a legibilidade;
  - **Imports:** Evitar importar pacotes inteiros, trazendo apenas classes específicas ou utilizar asterisco apenas para *imports* estáticos.
- **Front-End** Para o desenvolvimento *front-end*, adotaremos as principais convenções de código:
    - **Nomenclatura de variáveis:** Utilizaremos camelCase, começando com letra minúscula;
    - **Nomenclatura de funções:** Utilizaremos PascalCase, começando com letra maiúscula;
    - **Declaração de variáveis:** Preferimos utilizar **let** ou **const** em vez de **var** para declarar variáveis, pois isso ajuda a evitar problemas de escopo;
    - **Convenções de nome para constantes:** Nomearemos constantes utilizando letras maiúsculas e palavras separadas por sublinhados;
    - **Comentários:** Faremos uso de comentários para documentar o código, explicando o propósito de funções e partes importantes do código;
    - **Tratamento de erros:** Faremos uso de blocos try-catch para capturar e lidar com exceções, quando necessário;
    - **Ponto e vírgula:** Apesar de o JavaScript permitir a omissão de ponto e vírgula no final das declarações, incluiremos sempre para evitar comportamentos inesperados;
    - **Uso de aspas:** Utilizaremos aspas duplas de forma consistente para *strings*.

### 5.5.2 Ferramentas de testes

Os testes de *software* são uma etapa fundamental no ciclo de desenvolvimento de *software*, cujo objetivo é garantir a qualidade e a confiabilidade do produto final. Essas

atividades consistem em verificar se o *software* atende aos requisitos estabelecidos, se funciona corretamente em diferentes cenários de uso e se está livre de defeitos e erros. Os testes de *software* podem abranger diversas áreas, como funcionalidade, desempenho, segurança e usabilidade, e são conduzidos por meio de técnicas e ferramentas específicas.

Para a realização dos testes utilizaremos as seguintes ferramentas:

- **JUnit:** Será utilizado no *back-end* o *framework* JUnit como ferramenta de testes automatizados. Ele oferece recursos que facilitam a criação e execução de testes, incluindo suporte para testes parametrizados, testes de exceção e integração com ferramentas que utilizaremos para o desenvolvimento.
- **Jest:** Para os testes automatizados no *front-end*, escolhemos o Jest como nossa ferramenta principal. Ele abrange uma ampla gama de cenários de teste, desde os unitários mais simples até os de integração mais complexos, destacando-se pela sua facilidade e rapidez de uso.
- **Typescript ESLint:** Para conduzir uma análise estática abrangente em nosso *front-end*, decidimos adotar o TypeScript ESLint. Essa escolha se deve ao fato de que essa ferramenta combina as vantagens de *linting* de código, e nos permite definir regras personalizadas, garantindo que sigamos as melhores práticas e padrões de codificação estabelecidos.
- **SonarQube:** Utilizaremos o SonarQube como nossa ferramenta de análise estática do *back-end*. Ele realiza uma avaliação detalhada do código, identificando potenciais problemas, vulnerabilidades de segurança, *bugs*, duplicações de código e padrões de código não conformes.

### 5.5.3 Design Patterns

Optamos por adotar o padrão **SOLID** como base para o desenvolvimento deste sistema, pois ele encapsula um conjunto de cinco princípios essenciais de design de software, idealizados por Robert C. Martin. Esses princípios são fundamentais para criar um código mais limpo, modular e fácil de manter. Ao seguirmos esses princípios, garantimos que nossas classes e módulos tenham responsabilidades bem definidas, promovendo a coesão e o baixo acoplamento, o que facilita tanto a compreensão do código quanto sua manutenção no futuro.

Também utilizaremos o padrão [Data Transfer Object \(DTO\)](#) para permitir a comunicação eficiente de dados entre diferentes partes do nosso sistema. Para representar esses objetos de transferência de dados, optaremos por utilizar *records*, aproveitando as suas vantagens de imutabilidade e clareza.

## 5.6 Segurança, Privacidade e Legislação

A segurança representa um aspecto essencial na arquitetura da Vocco. Dessa forma, nesta seção serão citadas as boas práticas e tecnologias usadas para garantir uma maior confiabilidade e segurança para os usuários de nossa aplicação.

### 5.6.1 Lei Geral de Proteção de Dados (LGPD)

A [Lei Geral de Proteção de Dados \(LGPD\)](#) (Congresso Nacional, 2018), foi promulgada com o objetivo de proteger os direitos fundamentais de liberdade e de privacidade, e a livre formação da personalidade de cada indivíduo, atuando sobre o tratamento de dados pessoais, incluindo em meios digitais. Com isso, durante o planejamento da estruturação da plataforma, foram definidas ações visando atender a essas necessidades de segurança, para dessa forma assegurar que a aplicação estivesse em conformidade com essa legislação e atendendo ao compromisso de proteger nossos usuários.

A aplicação respeita o livre acesso a esses dados, definido no Art. 6º, IV, e requisitará consentimento dos usuários para armazenar os dados fornecidos, o que confere a ela conformidade com o Art. 7º, I.

### 5.6.2 Autenticação e Autorização

A aplicação conta com etapas de autenticação e autorização à depender do usuário e seus privilégios, para garantir maior privacidade às informações e também uma melhor gestão de como os recursos serão acessados tanto no *back-end* como no *front-end*.

A autenticação como sendo o login do usuário no sistema e a autorização como sendo o processo posterior, em que é verificado se o usuário tem permissão de acesso a um determinado recurso. O Spring Security é um *framework* do Java que possui um sistema de autenticação e autorização para aplicações Java, como é o caso da Vocco, além disso conta com proteção contra ataques como *session fixation*, *clickjacking* e protege contra injeção de DDR, se mostrando uma ferramenta madura e qualificada para proteger a aplicação.

Para implementar o fluxo de autenticação, é usado o *Bearer Token*, que é gerado após a validação das credenciais pelo *back-end* Java Spring Boot, garantindo a autenticidade e evitando conflitos. O *Bearer Token* ou "token de portador", é um tipo de token de acesso usado para acessar recursos protegidos, como [API](#). Esse *token* é gerado pelo servidor em resposta a uma solicitação de login, e deve ser incluído no cabeçalho de autorização das solicitações [HTTP](#) para recursos protegidos.

### 5.6.3 Criptografia de Dados

A criptografia é uma alternativa ideal para anonimização de dados, reduzindo as chances de violação de dados e as multas que a [LGPD](#) pode impor. Desempenha um papel fundamental na proteção das informações, tornando-as ininteligíveis para terceiros não autorizados, impedindo que pessoas não autorizadas utilizem as informações para benefício próprio, embaralhando os dados de forma que não sejam legíveis, tanto por humanos quanto por sistemas projetados para interpretar informações.

Para aplicar essa medida de segurança na aplicação Vocco, será usada a ferramenta **BCrypt**, que tem o intuito de esconder senhas criadas pelos usuários em forma de texto puro em dados indecifráveis, utilizando o algoritmo *hash*. Será utilizada essa ferramenta para criptografar as senhas dos usuários no banco de dados.

## 5.7 Viabilidade Financeira

A análise da viabilidade financeira demonstra o custo operacional estimado para o primeiro ano após o lançamento da plataforma. Esta avaliação considera os custos de mão de obra de desenvolvimento, registro de domínio, utilização dos serviços oferecidos pela plataforma de desenvolvimento na nuvem Railway para hospedar o *back-end*, levando em conta um aumento de 30% no processamento e uso dos recursos da mesma e também pela utilização da plataforma Vercel para hospedar o *front-end*. Com base nesses parâmetros, segue a estimativa dos custos operacionais para o primeiro ano após o lançamento da plataforma.

Tabela 9 – Estimativa dos Custos Operacionais

Descrição	Valor Anual (R\$)	Valor Mensal (R\$)
Custo de Desenvolvimento	42.240,00	3.520,00
Registro de Domínio	40,00	3,33
Plataforma Railway	702,92	58,58
Plataforma Vercel	1.225,2	102,10
<b>Total</b>	<b>44.208,12</b>	<b>3.684,01</b>

**Fonte:** Elaborado pelo autor.

A estimativa dos custos operacionais detalhada, apresenta que o Custo de Desenvolvimento anual será calculado com base em 1.320 horas anuais de trabalho, cumpridas por uma gerente e quatro desenvolvedoras. A gerente será remunerada em R\$8,00 por hora, enquanto cada desenvolvedora será remunerada em R\$6,00 por hora. O valor de registro de domínio será de R\$40,00 o qual é uma taxa paga anualmente. Além disso, há o custo com a plataforma Railway em que são considerados os valores de utilização do plano mensal Hobby de R\$ 25,33, mais o valor adicional de R\$ 398,96 para o aumento de 30% no uso dos recursos da plataforma, e esse valor é dividido em 12 meses. Por fim, o

custo com a plataforma Vercel será de R\$102,10 com o plano mensal Pró. Esses valores são apresentados na Tabela 10 a seguir:

Tabela 10 – Detalhamento dos Custos Operacionais

Item	Descrição	Valor Anual (R\$)	Valor Mensal (R\$)
Desenvolvimento	1 Gerente	10.560,00	880,00
Desenvolvimento	4 Desenvolvedoras	31.680,00	2.640,00
Registro de Domínio	Plano Fixo Anual	40,00	3,33
Plataforma Railway	Plano mensal Hobby	303,96	25,33
Plataforma Railway	30% de uso adicional	398,96	33,25
Plataforma Vercel	Plano Pró	1.225,20	102,10
<b>Total</b>		<b>44.208,12</b>	<b>3.684,01</b>

**Fonte:** Elaborado pelo autor.

Dessa forma, a estimativa de Custo Operacional anual será de R\$44.208,12 e mensal de R\$3.684,01, como apresentado nas tabelas anteriores.

### 5.7.1 Monetização

A plataforma oferecerá suas funcionalidades gratuitamente, visando garantir a acessibilidade e engajamento dos usuários, porém apresentará campos com anúncios, com o intuito de cobrir os custos operacionais e sustentar o desenvolvimento contínuo. Dessa forma, a estratégia de monetização planejada envolve o estabelecimento de parcerias com cursos pré-vestibular visando que a receita gerada cubra os custos necessários.

A monetização será feita a partir de anúncios e conteúdo patrocinado, provenientes inicialmente de seis empresas de cursos pré-vestibular. O valor cobrado estimado será composto por uma taxa fixa de R\$ 600,00 e por uma comissão de 1,5% sobre as vendas realizadas através dos *links* disponibilizados na plataforma, considerando o valor médio de R\$ 253,00 em cada curso vendido. Os valores totais são apresentados na Tabela 11 a seguir:

Tabela 11 – Estimativa de Receitas Operacionais

Descrição	Valor Anual (R\$)	Valor Mensal (R\$)
Receita Fixa de Anúncios	43.200,00	3.600,00
Comissão de Vendas (1,5% sobre vendas)	2.736,00	228,00
<b>Total de Receita</b>	<b>45.936,00</b>	<b>3.828,00</b>

**Fonte:** Elaborado pelo autor.

Levando em consideração uma base estimada de 1.000 usuários ativos mensalmente e assumindo uma taxa de conversão de 1% desses usuários em vendas efetivas, projetamos uma receita mensal fixa no valor de R\$ 3.600,00 e comissão de vendas no valor de R\$



228,00. Dessa forma, a estimativa total de receita anual será de R\$ 45.936,00 e mensal de R\$ 3.828,00, como foi apresentado pela Tabela 11.

## 5.8 Fases de Entrega

As fases de entrega delimitam as datas das principais entregas do sistema para a disciplina, sendo elas a Prova de Conceito, o Produto Mínimo Viável e o Produto Final.

### 5.8.1 Prova de Conceito (POC)

Para a prova de conceito desenvolvemos o requisito de manutenção das [IES](#) cadastradas na plataforma Vocco, concebida como uma parte essencial do produto final planejado. Destinada aos administradores do sistema, essa funcionalidade foi desenvolvida para proporcionar uma gestão eficiente das instituições de ensino cadastradas, englobando desde o cadastro até associações com cursos e políticas públicas.

As funcionalidades entregues foram:

- **Tela inicial:** Tela disponibilizada para o administrador gerenciar as entidades da aplicação.
- **Cadastro:** Incluir informações sobre a instituição.
- **Listagem:** Visualizar todas as instituições ativas e inativas.
- **Detalhamento:** Acessar informações específicas de cada instituição.
- **Alteração:** Modificar dados cadastrais conforme necessário.
- **Exclusão:** Implementamos exclusão lógica, definindo o campo 'ativo' como falso.
- **Associação com Curso:** Vincular instituições a cursos específicos.
- **Associação com Política Pública:** Integrar instituições às políticas de entrada e permanência oferecidas.

Os protótipos das telas desenvolvidas podem ser acessados no apêndice [A](#).

### 5.8.2 Produto Mínimo Viável (MVP)

Na entrega do MVP, além do que foi desenvolvido na [PoC](#), será entregue a funcionalidade principal do sistema, que se trata do preenchimento do teste vocacional bem como a recomendação dos cursos e áreas que mais se encaixam no perfil do estudante, a depender das respostas enviadas. Além disso, para essa funcionalidade ser entregue é

necessário que o cadastro dos estudantes também já esteja implementado. Esta entrega será feita no mês de junho de 2024.

### 5.8.3 Produto Final

No Produto Final será entregue o projeto completo, contendo todas as funcionalidades propostas, como o teste vocacional, cadastro de administradores, cadastros de cursos e áreas, disponibilização das informações das Universidades estaduais do Estado de São Paulo, e as universidades Federais do Brasil, bem como todas as funcionalidades definidas anteriormente. É importante destacar que, durante o desenvolvimento do projeto, devido à sua complexidade e ao foco principal da aplicação, foi decidido descartar a integração com instituições estrangeiras que aceitam a nota do ENEM, conforme discutido na Seção 5.2 do Capítulo 6. Esta entrega será feita no mês de dezembro de 2024.

## 5.9 Métricas

Ao longo do desenvolvimento do projeto, o progresso foi acompanhado através de medições realizadas mensalmente tanto no repositório do projeto *front-end* quanto no repositório do projeto *back-end*. Conforme mostra o quadro foram analisados os itens: reuniões da equipe, posts no blog, vídeos no canal do youtube do projeto AdasTech, quantidade de requisitos, quantidade de entidades de Banco de Dados, Interfaces, classes, Tamanho do Projeto (em MegaBytes), quantidade de métodos, commits, Arquivos, linhas de código e informações sobre os testes unitários. De todos os elementos analisados na tabela abaixo, as informações sobre o número de classes, interfaces, métodos, atributos, arquivos e linhas de código, foram obtidas através da ferramenta MetricRunner. Os valores informados a cada mês são acumulativos.

Item	Fevereiro	Março	Abril	Maio	Junho	Julho
Reuniões	0	5	9	11	12	14
Posts no blog	0	4	8	11	11	11
Vídeos no Canal	0	1	2	3	3	3
Requisitos	0	0	14	14	14	14
Entidades de BD	0	0	11	18	18	18
Interfaces	0	0	0	15	19	19
Classes	0	0	0	25	29	29
Tamanho do Projeto (MB)	0	0	0,139	0,258	0,258	0,961
Métodos	0	0	1	160	160	198
Atributos	0	0	41	60	60	60
Commits	2	2	18	78	80	80
Arquivos	0	0	101	192	201	216
Imagens	0	0	0	1	4	4
Linhas de Código	0	1	954	11.206	11.651	13.438
Teste Unitário Quantidade Classes	0	0	0	0	0	9
Testes Unitários Cobertura (%)	0	0	0	0	0	50%

Figura 10 – Tabela de métricas do projeto primeiro semestre 2024

As métricas representam numericamente o andamento do projeto, demonstrando quais foram os períodos de maior atividade do grupo e o tamanho do projeto como um todo.

### 5.9.1 Ferramentas Auxiliares

Com o intuito de simplificar o processo de análise do projeto como um todo, foi usada a ferramenta MetricRunner, que auxilia na contagem do número de classes, interfaces, métodos, atributos, arquivos e linhas de código do projeto Vocco. Para fazer uso desta ferramenta é necessário cumprir os seguintes passos:

- 1- Buscar no Maven Central a Biblioteca Reflections versão 0.9.9-RC1;
- 2- Adicionar a dependência da biblioteca indicada, no arquivo Pom;
- 3- Adicionar a classe MetricRunner à aplicação;
- 4- Alterar o valor da variável rootPackage para o nome do pacote raiz da aplicação;
- 5- Executar a classe MetricRunner.

### 5.9.2 Estatísticas do Projeto

Para obtenção de elementos visuais, e mais informações estatísticas à respeito de ambos os projetos (*back-end* e *front-end*), foi utilizada a ferramenta GitStats para a geração de gráficos e tabelas informativas.

Na figura 11, são apresentados os dados gerais à respeito do repositório GitHub referente ao projeto *back-end*.

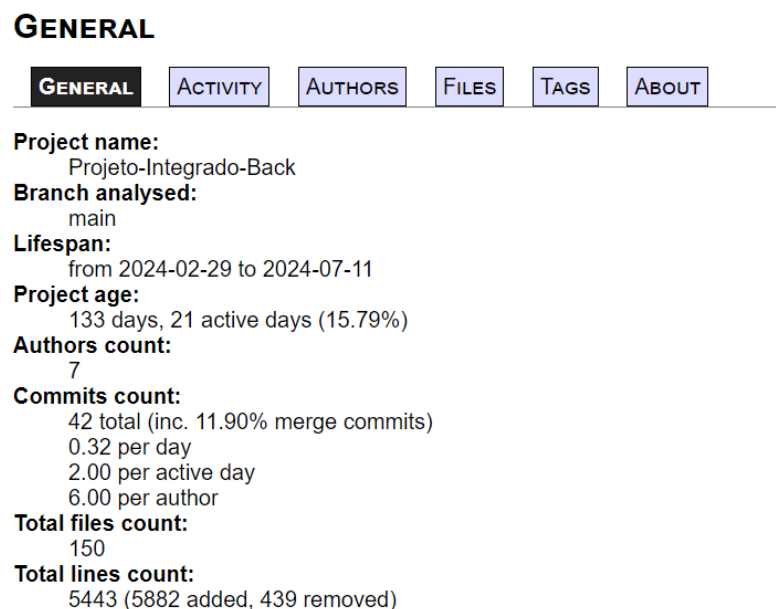


Figura 11 – Dados Gerais sobre o repositório *back-end* do GitHub

Na figura 12 são apresentados os dados referentes à relação entre o dia da semana, e o horário em que foram feitos os commits no repositório do projeto *back-end* do GitHub.

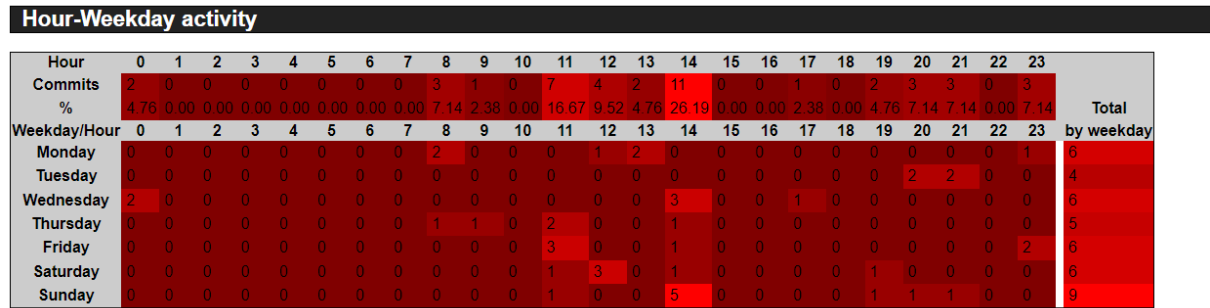


Figura 12 – Dados Gerais sobre o repositório *back-end* do GitHub

Na figura 13 são apresentados os dados referentes ao número de commits em cada mês do ano de 2024 referente ao repositório do projeto *back-end* do GitHub.

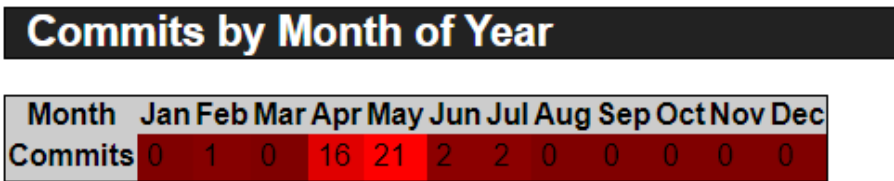


Figura 13 – Número de commits em cada mês do ano de 2024

Na figura 14 são apresentados os dados referentes ao número de commits por autor no repositório do projeto *back-end* do GitHub.

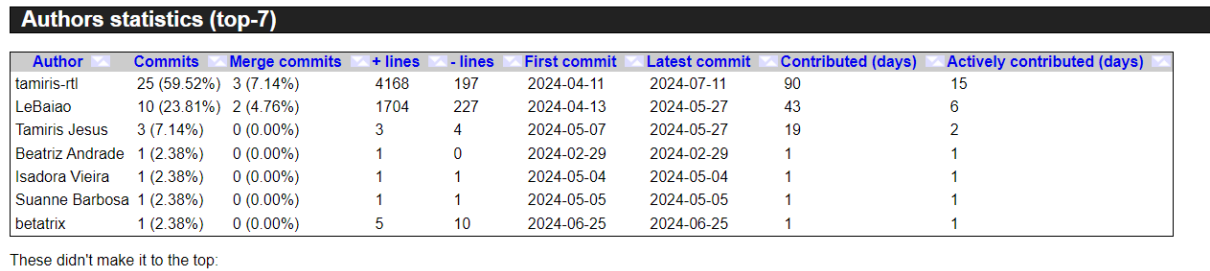


Figura 14 – Número de commits por autor

Na figura 15 é apresentado o gráfico referente ao número de linhas de código adicionadas por autor no repositório do projeto *back-end* do GitHub.

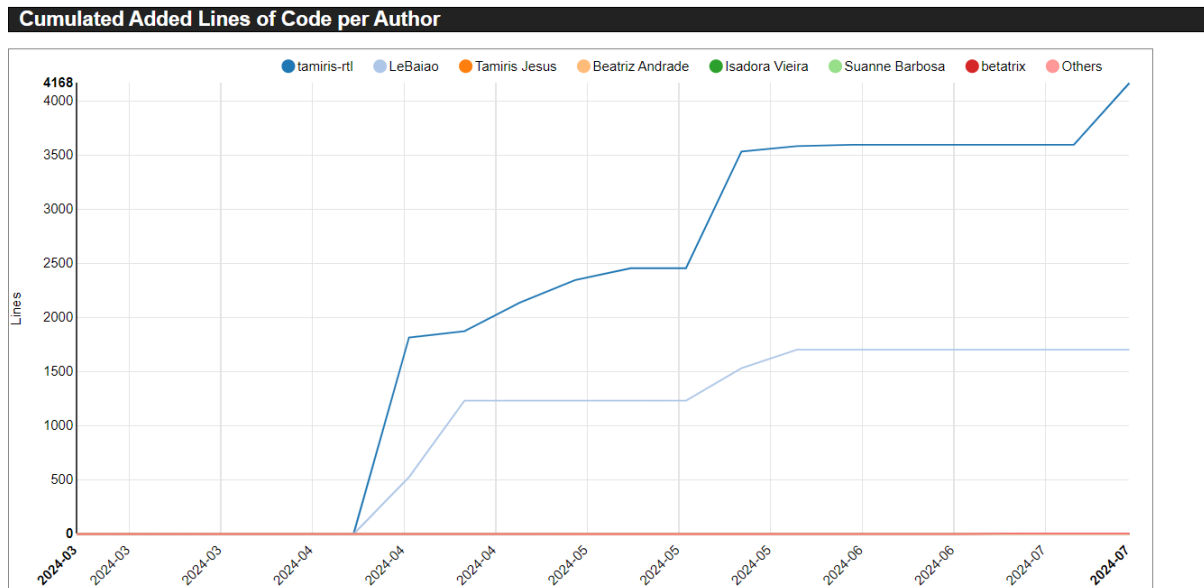


Figura 15 – Número de linhas de código por autor

Na figura 16 é apresentado o gráfico referente ao número de commits por autor no repositório do projeto *back-end* do GitHub.

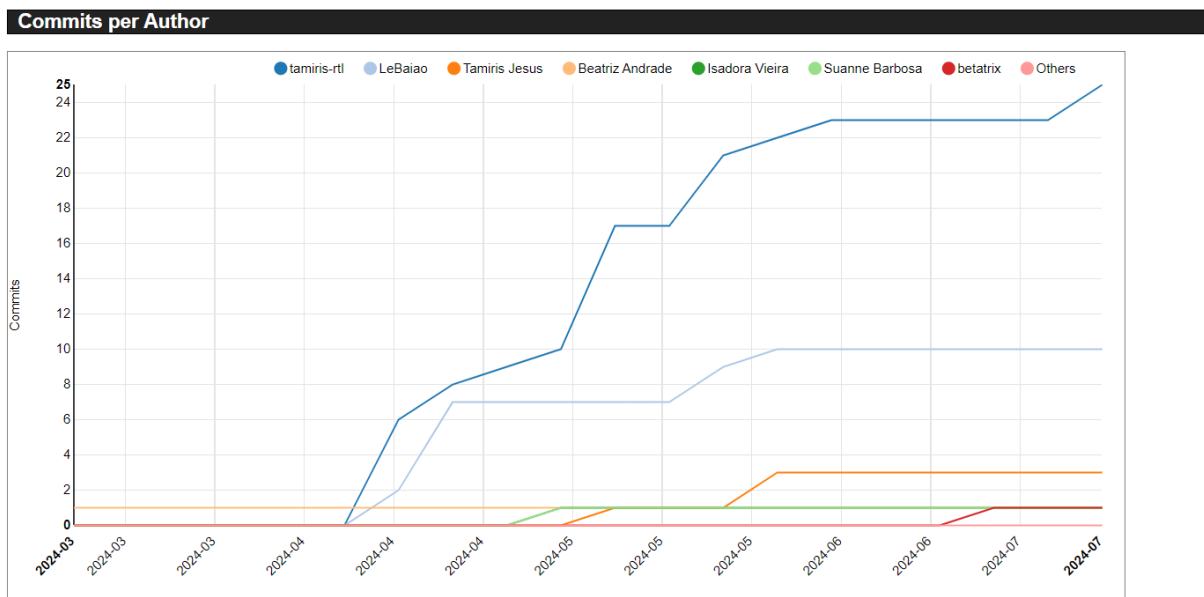


Figura 16 – Gráfico número de commits por autor

A figura 17 é um complemento do gráfico da figura 19, no qual é apresentado o gráfico referente ao número de commits por autor no repositório do projeto *back-end* do GitHub.

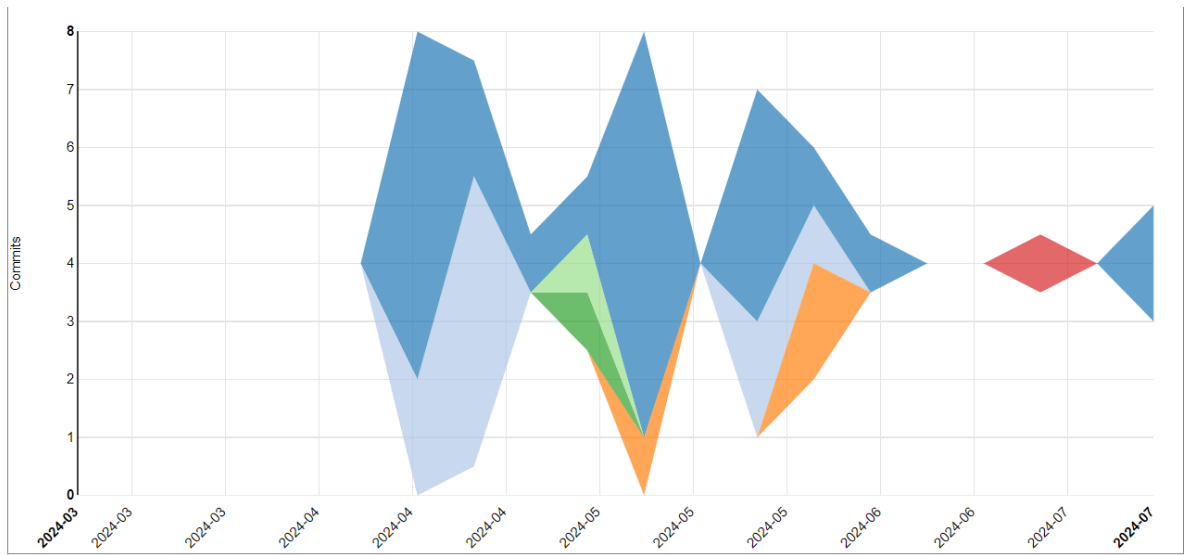


Figura 17 – Gráfico número de commits por autor

Na figura 18 é apresentado o ranking dos autores com o maior número de commits em cada mês do ano de 2024 no repositório do projeto *back-end* do GitHub.

Author of the Month				
Month	Author	Commits count	Next top 5	Total authors count
2024-07	tamiris-rtl	2 (of 2)		1
2024-06	tamiris-rtl	1 (of 2)	betatrix	2
2024-05	tamiris-rtl	13 (of 21)	Tamiris Jesus, LeBaiao, Suanne Barbosa, Isadora Vieira	5
2024-04	tamiris-rtl	9 (of 16)	LeBaiao	2
2024-02	Beatriz Andrade	1 (of 1)		1

Figura 18 – Tabela autor com o maior número de commits em cada mês do ano de 2024

Na figura 19 é apresentado o gráfico referente ao número de arquivos adicionados e número de linhas de código no repositório do projeto *back-end* do GitHub no decorrer do primeiro semestre de 2024.

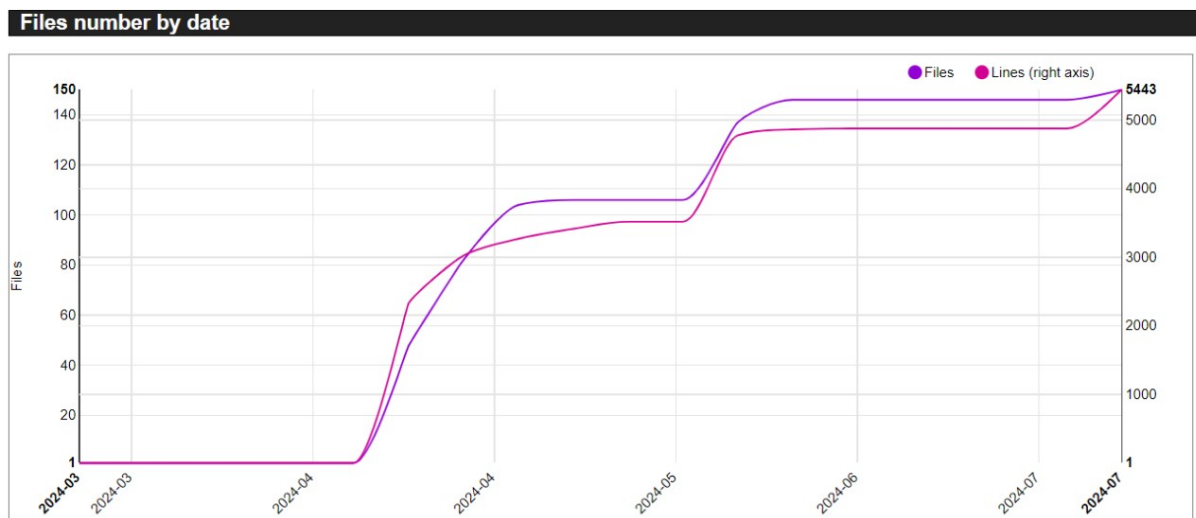


Figura 19 – Gráfico com a relação número de arquivos adicionados e linhas de código

Na figura 20 são apresentados os dados referentes ao tipo de arquivo presente no projeto *back-end*.

File types				
Extension	Count (%)	Byte Size (%)	Lines count (%)	Lines per file
java	142 (94.67%)	161515 (65.04%)	4748 (87.23%)	33.4
properties	2 (1.33%)	1502 (0.60%)	29 (0.53%)	14.5
.gitignore	1 (0.67%)	395 (0.16%)	33 (0.61%)	33.0
cmd	1 (0.67%)	7592 (3.06%)	205 (3.77%)	205.0
md	1 (0.67%)	19 (0.01%)	1 (0.02%)	1.0
mvnw	1 (0.67%)	11290 (4.55%)	308 (5.66%)	308.0
xml	1 (0.67%)	3475 (1.40%)	119 (2.19%)	119.0
jar	1 (0.67%)	62547 (25.19%)	binary	-

Figura 20 – Tipos de arquivos presentes no repositório do projeto *back-end* do GitHub



Na figura 21 são apresentados os dados gerais referente aos arquivos presentes no projeto *back-end*.

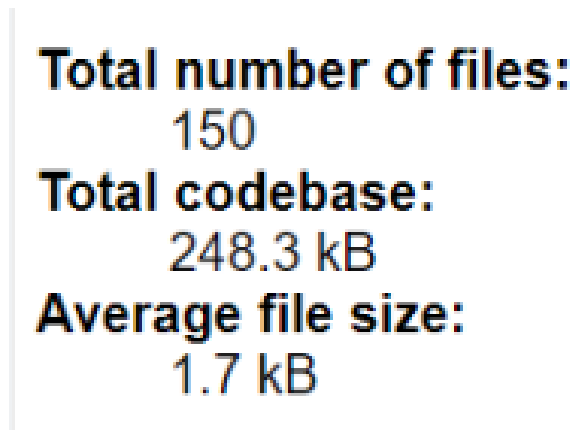


Figura 21 – Dados de arquivos presentes no repositório do projeto *back-end*

Na figura 22, são apresentados os dados gerais à respeito do repositório GitHub referente ao projeto *front-end*.

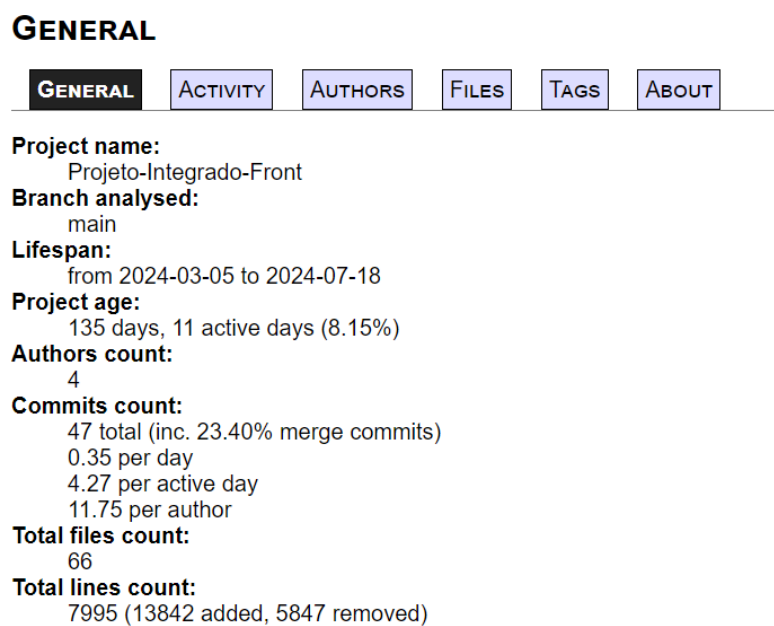


Figura 22 – Dados Gerais sobre o repositório *front-end* do GitHub

Na figura 23 são apresentados os dados referentes à relação entre o dia da semana, e o horário em que foram feitos os commits no repositório do projeto *front-end* do GitHub.

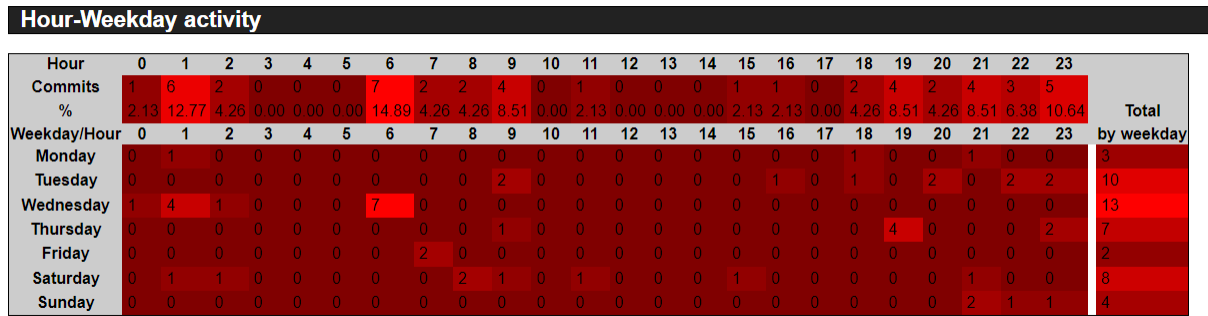


Figura 23 – Dados Gerais sobre o repositório *front-end* do GitHub

Na figura 24 são apresentados os dados referentes ao número de commits em cada mês do ano de 2024 referente ao repositório do projeto *front-end* do GitHub.

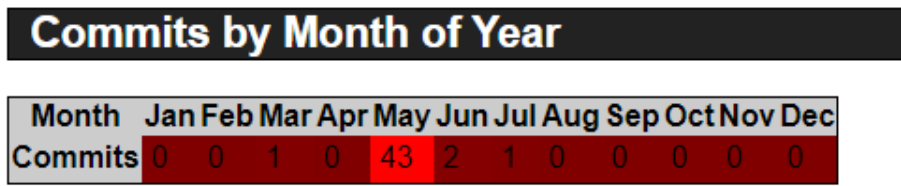


Figura 24 – Número de commits em cada mês do ano de 2024

Na figura 25 são apresentados os dados referentes ao número de commits por autor no repositório do projeto *front-end* do GitHub.

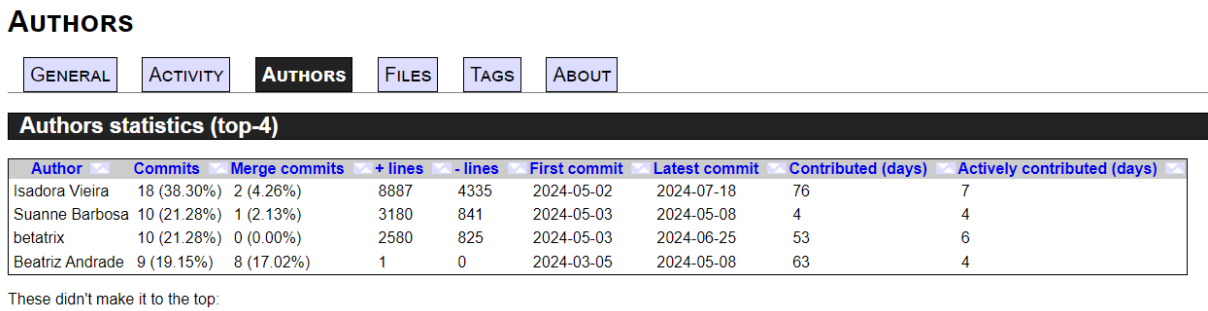


Figura 25 – Número de commits por autor

Na figura 26 é apresentado o gráfico referente ao número de linhas de código adicionadas por autor no repositório do projeto *front-end* do GitHub.

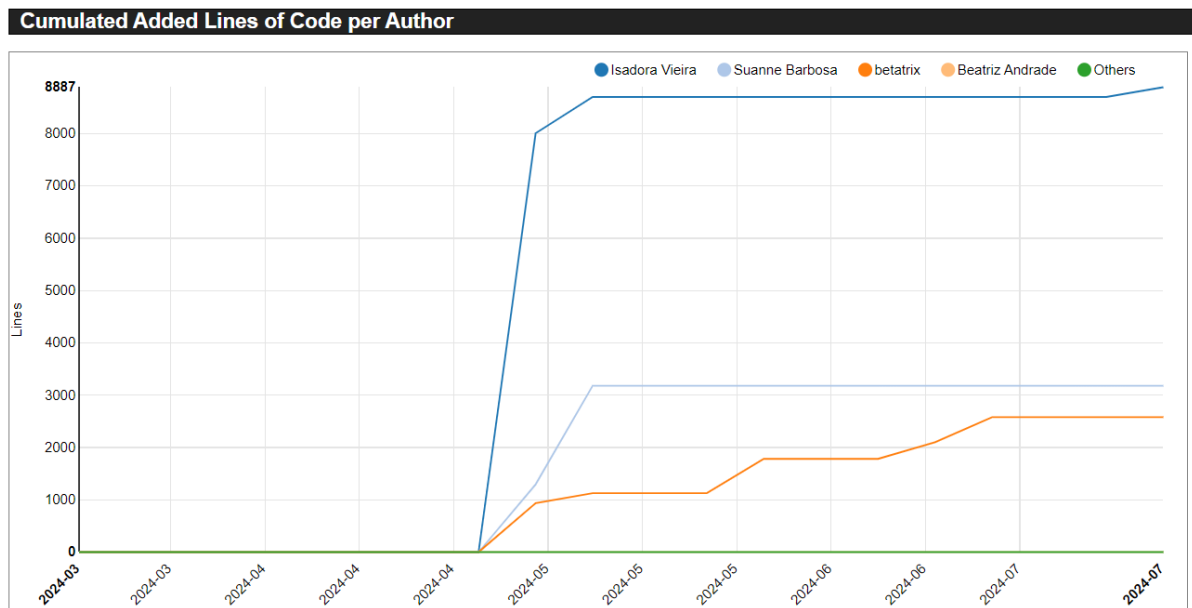


Figura 26 – Número de linhas de código por autor

Na figura 27 é apresentado o gráfico referente ao número de commits por autor no repositório do projeto *front-end* do GitHub.

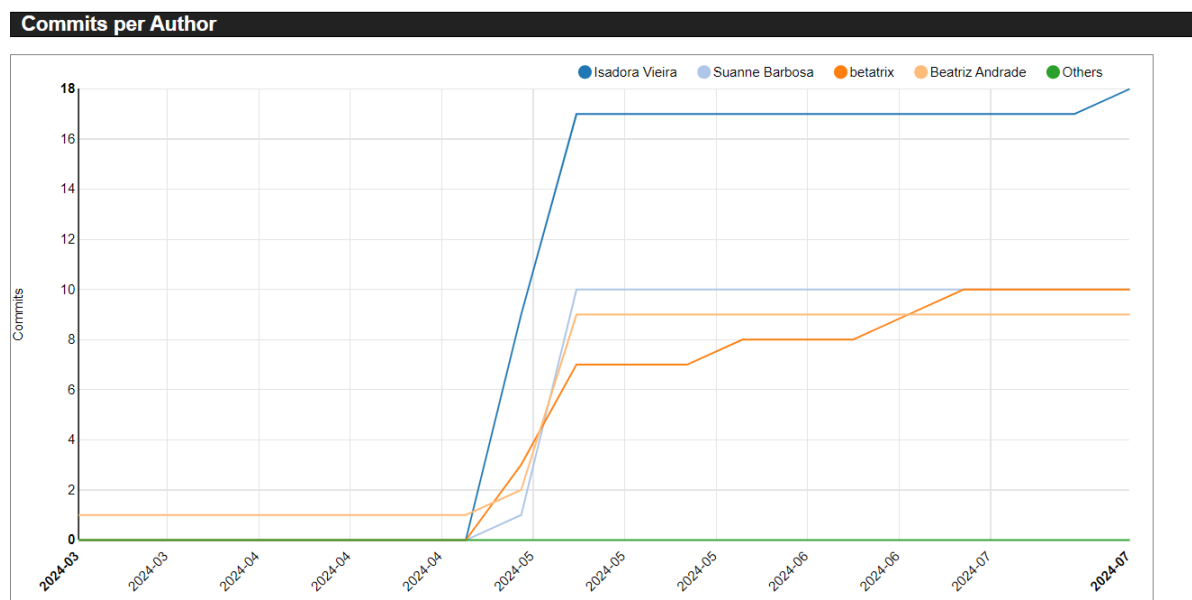


Figura 27 – Gráfico número de commits por autor

A figura 28 é um complemento do gráfico da figura 30, no qual é apresentado o gráfico referente ao número de commits por autor no repositório do projeto *front-end* do GitHub.

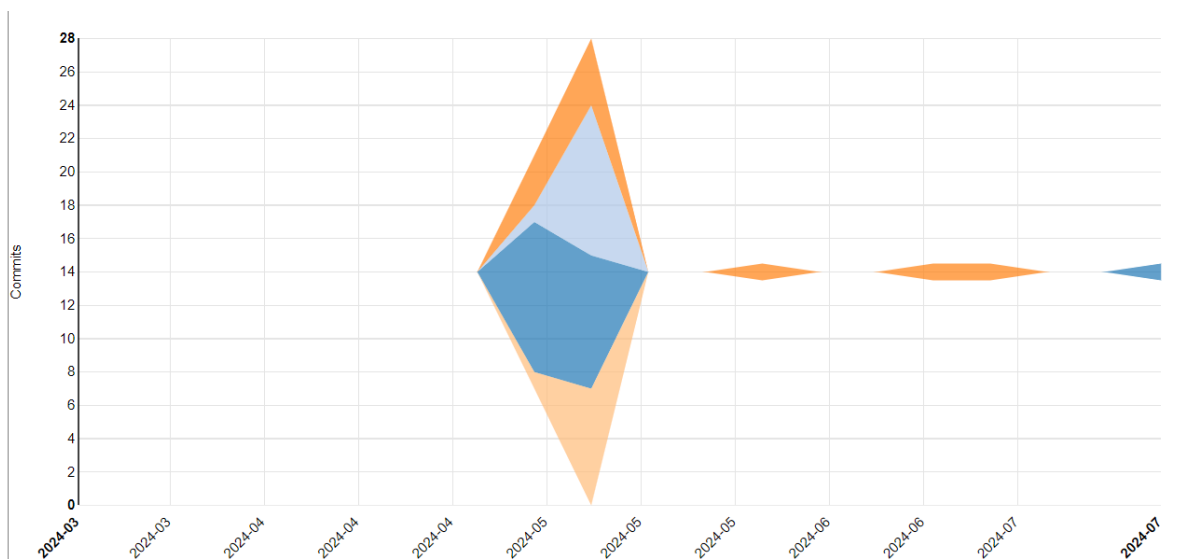


Figura 28 – Gráfico número de commits por autor

Na figura 29 é apresentado o ranking dos autores com o maior número de commits em cada mês do ano de 2024 no repositório do projeto *front-end* do GitHub.

Author of the Month				
Month	Author	Commits count	Next top 5	Total authors count
2024-07	Isadora Vieira	1 (of 1)		1
2024-06	betatrix	2 (of 2)		1
2024-05	Isadora Vieira	17 (of 43)	Suane Barbosa, betatrix, Beatriz Andrade	4
2024-03	Beatriz Andrade	1 (of 1)		1

Figura 29 – Tabela autor com o maior número de commits em cada mês do ano de 2024

Na figura 30 é apresentado o gráfico referente ao número de arquivos adicionados e número de linhas de código no repositório do projeto *front-end* do GitHub no decorrer do primeiro semestre de 2024.

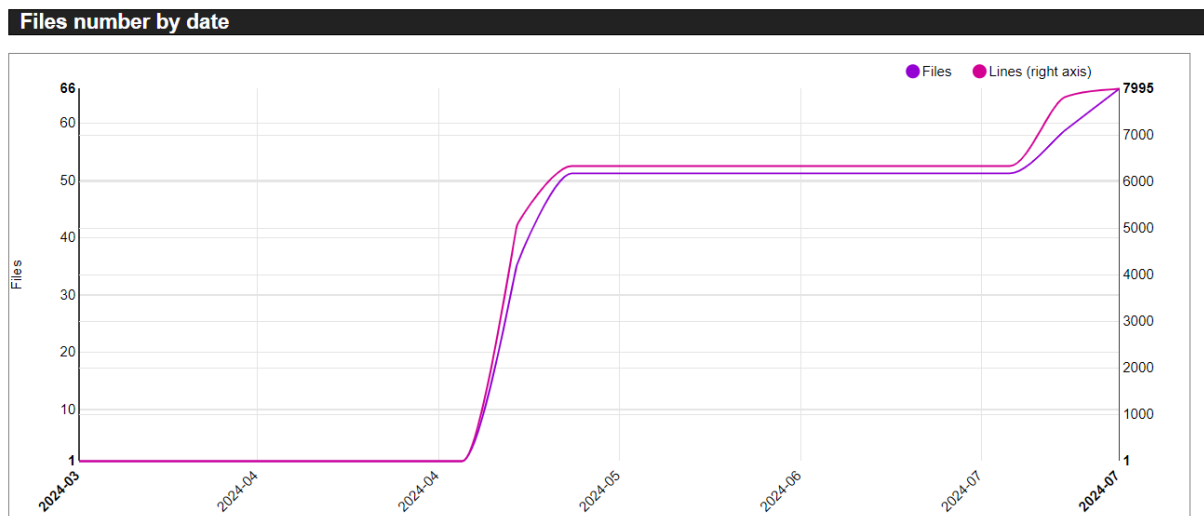


Figura 30 – Gráfico com a relação número de arquivos adicionados e linhas de código

Na figura 31 são apresentados os dados referentes ao tipo de arquivo presente no projeto *front-end*.

File types				
Extension	Count (%)	Byte Size (%)	Lines count (%)	Lines per file
ts	29 (43.94%)	22088 (5.08%)	996 (12.46%)	34.3
tsx	23 (34.85%)	97478 (22.44%)	2366 (29.59%)	102.9
json	4 (6.06%)	164681 (37.91%)	4519 (56.52%)	1129.8
svg	4 (6.06%)	147224 (33.89%)	4 (0.05%)	1.0
.gitignore	1 (1.52%)	257 (0.06%)	24 (0.30%)	24.0
cjs	1 (1.52%)	701 (0.16%)	25 (0.31%)	25.0
css	1 (1.52%)	320 (0.07%)	17 (0.21%)	17.0
example	1 (1.52%)	39 (0.01%)	1 (0.01%)	1.0
html	1 (1.52%)	293 (0.07%)	13 (0.16%)	13.0
md	1 (1.52%)	1300 (0.30%)	30 (0.38%)	30.0

Figura 31 – Tipos de arquivos presentes no repositório do projeto *front-end* do GitHub

Na figura 32 são apresentados os dados gerais referente aos arquivos presentes no projeto *front-end*.

**Total number of files:**  
66  
**Total codebase:**  
434.4 kB  
**Average file size:**  
6.6 kB

Figura 32 – Dados de arquivos presentes no repositório do projeto *front-end*

Com esses dados estatísticos à respeito dos repositórios do projeto Vocco, é possível obter uma visão ampla da evolução desenvolvimento e como a participação de cada integrante do grupo é essencial, auxiliando na construção do sistema como um todo.

## 6 DECISÕES

A seguir, justificaremos as escolhas feitas para a elaboração do projeto, incluindo as tecnologias utilizadas e as funcionalidades que optamos por não implementar ou que serão implementadas em versões futuras da aplicação.

### 6.1 Escolhas

Nesta seção, detalhamos as principais decisões técnicas tomadas durante a elaboração do projeto. Abordamos as razões por trás da seleção das tecnologias utilizadas, destacando os benefícios e vantagens que cada uma oferece para o desenvolvimento, manutenção e escalabilidade da aplicação.

#### 6.1.1 Utilização do Java

A escolha de utilizar Java como a linguagem de programação para o *back-end* foi baseada em sua robustez, segurança e ampla adoção no mercado. Java oferece uma vasta gama de bibliotecas e *frameworks* que facilitam o desenvolvimento de aplicações escaláveis e de alto desempenho. Além disso, sua portabilidade e forte suporte comunitário garantem uma base sólida para a manutenção e evolução contínua do projeto.

#### 6.1.2 Utilização do Spring Boot

O Spring Boot foi escolhido por sua capacidade de simplificar o desenvolvimento de aplicações Java baseadas em Spring. Ele oferece uma configuração mínima, permitindo que os desenvolvedores se concentrem nas funcionalidades da aplicação em vez de se preocupar com a infraestrutura. Além disso, o Spring Boot fornece uma série de ferramentas integradas para monitoramento, segurança e *deploy*, tornando o processo de desenvolvimento mais eficiente e ágil.

#### 6.1.3 Utilização do TypeScript

A adoção do TypeScript foi motivada pela necessidade de aumentar a produtividade e a qualidade do código no desenvolvimento *front-end*. TypeScript, ao adicionar tipagem estática ao JavaScript, ajuda a prevenir erros comuns durante a compilação, melhorando a robustez do código. Além disso, sua integração com ferramentas de desenvolvimento modernas e a capacidade de utilizar recursos mais avançados do JavaScript facilitam a criação de aplicações web complexas e escaláveis.

### 6.1.4 Utilização do React

O React foi escolhido como a biblioteca principal para a construção da interface do usuário devido à sua eficiência e flexibilidade. Com sua abordagem baseada em componentes, o React permite a criação de interfaces de usuário reutilizáveis e altamente interativas. Além disso, sua forte comunidade e vasta quantidade de recursos e bibliotecas auxiliares tornam o desenvolvimento mais rápido e eficaz, garantindo uma experiência de usuário final mais rica e responsiva.

## 6.2 Descartes

Durante a fase de elaboração do projeto, decidiu-se descartar a funcionalidade de recomendar cursos e instituições estrangeiras que aceitam a nota do [ENEM](#). A principal razão para essa decisão foi a complexidade adicional envolvida na integração e atualização constante das informações de instituições estrangeiras. Além disso, a maioria dos usuários da plataforma está interessada em oportunidades de educação dentro do Brasil, tornando essa funcionalidade de menor prioridade. Acreditamos que focar exclusivamente em cursos e universidades brasileiras permite uma experiência de usuário mais direcionada, alinhada com os principais objetivos do projeto.

## 6.3 Implementações futuras

Para o desenvolvimento contínuo da plataforma e futuras entregas, estão previstas várias melhorias e novas funcionalidades que irão aprimorar a experiência do usuário e expandir a extensão dos dados disponíveis. Entre essas futuras implementações, destacam-se:

### 6.3.1 Integração com API do e-MEC

Uma das principais funcionalidades a serem implementadas é a utilização de uma [API](#) para buscar dados diretamente no site do e-MEC. Essa integração permitirá que a plataforma acesse e atualize informações de universidades e cursos de forma automática, garantindo que os dados sejam sempre atuais e precisos. A [API](#) facilitará a obtenção de dados detalhados sobre as instituições de ensino superior, seus cursos oferecidos, localizações, avaliações, entre outros. A atualização dos dados seguirá a regra de negócio estabelecida de atualização semestral, alinhada com as datas de abertura dos processos seletivos das universidades, como o [Sistema de Seleção Unificada \(SISU\)](#) e o [Programa Universidade Para Todos \(PROUNI\)](#). Dessa forma, garantiremos que as informações disponibilizadas aos usuários estejam sempre atualizadas e em consonância com os períodos de ingresso nas instituições de ensino.



### 6.3.2 Inclusão de Dados de Cursos Técnicos e Outras Instituições

Atualmente, a plataforma foca principalmente em cursos de graduação. No entanto, em futuras implementações, pretendemos incluir dados de cursos técnicos e instituições públicas que os ofertem. Isso ampliará o escopo da plataforma, atendendo a uma gama mais ampla de usuários interessados em diferentes níveis e tipos de formação.

### 6.3.3 Funcionalidades de Busca Avançada

Outra melhoria significativa será a implementação de funcionalidades que permitam ao usuário realizar buscas mais rebuscadas e detalhadas nas bases de dados disponíveis. Os usuários poderão refinar suas pesquisas utilizando diversos filtros, como área de conhecimento, políticas públicas oferecidas, avaliação institucional, entre outros critérios específicos. Isso tornará a plataforma mais eficiente e personalizada, facilitando a busca por cursos e instituições que melhor atendam às necessidades e preferências individuais dos usuários.

## 7 TESTES

O presente capítulo tem como objetivo descrever quais foram os testes realizados durante o desenvolvimento do projeto Vocco a fim de garantir um controle de qualidade preciso da aplicação.

### 7.1 Plano de testes

Nesta seção, será detalhado o modo como serão realizados os testes do sistema. Assim garantindo que as funções principais estejam de acordo com os requisitos propostos.

#### 7.1.1 Teste de funcionalidade de Instituição

- **Cenário Ideal Cadastro de Instituição:** Descrição do cenário ideal de usabilidade do usuário no processo de cadastro de instituição.

Passos	Ações	Resultado Esperado	Resultado Efetivo	Observação	Situação do Fluxo (OK/NOK)
C#1.1.1	Acessar o botão de cadastrar uma IES	Será mostrada uma tela com os campos para preencher as informações necessárias			
C#1.1.2	Inserir nome, site, nota do Mec, e endereço	Habilita os campos para preenchimento			
C#1.1.3	Clicar em avançar	Redirecionar para a tela de associação de cursos			

Figura 33 – Cenário ideal cadastro de instituição

O processo ideal de cadastro de instituição percorre três ações para ser concluído.

- **Cenário de Exceção Cadastro de Instituição:** Descrição do cenário de exceções de usabilidade do usuário no processo de cadastro de instituição.

Passos	Ações	Resultado Esperado	Resultado Efetivo	Observação	Situação do Fluxo
C#1.2.1	Acessar o botão de cadastrar uma IES	Retorna uma tela com os campos para preencher as informações necessárias			
C#1.2.2	Deixar de informar alguma informação em um dos campos de preenchimento	Retornar nos campos um aviso informando que o preenchimento do campo é obrigatório para o cadastro.			
C#1.2.3	Informar uma sigla de uma IES já existente	Retorna uma mensagem de erro indicando que aquela IES já existe			

Figura 34 – Cenário de exceção de cadastro de instituição

O processo com exceções de cadastro de instituição percorre três ações para ser concluído.

- **Cenário Ideal Listagem de Instituição:** Descrição do cenário ideal de usabilidade do usuário no processo de listagem de instituição.

Passos	Ações	Resultado Esperado	Resultado Efetivo	Observação	Situação do Fluxo (OK/NOK)
C#2.1.1	Acessar a tela de gerenciamento de IES	Carrega a tela de gerenciamento com a listagem das instituições			
C#2.1.2	Retorna uma lista de todas as IES com as informações de id, nome, sigla, site e nota Mec	Retorna uma lista de todas as IES com as informações de id, nome, sigla, site e nota Mec			

Figura 35 – Cenário ideal de Listagem de instituição

O processo ideal de listagem de instituição percorre duas ações para ser concluído.

- **Cenário de Exceção Listagem de instituição:** Descrição do cenário de exceções de usabilidade do usuário no processo de listagem de instituição.

Passos	Ações	Resultado Esperado	Resultado Efetivo	Observação	Situação do Fluxo
C#2.2.1	Acessar a tela de gerenciamento de IES	Carrega a tela de gerenciamento com a listagem das instituições			
C#2.2.2	O site não retorna a listagem de todas as IES	O site exibe uma mensagem de erro clara e direta descrevendo o que pode ter ocasionado o erro			

Figura 36 – Cenário de exceção de listagem de instituição

O processo com exceções de listagem de instituição percorre duas ações para ser concluído.

- **Cenário Ideal Detalhamento de instituição:** Descrição do cenário ideal de usabilidade do usuário no processo de detalhamento de instituição.

Passos	Ações	Resultado Esperado	Resultado Efetivo	Observação	Situação do Fluxo (OK/NOK)
C#1.3.1	Entrar na tela de gerenciamento de IES	O site disponibilizar a listagem de todas as IES cadastradas			
C#1.3.2	Clicar no campo de alguma IES exibida	O site abre um modal com as informações de id, nome, sigla, site, nota Mec e endereço			

Figura 37 – Cenário ideal de detalhamento de instituição

O processo ideal de detalhamento de instituição percorre duas ações para ser concluído.

- **Cenário de Exceção Detalhamento de instituição:** Descrição do cenário de exceções de usabilidade do usuário no processo de detalhamento de instituição.

Passos	Ações	Resultado Esperado	Resultado Efetivo	Observação	Situação do Fluxo
C#2.3.1	Entrar na tela de detalhamento de IES	O site disponibilizar a listagem de todas as IES cadastradas			
C#2.3.2	Clicar no campo de alguma IES exibida	O site abre um modal com as informações de id, nome, sigla, site, nota Mec e endereço			
C#2.3.4	O site não exibe os detalhes da IES selecionada	O site retorna uma mensagem de erro indicando que a IES não está cadastrada			

Figura 38 – Cenário de exceção de detalhamento de instituição

O processo com exceções de detalhamento de instituição percorre três ações para ser concluído.

- **Cenário Ideal Associação de Instituição com Curso:** Descrição do cenário ideal de usabilidade do usuário no processo de associação de instituição com curso.

Passos	Ações	Resultado Esperado	Resultado Efetivo	Observação	Situação do Fluxo (OK/NOK)
C#1.4.1	Após cadastrar uma IES, acessar a tela de associação de curso à IES	A tela é carregada com uma lista de IES			
C31.4.2	Selecionar os cursos para associar com a IES previamente cadastrada e indicar a nota do Mec	A checkbox dos cursos são selecionadas e as notas Mec inseridas			
C31.4.3	Clicar em avançar	Após clicar em avançar, os cursos são associados à IES com sucesso			

Figura 39 – Cenário ideal de associação de instituição com curso

O processo ideal de associação de instituição com curso percorre três ações para ser concluído.

- **Cenário de Exceção Associação de Instituição com Curso:** Descrição do cenário de exceções de usabilidade do usuário no processo de associação de instituição com curso.

Passos	Ações	Resultado Esperado	Resultado Efetivo	Observação	Situação do Fluxo
C#2.4.1	Após cadastrar uma IES, acessar a tela de associação de curso à IES	A tela é carregada com uma lista de IES			
C#2.4.2	Selecionar os cursos para associar com a IES previamente cadastrada e indicar a nota do Mec	A checkbox dos cursos são selecionadas e as notas Mec inseridas			
C#2.4.3	Clicar em avançar	Os cursos não são associados à IES, e o site exibe uma mensagem de erro clara e direta descrevendo o que pode ter ocasionado o erro			

Figura 40 – Cenário de exceção associação de instituição com curso

O processo com exceções de associação de instituição com curso percorre três ações para ser concluído.



- **Cenário Ideal Exclusão de Instituição:** Descrição do cenário ideal de usabilidade do usuário no processo de exclusão de instituição.

Passos	Ações	Resultado Esperado	Resultado Efetivo	Observação	Situação do Fluxo (OK/NOK)
C#1.5.1	Entrar na tela de gerenciamento de IES	O site disponibilizar a listagem de todas as IES cadastradas			
C31.5.2	Retorna uma lista de todas as IES com um botão de exclusão no canto direito de cada nome de IES	Retorna uma lista de todas as IES com o botão de exclusão ao lado			
C31.5.3	Clicar no botão de exclusão de uma IES	A IES excluída ficará inativa			

Figura 41 – Cenário ideal de exclusão de instituição

O processo ideal de exclusão de instituição percorre três ações para ser concluído.

- **Cenário de Exceção Exclusão de Instituição:** Descrição do cenário de exceções de usabilidade do usuário no processo de exclusão de instituição.

Passos	Ações	Resultado Esperado	Resultado Efetivo	Observação	Situação do Fluxo
C#2.5.1	Entrar na tela de gerenciamento de IES	O site disponibilizar a listagem de todas as IES cadastradas			
C#2.5.2	Retorna uma lista de todas as IES com um botão de exclusão no canto direito de cada nome de IES	Retorna uma lista de todas as IES com o botão de exclusão ao lado			
C#2.5.3	Clicar no botão de exclusão da IES	A IES não ser inativada e o site exibe uma mensagem de erro clara e direta descrevendo o que pode ter ocasionado o erro			

Figura 42 – Cenário de exceção de exclusão de instituição

O processo com exceções de exclusão de instituição com curso percorre três ações para ser concluído.

A partir dos cenários de teste apresentados, torna-se viável simular um eventual caminho alternativo que o usuário final pode enfrentar durante a sua experiência navegando pelo site. Dessa forma, é possível realizar uma análise voltada à visão do usuário final, buscando suprir suas necessidades e melhorar a usabilidade do site.

## Referências

ALVARENGA, C. F.; SALES, A. P.; COSTA, A. D. da; COSTA, M. D. da; VERONEZE, R. B.; SANTOS, T. L. B. Desafios do ensino superior para estudantes de escola pública: um estudo na ufla. *Revista Pensamento Contemporâneo em Administração*, Universidade Federal Fluminense, v. 6, n. 1, p. 55–71, 2012. Citado na página 13.

ALVES, E. F. P. Tecnologia na educação: reflexão para uma prática docente technology in education: reflection for a teacher practice. *Brazilian Journal of Development*, v. 8, n. 1, p. 4227–4238, 2022. Citado na página 13.

ANDRADE, C. Y. de. Acesso ao ensino superior no brasil: equidade e desigualdade social. *Revista Ensino Superior Unicamp*, v. 6, p. 18–27, 2012. Citado 3 vezes nas páginas 20 e 21.

HOLLAND, J. L. *Making Vocational Choices: A Theory of Vocational Personalities and Work Environments*. 3rd. ed. Odessa: Psychological Assessment Resources, 1997. Citado na página 19.

LOPES, T. D. S. et al. Fim do ensino médio e os desafios para a escolha profissional. In: *Anais VIII CONEDU*. Campina Grande: Realize Editora, 2022. Acesso em: 05/04/2024. Disponível em: <<https://editorarealize.com.br/artigo/visualizar/88466>>. Citado na página 19.

MAGALHÃES, M.; MARTINUZZI, V.; TEIXEIRA, M. A. P. Relação entre estilos cognitivos e interesses vocacionais. *Revista Brasileira de Orientação Profissional*, v. 5, n. 2, p. 11–20, 2004. Citado na página 19.

MAGALHÃES, M. de O. Relação entre personalidades vocacionais e estilos interpessoais. *Revista Brasileira de Orientação Profissional*, Associação Brasileira de Orientação Profissional, v. 7, n. 1, p. 11–22, 2006. Citado na página 19.

MARTINS, C. R. *Psicologia do comportamento vocacional*. São Paulo: EPU, 1978. Citado na página 19.

MATOS, M. d. S.; PIMENTA, S. G.; ALMEIDA, M. I. d.; OLIVEIRA, M. A. d. C. O impacto do programa de inclusão social da universidade de são paulo no acesso de estudantes de escola pública ao ensino superior público gratuito. *Revista Brasileira de Estudos Pedagógicos*, INEP, v. 93, n. 235, p. 720–742, 2012. Citado na página 13.

## Apêndices

# APÊNDICE A – PROTOTIPAÇÃO

## A.1 Tela Inicial do administrador

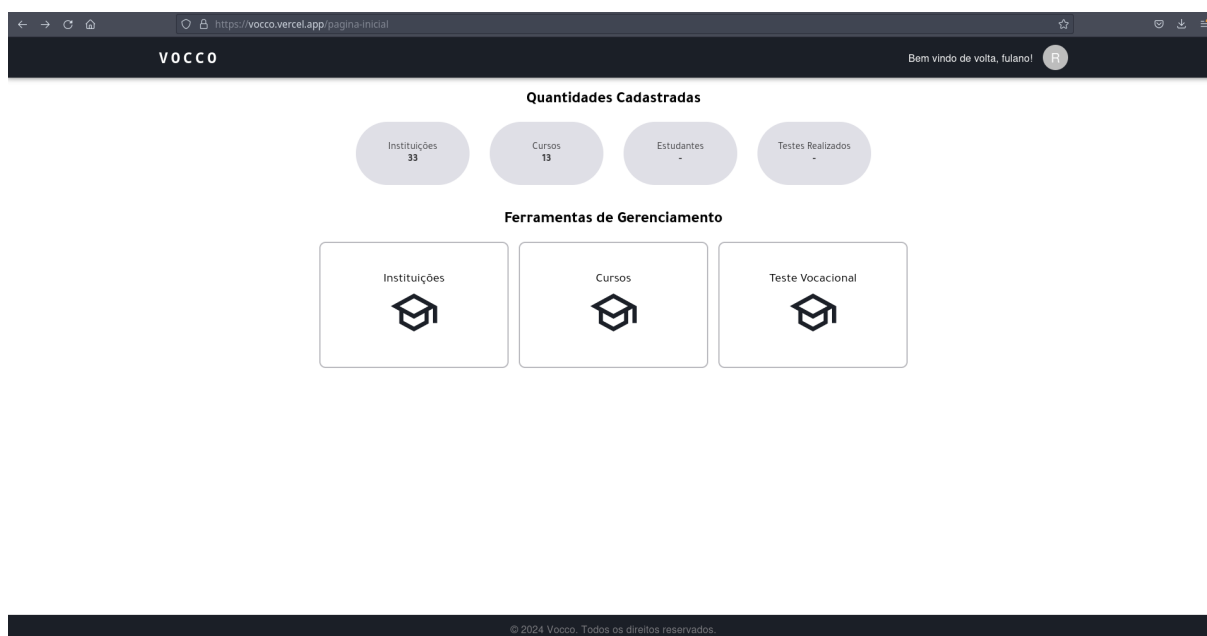


Figura 43 – Tela inicial do administrador

A.2 Tela de cadastro de dados gerais da instituição

VOCCO

Bem vindo de volta, fulano!

Cadastrar Dados da Instituição

Adicionar Cursos na Instituição

Adicionar Políticas Afirmativas na Instituição

Dados Gerais

Nome da Instituição \*

Site \*

Sigla \*

Nota MEC \*

Endereço

Logradouro \*

CEP \*

Número \*

Complemento

Bairro \*

Cidade \*

Estado \*

VOLTAR

AVANÇAR

© 2024 Vocco. Todos os direitos reservados.

Figura 44 – Cadastro de dados gerais da instituição

A.3 Tela de gerenciamento de instituições

VOCCO



















Bem vindo de volta, fulano!

Gerenciamento de Instituições

Pesquisar instituição

CADASTRAR INSTITUIÇÃO

EXCLUIR INSTITUIÇÕES

	ID	Nome	Ativo	
<input type="checkbox"/>	3	Instituto Federal de São Paulo	Inativo	 
<input type="checkbox"/>	4	UNIVERSIDADE FEDERAL DE ALAGOAS	Inativo	 
<input type="checkbox"/>	5	UNIVERSIDADE FEDERAL DE ALFENAS	Inativo	 
<input type="checkbox"/>	6	UNIVERSIDADE DE BRASÍLIA	Inativo	 
<input type="checkbox"/>	7	FUNDAÇÃO UNIVERSIDADE FEDERAL DA GRANDE DOURADOS	Ativo	 
<input type="checkbox"/>	8	UNIVERSIDADE FEDERAL DE GOIÁS	Inativo	 
<input type="checkbox"/>	9	UNIVERSIDADE FEDERAL DE MATO GROSSO	Inativo	 
<input type="checkbox"/>	10	UNIVERSIDADE FEDERAL DE MATO GROSSO DO SUL	Inativo	 
<input type="checkbox"/>	11	UNIVERSIDADE FEDERAL DE CATALÃO	Ativo	 

© 2024 Vocco. Todos os direitos reservados.

Figura 45 – Gerenciamento de instituições

A.4 Tela de detalhamento de uma instituição

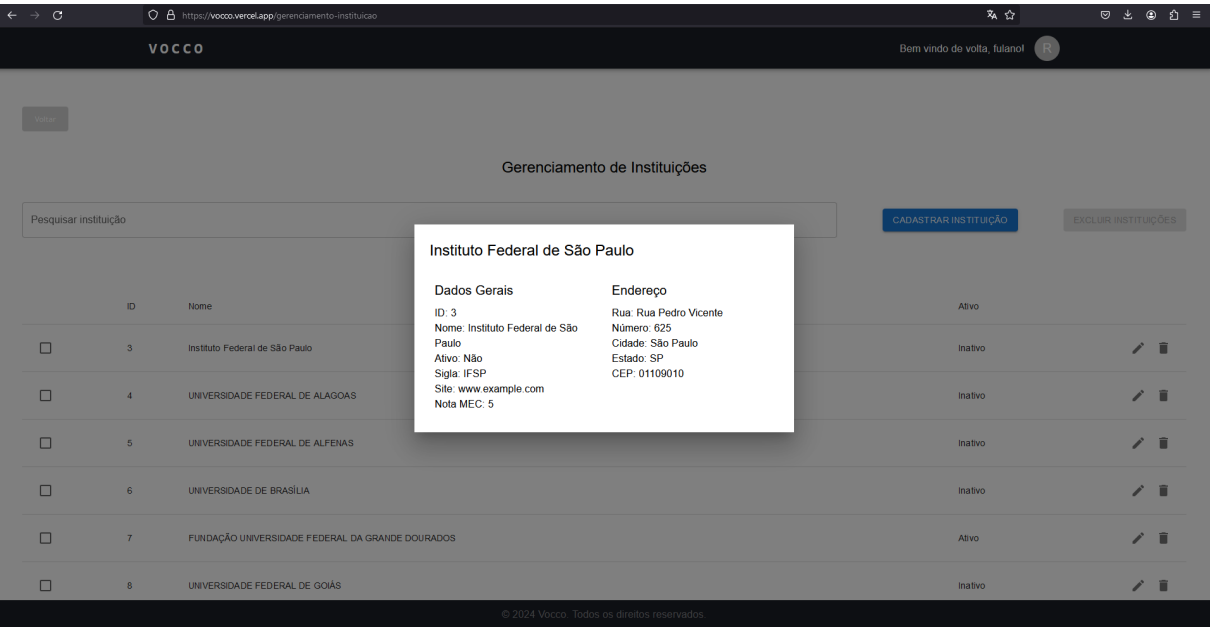


Figura 46 – Cadastro de dados gerais da instituição

A.5 Tela de edição de uma instituição

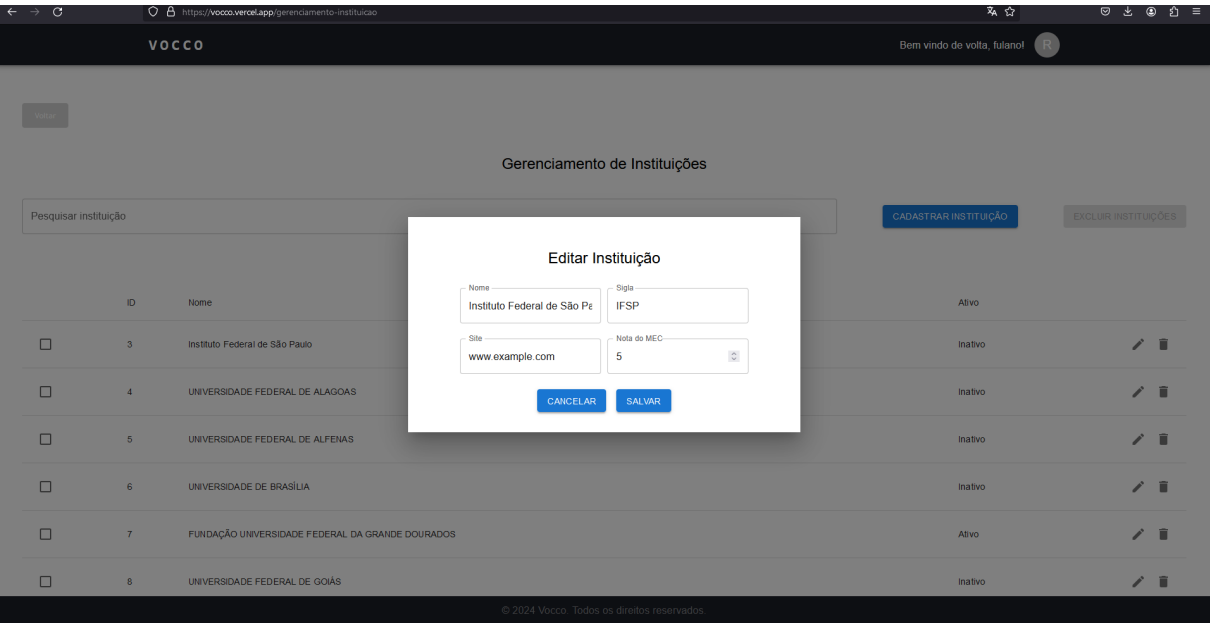


Figura 47 – Cadastro de dados gerais da instituição

A.6 Tela de exclusão de uma instituição

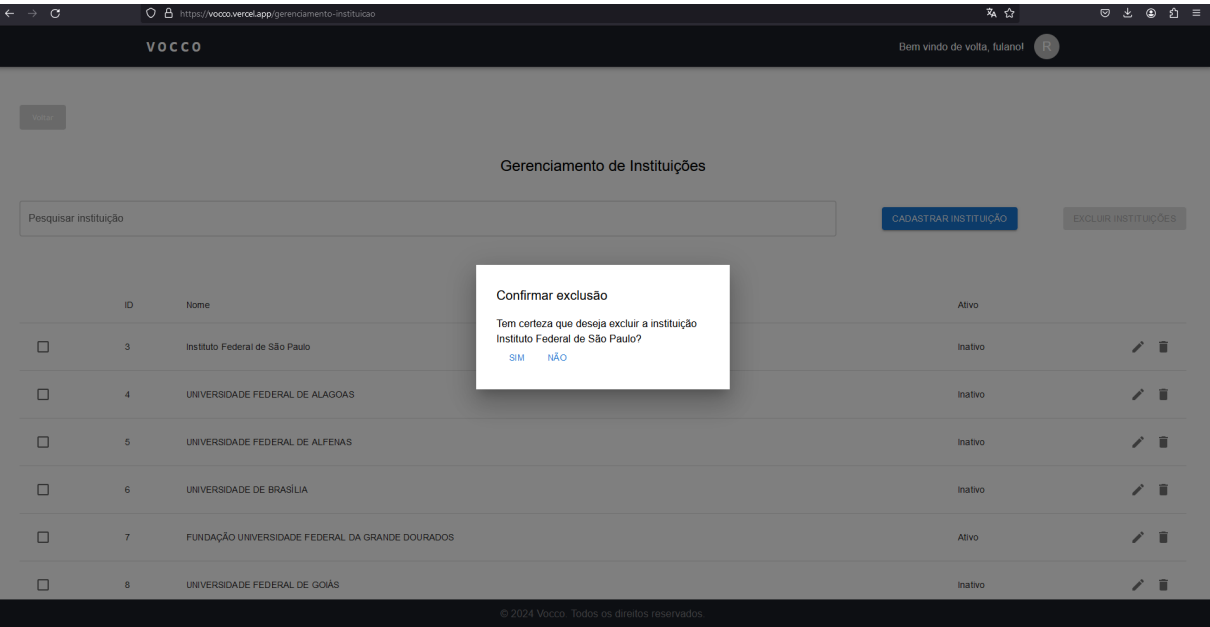


Figura 48 – Cadastro de dados gerais da instituição

A.7 Tela de associação de instituições com cursos

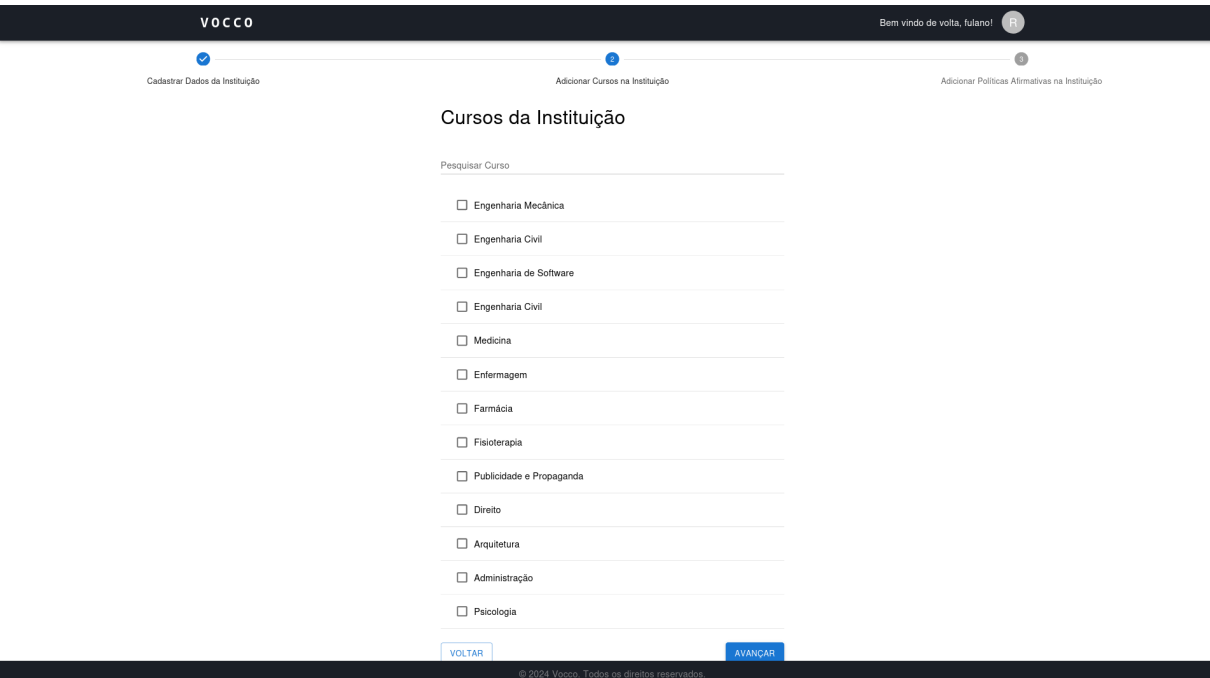


Figura 49 – Inserção dos cursos na instituição



## A.8 Tela de associação de instituições com políticas

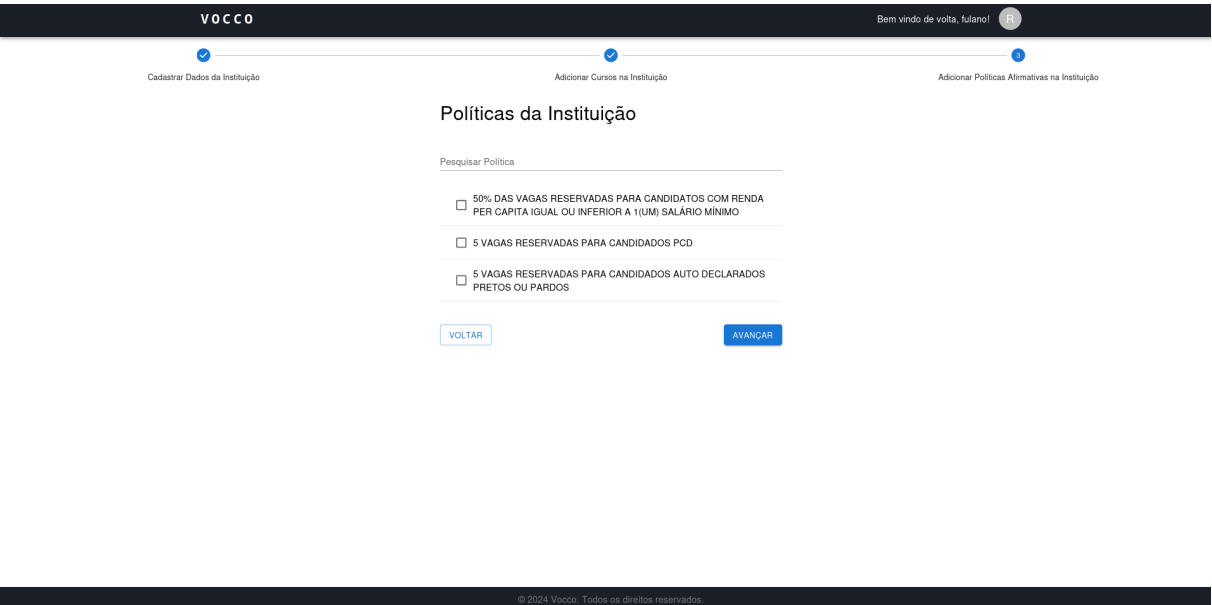


Figura 50 – Inserção das políticas na instituição

## APÊNDICE B – Cronograma

Para a organização e planejamento do desenvolvimento do projeto, dividimos as atribuições entre os integrantes do grupo. Dessa forma, conseguimos definir as datas de entrega e segui-las com mais facilidade. Com isso, o cronograma manteve um fluxo mais dinâmico e organizado, atendendo aos prazos estabelecidos para a entrega de cada fase do projeto.

### Desenho da Aplicação

Primeira fase de entrega, na qual se iniciou o desenvolvimento do projeto.

- **06 de março de 2024:** Primeira discussão com o orientador do projeto, onde foram apresentados possíveis temas a serem escolhidos;
- **08 de março de 2024:** Criação do blog e do canal no YouTube;
- **13 de março de 2024:** Início da escrita e pesquisas da documentação do desenho da aplicação;
- **27 de março de 2024:** Construção da apresentação do desenho da aplicação;
- **03 de abril de 2024:** Finalização da documentação e entrega;
- **10 de abril de 2024:** Apresentação do desenho da aplicação;

### Prova de Conceito

A segunda fase de entrega a qual foi demonstrado a capacidade das integrantes de utilizar as ferramentas definidas para a construção da aplicação final [Produto Mínimo Viável \(MVP\)](#).

- **17 de abril de 2024:** Início do desenvolvimento da aplicação básica;
- **18 de abril de 2024:** Início da construção da documentação dessa fase;
- **24 de abril de 2024:** Apresentação da prova de conceito;

### Apresentação Final do Projeto

A última fase do projeto no qual ocorrerá a apresentação do projeto final, incluindo o documento final e o [MVP](#).

- **25 de abril de 2024:** Início do desenvolvimento do MVP, após a apresentação da prova de conceito. Houveram várias reuniões de definição das principais funcionalidades e alinhamento do desenvolvimento;
- **08 de maio de 2024:** Início da construção da documentação final;
- **15 de maio de 2024** A partir desta data todos os esforços foram para finalização das tarefas do projeto, documentação e postagens do blog;

O desenvolvimento do projeto e a documentação são extensos e demandam reuniões constantes para verificar o progresso e ajustar o que ainda precisa ser feito. Dessa forma, as reuniões de planejamento e verificação continuam acontecendo para garantir que a entrega final do projeto seja realizada dentro do prazo.

## APÊNDICE C – Atas de Reuniões

### Reunião 10/03

#### Participantes

Beatriz Andrade, Isadora Câmara, Letícia Baião, Suanne Almeida, Tamiris Jesus.

#### Pauta

- Estudo de viabilidade do projeto, considerando o fluxo de atividades necessárias para a sua realização.
- Desenvolvimento e organização do documento de introdução.
- Divisão dos temas a serem abordados na apresentação da próxima semana.

#### Observações

Primeira reunião realizada, marcando o início oficial do projeto. Decisão de cada membro gravar sua parte da apresentação para posterior edição em um vídeo a ser publicado no canal da equipe no YouTube.

#### Ações Necessárias

- Definição da proposta da plataforma e seus objetivos.
- Análise das propostas para o desenvolvimento do cálculo de compatibilidade a ser utilizado no teste vocacional.
- Estudo de ideias para potenciais fontes de financiamento e oportunidades de parcerias.
- Transferência do documento de introdução para o Overleaf.
- Início da elaboração dos slides para a apresentação.

#### Pauta da Próxima Semana

Para a próxima semana, ficou definido que iremos unir as partes gravadas por cada integrante para o vídeo.

## Reunião 13/03

### Participantes

Beatriz Andrade, Isadora Câmara, Letícia Baião, Tamiris Jesus, Suanne Almeida.

### Pauta

- Acompanhamento da última reunião: Os vídeos de cada integrante foram agrupados em um único vídeo, o qual já foi publicado no canal do YouTube da equipe, conforme o proposto na reunião anterior.
- Novos assuntos:
  - Discussão sobre o algoritmo responsável pelo cálculo de compatibilidade a ser utilizado no teste vocacional.
  - Definição de cargos para cada membro da equipe.
  - Análise de estratégias para o controle de versão.
  - Análise de estratégias para a documentação.

### Observações

Não foi realizada uma reunião para consolidar as partes dos vídeos gravados por cada integrante em um único vídeo. Cada membro da equipe assumiu a responsabilidade de realizar pesquisas referentes à proposta do projeto e analisar o funcionamento dos testes vocacionais em plataformas concorrentes.

### Ações Necessárias

- Avaliação de perspectivas sobre o algoritmo a ser usado no cálculo de compatibilidade, juntamente com as suposições relacionadas às funcionalidades e ao fluxo.
- Análise das habilidades/afinidades individuais de cada integrante para determinar a área em que melhor se encaixa no projeto.
- Avaliação e decisão sobre qual método de ramificação utilizar para o controle de versão.
- Definição das ferramentas de documentação a serem utilizadas.

## Pauta da Próxima Semana

- Apresentar os resultados e sugestões de aprimoramento dos testes vocacionais a serem analisados.
- *Brainstorming* de funcionalidades para o sistema.
- Elaborar uma proposta de projeto para concorrer ao Prêmio Luz na Educação.

## Reunião 20/03

### Participantes

Beatriz Andrade, Isadora Câmara, Letícia Baião, Tamiris Jesus, Suanne Almeida.

### Pauta

- Acompanhamento da última reunião: Cada membro da equipe cumpriu sua responsabilidade ao realizar pesquisas sobre a proposta do projeto e analisar o funcionamento dos testes vocacionais em plataformas concorrentes.
- Novos assuntos:
  - Discussão sobre como funciona o [subversion](#) e como subiremos nossos arquivos, incluindo o compartilhamento do link do YouTube, a apresentação de slides e o documento de introdução.
  - Análise dos testes vocacionais realizados em plataformas concorrentes.

### Observações

O brainstorming de funcionalidades para o sistema e a elaboração de uma proposta de projeto para concorrer ao Prêmio Luz na Educação não foram realizados.

### Ações Necessárias

- Cada integrante compartilhou suas experiências com os testes vocacionais, discutindo o que mais gostaram, o que menos gostaram, as vantagens e as desvantagens.
- Análise do que de positivo podemos extrair dos testes vocacionais estudados para aplicar em nosso projeto.
- Pré-seleção dos requisitos para os testes vocacionais.

## Pauta da Próxima Semana

- Definição de requisitos, incluindo requisitos funcionais, não funcionais e regras de negócio.
- Brainstorming de funcionalidades para o sistema.
- Elaborar uma proposta de projeto para concorrer ao Prêmio Luz na Educação.

## Reunião 27/03

### Participantes

Beatriz Andrade, Isadora Câmara, Tamiris Jesus, Suanne Almeida.

### Pauta

- Novos assuntos:
  - Discussão sobre as funcionalidades do sistema e as entidades que serão implementadas.
  - Discussão sobre as vantagens das tecnologias que serão utilizadas no frontend.
  - Início da definição de requisitos, abrangendo requisitos funcionais, não funcionais e regras de negócio.

### Observações

A definição dos requisitos do sistema não foi concluída e a elaboração de uma proposta de projeto para concorrer ao Prêmio Luz na Educação não foi realizada. Foi sugerido que assistíssemos às aulas recomendadas por uma integrante do grupo sobre o uso do TypeScript em uma aplicação.

### Ações Necessárias

- Discutimos as funcionalidades que nosso sistema terá, avaliando quais são essenciais para a fase atual e quais pretendemos implementar posteriormente.
- No contexto das funcionalidades, analisamos quais entidades seriam necessárias em nosso sistema.
- Discutimos as vantagens do React e do TypeScript, bem como as otimizações que essas tecnologias trariam para o projeto.

- Elaboração de um documento que contém os requisitos, abrangendo tanto os requisitos funcionais quanto os não funcionais, além das regras de negócio.

## Pauta da Próxima Semana

- Continuação da definição de requisitos, incluindo requisitos funcionais, não funcionais e regras de negócio.
- Início do esboço do desenho da arquitetura da aplicação.
- Elaborar uma proposta de projeto para concorrer ao Prêmio Luz na Educação.

## Reunião 28/03

### Participantes

Beatriz Andrade, Isadora Câmara, Letícia Baião, Tamiris Jesus, Suanne Almeida.

### Pauta

- Novos assuntos:
  - Atualização para todos os membros sobre as entidades do sistema e esclarecimento de dúvidas/inseguranças referentes à implantação do sistema na Amazon Web Service.
  - Definição de todos os elementos que precisam ser incluídos no desenho da aplicação.
  - Elaboração da proposta e inscrição para o Prêmio Luz na Educação.

### Observações

Não foi proposta uma alternativa definitiva para substituir a implantação do sistema na **Amazon Web Service**.

### Ações Necessárias

- Discutimos os prós e contras de implantar a plataforma na **Amazon Web Service**.
- Dividimos as partes que devem ser escritas para elaborar a documentação do desenho da aplicação entre as integrantes.



- Lemos o rascunho da proposta elaborado pela integrante Tamiris e realizamos os ajustes e melhorias necessários.
- Lemos a documentação do Prêmio Luz na Educação e seguimos todas as etapas necessárias para a inscrição.

## Pauta da Próxima Semana

- Cada integrante será responsável por realizar a sua parte na documentação dos desenhos da aplicação.
- Ficou definido que as integrantes irão estudar e avaliar outras propostas para substituir a **Amazon Web Service** como plataforma de infraestrutura.

## Reunião 03/04

### Participantes

Beatriz Andrade, Isadora Câmara, Letícia Baião, Tamiris Jesus, Suanne Almeida.

### Pauta

- Novos assuntos:
  - Escolha de publicar toda a documentação no GitHub.
  - Divisão das responsabilidades de criação dos slides entre as integrantes.
  - Ajustes de configuração no **Overleaf**.

### Observações

Após análises e estudos, as integrantes ficaram em dúvida entre duas plataformas para a implantação do sistema: **Heroku** e **Railway**.

### Ações Necessárias

- Discutimos a divisão das responsabilidades na criação dos slides entre as integrantes.
- Foi feita a confirmação que a configuração do **Overleaf** que devemos utilizar é aquela enviada pelo professor.
- Nós definimos e validamos todos os requisitos do projeto.

- Decidimos que, a partir de agora, vamos incluir os áudios dos vídeos de apresentação diretamente nos slides do Canva.

## Pauta da Próxima Semana

- As integrantes assumiram o compromisso de cada uma realizar sua parte nos slides, incluindo a gravação dos áudios a serem incorporados no Canva.
- As integrantes realizarão um estudo comparativo entre as plataformas Heroku e Railway para decidir qual delas deve ser utilizada para a infraestrutura.

## Reunião 10/04

### Participantes

Beatriz Andrade, Isadora Câmara, Letícia Baião, Tamiris Jesus, Suanne Almeida.

### Pauta

- Novos assuntos:
  - Análise da Prova de Conceito.

### Observações

Reunião apenas para discutir a Prova de Conceito que deverá ser entregue.

### Ações Necessárias

- Realizamos uma reunião em que discutimos minuciosamente e dividimos todas as etapas necessárias para a conclusão bem-sucedida da Prova de Conceito.

## Pauta da Próxima Semana

- Ficou decidido que as integrantes começarão a trabalhar na documentação e se reunirão posteriormente para planejar o desenvolvimento do sistema.

## Reunião 24/04

### Participantes

Beatriz Andrade, Isadora Câmara, Letícia Baião, Tamiris Jesus, Suanne Almeida.

### Pauta

- Novos assuntos:
  - Escolha do modelo de referência para o código de personalidade que servirá de base para o teste vocacional.
  - Escolha da ferramenta de construção para o *frontend*.

### Observações

A reunião também serviu para alinhar todas as integrantes quanto aos estudos em andamento, tanto no desenvolvimento *backend* quanto no *frontend*.

### Ações Necessárias

- Realizamos uma pesquisa para identificar modelos de referência existentes para testes vocacionais ou avaliações de personalidade.
- Optamos por utilizar o modelo de código de personalidade de John Holland.
- Debates sobre as ferramentas de construção para o *frontend* e optamos por utilizar o Vite.

### Pauta da Próxima Semana

- Ficou decidido que as integrantes irão se reunir para elaborar como devem prosseguir no fluxo do desenvolvimento das telas.

## Reunião 28/04

### Participantes

Beatriz Andrade, Isadora Câmara, Letícia Baião, Tamiris Jesus, Suanne Almeida.

## Pauta

- Novos assuntos:
  - Desenvolvimento do *design* das telas que serão implementadas no fluxo do perfil do administrador.

## Observações

Como esses eram protótipos de baixa fidelidade, os desenhos das telas foram feitos no Canva.

## Ações Necessárias

- As integrantes que estavam responsáveis pelo *backend* orientaram as colegas encarregadas do *frontend* no desenvolvimento das telas.
- Optamos por utilizar o modelo de código de personalidade de John Holland.
- Analisamos completamente os requisitos específicos do perfil do administrador e os fluxos que as telas devem suportar.
- Criamos *wireframes* detalhados das telas.

## Pauta da Próxima Semana

- Ficou decidido que as integrantes se reunirão para alinhar o desenvolvimento das telas.

## Reunião 01/05

### Participantes

Beatriz Andrade, Isadora Câmara, Letícia Baião, Tamiris Jesus, Suanne Almeida.

## Pauta

- Novos assuntos:
  - Discussão sobre o desenvolvimento das telas conforme os *wireframes*.
  - Atualização do progresso individual de cada integrante no projeto.

## Observações

As telas estão sendo desenvolvidas de acordo com os protótipos de baixa fidelidade criados anteriormente. Cada integrante compartilhou seu progresso e contribuições até o momento.

## Ações Necessárias

- Implementação das alterações nos *wireframes* conforme discutido.
- Continuação do desenvolvimento das telas com base nos requisitos estabelecidos.

## Reunião 08/05

### Participantes

Beatriz Andrade, Isadora Câmara, Letícia Baião, Tamiris Jesus, Suanne Almeida.

### Pauta

- Novos assuntos:
  - Discussão sobre os *feedbacks* recebidos durante a apresentação.
  - Planejamento dos próximos passos para a próxima entrega.

## Observações

Como esses eram protótipos de baixa fidelidade, os desenhos das telas foram feitos no Canva.

## Ações Necessárias

- O projeto foi apresentado e bem recebido pelo professor orientador.
- Foram discutidas melhorias sugeridas pelos *feedbacks* recebidos.

### Pauta da Próxima Semana

- Ficou decidido que as integrantes se reunirão para alinhar o desenvolvimento da próxima entrega.

## APÊNDICE D – Blog

### Primeira Publicação do Blog

9 de março de 2024

#### 1º Semana – O início da nossa jornada.

Olá Ada's e Alan's ✨ ,

Nossa primeira reunião ocorreu hoje e nela decidimos levantar algumas pautas importantes para o início do projeto:

- Decidimos que o nome da nossa plataforma seria **Vocco**, ela faz referência a palavra em latim *voco* (vocação);
- Fizemos um estudo de viabilidade, analisando o que seria possível entregar neste primeiro semestre e que caminho iremos tomar para dar o início ao desenvolvimento;
- Iniciamos nossa documentação com o LaTeX, dando uma visão geral do projeto;
- Dividimos os temas que iremos abordar na apresentação da proposta nesta próxima quarta-feira.

Por estarmos com um tempo bem apertado neste início, decidimos gravar o vídeo pré-apresentação separadamente (vamos nos organizar para os próximos serem feitos por chamada via meet).

É isso, até semana que vem!

Figura 51 – Primeira Publicação do Blog

## Segunda Publicação do Blog

13 de março de 2024

### 2º Semana – Testes e algoritmos

Olá Ada's e Alan's ✨,

Nesta segunda semana definimos alguns pontos importantes,

- Discutimos sobre o algoritmo responsável pelo cálculo de compatibilidade a ser utilizado no teste vocacional;
- Definimos de cargos de desenvolvimento para cada membro da equipe;
- Analisamos de estratégias para o controle de versão e a documentação do projeto;
- Decidimos realizar pesquisas referentes à proposta do projeto e analisar o funcionamento dos testes vocacionais em plataformas concorrentes.

Nesta semana também tivemos a iniciativa de participar do movimento LED da Globo apresentando o nosso projeto como proposta. Planejamos trazer os resultados das pesquisas e análises como pauta para a próxima reunião.

É isso, até semana que vem!

Figura 52 – Segunda Publicação do Blog

## Terceira Publicação do Blog

24 de março de 2024

### 3º Semana – Pesquisas e análises.

Olá Ada's e Alan's ✨ ,

Nesta terceira semana, acompanhamos as ações anteriores e discutimos novos tópicos importantes:

- Cada membro da equipe realizou pesquisas sobre a proposta do projeto e analisou os testes vocacionais em plataformas concorrentes;
- Discutimos como funciona o SVN e como subiremos nossos arquivos, incluindo o link do YouTube, a apresentação de slides e o documento de introdução;
- Analisamos os testes vocacionais realizados em plataformas concorrentes.

Planejamos definir os requisitos, incluindo requisitos funcionais, não funcionais e regras de negócio, na próxima semana.

É isso, até semana que vem!

Figura 53 – Terceira Publicação do Blog



## Quarta Publicação do Blog

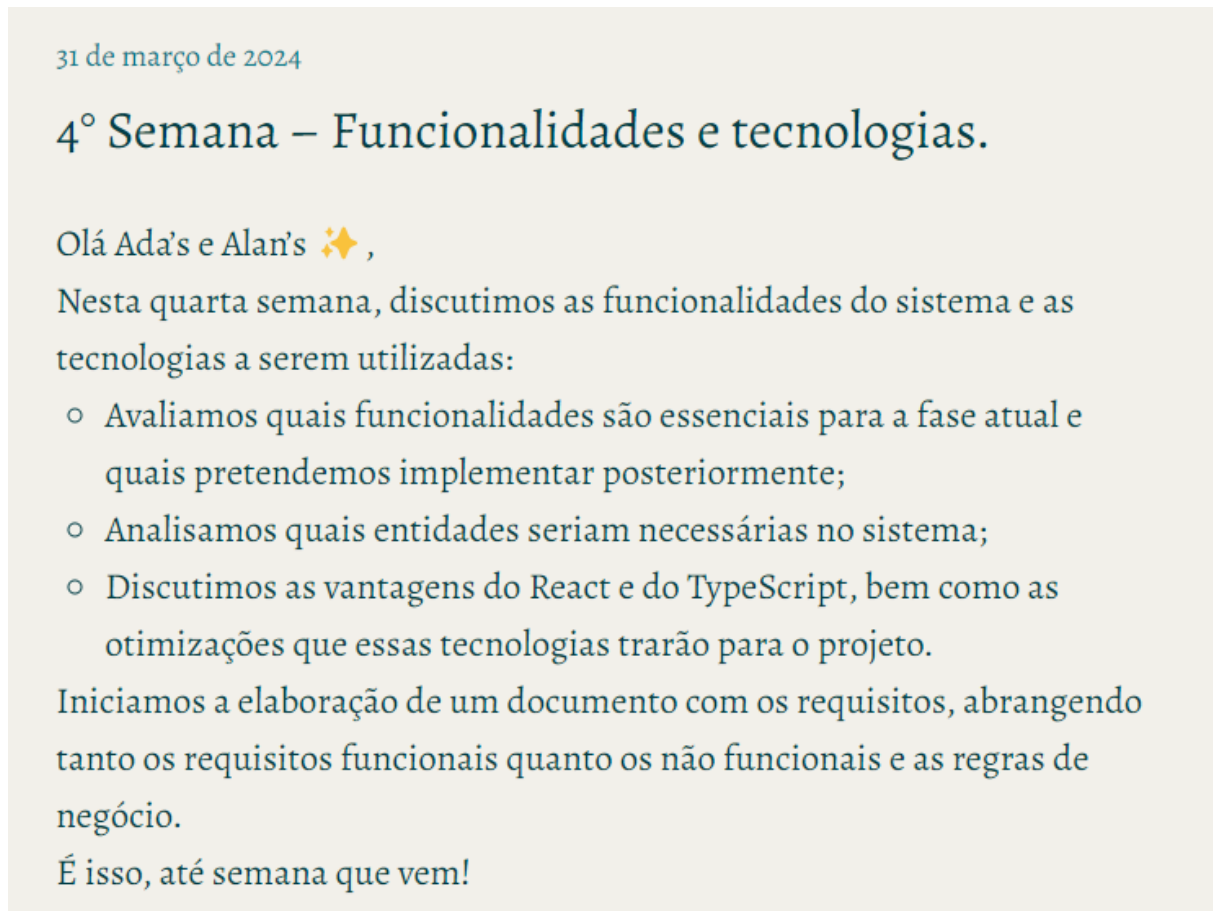


Figura 54 – Quarta Publicação do Blog

## Quinta Publicação do Blog

7 de abril de 2024

### 5º Semana – Entidades e proposta LED.

Olá Ada's e Alan's ✨ ,

Nesta quinta semana, fizemos uma atualização para todos os membros sobre as entidades do sistema e discutimos novos tópicos importantes:

- Atualizamos todos os membros sobre as entidades do sistema e esclarecemos dúvidas sobre a implantação na AWS;
- Definimos todos os elementos necessários no desenho da aplicação;
- Elaboramos a proposta e inscrição para o Prêmio LED.

Nesta semana conseguimos finalizar e realizar todas as etapas necessárias para a inscrição no Prêmio LED.

É isso, até semana que vem!

Figura 55 – Quinta Publicação do Blog

## Sexta Publicação do Blog

14 de abril de 2024

### 6º Semana – Publicação e infraestrutura.

Olá Ada's e Alan's ✨ ,

Nesta sexta semana, fizemos escolhas importantes e ajustes necessários:

- Decidimos publicar toda a documentação no GitHub;
- Dividimos as responsabilidades de criação dos slides entre as integrantes;
- Ajustamos a configuração no Overleaf.

Realizamos um estudo comparativo entre Heroku e Railway para decidir qual delas utilizar para a infraestrutura.

É isso, até semana que vem!

Figura 56 – Sexta Publicação do Blog

## Sétima Publicação do Blog

21 de abril de 2024

### 7º Semana – Prova de conceito.

Olá Ada's e Alan's ✨ ,

Nesta sétima semana, focamos na Prova de Conceito (POC):

- Discutimos e dividimos todas as etapas necessárias para a conclusão bem-sucedida da POC.

Nesta semana foi iniciada a construção da documentação e foi uma semana importante para o planejamento do desenvolvimento do sistema.

É isso, até semana que vem!

Figura 57 – Sétima Publicação do Blog

## Oitava Publicação do Blog

28 de abril de 2024

### 8º Semana – Modelos e ferramentas.

Olá Ada's e Alan's ✨,

Nesta oitava semana, fizemos escolhas importantes sobre modelos e ferramentas:

- Escolhemos o modelo de referência para o código de personalidade que servirá de base para o teste vocacional;
- Decidimos utilizar o modelo de código de personalidade de John Holland;
- Optamos por utilizar o Vite como ferramenta de construção para o frontend.

É isso, até semana que vem!

Figura 58 – Oitava Publicação do Blog

## Nona Publicação do Blog

5 de maio de 2024

### 9º Semana – Desenvolvimento de telas.

Olá Ada's e Alan's ✨ ,

Nesta nona semana, focamos no desenvolvimento das telas do sistema:

- Desenvolvemos o design das telas que serão implementadas no fluxo do perfil do administrador;
- Criamos wireframes detalhados das telas;
- Analisamos os requisitos específicos do perfil do administrador e os fluxos que as telas devem suportar.

As integrantes se reuniram para alinhar o desenvolvimento das telas.

É isso, até semana que vem!

Figura 59 – Nona Publicação do Blog

## Décima Publicação do Blog

12 de maio de 2024

### 10º Semana – Desenvolvimento contínuo.

Olá Ada's e Alan's ✨ ,

Nesta décima semana, discutimos o desenvolvimento das telas conforme os wireframes e atualizamos o progresso individual de cada integrante no projeto:

- Implementamos as alterações nos wireframes conforme discutido;
- Continuamos o desenvolvimento das telas com base nos requisitos estabelecidos;
- Cada integrante compartilhou seu progresso e contribuições até o momento.

É isso, até semana que vem!

Figura 60 – Décima Publicação do Blog

## Décima Primeira Publicação do Blog

19 de maio de 2024

### 11º Semana – Feedbacks e próximos passos.

Olá Ada's e Alan's ✨,

Nesta décima primeira semana, discutimos os feedbacks recebidos durante a apresentação e planejamos os próximos passos para a próxima entrega:

- O projeto foi apresentado e bem recebido pelo professor orientador;
- Foram discutidas melhorias sugeridas pelos feedbacks recebidos.

É isso, até o próximo post!

Figura 61 – Décima Primeira Publicação do Blog