

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

Campus São Paulo

Projeto Integrado I - PI1A5

Prova de conceito

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

8 de Maio 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

Prova de conceito

08 de Maio 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

Lista de ilustrações

Figura 1 – Código QR Blog	18
Figura 2 – Código QR Canal YouTube	19
Figura 3 – Caso de Uso: Funcionamento do Sistema	23
Figura 4 – Arquitetura da aplicação	32
Figura 5 – Modelo Entidade Relacionamento (MER) do projeto	35
Figura 6 – Diagrama Entidade Relacionamento (DER) do projeto	36
Figura 7 – Tela inicial do administrador	48
Figura 8 – Cadastro de dados gerais da instituição	49
Figura 9 – Gerenciamento de instituições	49
Figura 10 – Cadastro de dados gerais da instituição	50
Figura 11 – Cadastro de dados gerais da instituição	50
Figura 12 – Cadastro de dados gerais da instituição	51
Figura 13 – Inserção dos cursos na instituição	51
Figura 14 – Inserção das políticas na instituição	52

Lista de tabelas

Tabela 2 – Estimativa dos Custos Operacionais	43
Tabela 3 – Detalhamento dos Custos Operacionais	43
Tabela 4 – Estimativa de Receitas Operacionais	44

Lista de abreviaturas e siglas

DTO Data Transfer Object - Citado em [41](#)

IES Instituições de Ensino Superior - Citado em [9](#), [44](#)

Sumário

1	INTRODUÇÃO	9
1.1	Justificativa	10
1.2	Objetivos	10
1.3	Análise da Concorrência	10
1.3.1	Aplicativo Soutec	10
1.3.2	Kuau	11
1.3.3	Orientação Vocacional e Coaching	11
1.3.4	FutureMe	11
2	REVISÃO BIBLIOGRÁFICA	12
2.1	O processo de escolha profissional para jovens do Ensino Médio	12
2.2	Testes Vocacionais como Ferramenta de Orientação Profissional	13
2.3	Desigualdade no acesso ao ensino superior	14
3	GESTÃO DO PROJETO	16
3.1	Metodologia de gestão de projeto	16
3.2	Papéis e Atribuições	17
3.2.1	Ferramentas de gestão utilizadas	18
3.2.2	Canais de comunicação	18
4	DESENVOLVIMENTO DO PROJETO	20
4.1	Elicitação de Requisitos	20
4.1.1	Requisitos funcionais	20
4.1.2	Requisitos não funcionais	21
4.1.3	Regras de negócio	22
4.2	Escopo do projeto	22
4.2.1	Casos de Uso	22
4.3	Arquitetura	32
4.3.1	Back-End	33
4.3.1.1	Gerenciamento de dependências	33
4.3.2	Front-End	33
4.3.3	Banco de dados	34
4.3.3.1	Modelo Entidade Relacionamento	35
4.3.3.2	Diagrama Entidade Relacionamento	36
4.3.4	Integrações	37
4.3.5	Infraestrutura	37

4.3.6	Front - End	37
4.3.6.1	Back - End	38
4.3.7	Escalabilidade	38
4.3.8	Controle de versão	38
4.4	Manutenibilidade	39
4.4.1	Code Convention	39
4.4.2	Ferramentas de testes	40
4.4.3	Design Patterns	41
4.5	Segurança, Privacidade e Legislação	41
4.5.1	Lei Geral de Proteção de Dados (LGPD)	41
4.5.2	Autenticação e Autorização	42
4.5.3	Criptografia de Dados	42
4.6	Viabilidade Financeira	42
4.6.1	Monetização	44
4.7	Fases de Entrega	44
4.7.1	Prova de Conceito (POC)	44
4.7.2	Produto Mínimo Viável (MVP)	45
4.7.3	Produto Final	45

REFERÊNCIAS	46
------------------------------	-----------

APÊNDICES	47
----------------------------	-----------

	APÊNDICE A – PROTOTIPAÇÃO	48
A.1	Tela Inicial	48
A.2	Tela de cadastro de dados gerais da instituição	49
A.3	Tela de gerenciamento de instituições	49
A.4	Tela de detalhamento de uma instituição	50
A.5	Tela de edição de uma instituição	50
A.6	Tela de exclusão de uma instituição	51
A.7	Tela de associação de instituições com cursos	51
A.8	Tela de associação de instituições com políticas	52

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 com o objetivo de atender às necessidades dos alunos de escola pública em busca de orientações sobre o ensino superior. Ela disponibiliza informações sobre políticas de cotas e programas de permanência estudantil oferecidos pelas [Instituições de Ensino Superior \(IES\)](#), que são auxílios essenciais para que os alunos consigam se manter durante a graduação. Além disso, a plataforma fornece um teste vocacional para orientar alunos do ensino médio na escolha da carreira a seguir.

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

Este projeto tem como objetivo desenvolver um sistema web que ofereça teste vocacional personalizado e gestão de informações sobre Universidades Públicas Federais do Brasil e Estaduais de São Paulo, como cursos oferecidos, formas de ingresso e políticas públicas de acesso e permanência. Pretendendo facilitar o acesso de estudantes de escolas públicas a Universidades Públicas, e o desenvolvimento de um processo de escolha de ensino superior e de carreira mais preparado.

1.3 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.

1.3.1 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) e considerando a localização de interesse do usuário. Mesmo com as similaridades com a Plataforma Vocco, o foco da Soutec é predominantemente para o público que deseja 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.

1.3.2 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.

1.3.3 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. O site também apresenta recursos de orientação vocacional abrangentes, incluindo informações úteis para orientadores e orientandos.

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.

1.3.4 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

2 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.

2.1 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.

2.2 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, Martinuzzi, Teixeira (2004) e Martins (1978) forneceram descrições recentes dos seis tipos de personalidades vocacionais, enquanto (MAGALHÃES, 2006) ofereceu uma revisão abrangente 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, o estudo destaca 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, o estudo reconhece 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.

2.3 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 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.

3 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.

3.1 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, o Scrum 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.

3.2 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 enviável a codificação ser feita por apenas três integrantes.

3.2.1 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.

3.2.2 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, escolhemos 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 1



Figura 1 – Código QR Blog

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 2



Figura 2 – Código QR Canal YouTube

4 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.

4.1 Elicitação de Requisitos

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

4.1.1 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.

4.1.2 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).

4.1.3 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 de IES públicas.

4.2 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.

4.2.1 Casos de Uso

A seguir, apresenta-se o caso de uso desenvolvido para a plataforma Vocco, apresentado na Figura 3, acompanhado de suas descrições.

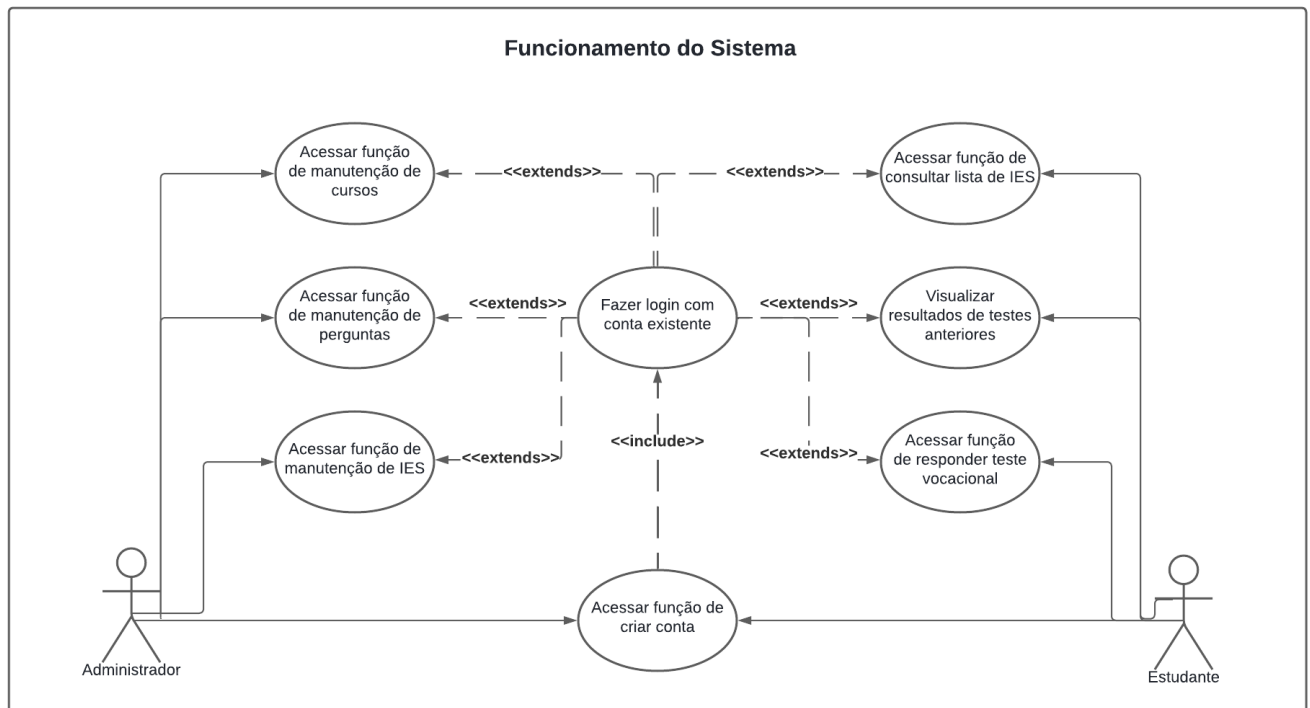


Figura 3 – Caso de Uso: Funcionamento do Sistema

- **Caso de Uso: Fluxo de 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:**

- * O administrador acessa a página de cadastro de administrador no sistema.
- * O sistema solicita ao administrador que insira seu nome, data de nascimento, escolaridade, telefone, CPF, endereço, cargo, e-mail e senha.
- * O administrador insere seus dados conforme o solicitado pelo sistema.
- * O sistema valida os dados inseridos pelo administrador.
- * O sistema verifica se o e-mail é único no sistema.
- * O sistema salva os dados do administrador no banco de dados.
- * O administrador recebe uma confirmação de que seu cadastro foi realizado com sucesso.

– **Fluxos Alternativos:**

- * Se o e-mail fornecido pelo administrador já estiver em uso, o sistema exibe uma mensagem de erro e solicita que o administrador insira um e-mail válido.
- * Se o administrador inserir um e-mail inválido, o sistema exibirá uma mensagem de erro e solicitará que o administrador corrija o e-mail.
- * Se o administrador inserir um CPF inválido, o sistema exibirá uma mensagem de erro e solicitará que o administrador corrija o CPF.
- * Se o administrador não preencher todos os campos obrigatórios, o sistema exibirá uma mensagem de erro e solicitará que o administrador complete os campos em falta.

– **Fluxo de Exceção:**

- * O sistema enfrenta uma falha técnica:
 - O sistema exibe uma mensagem de erro informando sobre a falha técnica.
 - O caso de uso é encerrado.

• **Caso de Uso: Fluxo de Cadastro de Instituição Pública**

– **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:**

- O Administrador possui cadastro na plataforma.
- A instituição pública possui as informações necessárias para o cadastro (nome, site, nota do Mec, endereço).

– **Fluxo Básico:**

- * O administrador acessa a funcionalidade de cadastro de instituição na plataforma.
- * O sistema solicita ao administrador que insira os dados da instituição pública, como nome da instituição, site da instituição, nota do MEC (Ministério da Educação) e endereço completo da instituição.
- * O administrador preenche os campos obrigatórios com as informações solicitadas.

- * O sistema verifica se os dados fornecidos são válidos e únicos no sistema.
- * O sistema salva os dados da instituição pública no banco de dados.
- * O administrador recebe uma confirmação de que a instituição pública foi cadastrada com sucesso.

– **Fluxos Alternativos:**

- * Se o administrador tentar cadastrar a instituição pública com um nome que já está em uso, o sistema exibirá uma mensagem de erro e solicitará um nome válido e único.
- * Se o administrador tentar cadastrar a instituição pública com um site que não é válido, o sistema exibirá uma mensagem de erro e solicitará um site válido.
- * Se o administrador tentar cadastrar a instituição pública com uma nota do MEC que não está dentro do intervalo válido, o sistema exibirá uma mensagem de erro e solicitará uma nota válida.
- * Se o administrador tentar cadastrar a instituição pública com um endereço incompleto ou inválido, o sistema exibirá uma mensagem de erro e solicitará um endereço válido e completo.

– **Fluxo de Exceção:**

- * O sistema enfrenta uma falha técnica:
 - O sistema exibe uma mensagem de erro informando sobre a falha técnica.
 - O caso de uso é encerrado.

• **Caso de Uso: Fluxo de Cadastro de Usuário (Estudante)**

– **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:**

- * O estudante acessa a página de cadastro de usuário
- * O sistema solicita ao estudante que preencha os campos obrigatórios, como nome, data de nascimento, escolaridade, telefone, e-mail e senha.
- * O estudante preenche os campos obrigatórios.

- * O sistema valida os dados inseridos pelo estudante.
- * O sistema cria uma conta para o estudante e salva os dados no banco de dados.
- * O estudante recebe uma confirmação de que seu cadastro foi realizado com sucesso.

– **Fluxos Alternativos:**

- * Se o estudante inserir um e-mail inválido, o sistema exibirá uma mensagem de erro e solicitará que o estudante corrija o e-mail.
- * Se o estudante não preencher todos os campos obrigatórios, o sistema exibirá uma mensagem de erro e solicitará que o estudante complete os campos em falta.

– **Fluxo de Exceção:**

- * O sistema enfrenta uma falha técnica:
 - O sistema exibe uma mensagem de erro informando sobre a falha técnica.
 - O caso de uso é encerrado.

• **Caso de Uso: Fluxo de Manter Perguntas do Teste Vocacional**

– **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:**

- * O administrador possui acesso ao sistema.
- * Para a funcionalidade de alterar e excluir perguntas, deve haver ao menos uma pergunta cadastrada no sistema.

– **Fluxo Básico:**

- * O administrador acessa a funcionalidade de manutenção de perguntas do teste vocacional.
- * O sistema exibe a lista de perguntas existentes.
- * O administrador escolhe entre incluir, alterar, excluir ou consultar uma pergunta.
 - **Incluir Pergunta:**
 - O sistema solicita ao administrador que insira o texto da nova pergunta.

- O administrador insere o texto da pergunta.
- O sistema salva a nova pergunta no banco de dados.
- **Alterar Pergunta:**
- O sistema exibe as perguntas existentes e suas opções de edição.
- O administrador seleciona a pergunta que deseja modificar.
- O sistema permite que o administrador edite o texto da pergunta selecionada.
- O administrador faz as alterações desejadas.
- O sistema salva as alterações no texto da pergunta.
- **Excluir Pergunta:**
- O sistema exibe as perguntas existentes e suas opções de exclusão.
- O administrador seleciona a pergunta que deseja excluir.
- O sistema solicita confirmação para excluir a pergunta selecionada.
- O administrador confirma a exclusão da pergunta.
- O sistema define a pergunta como inativa no banco de dados.
- **Consultar Pergunta:**
- O sistema exibe as perguntas existentes e suas opções de consulta.
- O administrador seleciona a pergunta que deseja consultar.
- O sistema exibe as informações detalhadas da pergunta, como o texto da pergunta e seu ID.
- * O sistema executa a ação escolhida pelo administrador e retorna à lista de perguntas, atualizando-a conforme necessário.
- **Fluxos Alternativos:**
- * Se o administrador tentar incluir uma pergunta com texto vazio, o sistema exibirá uma mensagem de erro e não permitirá a inclusão.
- * Se o administrador tentar alterar uma pergunta que não existe, o sistema exibirá uma mensagem de erro e não permitirá a alteração.
- * Se o administrador tentar excluir uma pergunta que não existe, o sistema exibirá uma mensagem de erro e não permitirá a exclusão.
- **Fluxo de Exceção:**
- * O sistema enfrenta uma falha técnica:
 - O sistema exibe uma mensagem de erro informando sobre a falha técnica.
 - O caso de uso é encerrado.
- **Caso de Uso: Manter Informações sobre Cursos**

- **Ator Principal:** Administrador.
- **Interessados e Interesses:**

O administrador deseja manter informações atualizadas sobre os cursos oferecidos pela instituição pública na plataforma.
- **Pré-Condições:**
 - * O administrador possui acesso à plataforma de cadastro.
 - * A instituição pública já está devidamente cadastrada na plataforma.
- **Fluxo Básico:**
 - * O administrador acessa a funcionalidade de manutenção de informações sobre cursos de uma instituição pública na plataforma.
 - * O sistema apresenta a lista de cursos cadastrados da instituição pública.
 - * O administrador escolhe entre incluir, alterar, excluir ou consultar um curso.
 - **Incluir Curso:**
 - O administrador insere os dados do novo curso, incluindo nome, descrição, grade curricular, formas de ingresso, empregabilidade, possíveis carreiras.
 - O sistema valida os dados fornecidos pelo administrador.
 - O sistema salva as informações do novo curso no banco de dados.
 - **Alterar Curso:**
 - O administrador seleciona o curso que deseja alterar.
 - O administrador faz as alterações necessárias nos dados do curso.
 - O sistema valida as alterações feitas pelo administrador.
 - O sistema salva as alterações no banco de dados.
 - **Excluir Curso:**
 - O administrador seleciona o curso que deseja excluir.
 - O sistema solicita confirmação para excluir o curso selecionado.
 - O administrador confirma a exclusão do curso.
 - O sistema define o curso como inativo no banco de dados.
 - **Consultar Curso:**
 - O administrador seleciona o curso que deseja consultar.
 - O sistema exibe as informações detalhadas do curso selecionado.
- **Fluxos Alternativos:**
 - * Se o administrador tentar incluir um curso com informações incompletas ou inválidas, o sistema exibirá uma mensagem de erro e solicitará informações válidas e completas.

- * Se o administrador tentar alterar um curso que não existe, o sistema exibirá uma mensagem de erro e solicitará a seleção de um curso válido.
- * Se o administrador tentar excluir um curso que não existe, o sistema exibirá uma mensagem de erro e solicitará a seleção de um curso válido.

– **Fluxo de Exceção:**

- * O sistema enfrenta uma falha técnica:
 - O sistema exibe uma mensagem de erro informando sobre a falha técnica.
 - O caso de uso é encerrado.

– **Pós-Condições:**

As informações sobre os cursos são atualizadas no sistema e podem ser consultadas pelos usuários da plataforma.

- **Caso de Uso: Responder o Teste Vocacional**

- **Ator Principal:**Estudante

- **Interessados e Interesses:**

Este caso de uso descreve como um estudante pode responder ao teste vocacional disponibilizado pela plataforma.

- **Pré-condições:**

O estudante deve estar autenticado no sistema para acessar a funcionalidade de responder ao teste vocacional.

- **Fluxo Básico:**

- O estudante acessa a funcionalidade de responder ao teste vocacional no sistema.
- O sistema apresenta as perguntas do teste vocacional ao estudante.
- O estudante lê e responde às perguntas do teste de acordo com suas preferências e características.
- Após responder todas as perguntas, o estudante finaliza o teste.
- O sistema registra as respostas do estudante no banco de dados, associando-as ao perfil do estudante.

- **Fluxos Alternativos:**

- Se o estudante interromper o teste antes de concluí-lo, o sistema preserva as respostas parciais registradas durante o teste, porém, somente as armazena no banco de dados após o estudante concluir integralmente o teste.

- **Fluxo de Exceção:**

- Se o sistema enfrentar uma falha técnica durante o teste:
 - * O sistema exibe uma mensagem de erro informando sobre a falha técnica.
 - * O estudante pode tentar responder ao teste novamente após a resolução do problema técnico.

- **Pós-condições:**

As respostas do estudante são registradas no sistema e podem ser utilizadas para gerar recomendações de cursos com base nos resultados do teste.

- **Caso de Uso: Fluxo de 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:**

- * O estudante realizou o teste vocacional e suas respostas estão registradas no sistema.
- * O sistema possui algoritmos de análise e recomendação de cursos configurados.

- **Fluxo Básico:**

- * O sistema analisa as respostas do teste vocacional do estudante, que estão previamente registradas e associadas no banco de dados.
- * Com base nas respostas analisadas, o sistema utiliza um algoritmo de recomendação para gerar uma lista de cursos que são mais adequados ao perfil do estudante.
- * O sistema apresenta a lista de cursos recomendados ao estudante após a conclusão do teste vocacional.

- **Fluxos Alternativos:**

Se o sistema não conseguir gerar uma recomendação com base nas respostas do teste, ele pode sugerir ao estudante que procure orientação adicional de um orientador educacional.

- **Fluxo de Exceção:**

- * O sistema enfrenta uma falha técnica ao gerar a recomendação de curso:

- O sistema exibe uma mensagem de erro informando sobre a falha técnica.
 - O caso de uso é encerrado.
- **Pós-Condições:**

O estudante recebe uma recomendação personalizada de cursos com base nas respostas do teste vocacional, auxiliando-o na escolha de sua formação acadêmica.
- **Caso de Uso: Fluxo de Consultar Informações das Instituições de Ensino Superior (IES)**
 - **Ator Principal:**Estudante
 - **Interessados e Interesses:**

O estudante deseja obter informações detalhadas sobre os cursos oferecidos pelas Instituições de Ensino Superior (IES) cadastradas na plataforma, permitindo filtrar as informações para encontrar cursos de seu interesse.
 - **Pré-Condições:**
 - * O estudante possui acesso à plataforma.
 - * As informações sobre os cursos e IES estão cadastradas e atualizadas na plataforma.
 - **Fluxo Básico:**
 - * O estudante acessa a funcionalidade de consultar informações sobre os cursos e IES na plataforma.
 - * O sistema exibe uma lista de cursos e IES cadastradas, apresentando informações básicas como nome do curso, nome da instituição, endereço, entre outras.
 - * O estudante pode utilizar filtros disponíveis na plataforma para refinar a busca, como por área de conhecimento, nome de curso, entre outros.
 - * O sistema atualiza dinamicamente a lista de cursos e IES conforme os filtros selecionados pelo estudante.
 - * O estudante visualiza as informações detalhadas dos cursos e IES selecionados na lista, incluindo grade curricular, notas de corte, empregabilidade, possíveis carreiras, entre outros detalhes relevantes.
 - **Fluxos Alternativos:**

Se não houver cursos ou IES cadastradas que correspondam aos critérios de busca do estudante, o sistema exibirá uma mensagem informando que não foram encontrados resultados.

– **Fluxo de Exceção:**

- * O sistema enfrenta uma falha técnica ao gerar a recomendação de curso:
 - O sistema exibe uma mensagem de erro informando sobre a falha técnica.
 - O caso de uso é encerrado.

4.3 Arquitetura

Nesse projeto, adotamos a **Clean Architecture** como o princípio norteador para o desenvolvimento do software. A decisão de seguir essa abordagem se fundamenta na necessidade de construir um sistema flexível e facilmente manutenível ao longo do tempo. Ao adotar a Clean Architecture, priorizamos a separação clara e bem definida das responsabilidades em diferentes camadas, como a de entidades de negócio, casos de uso, interfaces de usuário e infraestrutura. Isso não apenas promove uma estrutura organizada e modular, mas também facilita a realização de testes automatizados em cada uma dessas camadas, garantindo a qualidade do código e a detecção precoce de possíveis problemas. Além disso, ao desacoplar as regras de negócio das tecnologias específicas, tornamos o sistema mais adaptável a mudanças futuras, permitindo a evolução contínua do software de forma sustentável.

Na Figura 4 é possível observar a representação em diagrama da arquitetura escolhida para a aplicação.

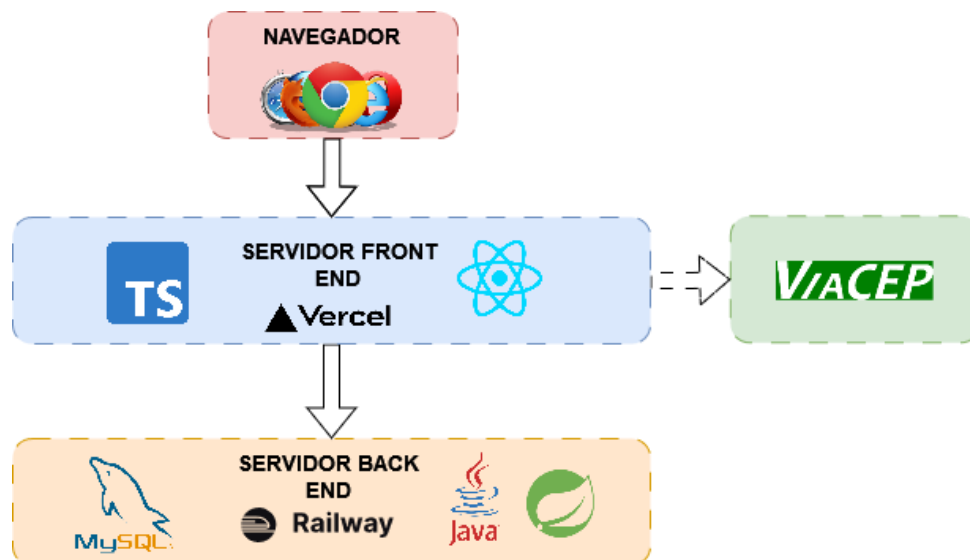


Figura 4 – Arquitetura da aplicação

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

4.3.1 Back-End

No que diz respeito ao back-end, adotamos o framework **Spring Boot** em conjunto com a arquitetura MSC (Model-Service-Controller). O Spring Boot é uma escolha popular para o desenvolvimento de aplicativos Java devido à sua facilidade de configuração e rápida inicialização. 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 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 CRUD (Create, Read, Update, Delete), 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.

4.3.1.1 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.

4.3.2 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. 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.

4.3.3 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.

4.3.3.1 Modelo Entidade Relacionamento

O modelo entidade-relacionamento (ER) 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 5

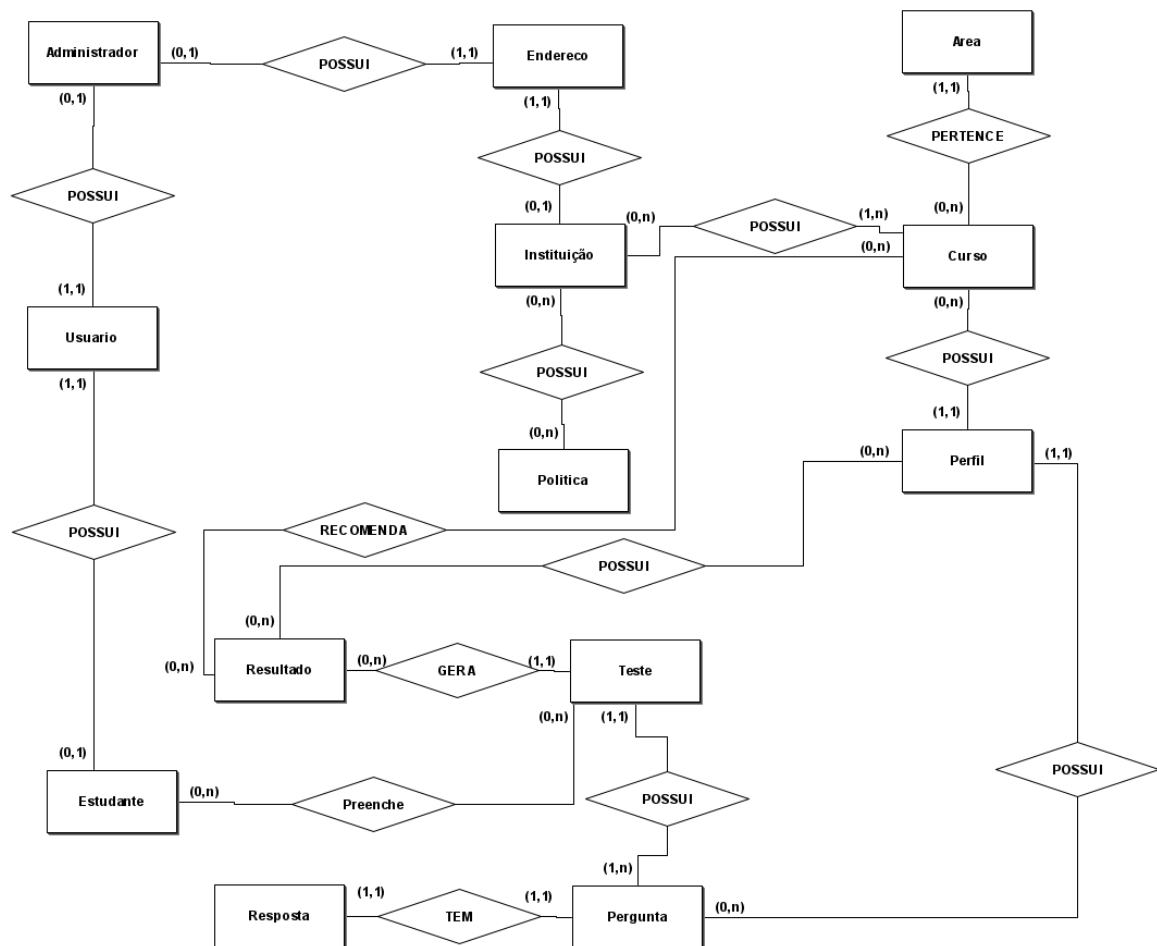


Figura 5 – Modelo Entidade Relacionamento (MER) do projeto

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.

4.3.3.2 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 6

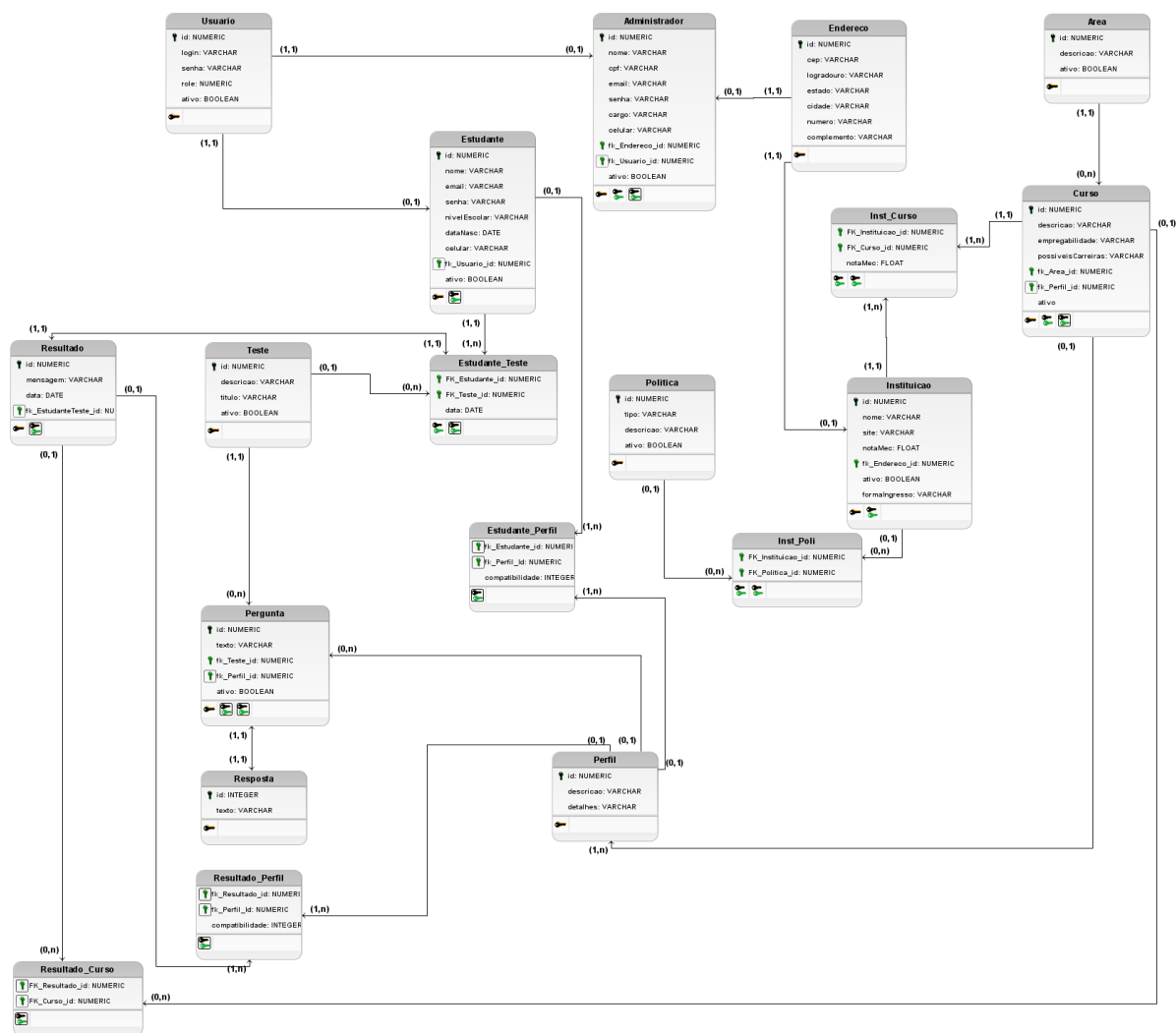


Figura 6 – Diagrama Entidade Relacionamento (DER) do projeto

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.

4.3.4 Integrações

Para aprimorar a experiência do usuário e garantir a precisão dos dados de endereço nos cadastros, optamos por integrar a 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 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.

4.3.5 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.

4.3.6 Front - End

Para hospedar o frontend, escolhemos a plataforma **Vercel**, conhecida por sua facilidade de uso e escalabilidade. O frontend 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 frontend se conecta ao backend através de requisições 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.

4.3.6.1 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.

4.3.7 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.

4.3.8 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 e implantação contínua (CI/CD).

4.4 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.

4.4.1 Code Convention

- **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 APIs;
 - **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.

4.4.2 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 potenci-

ais problemas, vulnerabilidades de segurança, bugs, duplicações de código e padrões de código não conformes.

4.4.3 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.

4.5 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.

4.5.1 Lei Geral de Proteção de Dados (LGPD)

A Lei Geral de Proteção de Dados Pessoais (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.

4.5.2 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 APIs. 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.

4.5.3 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 Lei Geral de Proteção de Dados (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.

4.6 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 2 – 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
Total	44.208,12	3.684,01

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 3 a seguir:

Tabela 3 – 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
Total		44.208,12	3.684,01

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.

4.6.1 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 4 a seguir:

Tabela 4 – 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
Total de Receita	45.936,00	3.828,00

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 4.

4.7 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.

4.7.1 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](#).

4.7.2 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.

4.7.3 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. Esta entrega será feita no mês de dezembro de 2024.

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 9.

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 9.

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 14 e 15.

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 13.

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 13.

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 9.

Apêndices

APÊNDICE A – PROTOTIPAÇÃO

A.1 Tela Inicial

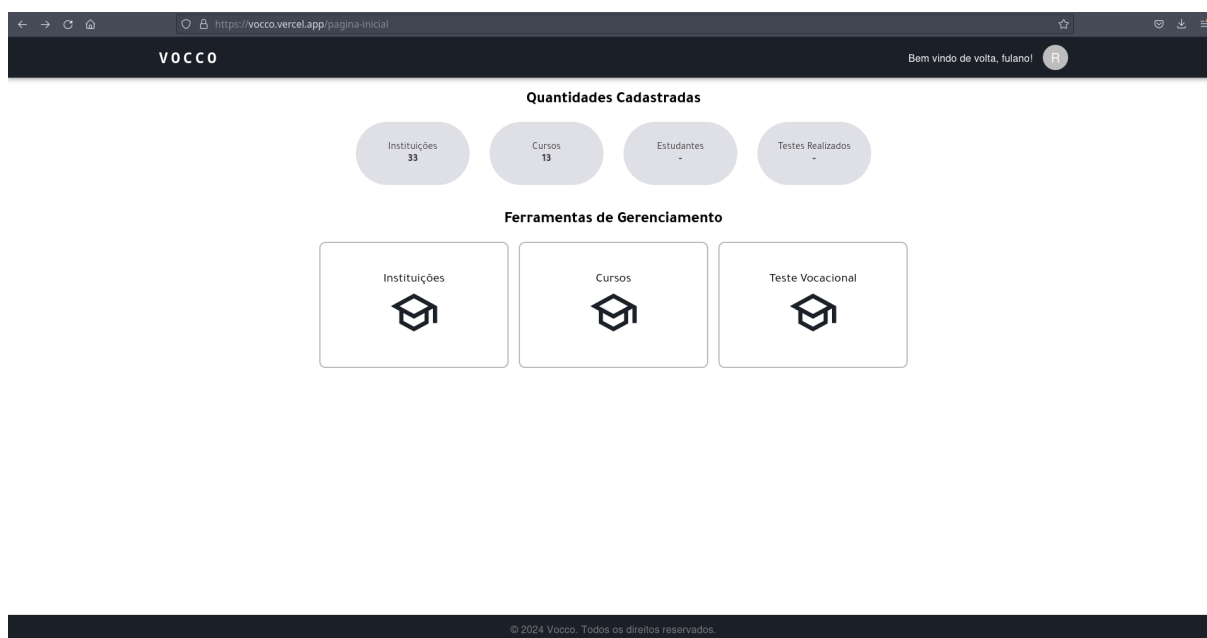


Figura 7 – Tela inicial do administrador

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 8 – 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 9 – Gerenciamento de instituições

A.4 Tela de detalhamento de uma instituição

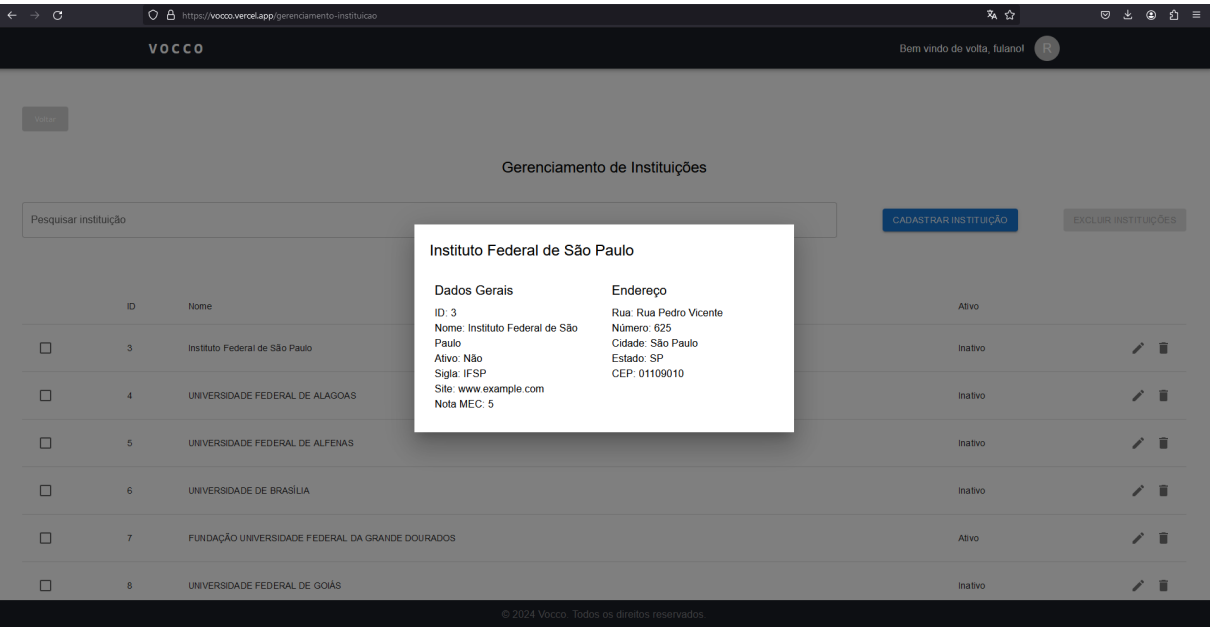


Figura 10 – Cadastro de dados gerais da instituição

A.5 Tela de edição de uma instituição

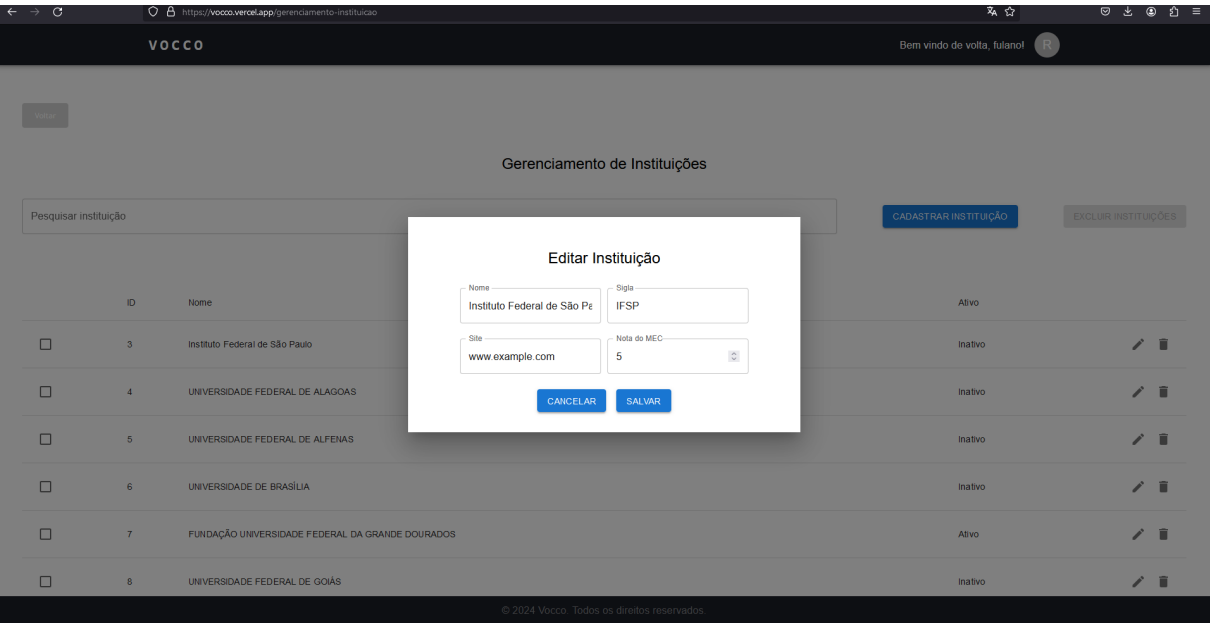


Figura 11 – Cadastro de dados gerais da instituição

A.6 Tela de exclusão de uma instituição

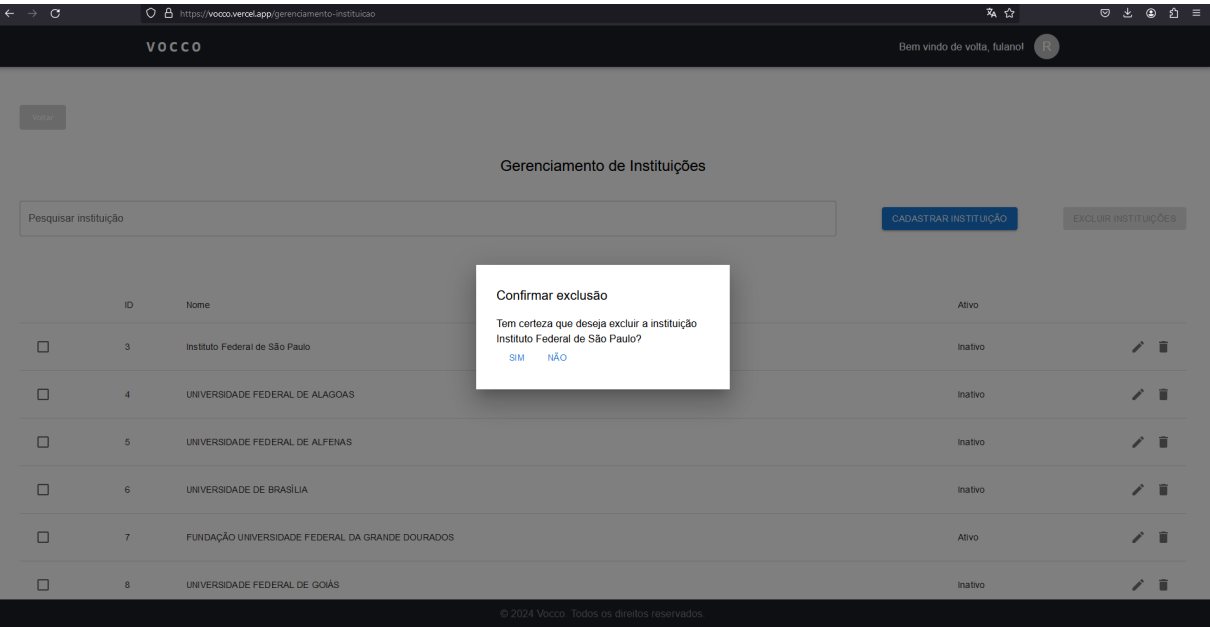


Figura 12 – Cadastro de dados gerais da instituição

A.7 Tela de associação de instituições com cursos

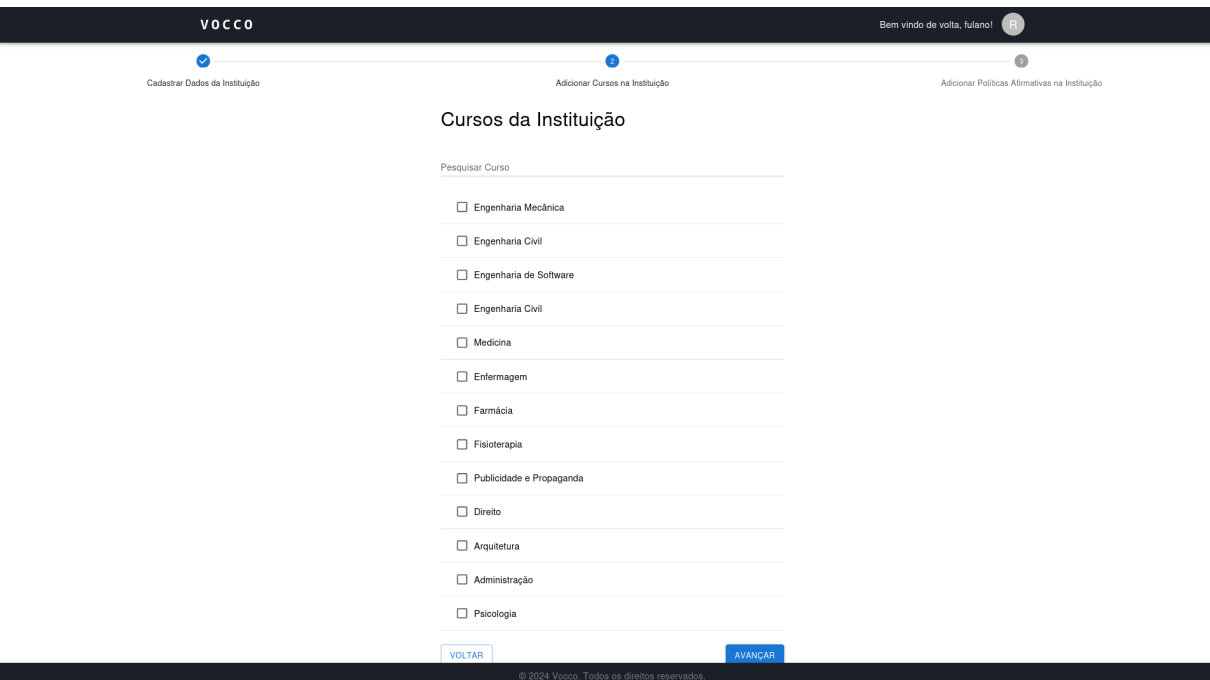


Figura 13 – Inserção dos cursos na instituição

A.8 Tela de associação de instituições com políticas

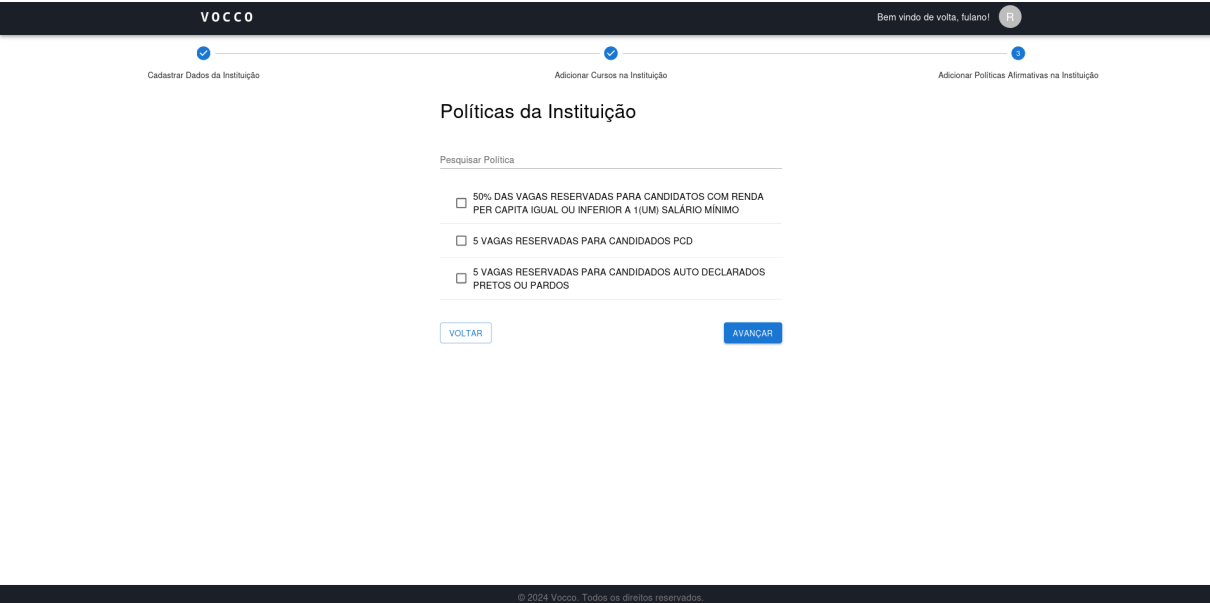


Figura 14 – Inserção das políticas na instituição