

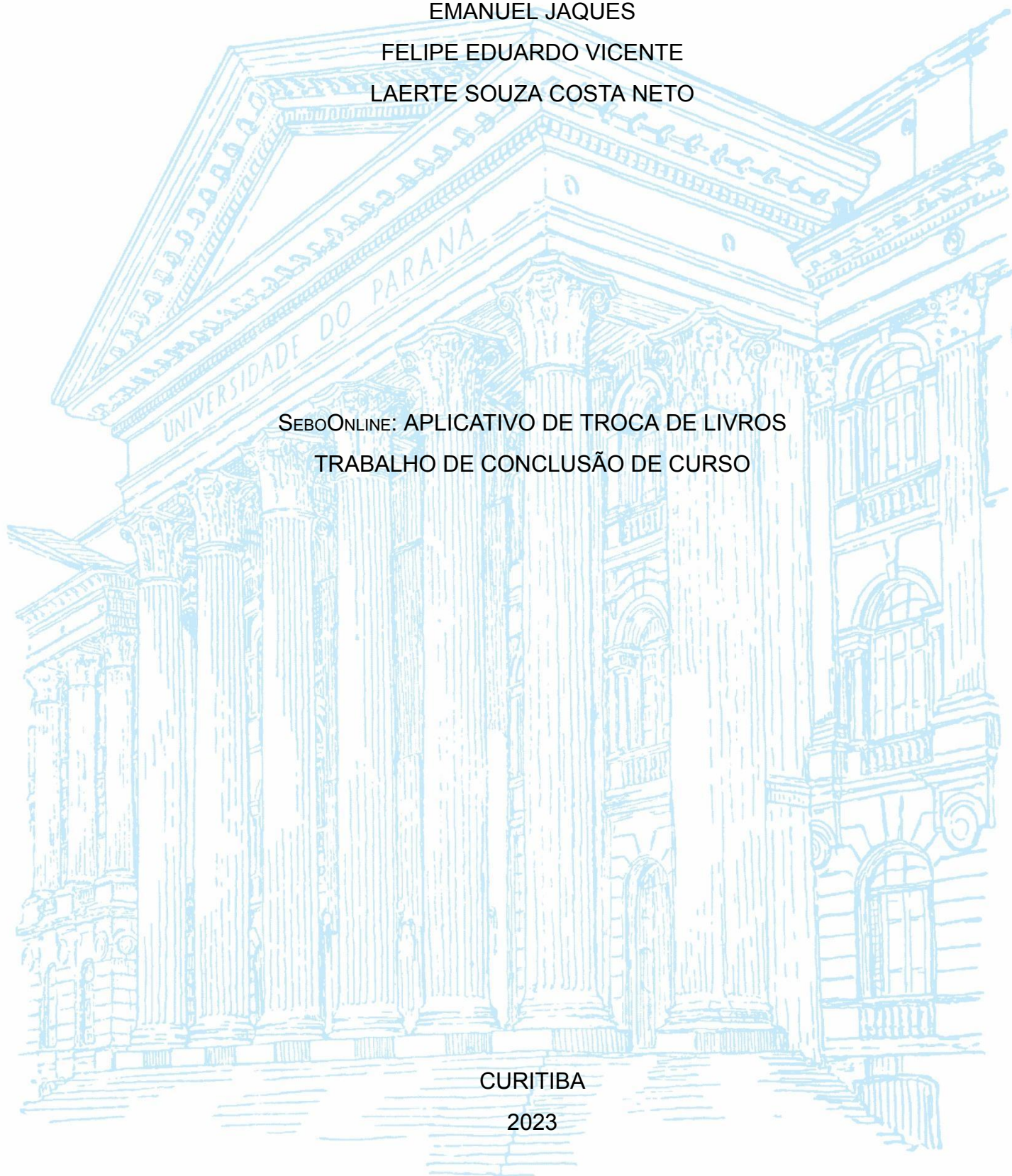
UNIVERSIDADE FEDERAL DO PARANÁ

EMANUEL JAQUES  
FELIPE EDUARDO VICENTE  
LAERTE SOUZA COSTA NETO

SEBOONLINE: APLICATIVO DE TROCA DE LIVROS  
TRABALHO DE CONCLUSÃO DE CURSO

CURITIBA

2023



EMANUEL JAQUES  
FELIPE EDUARDO VICENTE  
LAERTE SOUZA COSTA NETO

SEBOONLINE: APLICATIVO DE TROCA DE LIVROS

Trabalho de conclusão de curso apresentado ao curso de Tecnologia em Análise e Desenvolvimento de Sistemas, Setor de Educação Profissional e Tecnológica, Universidade Federal do Paraná, como requisito parcial à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador: Prof. Doutor Dieval Guizelini

CURITIBA

2023



UNIVERSIDADE FEDERAL DO PARANÁ  
SETOR DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA  
Rua Alcides Vieira Arcoverde 1225, - - Bairro Jardim das Américas, Curitiba/PR,  
CEP 81520-260  
Telefone: 3360-5000 - <http://www.ufpr.br/>

ATA DE REUNIÃO

**TERMO DE APROVAÇÃO**

EMANUEL DA SILVA JACQUES  
FELIPE EDUARDO VICENTE  
LAERTE SOUZA COSTA NETO

**SeboOnline: APLICATIVO DE TROCA DE LIVROS**

Monografia aprovada como requisito parcial à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas, do Setor de Educação Profissional e Tecnológica da Universidade Federal do Paraná.

Prof. Dr. Dieval Guizelini  
Orientador – SEPT/UFPR

Prof. Dr. Mario de Paula Soares Filho  
SEPT/UFPR

Prof. Dr. Mauro Antonio Alves Castro  
SEPT/UFPR



Documento assinado eletronicamente por **DIEVAL GUIZELINI, PROFESSOR ENS BASICO TECN TECNOLÓGICO**, em 12/12/2023, às 16:50, conforme art. 1º, III, "b", da Lei 11.419/2006.



Documento assinado eletronicamente por **MAURO ANTONIO ALVES CASTRO**,  
**PROFESSOR DO MAGISTERIO SUPERIOR**, em 12/12/2023, às 16:52, conforme art. 1º, III,  
"b", da Lei 11.419/2006.

---



Documento assinado eletronicamente por **MARIO DE PAULA SOARES FILHO**,  
**PROFESSOR ENSINO BAS/TEC/TECNOL**, em 12/12/2023, às 16:58, conforme art. 1º, III,  
"b", da Lei 11.419/2006.

---



A autenticidade do documento pode ser conferida [aqui](#) informando o código verificador  
**6273572** e o código CRC **AD35CA3F**.

---

“Com muita sabedoria, estudando muito, pensando muito, procurando compreender tudo e todos, um homem consegue, depois de mais ou menos 40 anos de vida, aprender a ficar calado.”

(Millôr Fernandes)

## RESUMO

A leitura desempenha um papel fundamental no desenvolvimento intelectual, emocional e social das pessoas. Além de ser uma fonte inesgotável de conhecimento, um dos pilares que motivou a criação desse sistema, visa conectar entusiastas de livros, leitores ávidos e doadores de maneira eficiente e prática. O sistema de empréstimo de livros e doações tem recursos que permitem aos usuários criarem contas pessoais, listar os livros que desejam compartilhar ou doar, conectar-se com outros usuários e solicitar empréstimos. A plataforma também fornece uma função de filtrar por categoria, promovendo a facilidade de trocas. Através de uma interface amigável e intuitiva, os usuários podem criar e gerenciar suas bibliotecas pessoais e controlar os livros emprestados. O sistema foi desenvolvido usando tecnologias atuais, tais como Java 17 com Spring Boot, Angular v16 e PostgreSQL.

**Palavras-chave:** Livros, trocas voluntárias; meios digitais; aplicativo;

## **ABSTRACT**

Reading plays a fundamental role in the intellectual, emotional, and social development of individuals. Besides being an endless source of knowledge, one of the pillars that motivated the creation of this system is to connect book enthusiasts, avid readers, and donors efficiently and practically. The book lending and donation system has features that allow users to create personal accounts, list the books they want to share or donate, connect with other users, and request loans. The platform also provides a filtering function by category, promoting ease of exchanges. Through a user-friendly and intuitive interface, users can create and manage their personal libraries and track borrowed books. The system was developed using current technologies such as Java 17 with Spring Boot, Angular v16, and PostgreSQL.

**Keywords:** Voluntary exchanges; books; digital media; application.

## LISTA DE FIGURAS

FIGURA 01 - Site web BookCrossing .....	13
FIGURA 02 – Site Estante Virtual .....	14
FIGURA 03 : MERCADO LIVRE(2023) .....	15
FIGURA 04 - Comparação Sites .....	15
FIGURA 05 – QUADRO KANBAN INICIAL .....	18
FIGURA 06 - Visão geral Projeto no GITHUB.....	33
FIGURA 07 - Ilustração Arquitetura .....	35
FIGURA 08 - Fluxo de Comunicação.....	39
FIGURA 09 - Fluxo de Comunicação 2 .....	40
FIGURA 10 – ESTRUTURA FRONT END(ANGULAR).....	41
FIGURA 11 – ESTRUTURA BACK END(JAVA/SPRING).....	42
FIGURA 12 – TELA DE LOGIN.....	43
FIGURA 13 – TELA DE RECUPERAR SENHA .....	44
FIGURA 14 – TELA DE CADASTRAR .....	44
FIGURA 15 – TELA DE HOME .....	45
FIGURA 16 – TELA DE HOME/COMENTÁRIOS .....	46
FIGURA 17 – TELA DE HOME/COMENTÁRIOS - RESPOSTAS AOS COMENTÁRIOS .....	46
FIGURA 18 – TELA DE HOME/FAVORITAR-DESFAVORITAR.....	47
FIGURA 19 – TELA DE PERFIL USUÁRIO.....	47
FIGURA 20 – TELA DE CADASTRO DE INFORMAÇÕES .....	48
FIGURA 21 – TELA DE ATUALIZAR SENHA .....	49
FIGURA 22 – TELA DE ADICIONAR CAPAS DE LIVROS .....	49
FIGURA 23 – TELA DE PESQUISAR NOVOS LEITORES .....	50

FIGURA 24 – TELA LOGIN.....	59
FIGURA 25 – TELA HOME/ SAIR .....	60
FIGURA 26 – TELA DE CADASTRO .....	62
FIGURA 27 – TELA TAG .....	66
FIGURA 28 – TELA DE ADICIONAR DADOS.....	68
FIGURA 29 – TELA GERENCIAR LIVRO .....	71
FIGURA 30 – TELA RECUPERAR SENHA .....	72
FIGURA 31 – TELA PEGAR EMPRESTADO.....	74
FIGURA 32 – TELA DELETAR CONTA.....	75

## **Lista de abreviaturas ou siglas**

API - Application Programming Interface

BD - Banco de Dados

DER - Diagrama Entidade Relacionamento

DTO - Data Transfer Object

HTTP - Hypertext Transfer Protocol (protocolo de transferência de hipertexto)

IDE - Integrated Development Environment (Ambiente de Desenvolvimento I integrado)

JVM - Java Virtual Machine (máquina virtual Java)

MVC - Model-View-Controller (Modelo-Visualização-Controlador)

ORM - Object-Relational Mapping (mapeamento objeto-relacionamento)

REST - Representational State Transfer (Transferência de Estado Representacional)

SEO - Search Engine Optimization (otimização para motores de busca)

SPA - Single Page Application (aplicação de uma única página)

SSG - Static Site Generation (geração de páginas estáticas)

SSR - Server-Side Rendering (renderização do lado servidor)

SQL - Structured Query Language (linguagem estruturada de consulta)

TADS - Tecnologia em Análise e Desenvolvimento de Sistemas

UML - Unified Modeling Language (Linguagem de Modelagem Unificada)

UFPR - Universidade Federal do Paraná

URI - Uniform Resource Identifier (identificador uniforme de recurso)

## SUMÁRIO

<b>1. INTRODUÇÃO</b> .....	<b>8</b>
<b>1.1. PROBLEMA</b> .....	<b>9</b>
<b>1.2. OBJETIVOS</b> .....	<b>9</b>
<b>1.3. JUSTIFICATIVA</b> .....	<b>10</b>
<b>2. FUNDAMENTAÇÃO TEÓRICA</b> .....	<b>12</b>
<b>2.1. APLICATIVOS SIMILARES</b> .....	<b>13</b>
<b>2.1.1. BookCrossing</b> .....	<b>14</b>
<b>2.1.2. Estante Virtual</b> .....	<b>14</b>
<b>2.1.3. Mercado Livre</b> .....	<b>15</b>
<b>2.1.4. Comparação das funcionalidades das soluções similares</b> .....	<b>16</b>
<b>3. MATERIAIS E MÉTODOS</b> .....	<b>17</b>
<b>3.1. METODOLOGIA DE DESENVOLVIMENTO</b> .....	<b>17</b>
<b>3.2. KANBAN</b> .....	<b>18</b>
<b>3.3. SCRUM</b> .....	<b>20</b>
<b>3.4. ADAPTAÇÃO DA METODOLOGIA SCRUM</b> .....	<b>21</b>
<b>3.5. FERRAMENTAS DE DESENVOLVIMENTO</b> .....	<b>21</b>
<b>3.5.1 BALSAMIQ</b> .....	<b>21</b>
<b>3.5.2 ASTAH</b> .....	<b>22</b>
<b>3.5.3 IDE INTELLIJ IDEA COMMUNITY</b> .....	<b>24</b>
<b>3.5.4 VS CODE</b> .....	<b>25</b>
<b>3.5.5 POSTMAN</b> .....	<b>26</b>
<b>3.5.6 POSTGRESQL</b> .....	<b>27</b>
<b>3.5.7 BROWSER MOZILLA FIREFOX</b> .....	<b>28</b>
<b>3.6. TECNOLOGIAS DE DESENVOLVIMENTO</b> .....	<b>30</b>
<b>3.6.1. Framework Spring</b> .....	<b>31</b>
<b>3.6.2. Angular</b> .....	<b>31</b>
<b>3.6.3. APIs em Spring</b> .....	<b>32</b>
<b>3.6.4. PostgreSQL</b> .....	<b>32</b>
<b>3.6.5. Git</b> .....	<b>33</b>
<b>3.6.6. GitHub</b> .....	<b>33</b>
<b>3.6.7. Versões das tecnologias utilizadas</b> .....	<b>35</b>
<b>3.7 ARQUITETURA EM CAMADAS</b> .....	<b>36</b>
<b>3.7.1. Topologia</b> .....	<b>36</b>
<b>3.8. INFRAESTRUTURA DE DESENVOLVIMENTO</b> .....	<b>38</b>
<b>4. RESULTADO: PROJETO E IMPLEMENTAÇÃO DO SISTEMA SeboOnline</b> .....	<b>39</b>
<b>4.1 ARQUITETURA DO SOFTWARE</b> .....	<b>39</b>
<b>4.2. AS INTERFACES DO SOFTWARE</b> .....	<b>44</b>
<b>4.3 IMPLANTAÇÃO E MELHORIAS FUTURAS</b> .....	<b>51</b>
<b>5 CONSIDERAÇÕES FINAIS</b> .....	<b>53</b>

<b>REFERÊNCIAS.....</b>	<b>54</b>
<b>APÊNDICE A – REQUISITOS DO SISTEMA.....</b>	<b>55</b>
<b>1.1 REQUISITOS FUNCIONAIS.....</b>	<b>55</b>
<b>APÊNDICE B – DIAGRAMA DE CASO DE USOS NÍVEL I.....</b>	<b>57</b>
<b>APÊNDICE C – DIAGRAMA DE CASOS DE USOS NÍVEL II.....</b>	<b>58</b>
<b>APÊNDICE D – ESPECIFICAÇÕES DE CASOS DE USOS.....</b>	<b>58</b>
<b>APÊNDICE E – DIAGRAMA DE CLASSES.....</b>	<b>77</b>
<b>APÊNDICE F – DIAGRAMAS DE SEQUÊNCIA.....</b>	<b>78</b>
<b>APÊNDICE G – DIAGRAMA ENTIDADE-RELACIONAMENTO.....</b>	<b>89</b>
<b>APÊNDICE H – DIAGRAMA DE OBJETOS.....</b>	<b>90</b>

## 1. INTRODUÇÃO

Em um mundo cada vez mais tecnológico, as relações humanas se diversificam extrapolando a dimensão física para coexistirem também na virtual. Isso resulta em facilidades, entre outras, a troca de informações instantânea. Contudo, o avanço da tecnologia e a crescente virtualização de nossas interações têm gerado impactos significativos em diversas esferas da vida, inclusive na forma como consumimos cultura e adquirimos conhecimento.

Em um ambiente cada vez mais digital, onde as distâncias físicas estão sendo reduzidas pelas redes sociais, onde a sociedade está sendo reorganizada constantemente em bolhas que compartilham interesses comuns, identificamos um grupo de pessoas, que demonstram grande interesse em leitura, principalmente de livros físicos. Uma parte dessas pessoas já estão organizadas em diferentes plataformas e redes (canais no youtube, grupos no telegram ou whatsapp, facebook, instagram etc). Essas redes não são especializadas no tema, mas possibilitam o agrupamento e o compartilhamento dos seus interesses comuns.

criação de um sistema que catalogue as obras disponíveis para doação, empréstimo, e/ou permuta de livros físicos poderá ampliar o acesso aos livros por mais pessoas e incentivar a leitura de um mesmo livro por mais pessoas e ao mesmo tempo em que valorizam e preservam o ato de leitura tradicional.

Essa abordagem permite que as pessoas escolham títulos, solicitem empréstimos e compartilhem obras com facilidade, bastando ter acesso a internet e a partir de qualquer lugar. Essa ferramenta promove a interação social e o compartilhamento de conhecimento, ao mesmo tempo em que valoriza o prazer de folhear um livro físico.

Além disso, esses sistemas de catálogo de livros físicos possibilitam a criação de comunidades de leitores, incentivando discussões e interações em torno das obras emprestadas. O sistema pode ser útil para os clubes de leitura, grupos de discussão e eventos relacionados à leitura de obras físicas, podendo prosperar, fortalecendo a importância da cultura do livro em papel na era digital.

Assim, este Trabalho de Conclusão do Curso de graduação em tecnologia em Análise e Desenvolvimento de Sistemas propõe o sistema inovador que potencializa e favorece o interesse por livros físicos, ao mesmo tempo em que mantém viva a tradição da leitura tangível.

### **1.1. PROBLEMA**

Muitos leitores de livros possuem uma vasta coleção, mas, cada vez mais, encontram dificuldades para compartilhar, trocar seus livros.. As feiras de livros vêm se tornando mais raras, assim os amantes de obras literárias precisam de um meio adequado para a amostragem e disponibilização de suas obras.

### **1.2. OBJETIVOS**

Fornecer um site na web que possibilite a postagem e a busca de obras literárias de acordo com seus interesses, de forma segura, dinâmica, propositiva para ambas as partes, estreitando o caminho entre os leitores de uma determinada região.

### 1.3. JUSTIFICATIVA

A criação deste sistema de catálogo de livros para empréstimo, permutação ou doações de livros, com características das redes sociais, é fundamentada em diversas razões relevantes e justificáveis, tanto no âmbito educacional quanto no aspecto social e tecnológico.

Inicialmente, a iniciativa surge da crescente necessidade de promover o acesso à leitura e à cultura, dados os desafios contemporâneos, como a falta de acesso a bibliotecas, livrarias e sebos, além da escassez de recursos educacionais. Este projeto busca contribuir para reduzir essa lacuna, incentivando a disseminação do conhecimento, proporcionando a troca de livros entre usuários e promovendo a cultura da leitura.

A plataforma proposta atende a uma demanda por acesso a recursos educacionais, mas também promove a sustentabilidade ao encorajar a reutilização de livros e reduzir a necessidade de compra de novos exemplares. Esta abordagem alinha-se com princípios ecológicos e de sustentabilidade, contribuindo para a redução do impacto ambiental.

Do ponto de vista tecnológico, a proposta deste Trabalho de Conclusão de Curso (TCC) abrange o desenvolvimento de um sistema que integra aspectos de redes sociais com funcionalidades específicas para gerenciamento de bibliotecas pessoais, empréstimos e doações. Isso representa um desafio inovador para a equipe, no contexto da Análise e Desenvolvimento de Sistemas, possibilitando explorar diferentes tecnologias e arquiteturas para alcançar a eficiência, segurança e escalabilidade necessárias.

Destacamos que a abordagem ética e legal explorada ao longo do trabalho é crucial para garantir a privacidade dos usuários e lidar com questões de segurança e proteção de dados, além das complexidades inerentes ao processo de compartilhamento de livros e interesses pessoais. Portanto, o sistema *SeboOnline* aqui apresentado atende os objetivos da disciplina de TCC do Curso, que possibilitou o desenvolvimento em equipe, de forma semelhante que ocorre no mercado de trabalho, permitindo aos desenvolvedores experimentarem novas ferramentas e novas técnicas de desenvolvimento, mas principalmente, permitiu

proponer una tecnología que puede ser relevante, se aplicada, a promoción da lectura e valorización da educación e da cultura, contribuyendo para a sustentabilidade social e preservando os aspectos éticos e legais.

## 2. FUNDAMENTAÇÃO TEÓRICA

A leitura desempenha um papel fundamental no desenvolvimento educacional, intelectual e social das pessoas. Sua importância é respaldada pelo(a):

a) Desenvolvimento Cognitivo e Intelectual:

- Estudos indicam que a leitura estimula áreas do cérebro relacionadas à compreensão, interpretação e criatividade.

b) Capacidade de Comunicação e Expressão:

- A leitura frequente contribui para a ampliação do vocabulário e melhora as habilidades de comunicação escrita e oral.

c) Empoderamento e Desenvolvimento Pessoal:

- Os livros podem proporcionar conhecimento sobre diferentes culturas, expandindo a visão de mundo e promovendo a empatia.

### **Estatísticas e Dados sobre Hábitos de Leitura**

Os dados da UNESCO apresentados no Quadro 1 colocam a América Latina e conseqüentemente o Brasil, como regiões que carecem de hábitos de leitura.

#### **QUADRO 1 - Contexto Global de Leitura**

Estatísticas	Dados
Porcentagem da população mundial que lê regularmente	Cerca de 27% a 35% (Fonte: UNESCO)
Países com maiores índices de leitura	Noruega, Islândia, Finlândia, Suécia
Países com desafios no acesso à leitura	Algumas regiões da África e América Latina

Fonte: os autores adaptado da UNESCO (2014)

Por outro lado, os dados estatísticos relacionam o hábito da leitura com o desempenho acadêmico, conforme descrito no Quadro 2.

## QUADRO 2 - Impacto da Leitura na Educação

Estatísticas	Dados
Alunos que leem regularmente têm melhor desempenho acadêmico	Estudo da OCDE mostrou que alunos que leem diariamente têm melhor desempenho em matemática, ciências e compreensão de texto.
Acesso a livros e recursos influencia o aprendizado	Em áreas com acesso limitado a livros, os estudantes têm menor desempenho escolar.

Fonte: os autores

### Tendências e Inovações no Empresto e Compartilhamento de Livros

Plataformas Digitais de Compartilhamento de Livros:

- Crescimento de aplicativos e sites que facilitam o empréstimo e doação de livros entre usuários, aumentando o acesso à leitura.

Impacto da Tecnologia na Leitura:

- E-books e dispositivos eletrônicos têm modificado os hábitos de leitura, oferecendo praticidade, acessibilidade e diversidade de conteúdo.

É indiscutível a importância da leitura na sociedade e os benefícios do acesso a livros para o desenvolvimento educacional e cultural. O sistema *SeboOnline* busca alinhar-se com essas tendências e necessidades, fornecendo um site inovador que promove o compartilhamento de livros e a disseminação do conhecimento, contribuindo assim para a promoção da cultura da leitura.

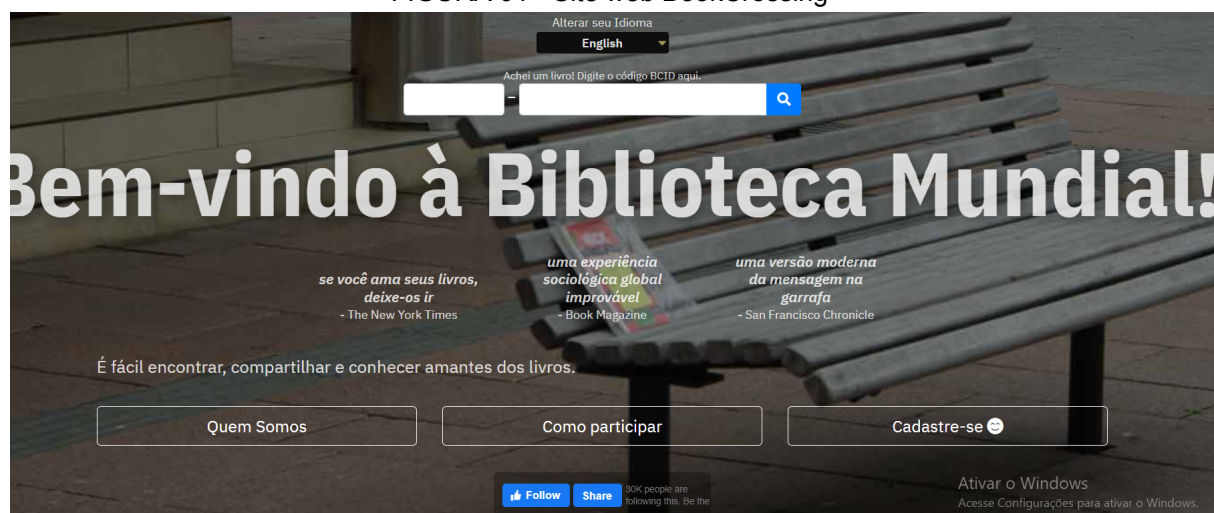
#### 2.1. APLICATIVOS SIMILARES

Neste subcapítulo são listados alguns aplicativos web com o objetivo similar ao SeboOnline, onde nos inspiramos e nos promoveram novas ideias.

### 2.1.1. BookCrossing

BookCrossing é uma comunidade global que permite registrar livros e liberá-los em locais públicos para que outros usuários os encontrem, leiam e depois os liberem novamente. A plataforma incentiva o compartilhamento e a disseminação de livros, rastreando seu movimento entre leitores ao redor do mundo.

FIGURA 01 - Site web BookCrossing



Fonte : BookCrossing

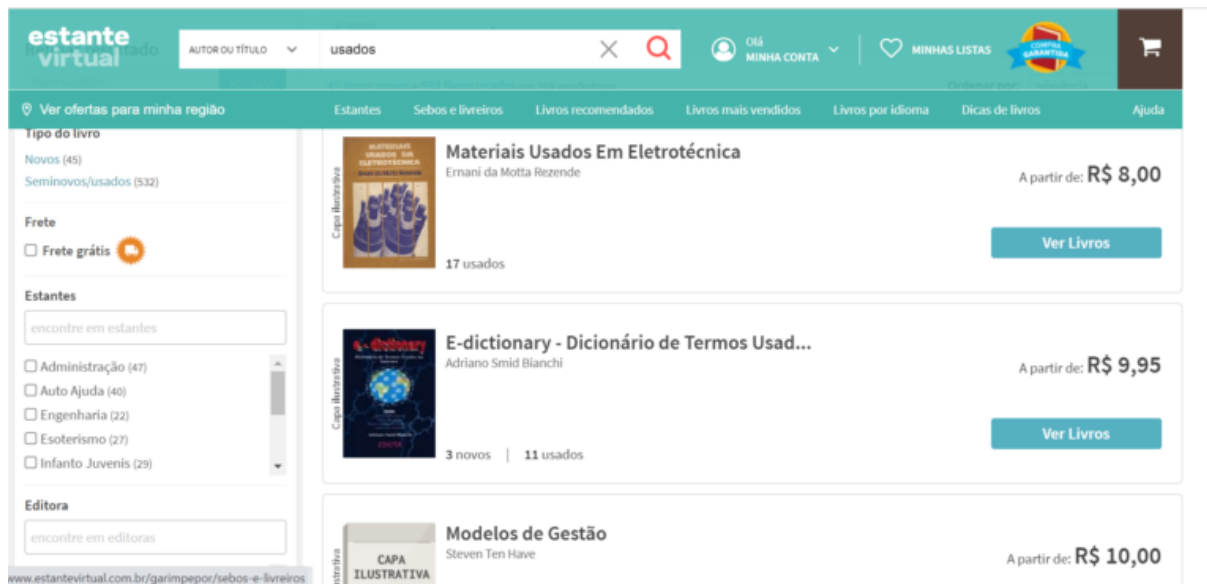
### 2.1.2. Estante Virtual

A [Estante Virtual](#) é uma plataforma digital especializada na venda e compra de livros, novos e usados. Nela, qualquer pessoa pode colocar à venda seus livros usados, bem como editoras, sebos e livreiros. Ela também exige um cadastro que pode ser feito como pessoa física ou jurídica.

Existe uma mensalidade em que é voltada para a comercialização exclusiva de livros. Essa é uma das vantagens, já que as pessoas que vão até o site são potenciais clientes que estão procurando por um acervo de livros.

Um ponto a se considerar é que o cadastro na Estante Virtual permite a disponibilização de até 2.000 livros, e isso pode gerar uma competição desvantajosa para os vendedores que contam com um acervo pequeno.

FIGURA 02 – Site Estante Virtual



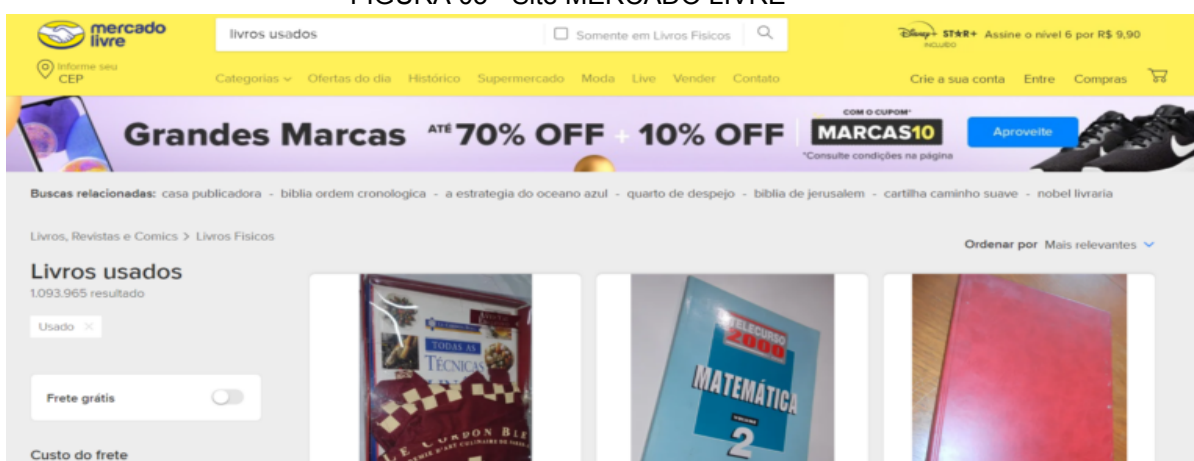
Fonte : EstanteVirtual

### 2.1.3. Mercado Livre

O Mercado Livre é um marketplace para a venda de qualquer produto, principalmente para os livros usados. Contar com inúmeros visitantes é um bom motivo para estar no Mercado Livre.

Existe a possibilidade de anunciar gratuitamente e pagar só quando a compra for realizada ou de fazer anúncios pagos, em que a plataforma vai potencializar o alcance das suas ofertas.

FIGURA 03 - Site MERCADO LIVRE



FONTE: Mercado Livre

#### 2.1.4. Comparação das funcionalidades das soluções similares

Com base nas análises dos sites listados anteriormente e de outros sites consultados, apresentamos na Figura 04 a tabela comparativa das funcionalidades que identificamos interessantes para o sistema aqui proposto.

FIGURA 04 - Comparação Sites

Funcionalidade	BookCrossing	Biblioteca do Estado de São Paulo	Amazon Kindle Unlimited	Biblioteca Solidária do Brasil	Estante Virtual	Mercado Livre
Empréstimo de livros físicos	X	-	-	X	X	X
Doação de livros físicos	X	-	-	X	X	X
Cadastro de bibliotecas	-	X	-	-	X	X
Classificação de livros por gênero	-	-	-	X	X	X
Notícias sobre literatura	-	-	-	X	X	X
Acesso a livros digitais	-	X	X	-	-	-
Leituras ilimitadas	-	-	X	-	-	-
Download de livros para leitura offline	-	-	X	-	-	-
Sistema de recomendações de livros	-	-	-	X	X	X
Ferramentas de análise de leitura	-	-	-	X	X	-

Fonte: Os Autores(2023)

### 3. MATERIAIS E MÉTODOS

O processo de desenvolvimento de software é um fator fundamental para o sucesso, ou não do produto final. A criação de programas e aplicativos apresenta complexidades, tanto quanto entender como funciona o processo a ser informatizado. Conhecer as ferramentas e metodologias de desenvolvimento de software ajuda (e muito!) na criação de produtos de qualidade, possibilitando que sejam finalizados dentro do prazo estabelecido e capazes de oferecer um retorno ao capital mobilizado.

este projeto web do SeboOnline, optamos por seguir as principais etapas do processo de desenvolvimento de software descritas por Pressman (Engenharia de Software, 2011), adaptadas às necessidades da equipe e incorporadas algumas práticas do desenvolvimento ágil:

- Levantamento e análise de requisitos
- Design do projeto
- Implementação
- Teste
- Documentação
- Debug, manutenção e atualização

#### 3.1. METODOLOGIA DE DESENVOLVIMENTO

Os métodos ágeis são uma abordagem alternativa ao modelo de gestão tradicional de projetos. Onde se tinha o desenvolvimento de produtos dividido por etapas bem definidas. Já os Métodos Ágeis possuem iterações curtas, onde o resultado é medido através de funcionalidades pronto.

O modelo de entrega ágil é baseado em ciclos iterativos e incrementais, o que traz flexibilidade e adaptabilidade, semelhante a metodologia espiral. Uma característica importante é a inspeção e adaptação dos ciclos e iterações, focados em gerar melhoria contínua para as equipes e processos.

Alguns autores propõem que os Métodos Ágeis são contra a documentação e burocracia, porém eles estão focados no que realmente irá agregar valor ao processo produtivo.

Ambientes ágeis de desenvolvimento normalmente são construídos por equipes que têm autonomia e que são capazes de se auto-organizar em busca de objetivos e metas.

Atualmente a utilização dos Métodos Ágeis é global. Isso pelo motivo das metodologias ágeis serem mais produtivas e eficazes do que os modelos tradicionais.

Entretanto, os métodos ágeis requerem experiência da equipe para evitar falhas de comunicação, retrabalho e foco nas funcionalidades e requisitos do sistema em desenvolvimento.

Por conseguinte, a seguir abordaremos as metodologias que utilizamos para a concretização do projeto.

### 3.2. KANBAN

Kanban, cartão em japonês, é o nome de uma metodologia de gestão de fluxo de trabalho. Seu uso pode receber variações conforme o segmento da empresa, mas de forma geral se utilizam cartões físicos ou digitais distribuídos em colunas que representam onde eles estão localizados no processo.

Cada cartão representa uma tarefa ou elemento que deve ser trabalhado. Também deve-se preencher com informações como nomes dos responsáveis, detalhamentos do que vai ser feito, prazos das micro-tarefas

Esse tipo de método permite otimizar a entrega de trabalhos que estejam divididos entre mais de um profissional ou lidar com projetos mais complexos.

No começo, era usado pela manufatura, mas foi abraçado pelas equipes de desenvolvimento de software Ágil e também acoplado como funcionalidade em plataformas como ERPs Industriais.

O Kanban foi criado pelo Taiichi Ohno em 1953 e faz parte do sistema Toyota de produção, onde tem um papel fundamental no [sistema de produção](#) puxada e no conceito [Just-in-time](#).

A história é curiosa: os japoneses foram aos Estados Unidos para conhecer as fábricas norte-americanas e se inspiraram nos supermercados da forma como conhecemos hoje, em que os consumidores circulavam com o carrinho e tinham livre acesso aos estoques de produtos.

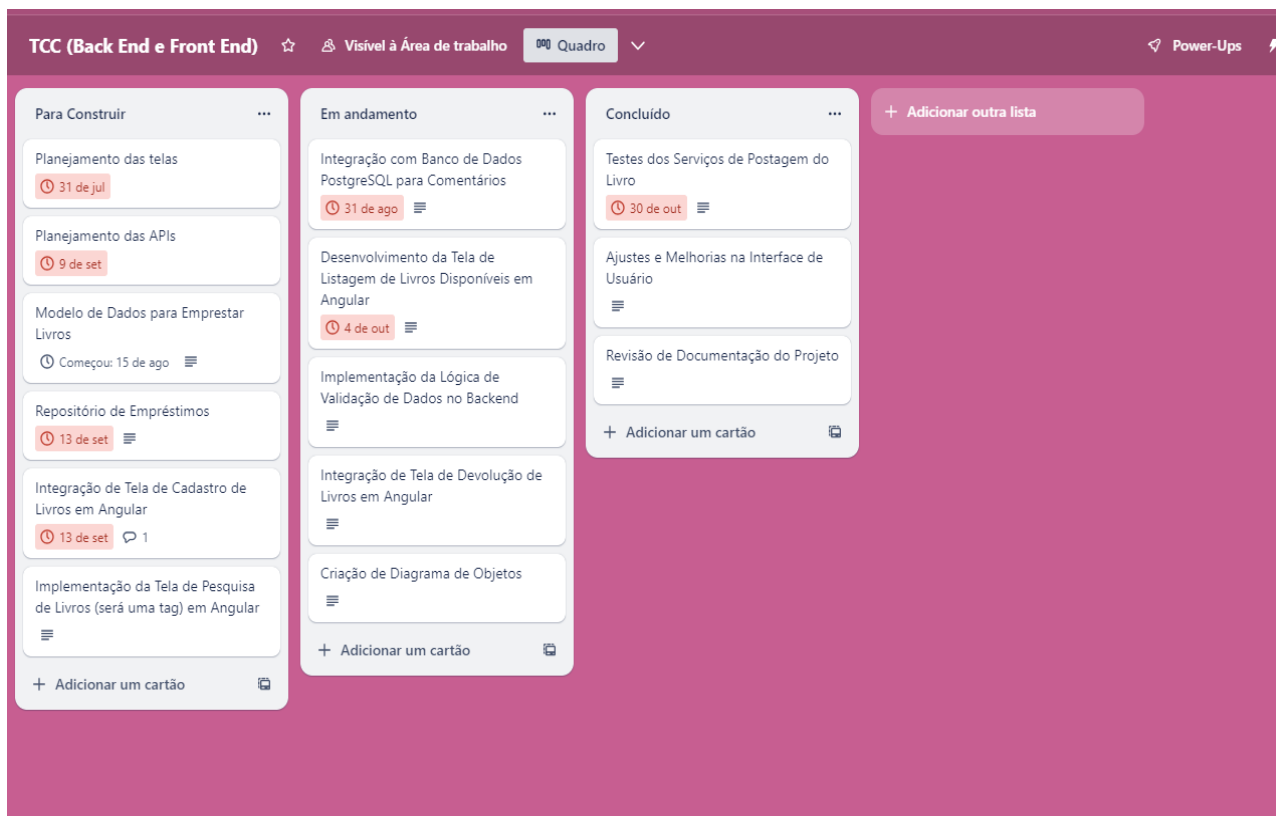
Na época, os mercados funcionavam como os almoxarifados, você entregava a lista e o funcionário do mercado (que parecia um almoxarife) ia pegar no estoque.

É até difícil imaginar hoje um mercado que funcione desta forma e imagino que no futuro vai ser difícil encontrar almoxarifados na forma como conhecemos hoje.

No Brasil o método começou a ser usado na década de 80 e também é conhecido como “Gestão Visual”, onde controla-se o estoque e o fluxo de peças utilizando cartões ou estoques intermediários visuais.

No nosso trabalho, utilizamos o Trello para organizarmos as nossas tarefas e implementações de funcionalidades.

FIGURA 05 – QUADRO KANBAN INICIAL



FONTE: Os Autores (2023).

### 3.3. SCRUM

Scrum é a resposta moderna para a gestão de projetos. É uma abordagem baseada em trabalho em equipe, colaboração e comunicação. Com isso, é possível gerar valor por meio de soluções adaptáveis para problemas complexos.

Em termos simples, trata-se de um conjunto de boas práticas que capacitam os times a trabalharem melhor em conjunto.

Ao ouvir falar deste método, é comum mencioná-lo como parte do framework ágil. O motivo é simples: ele é baseado em conceitos como ciclos de feedback, melhoria contínua e empirismo (ou seja, valoriza a experiência em si).

De acordo com o estudo State of Agile, esse é de longe o método ágil de gestão de projetos mais utilizado em todo o mercado corporativo.

Os principais componentes do Scrum incluem:

- Time Scrum: É a equipe que trabalha para entregar o produto final. Geralmente, é um grupo multidisciplinar e auto-organizado que trabalha em conjunto para alcançar os objetivos do projeto.
- Product Owner: É responsável por representar as necessidades dos stakeholders e clientes. Ele prioriza o backlog do produto, define as funcionalidades e decide o que deve ser feito no projeto.
- Scrum Master: É responsável por garantir que a equipe siga os princípios e práticas do Scrum. Ele remove obstáculos, facilita as reuniões e ajuda a equipe a operar de maneira eficaz.
- Backlog do Produto: É uma lista de todos os itens necessários para o produto, organizados por prioridade. É mantido e priorizado pelo Product Owner.
- Backlog do Sprint: É uma lista de itens selecionados pelo time Scrum para serem completados durante o sprint atual. Esses itens são retirados do backlog do produto e devem ser entregues ao final do sprint.
- Sprint: É um período de tempo fixo, geralmente de 1 a 4 semanas, durante o qual o time trabalha para entregar um incremento de produto funcional e testável.

### 3.4. ADAPTAÇÃO DA METODOLOGIA SCRUM

Conforme apresentado e explicitado acima, o Scrum segue algumas práticas tais como reuniões diárias (daily), retrospectivas , envolvendo todo um cerimonial , e devido a nossa rotina de trabalho e obrigações tivemos de adaptar a metodologia SCRUM de acordo com a nossa realidade , onde nos encontrávamos a cada dois dias às 18:00 para reunião e codificação, e aos sábados e domingos fazíamos o levantamento dos requisitos faltantes e ou codificação do sistema em si.

Quanto aos papéis durante todo o desenvolvimento, todos os integrantes assumiram o papel de desenvolvedor, e às vezes de Product Owner, o Scrum master ficou a cargo do Felipe fazer a gestão.

Scrum Master : Felipe

Product Owner : Dieval

Desenvolvedores:

Laerte

Emanuel

Felipe

### 3.5. FERRAMENTAS DE DESENVOLVIMENTO

#### 3.5.1 BALSAMIQ

O Balsamiq é uma ferramenta de software de prototipagem rápida e criação de wireframes, bastante popular no campo do design de interface do usuário (UI) e experiência do usuário (UX). Essa ferramenta é utilizada para esboçar e visualizar rapidamente ideias para interfaces de aplicativos, sites e outras soluções digitais.

O principal objetivo do Balsamiq é permitir que os designers e desenvolvedores criem esboços e mockups de interfaces de forma rápida e simples, sem se preocupar com detalhes de design complexos. Ele se destaca por sua

interface simples e intuitiva, que permite aos usuários arrastar e soltar componentes pré-definidos para criar esboços de páginas.

Principais características do Balsamiq:

- Biblioteca de Componentes Pré-definidos: Oferece uma ampla variedade de elementos de interface (botões, caixas de texto, menus, etc.) para arrastar e soltar na tela.
- Interface Simples e Intuitiva: Fácil de usar, sem necessidade de habilidades avançadas em design.
- Prototipagem Rápida: Permite criar esboços funcionais rapidamente, facilitando a comunicação de ideias e conceitos.
- Facilidade de Colaboração: Possibilita compartilhar e colaborar em projetos de forma eficiente.
- Foco na Funcionalidade, não no Design Detalhado: O Balsamiq prioriza a disposição e a funcionalidade dos elementos, não detalhes estilísticos.
- Adaptabilidade: Útil em todas as etapas do processo de design, desde esboços iniciais até a fase de refinamento.

Por ser uma ferramenta voltada para a criação de wireframes e protótipos, o Balsamiq é valorizado por sua capacidade de ajudar os profissionais de design e desenvolvimento a visualizar e comunicar ideias de forma rápida e eficiente, antes de iniciar o trabalho de codificação e design detalhado.

### 3.5.2 ASTAH

O Astah, anteriormente conhecido como Jude, é uma ferramenta de modelagem UML (Unified Modeling Language) é um software de diagramação utilizado principalmente para projetar e visualizar sistemas orientados a objetos e processos de desenvolvimento de software.

Essa ferramenta é usada por analistas de sistemas, engenheiros de software, desenvolvedores e outros profissionais da área de tecnologia para criar diferentes tipos de diagramas que ajudam a entender, comunicar e planejar sistemas de software complexos.

Principais recursos do Astah:

- Modelagem UML: Oferece suporte abrangente para a criação de diferentes tipos de diagramas UML, como diagramas de classe, diagramas de sequência, diagramas de caso de uso, diagramas de atividades, entre outros.
- Facilidade de Uso: Interface intuitiva e amigável, facilitando a criação e edição dos diagramas.
- Geração de Código: Permite a geração de código a partir dos diagramas UML criados, agilizando o processo de desenvolvimento.
- Colaboração: Possibilita a colaboração entre os membros da equipe, permitindo compartilhar e trabalhar em projetos de forma colaborativa.
- Compatibilidade: Suporta a importação e exportação de diferentes formatos de arquivo, facilitando a integração com outras ferramentas e ambientes de desenvolvimento.

No contexto do Trabalho de Conclusão de Curso (TCC) de SeboOnline, utilizamos o Astah como uma ferramenta fundamental para criar e visualizar os diferentes tipos de diagramas essenciais no processo de desenvolvimento do nosso software.

Diagrama de Casos de Uso:

No início do projeto, utilizamos o Astah para elaborar os diagramas de casos de uso. Esses diagramas foram cruciais para identificar e descrever as interações entre os usuários e o sistema, auxiliando na compreensão dos requisitos funcionais e na definição do escopo do projeto.

Diagrama de Classes:

O Astah foi empregado na criação dos diagramas de classes, permitindo a representação visual das entidades e relacionamentos do sistema proposto. Esses

diagramas foram fundamentais para modelar a estrutura do software, exibindo as classes, seus atributos, métodos e associações, oferecendo uma visão clara da organização do código.

Diagrama de Objetos e de Sequência:

Durante a fase de design e planejamento detalhado, o Astah foi utilizado para criar os diagramas de objetos e de sequência. Esses diagramas foram valiosos para representar as interações entre os objetos do sistema e ilustrar visualmente o fluxo de mensagens entre eles, ajudando a compreender melhor o comportamento dinâmico do software.

Ao utilizar o Astah para criar esses diferentes tipos de diagramas, pudemos não apenas visualizar e compreender melhor a estrutura e o comportamento do sistema proposto, mas também comunicar de forma mais eficaz as ideias e decisões de design para os membros da equipe e para os orientadores do TCC.

A ferramenta foi essencial para produzir representações visuais claras e coerentes, o que contribuiu significativamente para o desenvolvimento do TCC, desde a fase inicial de planejamento até a implementação e documentação do software.

### 3.5.3 IDE INTELLIJ IDEA COMMUNITY

O IntelliJ IDEA Community é uma versão gratuita e de código aberto do popular ambiente de desenvolvimento integrado (IDE) IntelliJ IDEA, desenvolvido pela JetBrains. Este IDE é projetado especialmente para programadores que trabalham em projetos de software usando várias linguagens de programação, como Java, Kotlin, Groovy, Scala, entre outras.

As principais características do IntelliJ IDEA Community incluem:

- Suporte multiplataforma: Ele é compatível com Windows, macOS e Linux, permitindo que os desenvolvedores o utilizem na plataforma de sua preferência.

- Edição de código inteligente: Oferece recursos de auxílio ao programador, como destaque de sintaxe, conclusão automática de código, refatoração de código, navegação inteligente, entre outros.
- Ferramentas específicas para linguagens de programação: Fornece ferramentas e assistência específicas para várias linguagens de programação, facilitando o desenvolvimento e a depuração de aplicativos.
- Integração com tecnologias populares: Possui integração com uma variedade de tecnologias e frameworks, como Spring, Android, JUnit, Git, entre outros, facilitando o desenvolvimento em diferentes ambientes e plataformas.
- O IntelliJ IDEA Community é uma opção muito útil para desenvolvedores que buscam um ambiente de desenvolvimento robusto e repleto de recursos, especialmente para projetos de código aberto e desenvolvimento de software em várias linguagens de programação.

#### 3.5.4 VS CODE

O Visual Studio Code (também conhecido como VS Code) é um ambiente de desenvolvimento leve, gratuito e altamente personalizável desenvolvido pela Microsoft. Ele é amplamente utilizado por desenvolvedores devido à sua interface amigável, extensibilidade e suporte a uma ampla variedade de linguagens de programação.

Para o desenvolvimento front-end com Angular, o VS Code oferece várias vantagens:

- Extensões para Angular: O VS Code possui extensões específicas para o desenvolvimento com Angular, oferecendo suporte a recursos como IntelliSense (completar automaticamente o código), realce de sintaxe, snippets e depuração específica para aplicativos Angular.
- Depuração Integrada: O VS Code oferece ferramentas de depuração integradas que podem ser usadas para encontrar e corrigir bugs no código Angular de maneira eficiente.

- Terminal Integrado: Ele possui um terminal integrado que permite aos desenvolvedores executar comandos diretamente no VS Code, facilitando a execução de scripts, testes e comandos de gerenciamento de pacotes.
- Git Integrado: Oferece suporte nativo ao Git, permitindo aos desenvolvedores gerenciar facilmente o controle de versão e realizar operações como commit, push, pull, etc., diretamente do VS Code.
- Extensibilidade e Personalização: O VS Code é altamente personalizável por meio de uma variedade de extensões disponíveis na sua marketplace. Isso permite que os desenvolvedores adicionem funcionalidades e ajustem o ambiente de acordo com suas preferências e necessidades específicas.
- Performance e Leveza: O VS Code é conhecido por ser leve em comparação com outros IDEs, o que significa que consome menos recursos do sistema e responde de maneira rápida, tornando-o ideal para o desenvolvimento em Angular, onde a agilidade é importante.
- Suporte à Comunidade: Como é amplamente adotado pela comunidade de desenvolvimento, você pode encontrar uma ampla variedade de recursos, tutoriais e suporte disponíveis online para ajudar no desenvolvimento com Angular usando o VS Code.

### 3.5.5 POSTMAN

O Postman é uma plataforma amplamente utilizada para testar, desenvolver e documentar APIs. Ele oferece uma interface amigável que permite aos desenvolvedores enviar solicitações HTTP para APIs, testar diferentes endpoints, gerenciar variáveis de ambiente, automatizar testes e colaborar em equipes de desenvolvimento.

Quanto ao uso específico com APIs Spring Boot, o Postman oferece várias vantagens:

- Criação e Teste de Requisições RESTful: Com o Postman, os desenvolvedores podem criar e enviar facilmente solicitações HTTP (como GET, POST, PUT, DELETE, etc.) para os endpoints da API Spring Boot. Isso

permite testar e verificar o comportamento dos endpoints da API de maneira eficiente.

- **Ambientes e Variáveis:** O Postman permite configurar ambientes e variáveis, facilitando a manipulação de diferentes ambientes (como desenvolvimento, produção, teste) e a reutilização de valores em várias solicitações. Isso é útil ao testar diferentes configurações de sua API Spring Boot.
- **Testes Automatizados:** O Postman permite criar testes automatizados para suas APIs usando a linguagem JavaScript. Isso permite verificar automaticamente a resposta da API, validar dados e garantir que a API Spring Boot esteja funcionando conforme o esperado.
- **Documentação e Colaboração:** Ele oferece a possibilidade de gerar documentação automática da API com base nas solicitações feitas. Isso facilita a compreensão da estrutura da API Spring Boot e suas funcionalidades. Além disso, o Postman possui recursos de colaboração que permitem que equipes trabalhem juntas, compartilhem coleções de APIs e colaborem nos testes.
- **Monitoramento e Histórico:** O Postman oferece recursos para monitorar o desempenho das APIs, permitindo rastrear e analisar o histórico de solicitações e respostas. Isso é útil para identificar possíveis problemas de desempenho ou erros na API Spring Boot.

### 3.5.6 POSTGRESQL

O PostgreSQL é um sistema de gerenciamento de banco de dados relacional (RDBMS) de código aberto e gratuito. É um dos RDBMS mais populares do mundo, sendo usado por empresas de todos os tamanhos, desde startups a grandes corporações.

Vantagens ao usá-lo com Spring:

- **Compatibilidade com o Spring:** O Spring Framework tem um amplo suporte para o PostgreSQL. O Spring Data JPA e o Spring JDBC oferecem recursos para interagir com o PostgreSQL de maneira simplificada, facilitando a persistência de dados e operações de banco de dados.
- **Suporte a Recursos Avançados:** O PostgreSQL é conhecido por ser um banco de dados relacional poderoso e altamente extensível. Oferece suporte a recursos avançados, como consultas complexas, gatilhos (triggers), funções armazenadas, índices avançados, chaves estrangeiras, além de tipos de dados personalizados e extensíveis.
- **Confiabilidade e Desempenho:** O PostgreSQL é reconhecido por sua confiabilidade, estabilidade e conformidade com padrões ANSI SQL. Além disso, é conhecido por oferecer bom desempenho, especialmente quando configurado corretamente e otimizado para cargas de trabalho específicas.
- **Suporte a JSON e NoSQL:** O PostgreSQL possui suporte integrado para operações relacionadas a JSON, permitindo armazenar, consultar e manipular dados no formato JSON diretamente no banco de dados. Isso é particularmente útil para aplicativos modernos que utilizam estruturas de dados flexíveis, como aplicações NoSQL.
- **Comunidade Ativa e Documentação Abundante:** O PostgreSQL possui uma comunidade ativa de usuários e uma documentação extensa. Isso significa que os desenvolvedores que utilizam o Spring com o PostgreSQL podem encontrar facilmente recursos, tutoriais, fóruns e soluções para problemas comuns.
- **Licença Open Source:** O PostgreSQL é um banco de dados de código aberto, o que significa que é gratuito para uso e oferece liberdade para os desenvolvedores utilizarem, modificarem e distribuírem conforme necessário, sem custos de licenciamento.

### 3.5.7 BROWSER MOZILLA FIREFOX

O Mozilla Firefox, ou simplesmente Firefox, é um navegador web gratuito e de código aberto desenvolvido pela Mozilla Foundation e sua subsidiária, a Mozilla Corporation. Ele usa o mecanismo de renderização Gecko para exibir páginas da web, que implementam os padrões da web atuais e esperados. O Firefox está disponível para os sistemas operacionais Windows, macOS, Linux, Android e iOS.

Principais recursos do Firefox:

- Código aberto: O Firefox é de código aberto, o que significa que seu código está livremente disponível para qualquer pessoa inspecionar, modificar e redistribuir. Isso torna o Firefox um navegador mais transparente e responsável do que muitos de seus concorrentes.
- Focado na privacidade: O Firefox tem um forte foco na privacidade e inclui uma série de recursos para proteger os dados dos usuários. Por exemplo, o Firefox bloqueia rastreadores de terceiros por padrão e tem um modo de navegação privada que apaga todo o histórico de navegação e cookies quando o navegador é fechado.
- Personalizável: O Firefox é altamente personalizável, com uma ampla gama de temas, extensões e complementos disponíveis. Isso permite aos usuários personalizar sua experiência de navegação e adicionar novos recursos ao navegador.
- Rápido e eficiente: O Firefox é um navegador rápido e eficiente, e consistentemente se classifica entre os melhores navegadores em termos de desempenho.

Benefícios do uso do Firefox:

- Privacidade aprimorada: Os recursos de privacidade do Firefox podem ajudar a proteger seus dados de serem rastreados e coletados por anunciantes e outras partes interessadas de terceiros.
- Maior controle sobre sua experiência de navegação: As opções de personalização do Firefox permitem que você personalize o navegador para atender às suas necessidades e preferências específicas.

- Acesso a uma ampla gama de extensões e complementos: O Firefox tem uma grande comunidade de desenvolvedores que criam extensões e complementos que podem adicionar novas funcionalidades ao navegador.
- Suporte a software de código aberto: Ao usar o Firefox, você está apoiando o desenvolvimento de software de código aberto.

No geral, o Firefox é um navegador web poderoso e versátil que oferece uma série de vantagens sobre seus concorrentes.

Nesse projeto utilizamos de forma majoritária o Firefox para exibição do lado cliente das telas nas quais desenvolvemos em angular.

### 3.6. TECNOLOGIAS DE DESENVOLVIMENTO

O desenvolvimento de software é uma área que está em constante evolução, com a constante introdução de novas tecnologias e frameworks. Para o desenvolvimento de um sistema de informação, é importante escolher as tecnologias que melhor atendam às necessidades do projeto.

Neste projeto de TCC, foi escolhido utilizar a tecnologia Java Spring no backend com Angular no frontend e banco de dados PostgreSQL. Essas tecnologias foram escolhidas por diversos motivos, incluindo:

- Java Spring é um framework de desenvolvimento web completo e robusto, que oferece uma ampla gama de recursos e funcionalidades.
- Angular é um framework de desenvolvimento web SPA (Single Page Application) moderno e eficiente.
- APIs em Spring são uma forma de exposição de dados e funcionalidades de um sistema de informação, que facilitam a integração com outros sistemas.
- PostgreSQL é um banco de dados relacional robusto e escalável.

A seguir, serão apresentados os principais motivos para a escolha de cada uma dessas tecnologias.

### 3.6.1. Framework Spring

O framework Spring é um framework de desenvolvimento web completo e robusto, que oferece uma ampla gama de recursos e funcionalidades. Entre os principais recursos oferecidos pelo framework Spring, destacam-se:

- Injeção de dependência: O Spring permite que os objetos de um sistema sejam instanciados e gerenciados automaticamente, o que facilita o desenvolvimento e a manutenção do código.
- MVC: O Spring implementa o padrão MVC (Model-View-Controller), que facilita o desenvolvimento de sistemas de informação modulares e reutilizáveis.
- Autenticação e autorização: Oferece recursos para autenticação e autorização de usuários, o que é essencial para a segurança de um sistema de informação.
- Testes unitários: O framework Spring oferece suporte a testes unitários, o que facilita a verificação da qualidade do código.

### 3.6.2. Angular

Angular é um framework de desenvolvimento web SPA (Single Page Application) moderno e eficiente. Entre os principais recursos oferecidos por Angular, destacam-se:

- Componentes: Angular é baseado no conceito de componentes, que são unidades de código reutilizáveis que podem ser combinadas para criar interfaces de usuário complexas.
- Data binding: Angular oferece recursos de data binding que permitem sincronizar os dados do modelo com a interface de usuário.
- Routing: Angular oferece recursos de routing que permitem navegar entre diferentes páginas de um sistema de informação.
- Testes: Angular oferece suporte a testes, o que facilita a verificação da qualidade do código.

### 3.6.3. APIs em Spring

APIs em Spring são uma forma de exposição de dados e funcionalidades de um sistema de informação. APIs são essenciais para a integração com outros sistemas, como aplicações móveis, sistemas de BI (Business Intelligence) e sistemas ERP (Enterprise Resource Planning).

O Spring oferece uma ampla gama de recursos para o desenvolvimento de APIs, incluindo:

- Padrões de design: Spring implementa diversos padrões de design, que facilitam o desenvolvimento de APIs robustas e escaláveis.
- Documentação: Oferece recursos para a geração de documentação automática para APIs, o que facilita a sua utilização por outros sistemas.
- Testes: Oferece suporte a testes, o que facilita a verificação da qualidade das APIs.

### 3.6.4. PostgreSQL

PostgreSQL é um banco de dados relacional robusto e escalável. PostgreSQL oferece uma ampla gama de recursos e funcionalidades, incluindo:

- SQL: PostgreSQL é um banco de dados SQL, o que facilita o seu uso por desenvolvedores familiarizados com esse padrão.
- Extensibilidade: PostgreSQL é um banco de dados extensível, o que permite a integração de novos recursos e funcionalidades.
- Recursos avançados: Oferece recursos avançados, como transações ACID, suporte a índices geográficos e suporte a colunas com valores nulos.

A escolha das tecnologias apresentadas neste tcc foi baseada nos requisitos e necessidades do projeto. Acreditamos que essas tecnologias são as mais adequadas para o desenvolvimento de um sistema de informação robusto, eficiente e escalável voltado para a web.

### 3.6.5. Git

O Git é um sistema de controle de versão distribuído usado principalmente para rastrear mudanças no código-fonte durante o desenvolvimento de software. Foi criado por Linus Torvalds em 2005 e se tornou amplamente utilizado devido à sua eficiência, velocidade e flexibilidade. O Git permite que várias pessoas trabalhem em um mesmo projeto simultaneamente, acompanhando as mudanças feitas em arquivos, revertendo para versões anteriores, ramificando o código para o desenvolvimento paralelo e mesclando as alterações de diferentes colaboradores.

Características principais do Git:

- **Distribuído:** Cada usuário tem uma cópia local do repositório inteiro, o que facilita o trabalho offline e permite que as alterações sejam feitas localmente antes de serem enviadas para o repositório remoto.
- **Rastreamento de Alterações:** O Git mantém um histórico completo de todas as alterações feitas nos arquivos, permitindo visualizar quem fez quais alterações e quando.
- **Branching e Merging:** Permite que os desenvolvedores criem ramificações (branches) para trabalhar em recursos ou correções de bugs separadamente do código principal (branch principal ou master), facilitando o gerenciamento de diferentes versões do código. Depois, é possível mesclar (merge) as mudanças de volta ao branch principal.
- **Velocidade e Desempenho:** O Git é conhecido por sua rapidez no gerenciamento de grandes projetos, tornando mais eficiente o processo de commit, merge e outras operações.

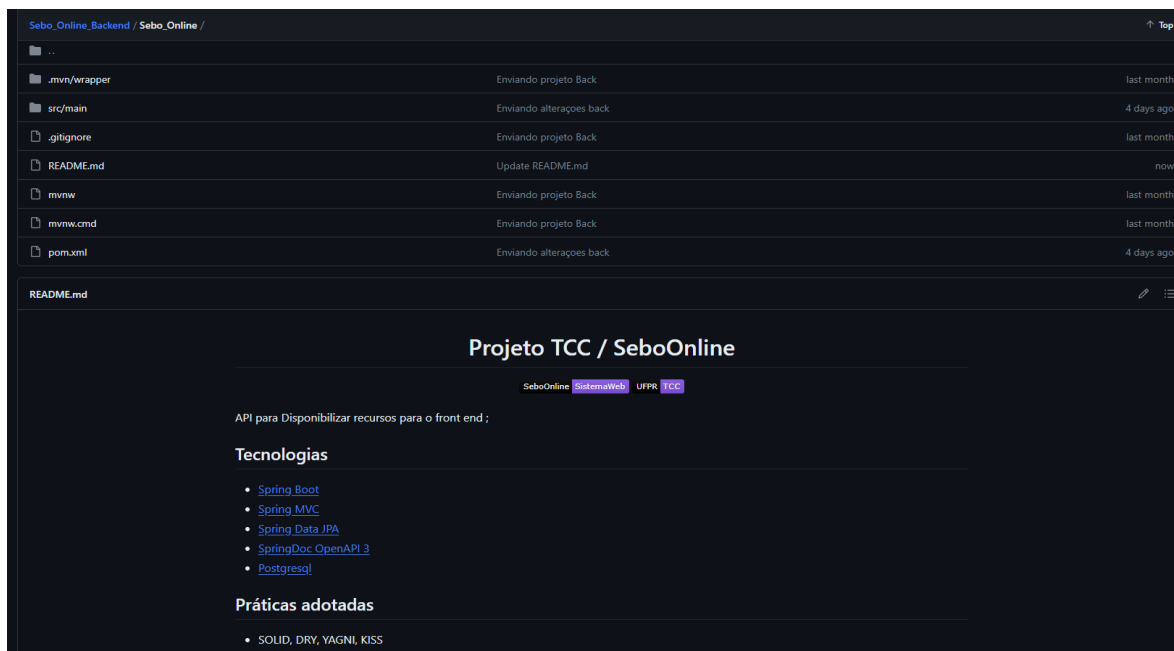
### 3.6.6. GitHub

GitHub é uma plataforma de hospedagem baseada na web para repositórios Git. Ele oferece funcionalidades adicionais ao Git, como um ambiente colaborativo para desenvolvimento de software, controle de acesso, rastreamento de problemas (issues), gerenciamento de projetos e integração contínua.

#### Características do GitHub:

- Hospedagem de Repositórios: Permite que os desenvolvedores hospedem seus repositórios Git remotamente, facilitando o compartilhamento do código com outros colaboradores.
- Colaboração: Oferece recursos para que várias pessoas possam trabalhar juntas em projetos, como solicitações de pull (pull requests) para revisar e mesclar alterações de branches diferentes, ferramentas de discussão e revisão de código.
- Rastreamento de Problemas: Permite que os usuários relatem problemas, solicitem novos recursos e acompanhem o progresso das tarefas por meio de issues.
- Integração Contínua e Implantação Contínua: Integração com várias ferramentas de CI/CD (Continuous Integration/Continuous Deployment), facilitando a automação de testes e implantação do código.
- Comunidade e Social: Funciona como uma rede social para desenvolvedores, permitindo que eles sigam outros usuários, contribuam para projetos de código aberto e participem de discussões sobre tópicos relacionados ao desenvolvimento de software.

FIGURA 06 - Visão geral Projeto no GITHUB



FONTE: Os Autores (2023)

### 3.6.7. Versões das tecnologias utilizadas

Tecnologia	Versão	Descrição
Frontend		
Angular	14.0.0	Framework de desenvolvimento web SPA (Single Page Application) moderno e eficiente.
Node.js	18.16.0	Ambiente de execução para JavaScript fora do navegador.
NPM	9.5.2001	Gerenciador de pacotes para Node.js.
Git	2.36.0	Sistema de controle de versão.
Backend		
Java	JDK 18.0.2	Linguagem de programação orientada a objetos.
Spring	5.3.18	Framework de desenvolvimento web completo e robusto.
Tomcat	10.0.16	Container de servlets para Java.
Hibernate	6.0.5.Final	Framework de persistência ORM (Object-Relational Mapping).
PostgreSQL	14.2	Banco de dados relacional robusto e escalável.

FIGURA 07 - TECNOLOGIAS UTILIZADAS PARA O SEBOONLINE

FONTE:Os Autores (2023)

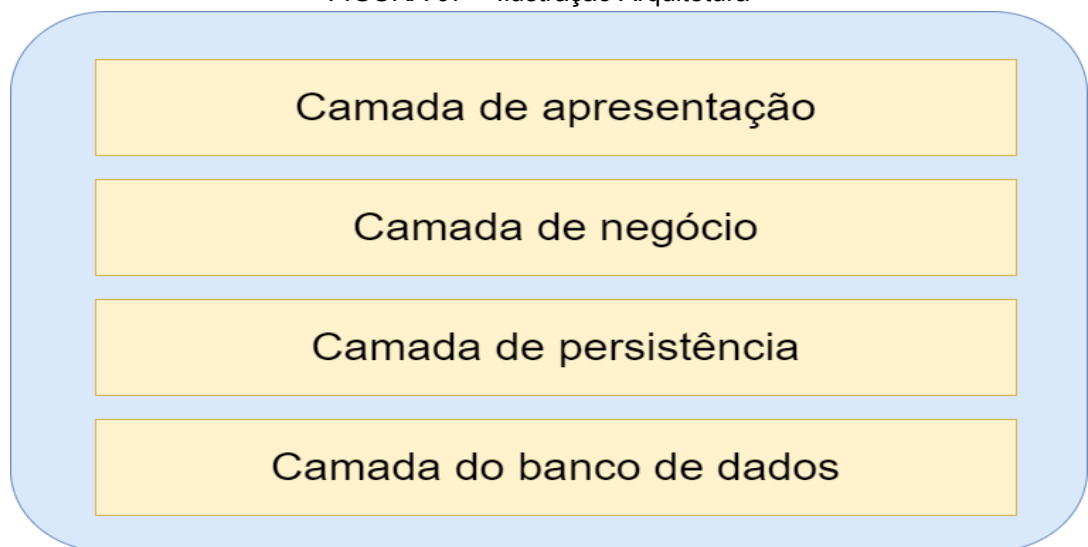
## 3.7 ARQUITETURA EM CAMADAS

A arquitetura em camadas é um estilo de arquitetura de software que divide um sistema em camadas, cada uma com uma responsabilidade específica. Essa abordagem permite que os sistemas sejam projetados e desenvolvidos de forma modular e reutilizável.

### 3.7.1. Topologia

A topologia desse estilo de arquitetura pode ser representada da seguinte forma:

FIGURA 07 - Ilustração Arquitetura



FONTE: DesignPattern(2021)

Essa arquitetura é comumente aplicada em projetos usando o framework Spring (ou qualquer outro framework similar) para separar e organizar diferentes responsabilidades dentro do código. Aqui está uma explicação geral de cada componente na arquitetura em camadas:

- DTO (Data Transfer Object): São objetos utilizados para transferir dados entre diferentes subsistemas do software. Eles são frequentemente usados para enviar dados entre o backend e o frontend ou entre diferentes partes do sistema.
- Services: Os Services (ou Services Layer) representam a lógica de negócios da aplicação. Eles contêm métodos que implementam a lógica de processamento e regras de negócio. Essa camada é responsável por orquestrar as operações e chamar os métodos apropriados nos Repositories.
- Repositories: Os Repositories são responsáveis pela comunicação com a fonte de dados, como um banco de dados. Eles encapsulam a lógica para recuperar, armazenar, atualizar e excluir dados do banco de dados ou de outras fontes de persistência.
- Controllers: Os Controllers são responsáveis por lidar com as requisições HTTP, receber dados do cliente, chamar os métodos apropriados nos Services e retornar as respostas adequadas ao cliente. Eles atuam como pontos de entrada para a aplicação.
- Model: Os Models representam a estrutura de dados da aplicação, incluindo entidades, objetos de dados e outras classes relacionadas à representação dos dados.
- Utils: A pasta Utils geralmente contém classes utilitárias que fornecem funcionalidades genéricas ou auxiliares para outras partes da aplicação.

### 3.8. INFRAESTRUTURA DE DESENVOLVIMENTO

Para o projeto do sistema web SeboOnline utilizamos nossas máquinas pessoais ,nos quais foram utilizados 3 notebooks aos quais possuem as configurações:

#### **Notebook 1:**

MacBook Air 2017:

- Processador: Intel Core i5 de 7<sup>a</sup> geração
- Memória RAM: 8 GB
- Armazenamento: 128 GB SSD
- Tela: Retina de 13,3 polegadas com resolução de 2560 x 1600 pixels
- Câmera: 720p
- Bateria: até 12 horas de duração

#### **Notebook 2:**

Dell Latitude 3420

- Processador: Intel Core i5 de 11<sup>a</sup> geração
- Memória RAM: 16 GB
- Armazenamento: 256 GB SSD
- Tela: de 14 polegadas com resolução de (1920 x 1080)
- Câmera: 720p
- Bateria: até 8 horas de duração

#### **Notebook 3:**

Dell G15

- Processador: Intel Core i7 de 11<sup>a</sup> geração
- Memória RAM: 16 GB
- Armazenamento: 512 GB SSD
- Tela: de 15.6" polegadas com resolução de (1920 x 1080)
- Câmera: 720p
- Bateria: até 6 horas de duração

## 4. RESULTADO: PROJETO E IMPLEMENTAÇÃO DO SISTEMA SEBOONLINE

O projeto SeboOnline é um sistema web para facilitar a interação das pessoas que queiram manter ou ampliar o hábito da a leitura, de forma mais sustentável, através de empréstimos, permuta, doações ou venda e revenda das obras literarias com a comunidade Na seção a seguir apresentamos a estrutura do sistema e as descrições das respectivas funcionalidades.

### 4.1 ARQUITETURA DO SOFTWARE

No nosso sistema optamos em empregar a arquitetura em camadas para facilitar o manuseio, correções de bugs e facilitar atualizações ou mudanças no código.

O SeboOnline foi desenvolvido em uma arquitetura multicamada, onde cada camada desempenha um papel específico na estrutura e funcionamento do sistema, são elas:

- **Controllers:** Do padrão de projeto Model-View-Controller (MVC), essa camada atua como ponto de entrada para o sistema, lidando com as requisições dos clientes(angular) e direcionando o fluxo de dados para os serviços correspondentes. Os controllers são responsáveis por receber e processar requisições HTTP, interagindo com os serviços para executar operações e retornando as respostas adequadas aos usuários.
- **Services:** A camada de serviços contém a lógica de negócios do sistema, ela difere ligeiramente da camada “modelo” do padrão MVC, por ser desenvolvida na forma de serviços, seguindo o modelo oferecido pelo Spring boot. Nesta camada são implementados os serviços que executam as operações específicas, como manipulação de dados, regras de negócio e processamento de informações. Nesta camada são realizadas as operações intermediárias entre os controllers e os repositories.
- **Repositories:** A camada de repositories é a camada de persistência dos dados no sistema, é a camada responsável pela interação com o Sistema de

Gerenciamento de Banco de Dados (SGBD) PostgreSQL. Ela encapsula a lógica de acesso e manipulação dos dados armazenados, fornecendo métodos para recuperar, armazenar, atualizar e deletar informações do banco de dados.

- **Models e DTOs:** As camadas de Models e DTOs representam a estrutura de dados da aplicação. Os Models são responsáveis por definir a estrutura dos objetos de dados, enquanto os DTOs (Data Transfer Objects) são utilizados para transferir dados entre diferentes partes do sistema, contribuindo para a integridade e consistência das informações.

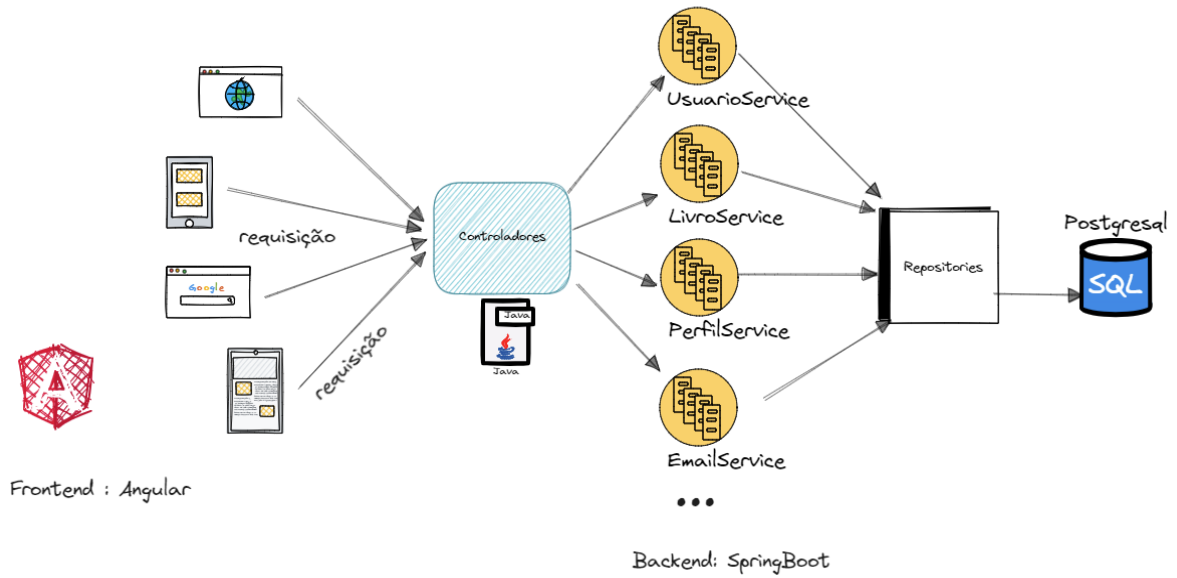
A utilização dessa arquitetura em camadas no SeboOnline não apenas promove uma melhor organização do código-fonte, mas também facilita a manutenção, escalabilidade e colaboração entre os membros da equipe de desenvolvimento. Esta estrutura proporciona uma clara separação de responsabilidades, permitindo o desenvolvimento ágil e eficiente do sistema.

Na sequência desse relatório mostraremos detalhadamente cada uma das camadas da arquitetura em camadas adotada no SeboOnline, destacando sua importância, inter-relações e contribuições para o funcionamento do sistema como um todo.

As Figuras 08 e 09, respectivamente, demonstram o modelo de comunicação baseado em requisição e resposta do protocolo HTTP, interagindo com as camadas e componentes do sistema, da forma que o implementamos.

Requisição:

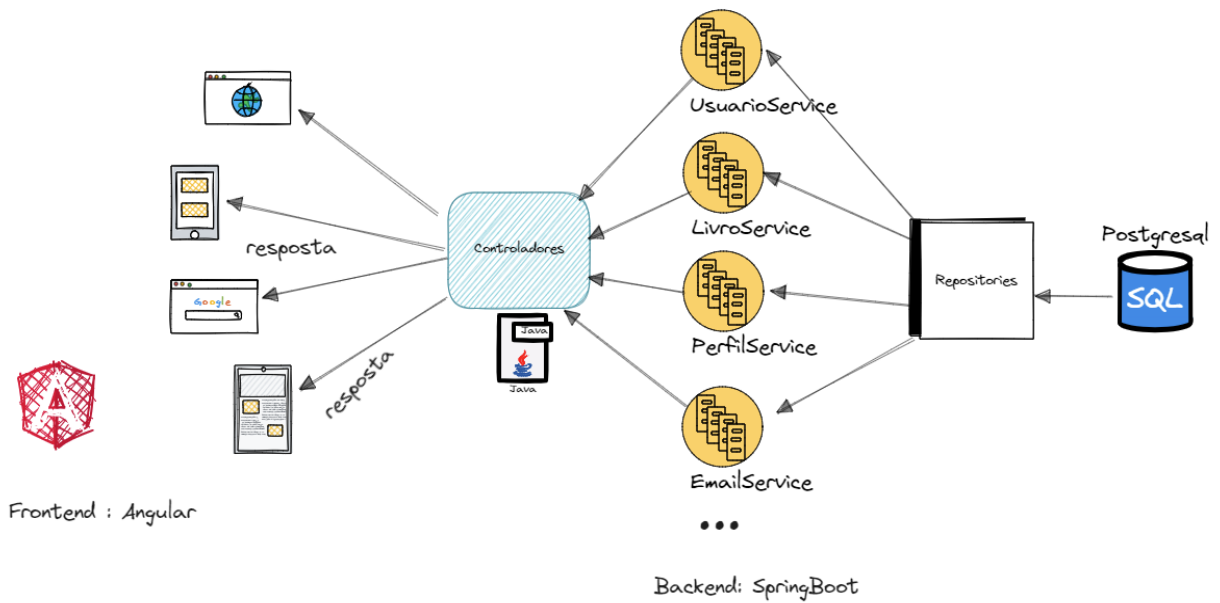
FIGURA 08 - Fluxo de Comunicação



FONTE: Os Autores(2023)

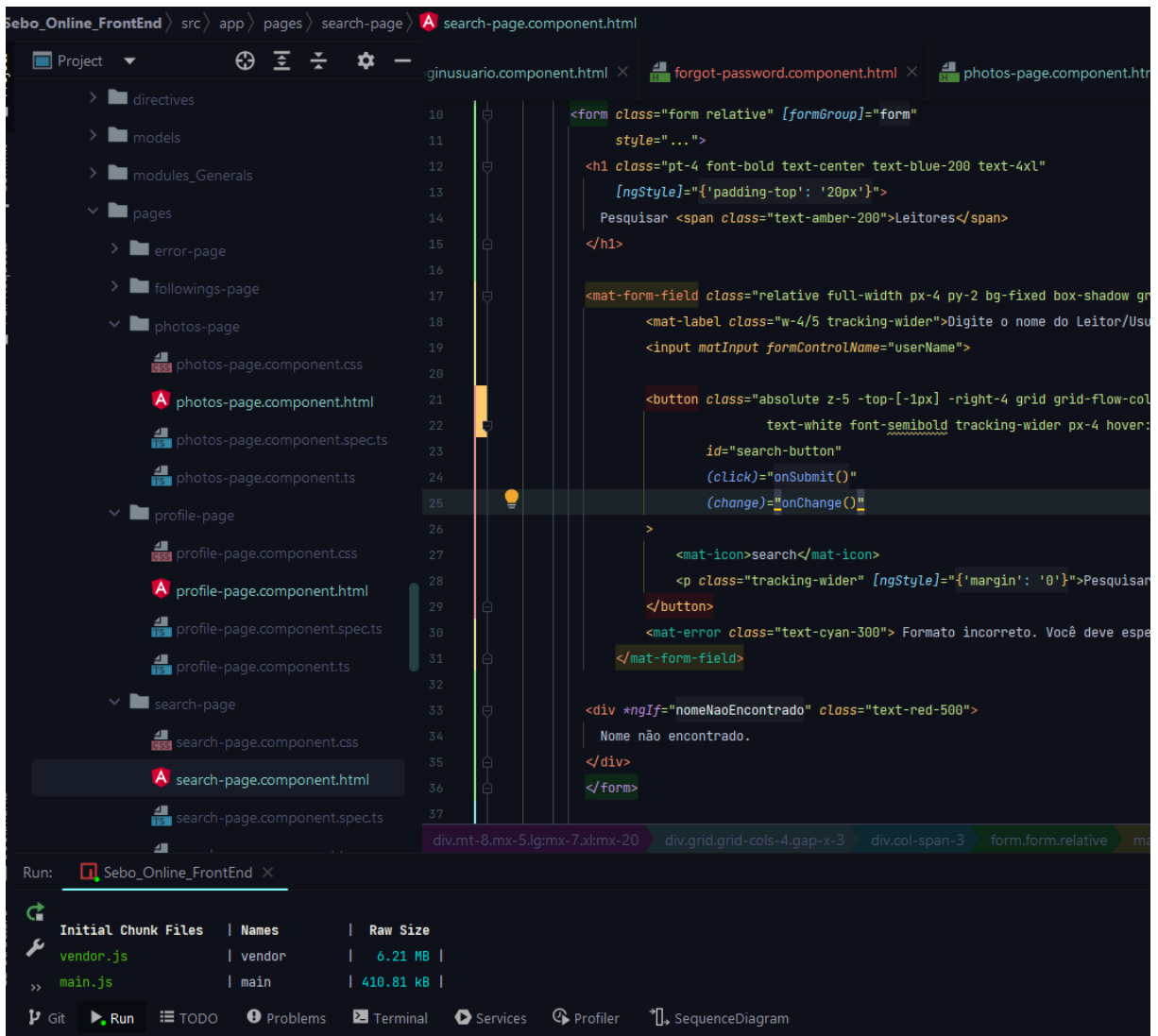
Resposta :

FIGURA 09 - Fluxo de Comunicação 2



FONTE: Os Autores(2023)

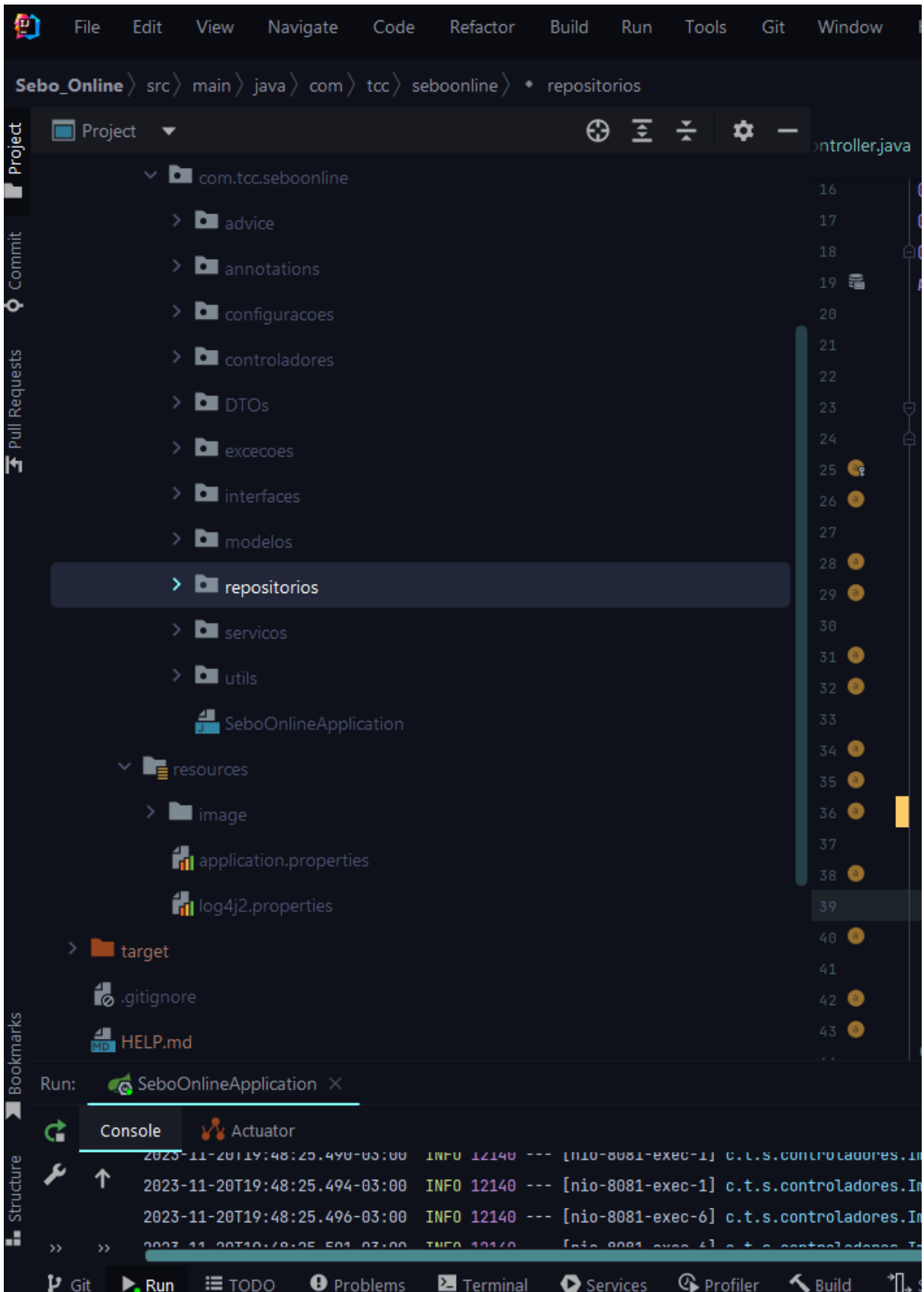
FIGURA 10 – ESTRUTURA FRONT END(ANGULAR)



FONTE: Os Autores (2023)

A camada do *front end*, como mencionado antes, foi feito em angular e por isso pra usar todo o poder e eficiência do mesmo, separamos as telas em componentes.

FIGURA 11 – ESTRUTURA BACK END(JAVA/SPRING)



FONTE: Os Autores (2023)

A imagem na Figura 11 é uma foto que descreve a organização do back end, no qual para desenvolvermos as APIs, optamos por utilizar a arquitetura em camadas.

## 4.2. AS INTERFACES DO SOFTWARE

Nesta seção apresentaremos as interfaces e suas respectivas funcionalidades. Abrangemos todas as telas do sistema, acompanhadas de uma breve descrição sobre o funcionamento.

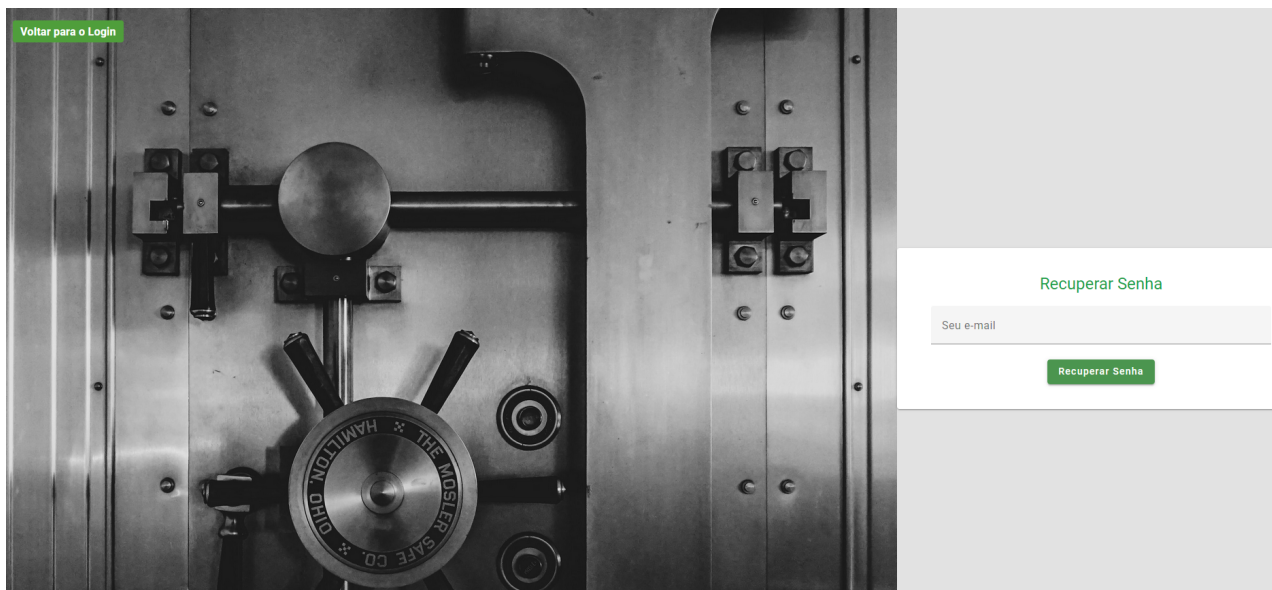
FIGURA 12 – TELA DE LOGIN



FONTE: Os Autores (2023)

Na figura 12 temos a tela de Login, após o usuário acessá-la e preencher os campos, será encaminhado para a tela principal (home), onde ficam listados todos os livros cadastrados em numa única página. antes porém tem a opção caso tenha esquecido a senha, de recuperá-la ao clicar no link 'Recuperar Senha'

FIGURA 13 – TELA DE RECUPERAR SENHA



FONTE: Os Autores (2023)

A tela acima é para recuperar a senha caso o usuário/leitor tenha esquecido será encaminhado uma nova senha para o mesmo.

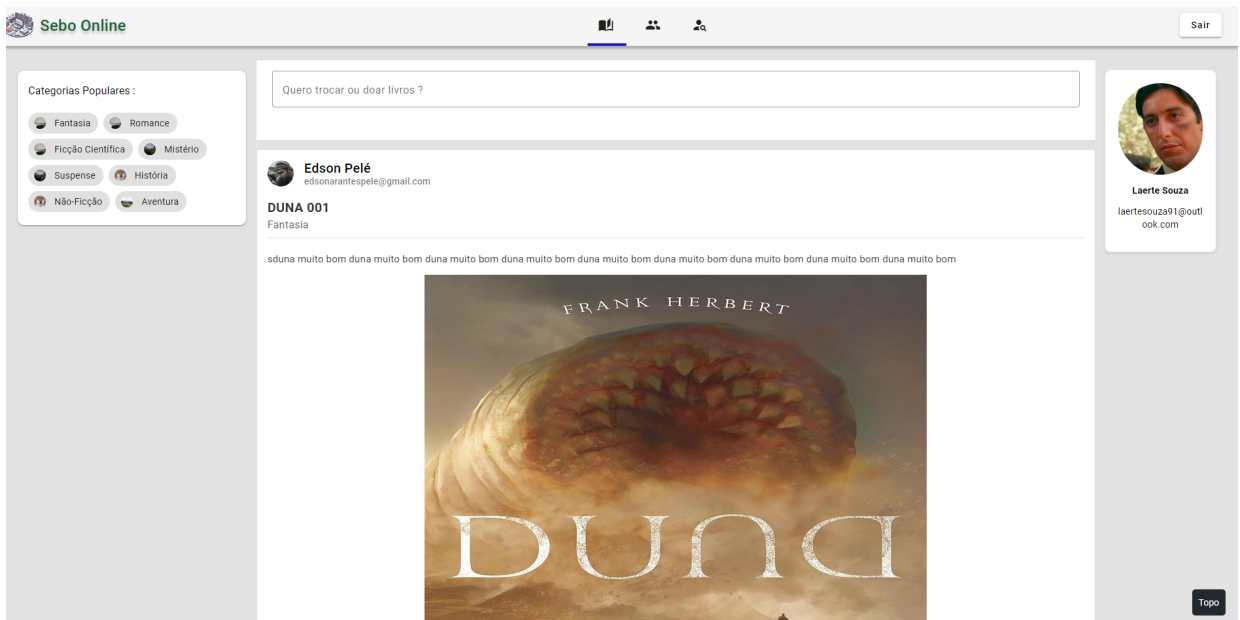
FIGURA 14 – TELA DE CADASTRAR



FONTE: Os Autores (2023)

Se acaso um indivíduo tiver a vontade de utilizar o sistema poderá cadastrar seus dados e se tornar um usuário/leitor.

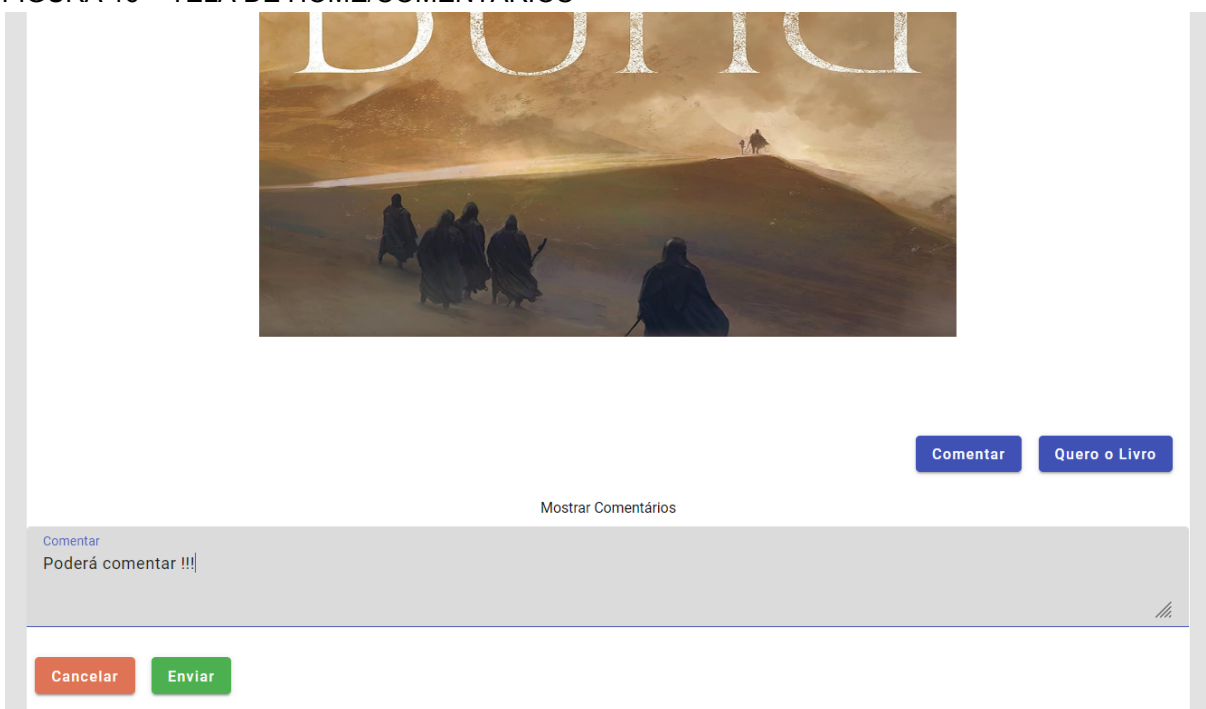
FIGURA 15 – TELA DE HOME



FONTE: Os Autores (2023)

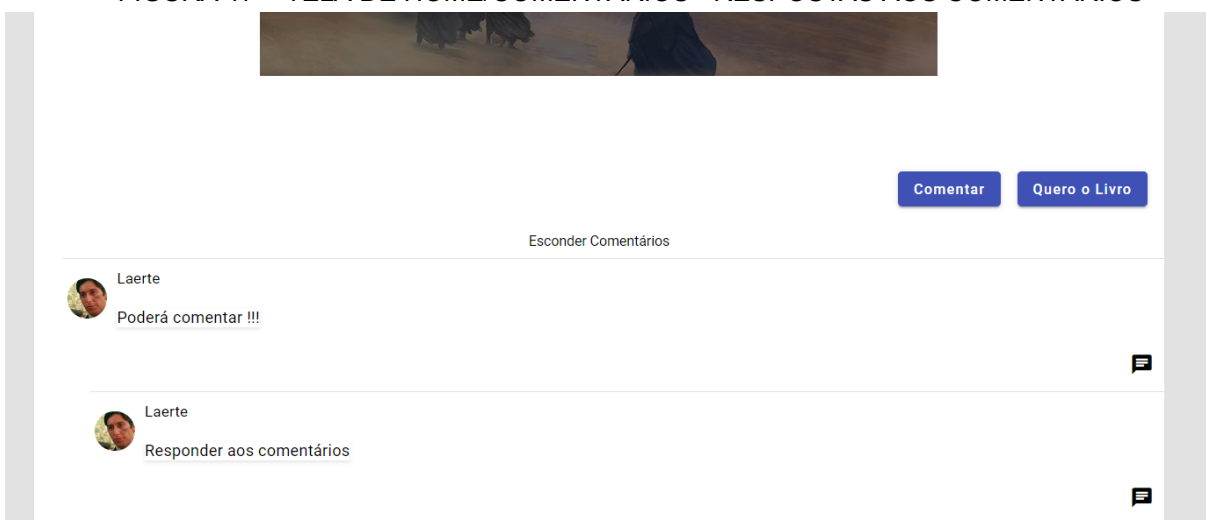
Essa tela acima é a tela home , onde ficam todos os livros cadastrados no sistema, as pessoas poderão visualizá-los e interagir com outros leitores da comunidade.

FIGURA 16 – TELA DE HOME/COMENTÁRIOS



FONTE: Os Autores (2023)

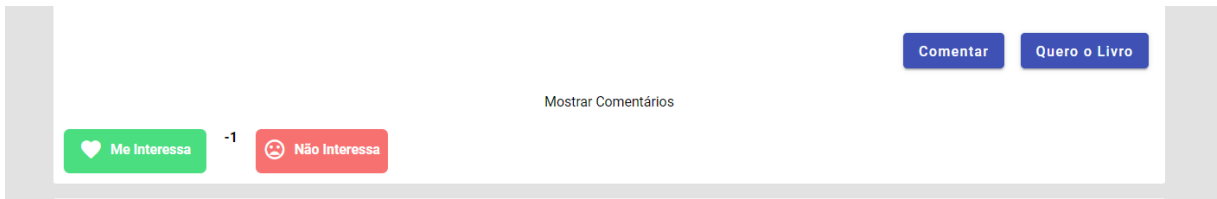
FIGURA 17 – TELA DE HOME/COMENTÁRIOS - RESPOSTAS AOS COMENTÁRIOS



FONTE: Os Autores (2023)

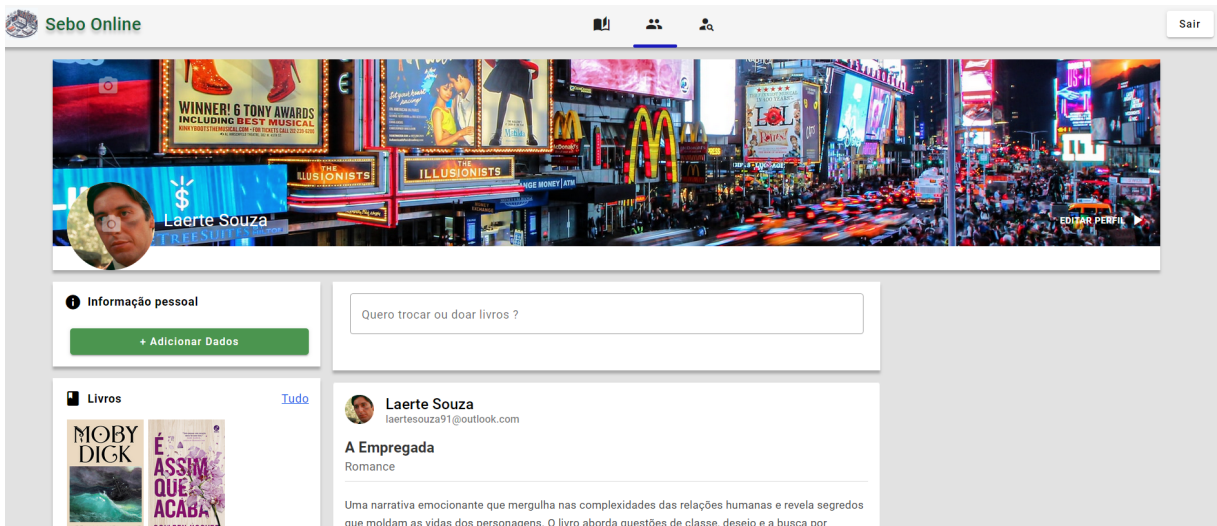
As telas acima em cada livro publicado o usuário poderá interagir através de comentários com outros indivíduos e demonstrar interesse caso queira emprestado, e até mesmo responder aos comentários caso queira.

FIGURA 18 – TELA DE HOME/FAVORITAR-DESFAVORITAR



FONTE: Os Autores (2023)

FIGURA 19 – TELA DE PERFIL USUÁRIO



FONTE: Os Autores (2023)

A tela de perfil do usuário, o usuário poderá fazer manipulações no seu perfil, tais como adicionar sua foto, uma foto de fundo como paisagem, adicionar capas de livros como uma 'biblioteca particular' ou mesmo gerenciar seus Livros publicados.

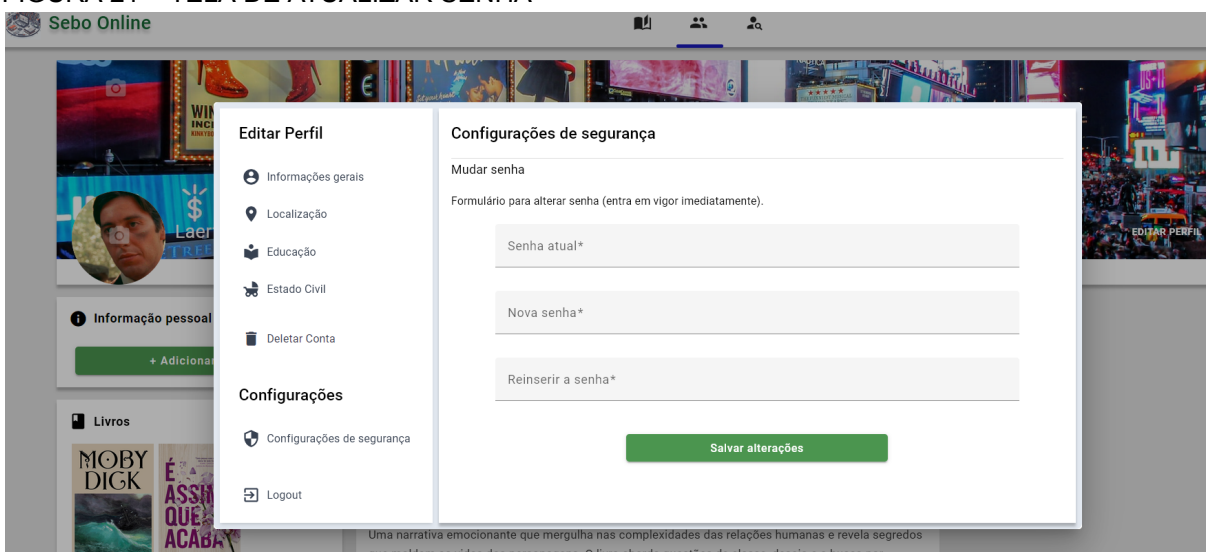
FIGURA 20 – TELA DE CADASTRO DE INFORMAÇÕES

The image shows a web interface for 'Sebo Online'. On the left, there is a sidebar menu under 'Editar Perfil' with the following items: 'Informações gerais', 'Localização', 'Educação', 'Estado Civil', 'Deletar Conta', 'Configurações', and 'Logout'. The 'Configurações' section includes 'Configurações de segurança'. The main area is titled 'Informações Gerais' and contains a form with the following fields: 'Nome\*' (Laerte), 'Sobrenome\*' (Souza), 'Gênero' (Masculino), 'Data de nascimento', 'Email\*' (laertesouza91@outlook.com), and 'Número do telefone' (4159595959). A green button labeled 'Salvar alterações' is at the bottom of the form. The background shows a book cover for 'MOBY DICK' and 'É ASSIM QUE ACABA'.

FONTE: Os Autores (2023)

A tela supracitada é onde um usuário poderá adicionar mais dados ao seu perfil, para que outros possam entender e conhecê-lo melhor.

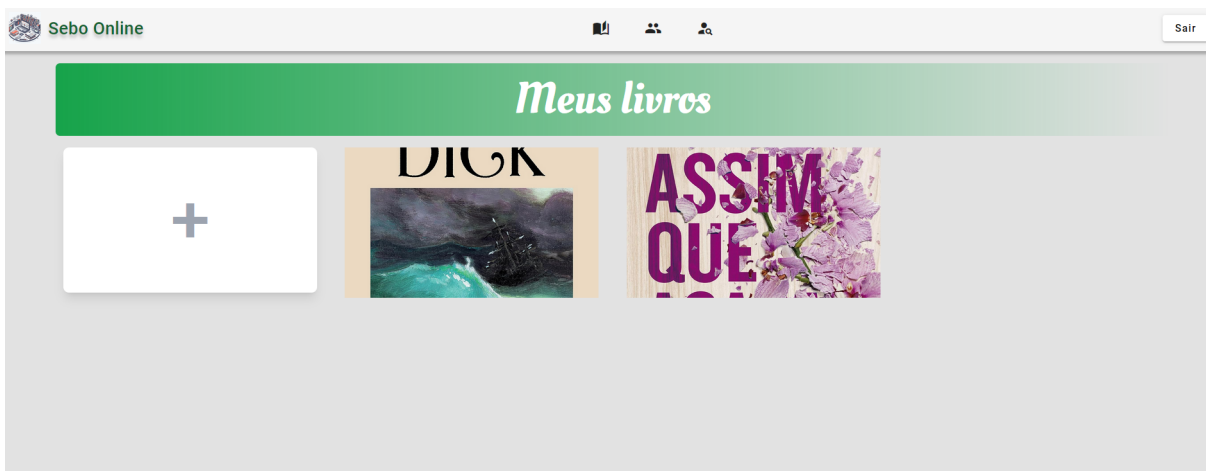
FIGURA 21 – TELA DE ATUALIZAR SENHA



FONTE: Os Autores (2023)

Caso o leitor queira atualizar sua senha, poderá atualizar por uma outra senha.

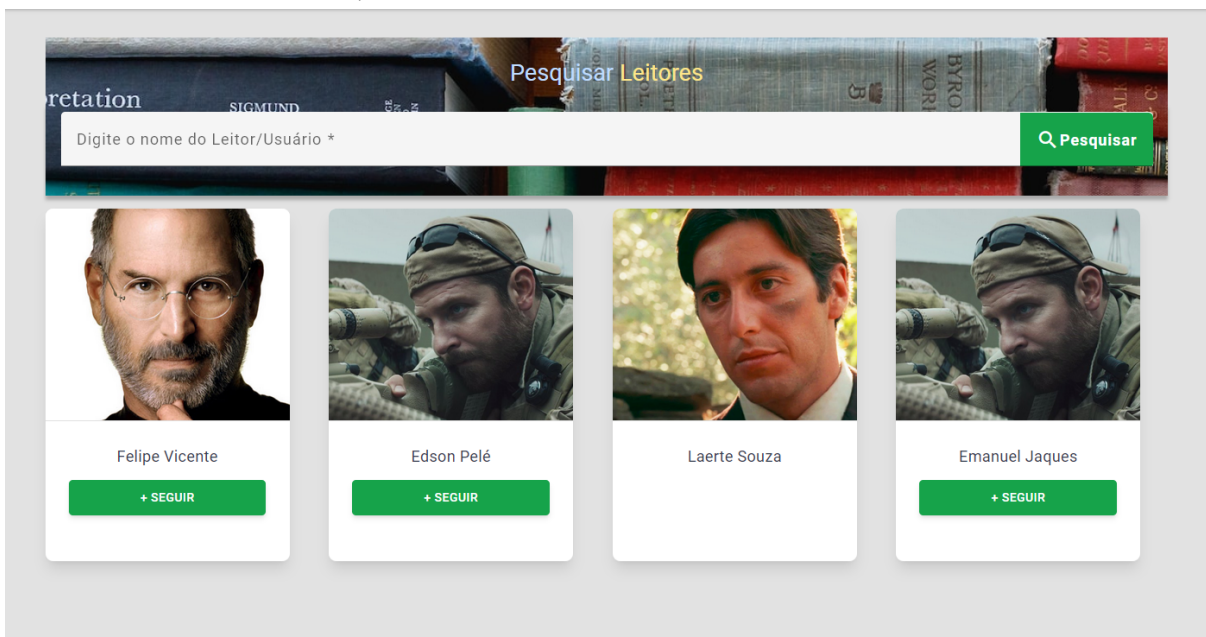
FIGURA 22 – TELA DE ADICIONAR CAPAS DE LIVROS



FONTE: Os Autores (2023)

Na tela acima o usuário poderá adicionar capas dos seus livros ou coleções, para deixar visível para outros usuários.

FIGURA 23 – TELA DE PESQUISAR NOVOS LEITORES



FONTE: Os Autores (2023)

Essa tela foi desenvolvida com o intuito do leitor procurar por novos amigos/leitores também.

#### 4.3 IMPLANTAÇÃO E MELHORIAS FUTURAS

A etapa de implantação do sistema "SeboOnline" foi realizada com sucesso, oferecendo uma base sólida para futuras implementações e aprimoramentos. Vislumbrando o constante desenvolvimento e aprimoramento da plataforma, identificamos áreas para expansão que poderão agregar maior valor à experiência do usuário e à funcionalidade do sistema. Entre outras, estão identificadas:

a) Conexão via Google para Autenticação:

Como parte do aprimoramento da segurança e comodidade no acesso dos usuários, uma implementação planejada consiste na integração de autenticação via Google. Esta adição permitirá que os usuários se autenticem utilizando suas contas do Google, oferecendo uma opção de login alternativa e simplificada, tornando o acesso à plataforma mais intuitivo.

b) Adição de Notificações:

Considerando a importância da comunicação e interação com os usuários, planejamos incorporar um sistema de notificações. Essas notificações seriam utilizadas para informar os usuários sobre atualizações importantes, transações realizadas, interações relevantes ou novidades dentro da plataforma, aprimorando a experiência do usuário e mantendo-os informados em tempo real.

c) Integração de Chat com Banco de Dados Não Relacional:

Para promover uma interação mais dinâmica entre os usuários, está prevista a implementação de um sistema de chat. Este chat será integrado com um banco de dados não relacional, possibilitando a troca de mensagens em tempo real entre os usuários, oferecendo uma experiência de comunicação mais imediata e eficiente.

Estas melhorias propostas têm o objetivo de enriquecer a plataforma "SeboOnline", ampliando sua funcionalidade e proporcionando aos usuários uma experiência mais completa e interativa. A implementação destas funcionalidades está alinhada com a visão de evolução contínua do sistema, garantindo maior satisfação dos usuários e mantendo-o atualizado com as demandas e tendências tecnológicas.

## 5 CONSIDERAÇÕES FINAIS

Ao longo do desenvolvimento deste sistema, nosso objetivo principal foi não apenas concluir nossa formação, mas também criar algo que fosse útil e potencialmente viável para uso futuro. Buscamos criar uma plataforma de trocas de livros que poderá ser lançada como uma plataforma gratuita ou incorporada a uma plataforma pré-existente. Não incluímos campos para sugestão de valor, pois nossa intenção é que seja sempre livre e sem custos.

No entanto, durante esse percurso, nos deparamos com desafios significativos. Um deles foi garantir a fluidez ao adicionar um livro e permitir essa ação de forma transparente para o usuário. Além disso, enfrentamos a dificuldade de encontrar o layout ideal: aquele que fosse não apenas intuitivo, mas também esteticamente agradável.

Apesar desses contratemplos, persistimos, encorajando uns aos outros ao longo do processo. Assim, a obra que apresentamos agora está finalizada e pronta para ser utilizada. Estamos cientes que dentro do ciclo de vida de um sistema, após sua real implantação, tenderá a surgir novas demandas e situações que não estavam previstas nos requisitos e para isso, será conveniente que a equipe, ou analistas conhecedores das tecnologias utilizadas, acompanhem e monitorem a implantação desse sistema.

## REFERÊNCIAS

**Download Astah Software.** Disponível em: <<https://astah.net/downloads/>>.

**HIBERNATE. Hibernate. Everything data. - Hibernate.** Disponível em: <<https://hibernate.org/>>.

**INITIALIZR, S.** Disponível em: <<https://start.spring.io/>>.

**JDK, Z. Azul Downloads.** Disponível em: <<https://www.azul.com/downloads/#zulu>>. Acesso em: 25 set. 2023.

**JETBRAINS. IntelliJ IDEA.** Disponível em: <<https://www.jetbrains.com/idea/>>.

**JPA. Arquitetura Java Persistence API (JPA).** Disponível em: <[https://www.ibm.com/docs/pt-br/was/8.5.5?topic=SSEQTP\\_8.5.5/com.ibm.websphere.nd.multiplatform.doc/ae/cejb\\_persistence.htm](https://www.ibm.com/docs/pt-br/was/8.5.5?topic=SSEQTP_8.5.5/com.ibm.websphere.nd.multiplatform.doc/ae/cejb_persistence.htm)>.

**KANBAN. Kanban: entenda o que é e como funciona esse método.** Disponível em: <<https://rockcontent.com/br/blog/kanban/>>.

**LIVRE, M. Mercado Livre Brasil.** Disponível em: <<https://www.mercadolivre.com.br/>>.

**MICROSOFT. Visual Studio Code.** Disponível em: <<https://code.visualstudio.com/>>. Acesso em: 25 nov. 2022.

**NODE. npm | build amazing things.** Disponível em: <<https://www.npmjs.com/>>.

**NPM. npm | build amazing things.** Disponível em: <<https://www.npmjs.com/>>.

**POSTMAN. Postman | The Collaboration Platform for API Development.** Disponível em: <<https://www.postman.com/>>.

SCRUM. **Scrum: como usar a metodologia ágil em seus projetos.** Disponível em: <<https://rockcontent.com/br/blog/scrum/>>.

SPRING. **spring.io.** Disponível em: <<https://spring.io/>>. Acesso em: 3 mar. 2020.

TEAMS. **Acesse sua conta | Microsoft Teams.** Disponível em: <<https://www.microsoft.com/pt-br/microsoft-teams/log-in>>.

THE POSTGRESQL GLOBAL DEVELOPMENT GROUP. **PostgreSQL.** Disponível em: <<https://www.postgresql.org/>>. Acesso em: 25 nov. 2021.

TRELLO. **Trello.** Disponível em: <<https://trello.com/pt-BR>>.

UNESCO. **Unesco.** Disponível em: <<https://unesdoc.unesco.org/ark:/48223/pf0000230725>>.

WHATSAPP. **Sobre o WhatsApp.** Disponível em: <[https://www.whatsapp.com/about/?lang=pt\\_br](https://www.whatsapp.com/about/?lang=pt_br)>.

## APÊNDICE A – REQUISITOS DO SISTEMA

### 1.1 REQUISITOS FUNCIONAIS

Login	
<b>IDENTIFICADOR:</b> RF 001	<b>Requisito :</b> Realizar Login
O usuário/leitor deverá entrar no sistema web , informando senha e e-mail cadastrados.	

Recuperar Senha	
<b>IDENTIFICADOR:</b> RF 002	<b>Requisito :</b> Recuperar Senha
O usuário/ leitor caso tenha perdido a senha atual , poderá clicar no link de 'Recuperar Senha' e preencher o seu e-mail cadastrado.	

Logout	
<b>IDENTIFICADOR:</b> RF 003	<b>Requisito :</b> Realizar Logout
Quando o usuário estiver conectado poderá clicar no botão 'Sair' , desconectando , o sistema redirecionará para a tela de login	

Cadastrar Usuário	
<b>IDENTIFICADOR: RF 004</b>	<b>Requisito : Realizar Cadastro de Usuário</b>
O usuário que queira utilizar o sistema seboOnline poderá fazer seu cadastro ao clicar na aba 'Cadastrar', e informar os dados conforme o formulário.	

Pesquisar/Filtrar Livro	
<b>IDENTIFICADOR: RF 005</b>	<b>Requisito : Filtrar Livro</b>
O usuário poderá clicar em tags que terão o título das categorias dos livros e selecionar a opção que desejar.	

Pesquisar Leitor/Usuário	
<b>IDENTIFICADOR: RF 006</b>	<b>Requisito : Pesquisar Leitor</b>
O usuário logado poderá clicar na aba de usuários , clicar e na sequência pesquisar um leitor que deseja conhecer melhor.	

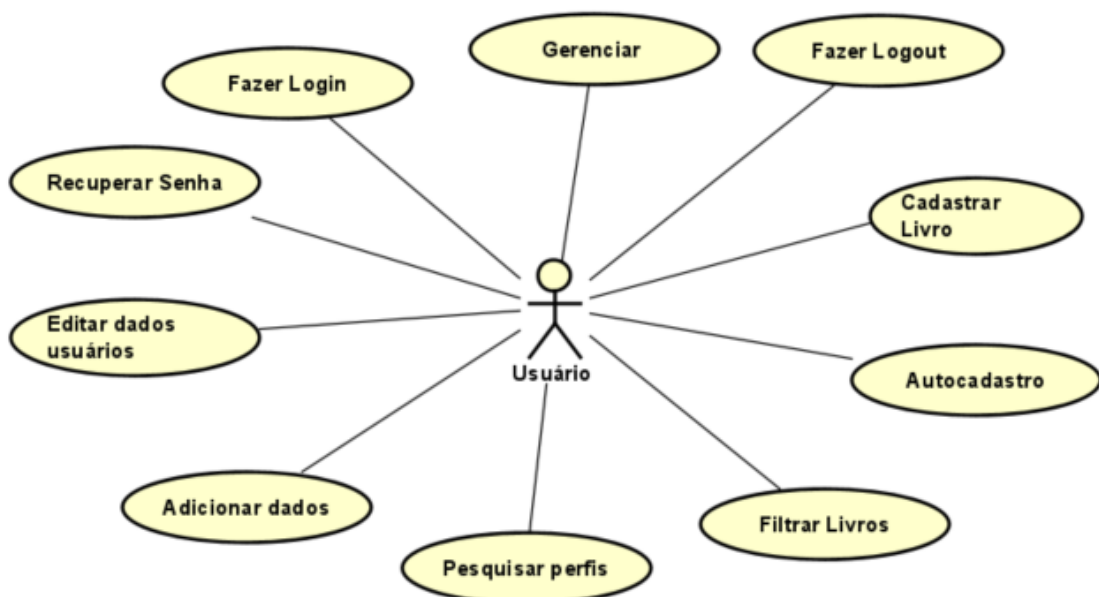
Remover Livro Publicado	
<b>IDENTIFICADOR: RF 007</b>	<b>Requisito : Remover Livro</b>
O usuário logado que após ter adicionado o livro e não queira por algum motivo deixar visível para outros usuários poderá deletar o livro .	

Ocultar Livro	
<b>IDENTIFICADOR: RF 008</b>	<b>Requisito :Ocultar</b>
O usuário logado poderá caso queira deixar oculto o livro , por algum motivo ou caso ja tenha algum interessado	

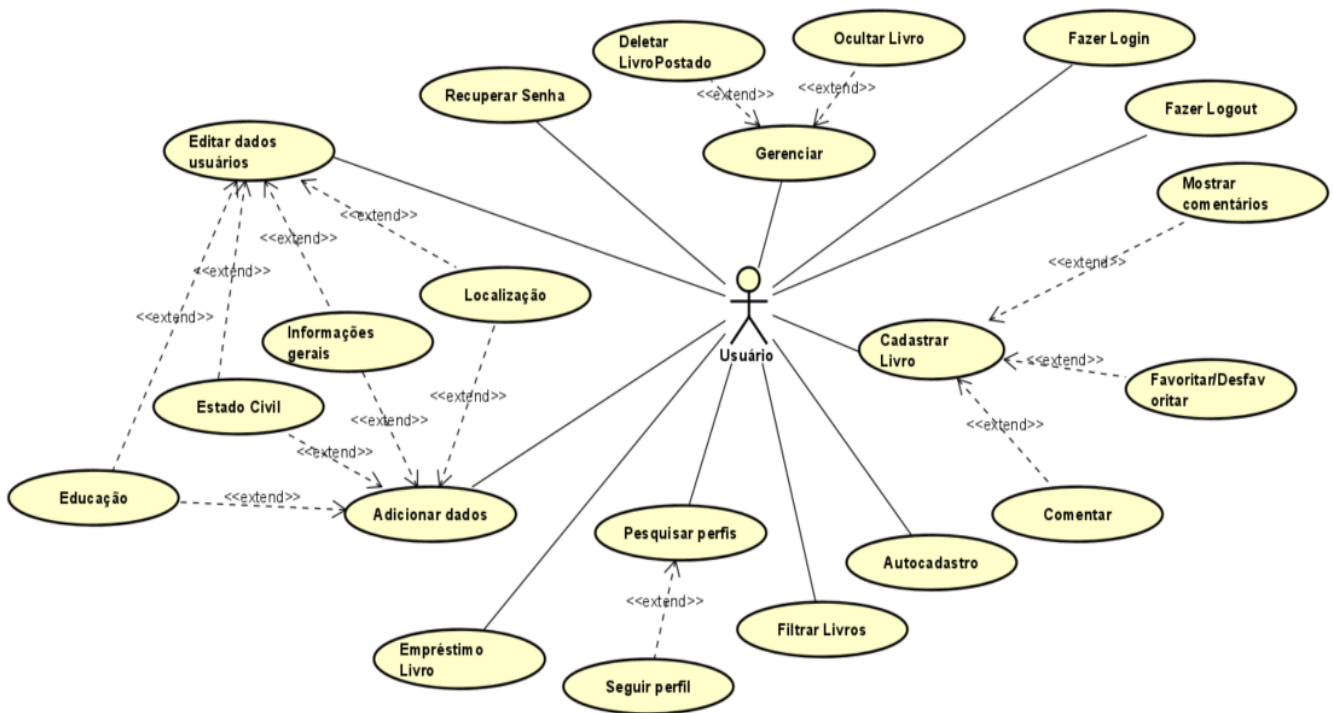
Adicionar Livro	
IDENTIFICADOR: RF 009	Requisito : Adicionar Livro
O usuário logado caso queira deixar exposto algum livro poderá clicar em adicionar livro e na sequência adicionar os dados tais como gênero , nome , descrição e uma foto de capa do livro.	

Comentar Livros	
IDENTIFICADOR: RF 010	Requisito : Comentários
O usuário logado caso se sinta com vontade , poderá interagir com outros usuários e comentar nos livros publicados deles.	

## APÊNDICE B – DIAGRAMA DE CASO DE USOS NÍVEL I

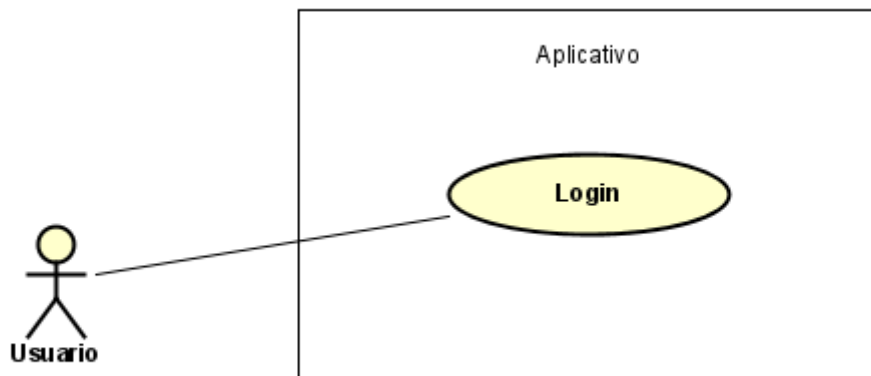


## APÊNDICE C – DIAGRAMA DE CASOS DE USOS NÍVEL II



## APÊNDICE D – ESPECIFICAÇÕES DE CASOS DE USOS

**Caso de uso:** UC001 - Realizar Login



**Ator principal:** Usuário

**Interesse e interessados:**

- O usuário deseja realizar o login no aplicativo para a utilização de suas funcionalidades.

**Pré condição:** O usuário ter acesso ao aplicativo e possuir um cadastro previamente criado.

**Pós condição:** O aplicativo efetua a autenticação do usuário e inicia a sua sessão.

**Cenário de sucesso principal:**

1. O usuário acessa o aplicativo e clicar em logar : e-mail e senha.
2. O aplicativo mostra a tela de login para o usuário.
3. O usuário entra com os seus dados para a autenticação.
4. O aplicativo efetua a autenticação do usuário e o redireciona para a tela inicial da aplicação.

**Cenário alternativo:**

1. O usuário acessa o aplicativo e clicar em logar.
2. O aplicativo mostra a tela de login para o usuário.
3. O usuário entra com os seus dados para a autenticação.
4. O aplicativo efetua a autenticação do usuário e retorna a mensagem de erro de usuário/senha invalido.

**Data-view 01:**

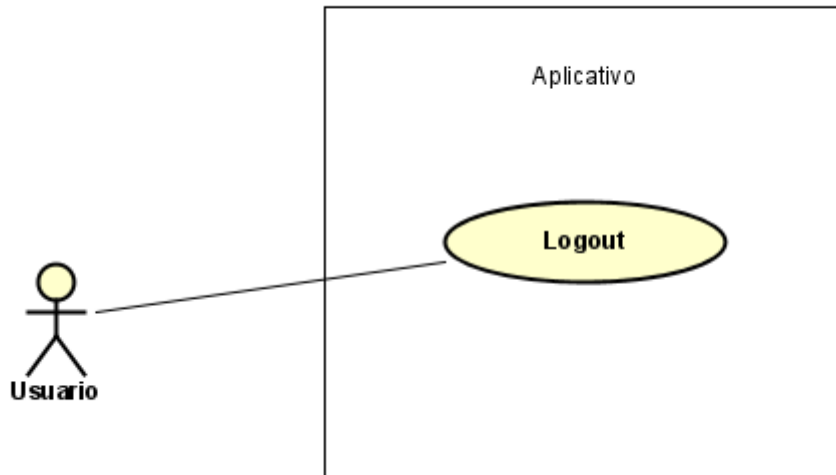
**dv1 - tela para login**

FIGURA 24 – TELA LOGIN



FONTE: Os Autores (2023)

## Caso de uso: UC 002 - Realizar logout



**Ator principal:** Usuário

**Interesse e interessados:**

- Usuário deseja realizar o logout da aplicação.

**Pré condição:** O usuário precisa estar logado no aplicativo.

**Pós condição:** O aplicativo efetua a desconexão do usuário e encerra sua sessão.

**Cenário de sucesso principal:**

1. Usuário logado acesso o aplicativo e clica em sair.
2. Aplicativo efetua a desconexão do usuário.
3. Aplicativo redireciona para a tela de login.

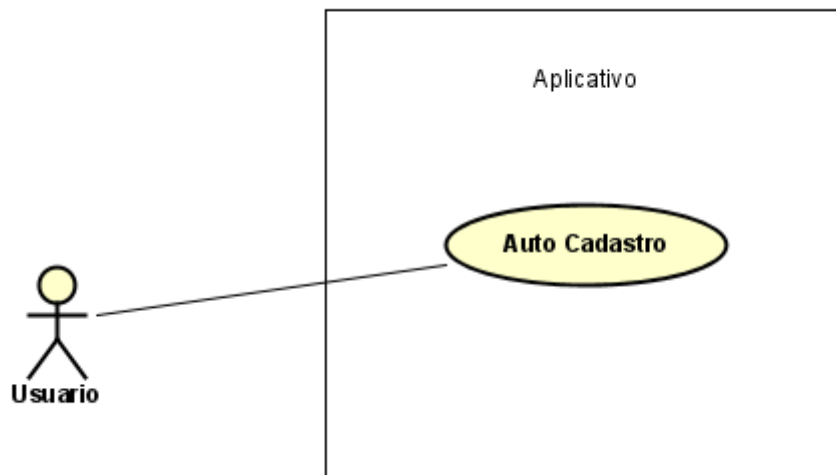
**Data-view 02:**

FIGURA 25 – TELA HOME/ SAIR



FONTE: Os Autores (2023)

## Caso de uso: UC 003 - Realizar auto cadastro



**Ator principal:** Usuário

**Interesse e interessados:**

- Usuário deseja realizar o auto cadastro para utilizar a aplicação.

**Pré condição:** O usuário tem acesso ao aplicativo.

**Pós condição:** O aplicativo efetua o cadastro do usuário.

**Cenário de sucesso principal:**

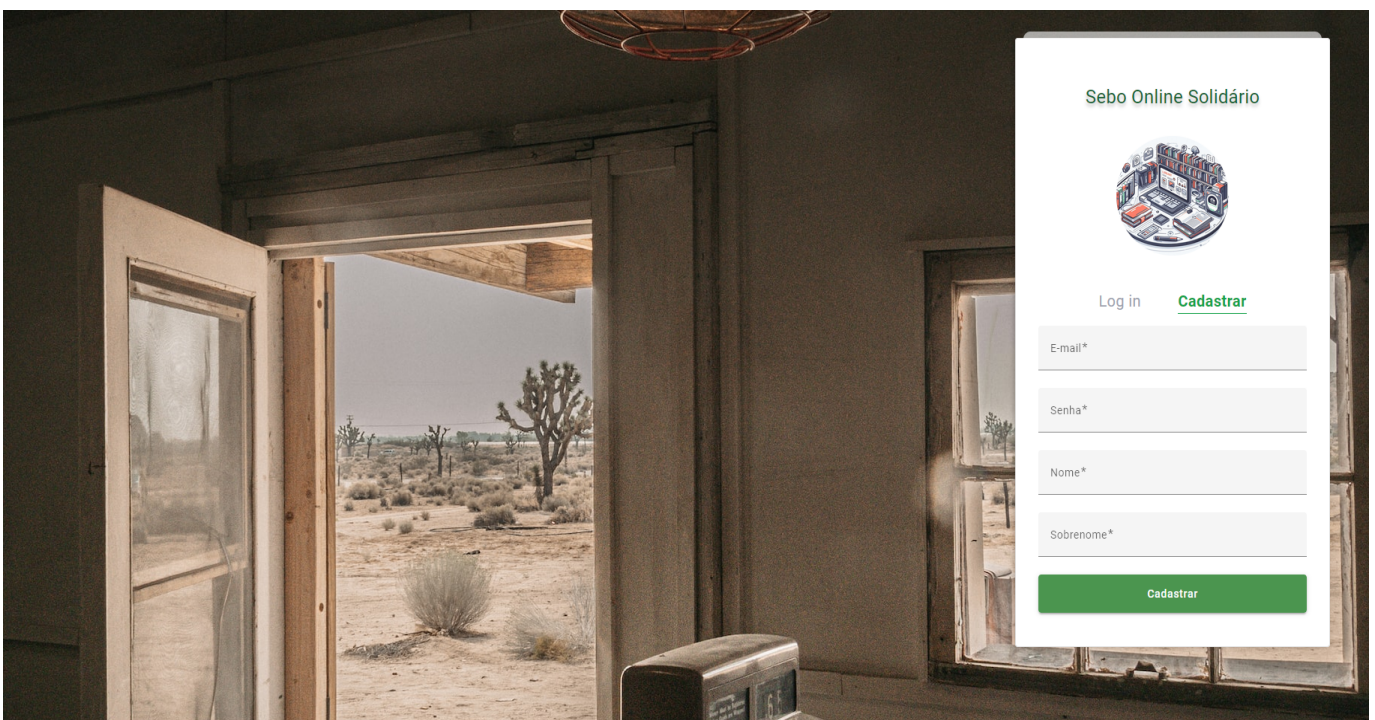
1. O usuário acessa o aplicativo e pressiona para criar um cadastro.
2. O aplicativo mostra a tela de efetuar cadastro.
3. Usuário insere seus dados cadastrais e clica em cadastrar.
4. O aplicativo recebe os dados e efetua o cadastro.
5. Logo na sequência o sistema encaminha para a tela de Login.

**Cenário alternativo:**

1. O usuário acessa o aplicativo e clica em criar um cadastro.
2. O aplicativo mostra a tela de efetuar cadastro.
3. Usuário insere seus dados cadastrais e clica em cadastrar.
4. O aplicativo indefere a solicitação e mostra uma mensagem de erro sobre algum dado cadastral inserido pelo usuário e retorna a tela de auto cadastro.

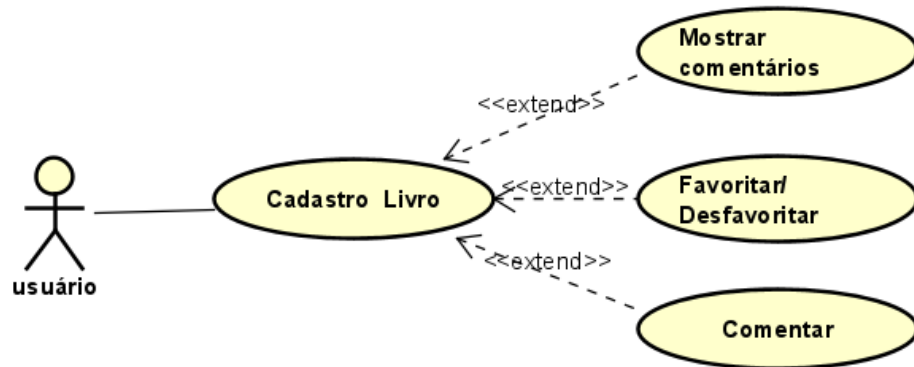
### Data-view 03:

FIGURA 26 – TELA DE CADASTRO



FONTE: Os Autores (2023)

## Caso de uso: UC 004 - Cadastro de livros



**Ator principal:** Usuário

**Interesse e interessados:**

- Usuário deseja realizar o cadastro de um livro

**Pré condição:** O usuário deve estar logado no aplicativo.

**Pós condição:** Os livros cadastrados pelo usuário aparecerão no aplicativo como disponíveis para troca, possibilitando buscar livros e sugestões de trocas na tela inicial/home do usuário.

**Cenário de sucesso principal (cadastro de lista de desejos):**

1. O usuário clica em cadastrar livros.
2. O aplicativo mostra o formulário de cadastro.
3. O usuário cadastra o nome do livro.
4. o usuário cadastra o gênero do livro.
5. O Usuário clica em adicionar imagem/capa do livro.
6. Após isso clica em Salvar livro

**Cenário alternativo :**

1. O usuário clica em cadastrar livros.
2. Caso exista um campo vazio não o deixará salvar.

**Cenário de sucesso principal (cadastro de livro):**

1. O usuário clica em cadastrar livros.
2. O aplicativo mostra o formulário de cadastro.
3. O usuário entra com os dados cadastrais do livro .
4. O Usuário clica em cadastrar livro.
5. O aplicativo salva o cadastro do livro e o adiciona à lista de disponíveis para troca.
6. O usuário poderá comentar nos livros.
7. O usuário poderá favoritar ou desfavoritar.
8. O usuário poderá clicar no botão de expandir comentários ou ocultá-los.

**Cenário alternativo (cadastro de livro):**

1. O usuário clica em cadastrar livros.
2. O aplicativo mostra o formulário de cadastro.
3. O usuário entra com os dados cadastrais do livro e clica em cadastrar.
4. O aplicativo mostra uma mensagem de erro sobre algum dado cadastral do livro inserido pelo usuário ,faltante, deixando o botão de Cadastrar desabilitado.

**Data-view 04:**

FIGURA 27 – TELA DE CADASTRO

The image shows a registration form for a book. It contains three main input fields, each with a red border and a red asterisk indicating it is mandatory. The first field is for the book title, the second is a dropdown menu for the genre, and the third is a text area for the description with a character count of 0/350. Below the description field is a file upload button labeled 'Procurar...' with the text 'Nenhum arquivo selecionado.' To the right of the form are two buttons: 'Cancelar' and 'Publicar'.

FONTE: Os Autores (2023)

## Caso de uso: UC 005 - Filtrar Livros

**Ator principal:** Usuário

**Interesse e interessados:**

- Usuário deseja efetuar a busca de livros.

**Pré condição:** O usuário deve estar logado no aplicativo.

**Pós condição:** O aplicativo mostra uma lista com os livros pesquisados (caso seja encontrado alguma categoria pesquisada) e mostra a opção de interação pelos comentários ou solicitar, dependendo do caso.

**Cenário de sucesso principal (Busca de livro):**

1. Usuário clica em uma das tags dos filtros.
2. Na tela de home aparecerá uma lista dos livros por categoria selecionada.
3. O usuário entra com os dados na tela e clica em buscar.

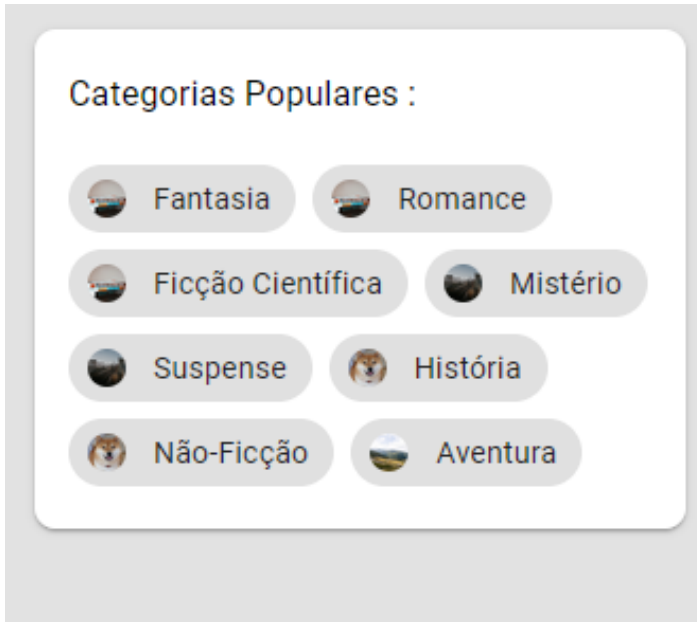
**Cenário alternativo (Busca de livro):**

1. Usuário clica em buscar livros.

2. A lista retorna vazia para a categoria selecionada , mantendo a lista previamente carregada.

**Data-view 005 :**

FIGURA 28 – TELA TAG



FONTE: Os Autores (2023)

**Caso de uso:** UC 006 - Adicionar dados do usuário.

**Ator principal:** Usuário

**Interesse e interessados:** Usuário deseja adicionar mais dados que a tela de cadastro (detalhados).

**Pré condição:** O usuário deve estar logado no aplicativo.

**Pós condição:** Após o usuário cadastrar os seus dados eles serão salvos com sucesso e mostrados no card lateral.

**Cenário de sucesso principal:**

1. O usuário clica na aba perfil .

2. Uma tela de usuário é mostrada com opções de adicionar dados.
3. O usuário clica no botão adicionar Dados.
4. Uma modal irá abrir com campos para editar perfil separados por categorias (Informações Gerais,Localização,Educação,Estado Civil).
5. O usuário clica na aba Informações Gerais e é exibido um formulário com campos : Nome , Sobrenome,Gênero, Data de Nascimento,E-mail,Telefone.
6. Após preenchê-los , o usuário clica no botão salvar.
7. O usuário clica na aba Localização e é exibido um formulário com campos : País Natal, Cidade Atual, País natal, Cidade natal.
8. Após preenchê-los , o usuário clica no botão salvar.
9. O usuário clica na aba Educação e é exibido um formulário com campos : Locais .
10. Após preenchê-los, o usuário clica no botão salvar.
11. O usuário clica na aba Estado Civil e é exibido um formulário com campos: Estado Civil.
12. Após preenchê-los, o usuário clica no botão salvar.
13. O usuário clica na aba Configurações de Segurança e é exibido um formulário com campos : Senha Atual, Nova Senha e Inserir senha.

**Cenário alternativo:**

1. O usuário clica na aba Adicionar dados .
2. O modal de Informações gerais do Usuário é aberto.
3. O usuário deixa campos , o formulário não deixa salvar os dados.
4. O usuário preenche os campos e deixa alguns vazios impossibilitando salvar os dados também.

## Data-view 006:

FIGURA 29 – TELA DE ADICIONAR DADOS

The screenshot shows a web interface for editing a user profile. On the left is a sidebar with navigation options: 'Editar Perfil' (with sub-items: 'Informações gerais', 'Localização', 'Educação', 'Estado Civil', 'Deletar Conta'), 'Configurações' (with sub-items: 'Configurações de segurança'), and 'Logout'. The main content area is titled 'Informações Gerais' and contains the following elements:

- A heading 'Informações' and a note: 'Ao alterar o e-mail , o novo e-mail será necessário para o Login.'
- Two input fields for 'Nome\*' and 'Sobrenome\*'. Both have red error messages below them: 'Nome é obrigatório' and 'Sobrenome é obrigatório'.
- A dropdown menu for 'Gênero' with 'Outros' selected.
- A date input field for 'Data de nascimento' with a calendar icon.
- An input field for 'Email\*' with a red error message: 'Formato do e-mail Inválido'.
- An input field for 'Número do telefone'.
- A green button labeled 'Salvar alterações' at the bottom.

FONTE: Os Autores (2023)

## Caso de uso: UC 007 - Editar dados do Usuário.

**Ator principal:** Usuário

**Interesse e interessados:** Usuário deseja editar dados na tela de editar (detalhados).

**Pré condição:** O usuário deve estar logado no aplicativo.

**Pós condição:** Após o usuário editar os seus dados eles serão salvos com sucesso e mostrados no card lateral.

**Cenário de sucesso principal:**

1. O usuário clica na aba perfil .

2. Uma tela de usuário é mostrada com opções de editar dados.
3. O usuário clica no botão adicionar Dados.
4. Uma modal irá abrir com campos para editar perfil separados por categorias (Informações Gerais,Localização,Educação,Estado Civil).
5. O usuário clica na aba Informações Gerais e é exibido um formulário com campos : Nome , Sobrenome,Gênero, Data de Nascimento,E-mail,Telefone.
6. Após preenchê-los , o usuário clica no botão salvar.
7. O usuário clica na aba Localização e é exibido um formulário com campos : País Natal, Cidade Atual, País natal, Cidade natal.
8. Após preenchê-los , o usuário clica no botão salvar.
9. O usuário clica na aba Educação e é exibido um formulário com campos : Locais .
10. Após preenchê-los, o usuário clica no botão salvar.
11. O usuário clica na aba Estado Civil e é exibido um formulário com campos: Estado Civil.
12. Após preenchê-los, o usuário clica no botão salvar.
13. O usuário clica na aba Configurações de Segurança e é exibido um formulário com campos : Senha Atual, Nova Senha e Inserir senha.

**Cenário alternativo:**

1. O usuário clica na aba Adicionar dados .
2. O modal de Informações gerais do Usuário é aberto.
3. O usuário deixa campos , o formulário não deixa salvar os dados.
4. O usuário preenche os campos e deixa alguns vazios impossibilitando salvar os dados também.

FIGURA 30 – TELA DE EDITAR DADOS USUÁRIO

**Editar Perfil**

- Informações gerais
- Localização
- Educação
- Estado Civil
- Deletar Conta

**Configurações**

- Configurações de segurança
- Logout

**Informações Gerais**

Informações

Ao alterar o e-mail , o novo e-mail será necessário para o Login.

Nome\*  
Laerte

Sobrenome\*  
Souza

Gênero  
Masculino

Data de nascimento

Email\*  
laertesouza91@outlook.com

Número do telefone  
4159595959

Salvar alterações

FONTE: Os Autores (2023)

## Caso de uso: UC 008 - Gerenciar

**Ator principal:** Usuário.

**Interesse e interessados:**

- O usuário deseja gerenciar o estado do livro, ou deixar oculto para os demais usuários ou deletá-lo.

**Pré condição:** O usuário deve estar logado no aplicativo e ter ao menos um livro cadastrado para troca.

**Pós condição:** O aplicativo remove o livro do acervo disponível ou oculto.

**Cenário de sucesso principal:**

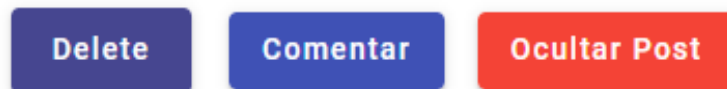
1. Usuário clica em Gerenciar.
2. Será aberto um modal com dois botões : Deletar e Ocultar.
3. O usuário pode clicar em Ocultar ou deletar livro.
4. O sistema executará : Ocultar / Deletar.

**Cenário alternativo:**

—

**Data-view 008 :**

FIGURA 31– TELA GERENCIAR LIVRO



FONTE: Os Autores (2023)

**Caso de uso:** UC 009 - Recuperar Senha de Usuário.

**Ator principal:** Usuário.

**Interesse e interessados:**

- Usuário deseja recuperar a senha perdida.

**Pré condição:** O sistema tem de estar disponível e usuário ter cadastro.

**Pós condição:** O sistema enviará um e-mail com uma senha temporária criptografada.

**Cenário de sucesso principal:**

1. Usuário clica em Recuperar Senha.
2. Aplicativo abre a página para recuperar senha.
3. O usuário digita o e-mail e clica em enviar e-mail com senha nova.

4. O usuário ao receber o e-mail com senha está apto a logar novamente.

**Cenário alternativo:**

1. Usuário ao informar o e-mail , caso não exista na base de dados, não será configurado nova senha.

**Data-view 009:**

FIGURA 32 – TELA RECUPERAR SENHA



FONTE: Os Autores (2023)

## **Caso de uso:** UC 010 - Desejo Solicitar Troca/Empréstimo/Doar

**Ator principal:** Usuário

**Interesse e interessados:**

- O usuário, caso fique interessado em algum livro , poderá clicar no botão “Quero o Livro” solicitando o mesmo.

**Pré condição:** O usuário deve estar logado.

**Pós condição:** Ao clicar no botão ‘Quero o Livro’ será enviado um e-mail ao dono do mesmo , contendo as informações do requerente ao livro , tais como email , nome, e telefone celular.

**Cenário de sucesso principal:**

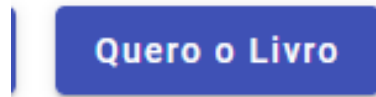
1. O usuário ao se interessar por um livro poderá clicar no botão ‘Quero o Livro’.
2. O aplicativo mostrará uma modal onde apresentará a informação que será enviado um e-mail com os seus dados ao dono do livro.
3. O aplicativo notificará o dono por uma mensagem via e-mail sobre o desejo de um usuário terceiro querer o livro caso demonstre interesse mútuo em emprestar ou doá-lo poderá entrar em contato através das informações do e-mail.

**Cenário alternativo:**

1. O usuário ao se interessar e clicar no botão ‘Quero o Livro’, o dono poderá não respondê-lo enviando um e-mail , tendo em vista que essa opção é apenas do dono que divulgou que gostaria de doar/emprestar o livro.

**Data-view 010: Colocar**

FIGURA 33 – TELA PEGAR EMPRESTADO



FONTE: Os Autores (2023)

**Caso de uso:** UC 011 - Deletar Conta de Usuário.

**Ator principal:** Usuário.

**Interesse e interessados:**

- Usuário deseja deletar a sua conta.

**Pré condição:** O sistema tem de estar disponível e usuário ter cadastro.

**Pós condição:** O sistema irá deletar no banco de dados tudo relacionado ao usuário e encaminhará para a tela de login.

**Cenário de sucesso principal:**

1. Usuário ao querer deletar sua conta no Sebo Online, poderá em seu perfil clicar no botão 'Deletar Perfil'
2. Após isso aparecerá um modal dizendo se realmente tem certeza da operação.

3. O sistema irá deletar tudo relacionado ao usuário na base de dados.
4. O sistema encaminhará na sequência para a tela de login(inicial).

**Cenário alternativo:**

1. O usuário ao clicar no botão de deletar conta , poderá cancelar a operação.
2. A conta continuará em operação , nada sendo afetado na base de dados.

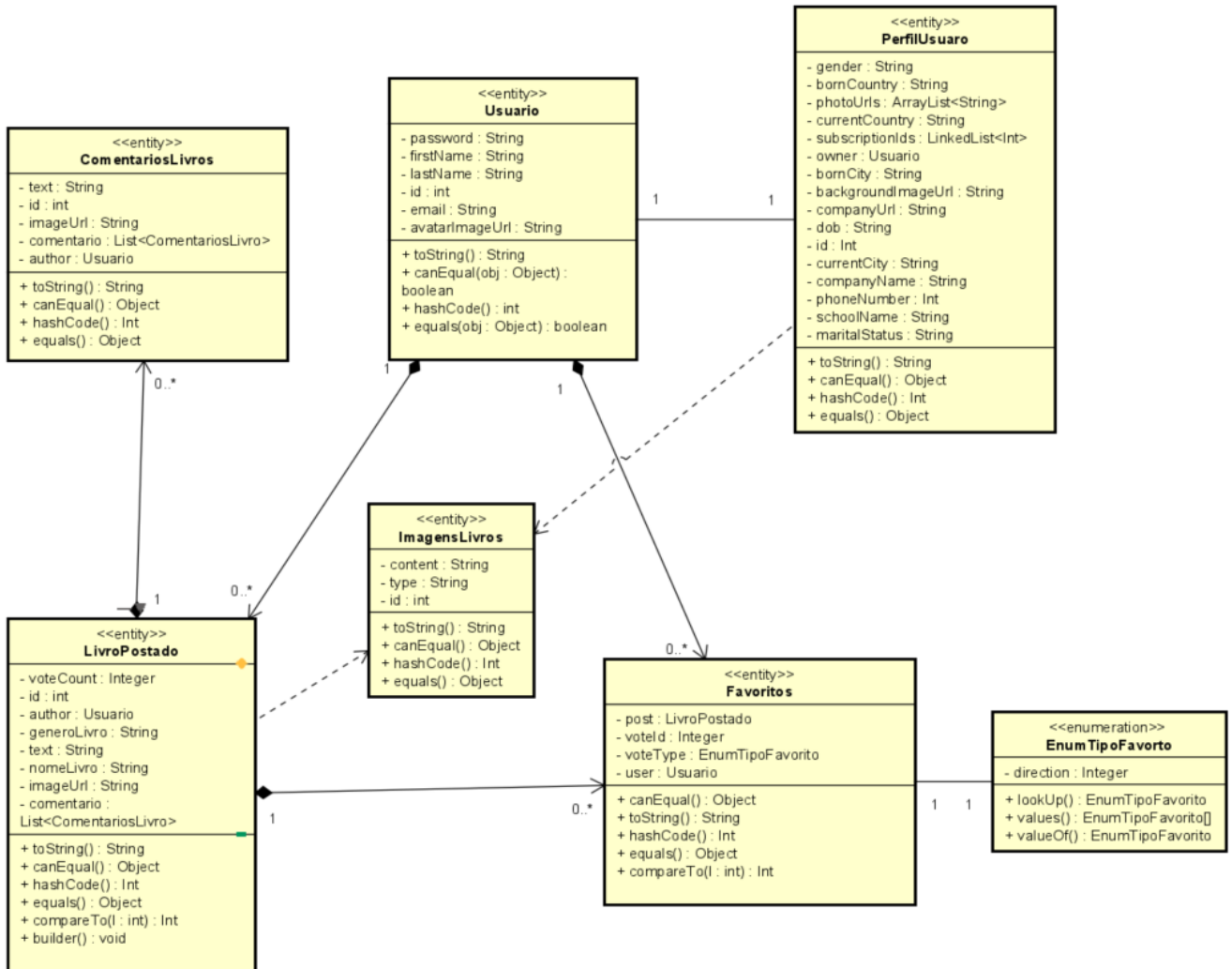
**Data-view 011 :**

FIGURA 34 – TELA DELETAR CONTA



FONTE: Os Autores (2023)

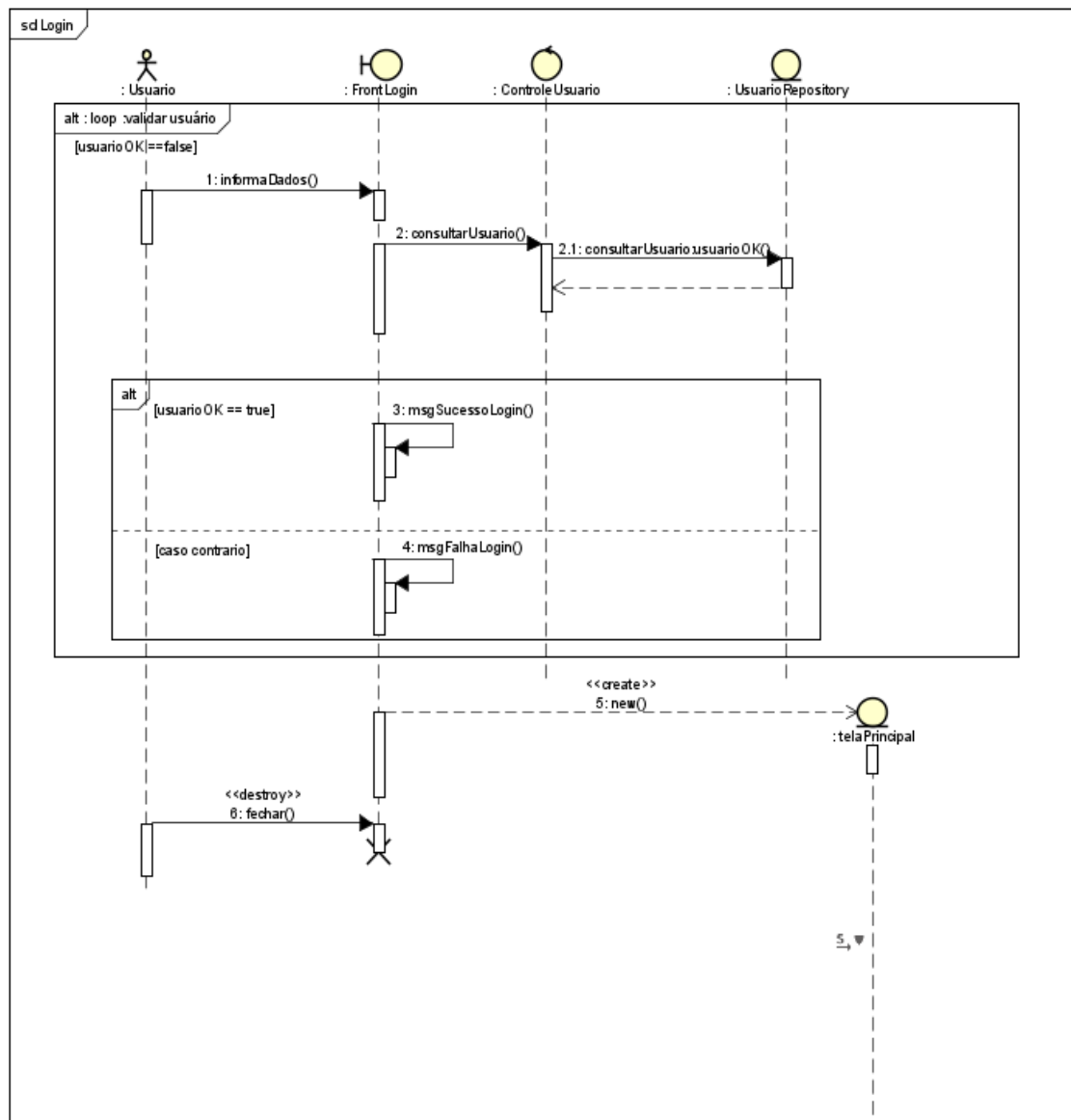
## APÊNDICE E – DIAGRAMA DE CLASSES



FONTE: Os Autores (2023)

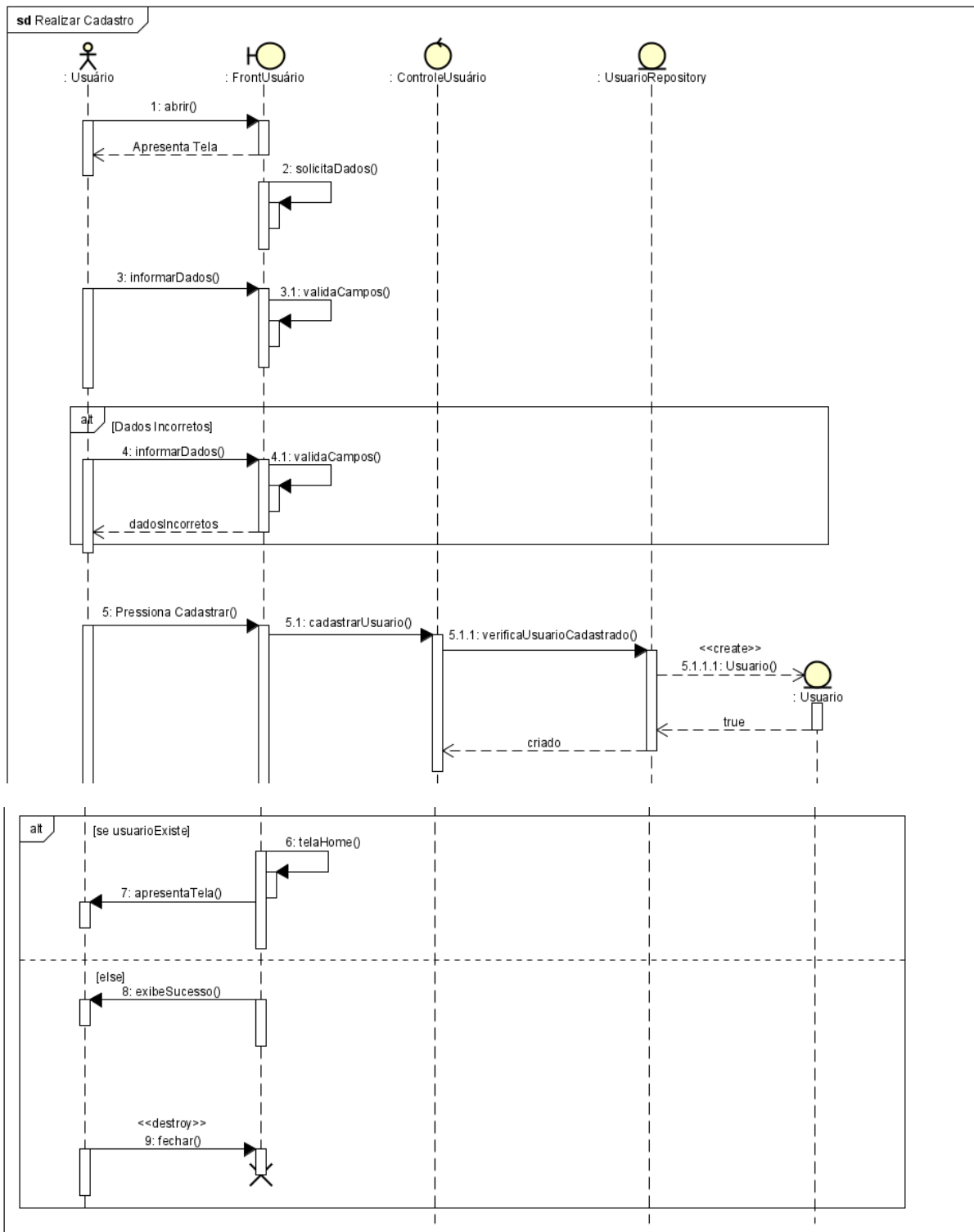
## APÊNDICE F – DIAGRAMAS DE SEQUÊNCIA

### DIAGRAMA DE SEQUÊNCIA – LOGIN



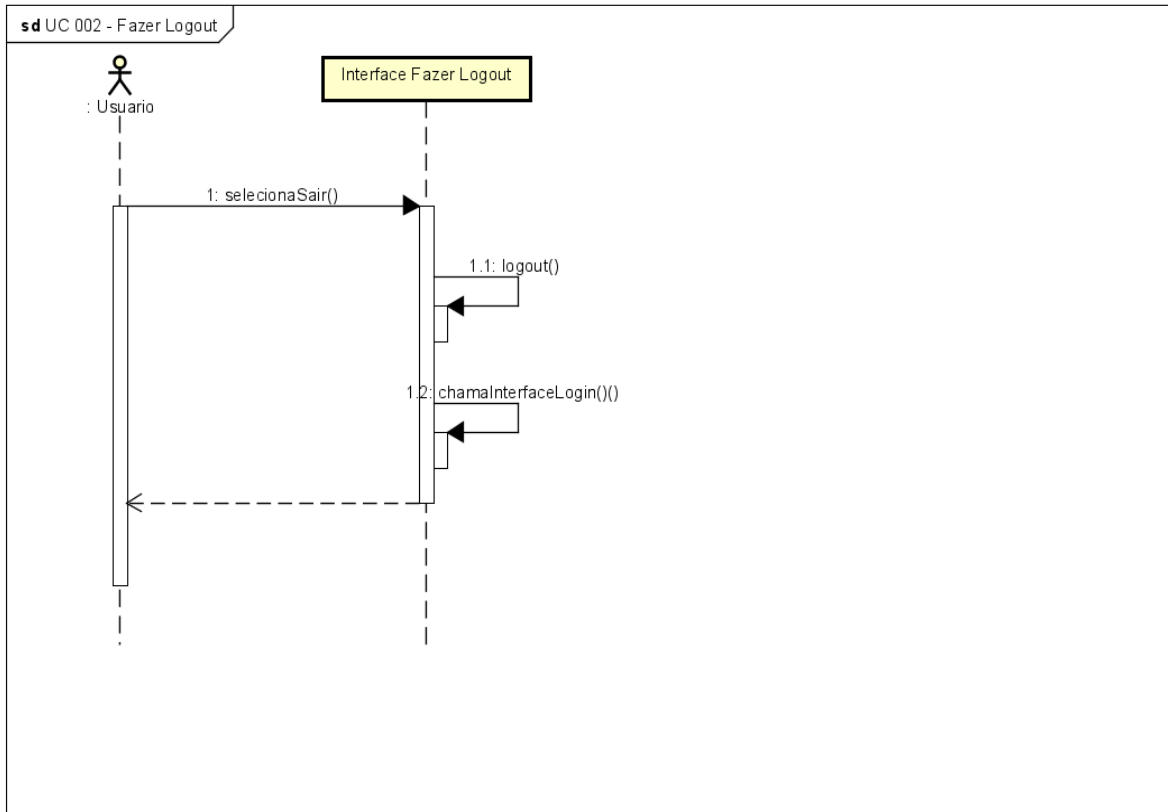
FONTE: Os Autores (2023)

## DIAGRAMA DE SEQUÊNCIA – CADASTRAR USUÁRIO



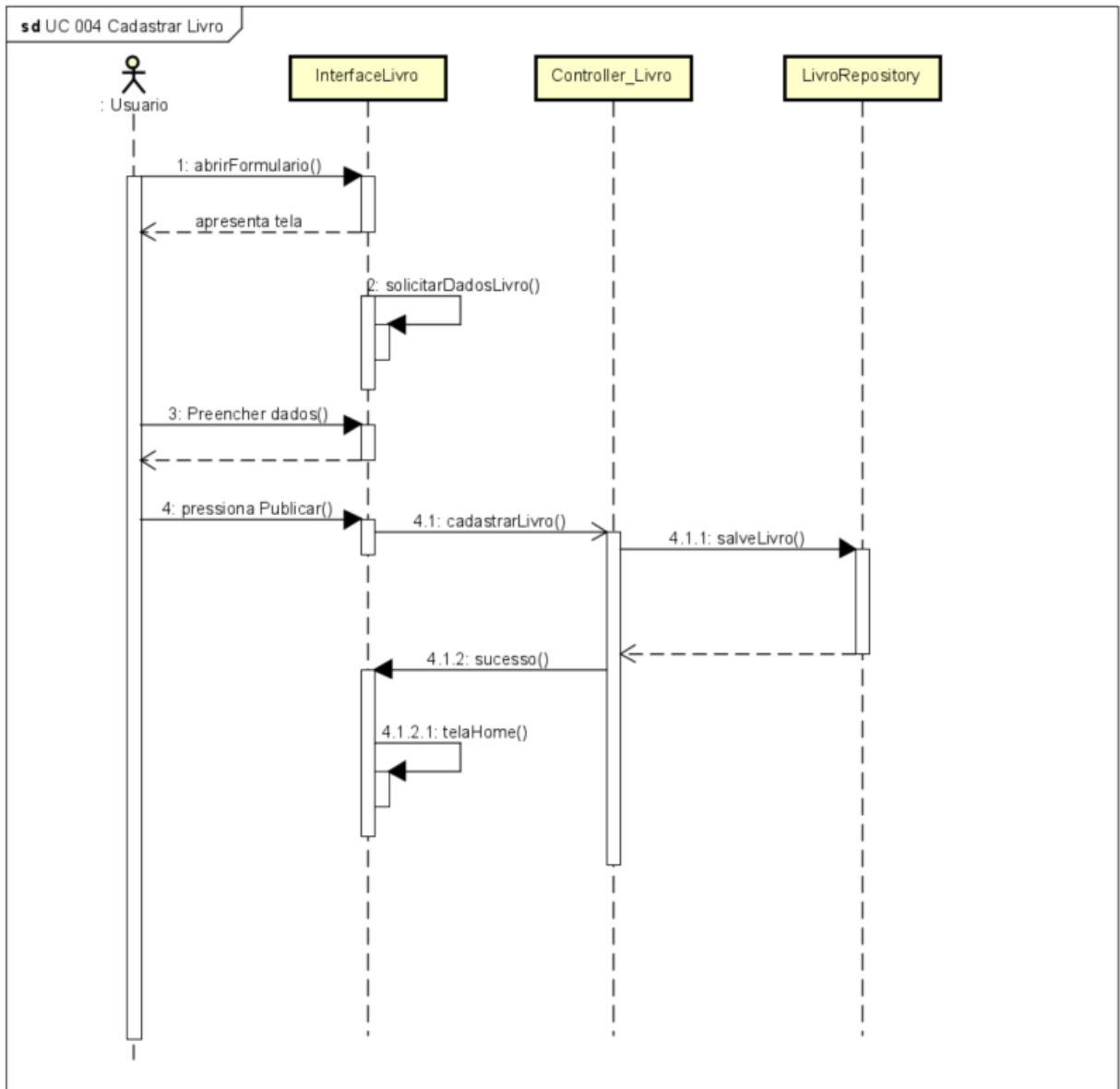
FONTE: Os Autores (2023)

## DIAGRAMA DE SEQUÊNCIA – LOGOUT



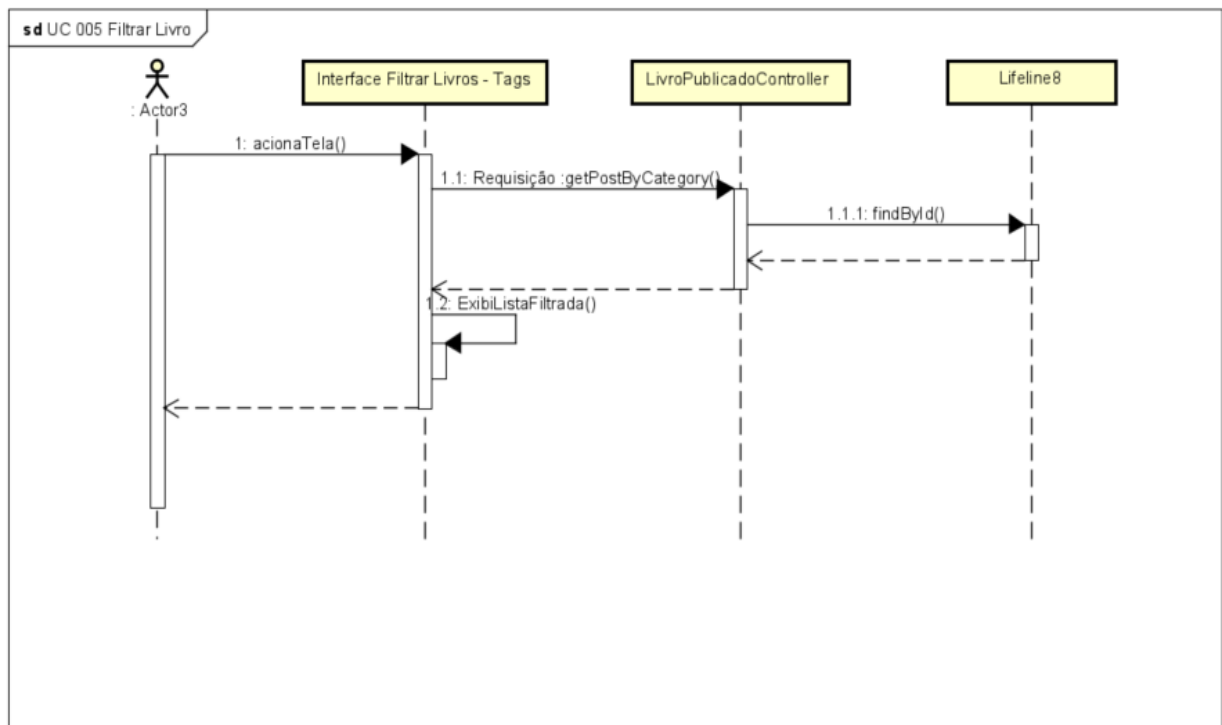
FONTE: Os Autores (2023)

## DIAGRAMA DE SEQUÊNCIA – CADASTRAR LIVRO



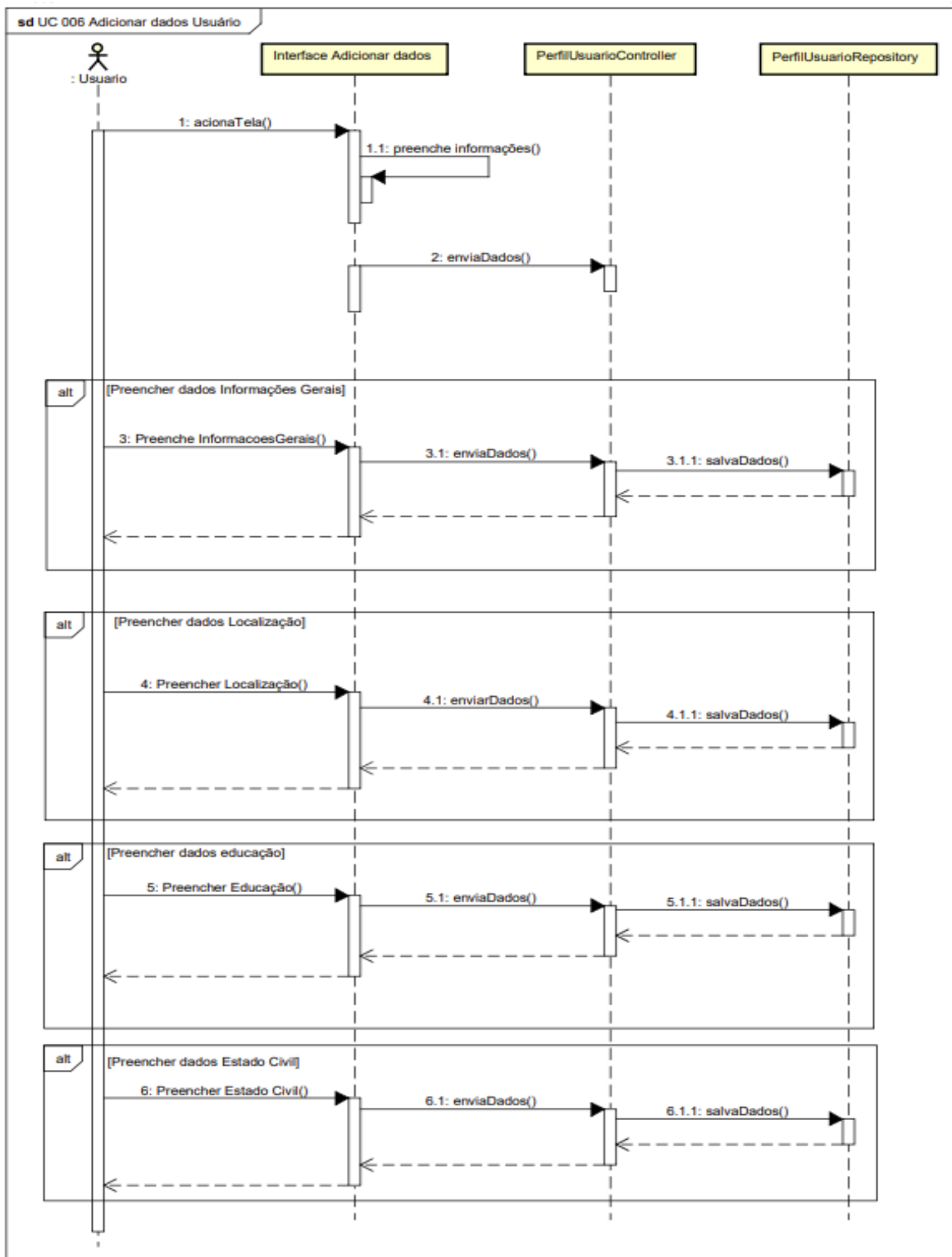
FONTE: Os Autores (2023)

## DIAGRAMA DE SEQUÊNCIA – FILTRAR LIVROS POR CATEGORIA



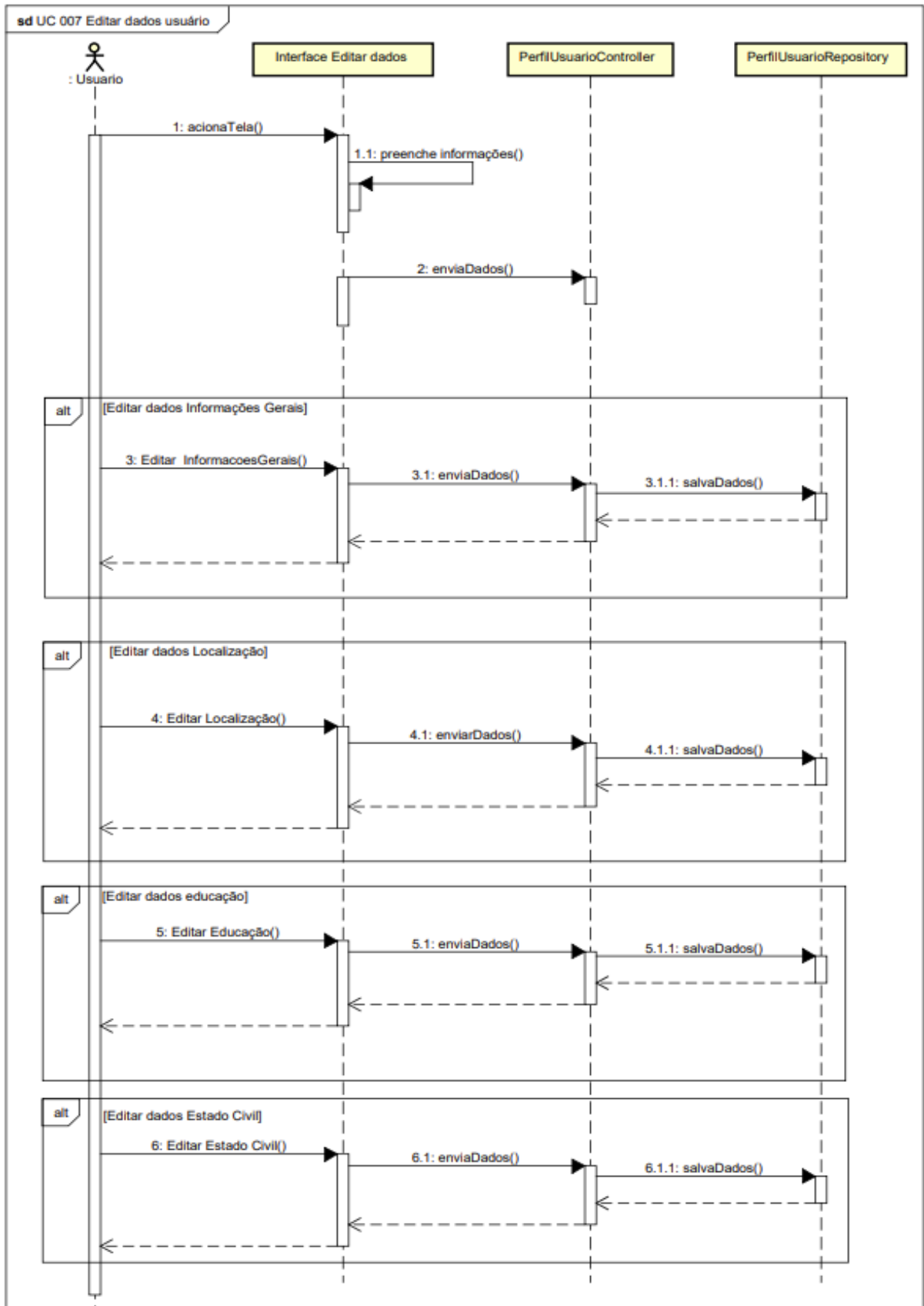
FONTE: Os Autores (2023)

## DIAGRAMA DE SEQUÊNCIA – ADICIONAR DADOS USUÁRIO



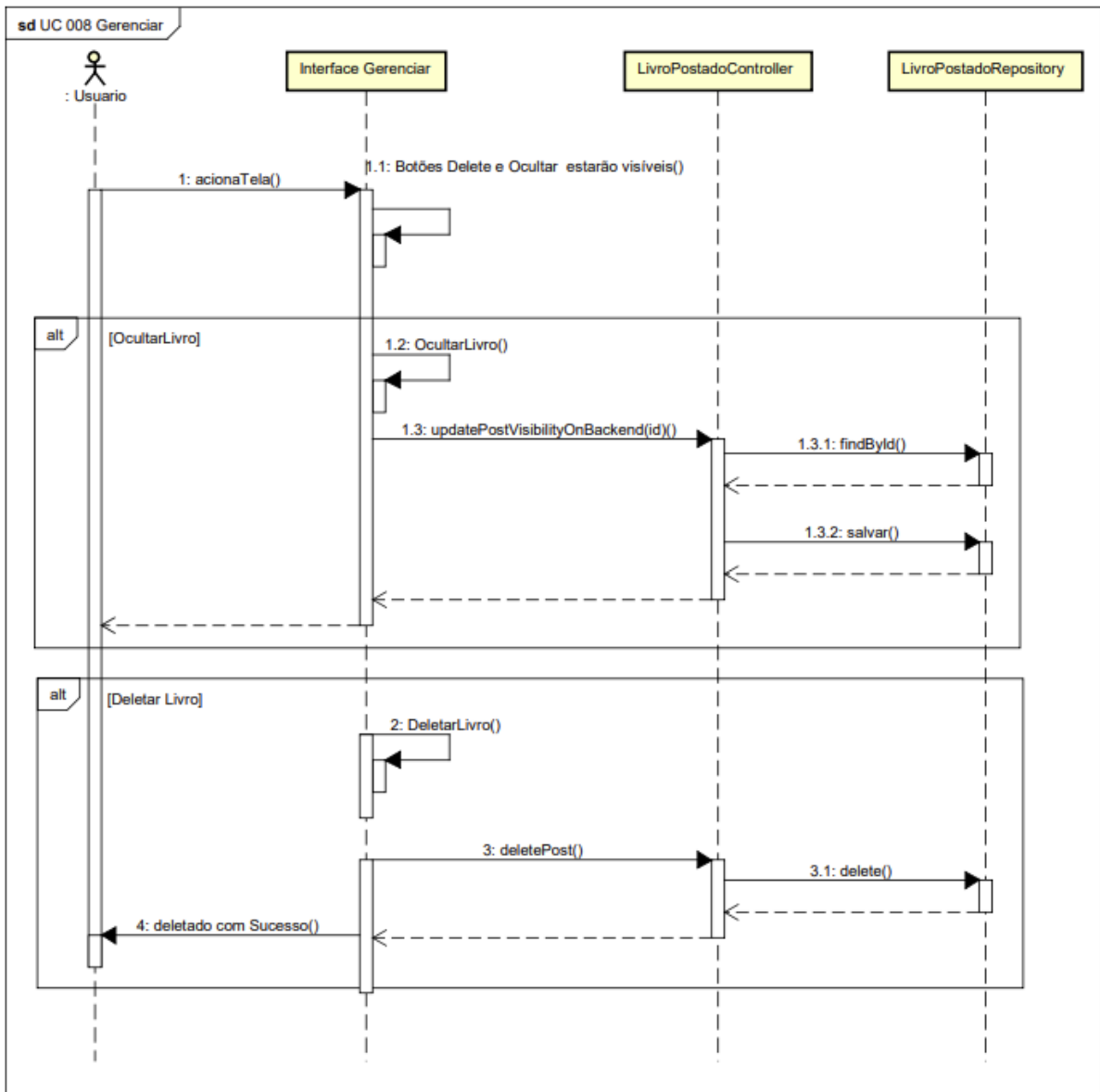
FONTE: Os Autores (2023)

## DIAGRAMA DE SEQUÊNCIA – EDITAR DADOS USUÁRIO



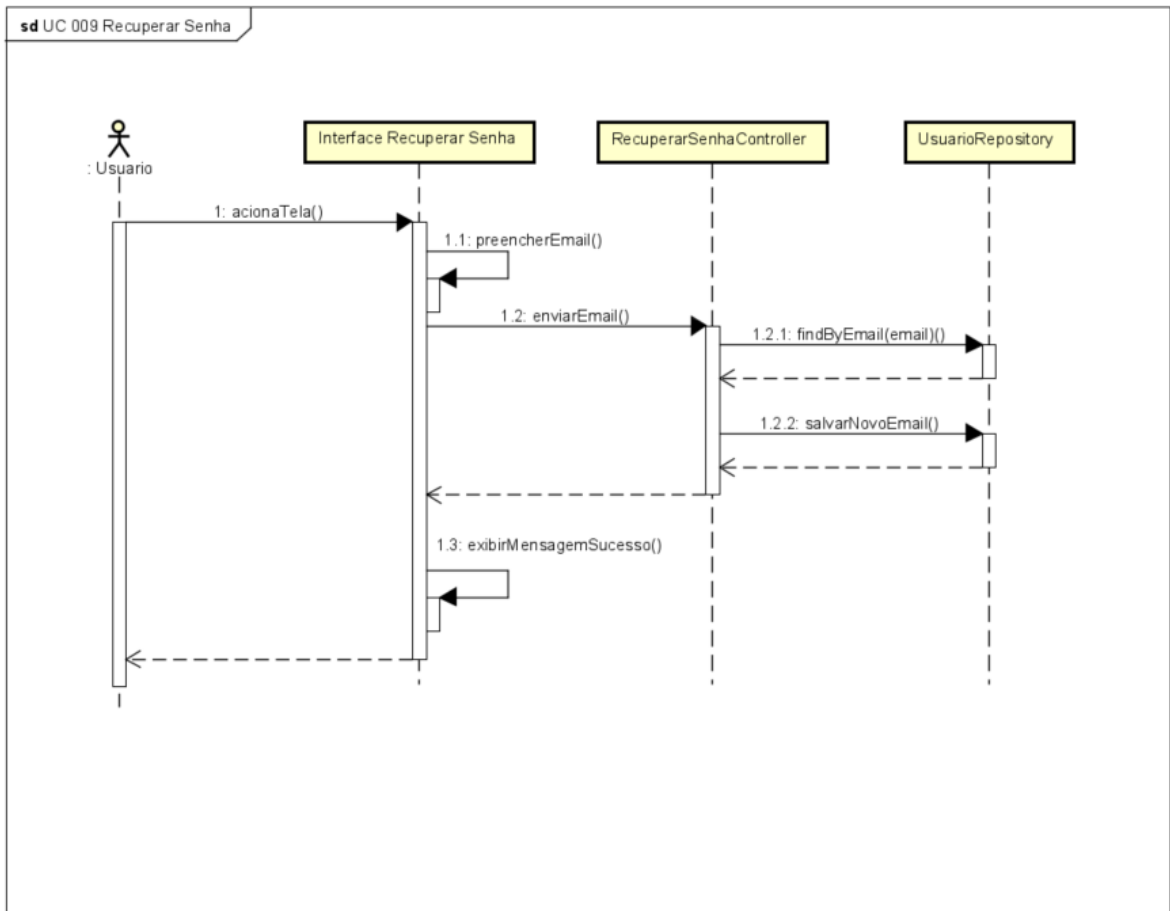
FONTE: Os Autores (2023)

**DIAGRAMA DE SEQUÊNCIA – GERENCIAR**



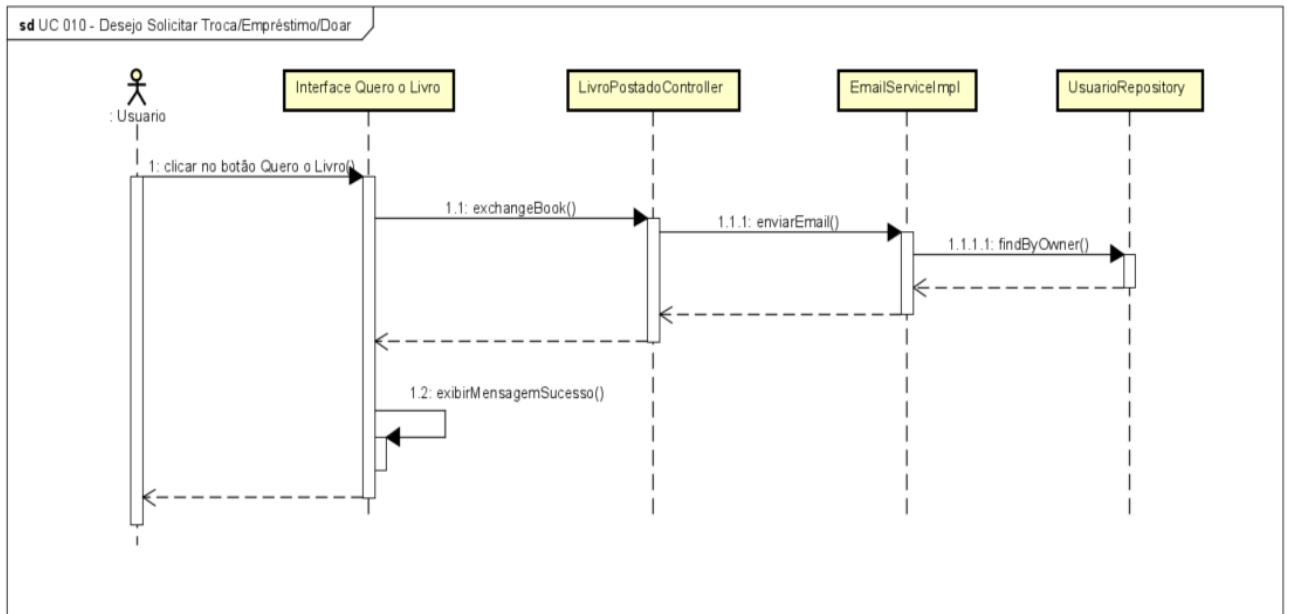
FONTE: Os Autores (2023)

## DIAGRAMA DE SEQUÊNCIA – RECUPERAR SENHA USUÁRIO



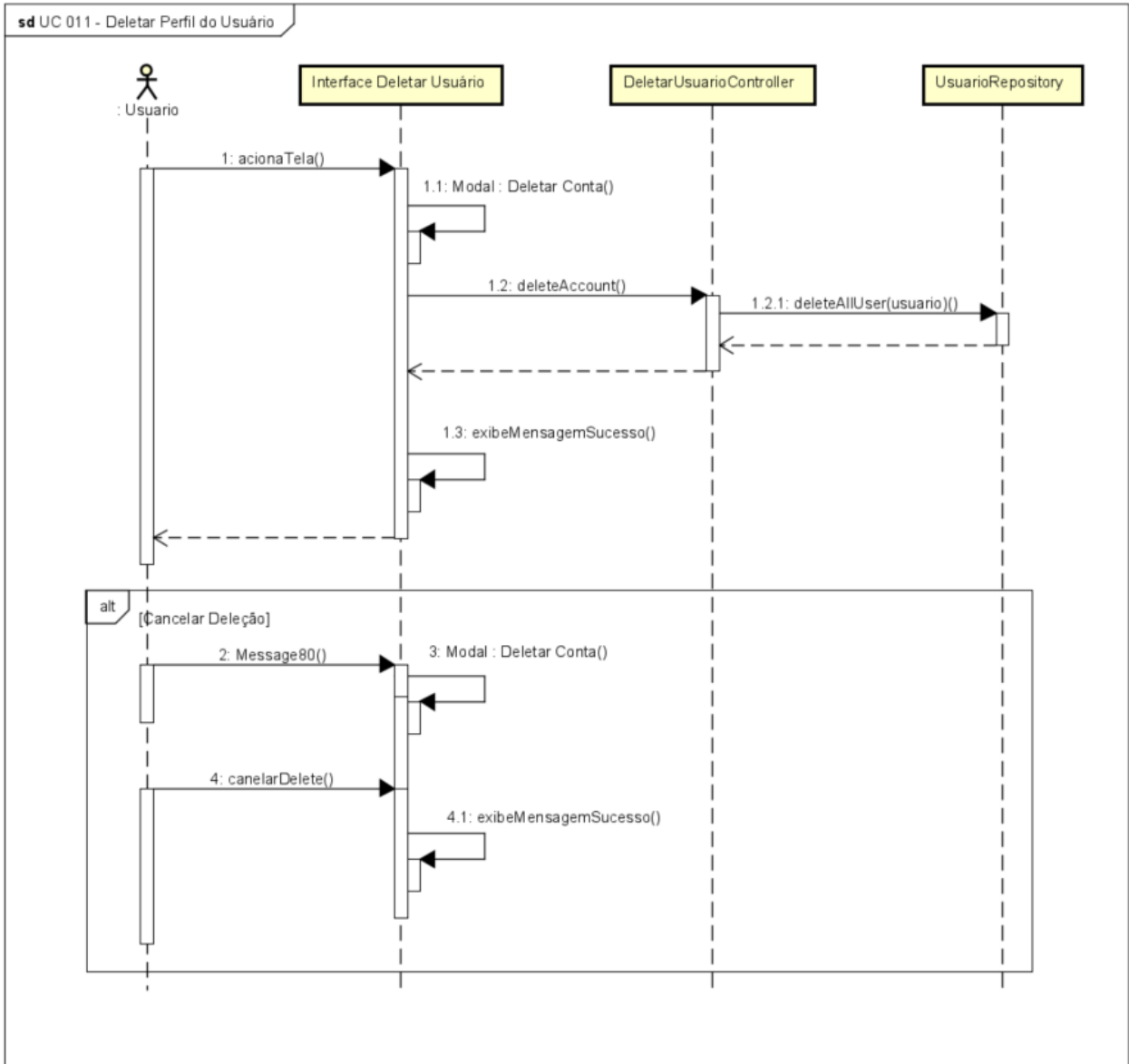
FONTE: Os Autores (2023)

**DIAGRAMA DE SEQUÊNCIA – Desejo Solicitar Troca/Empréstimo/Doar**



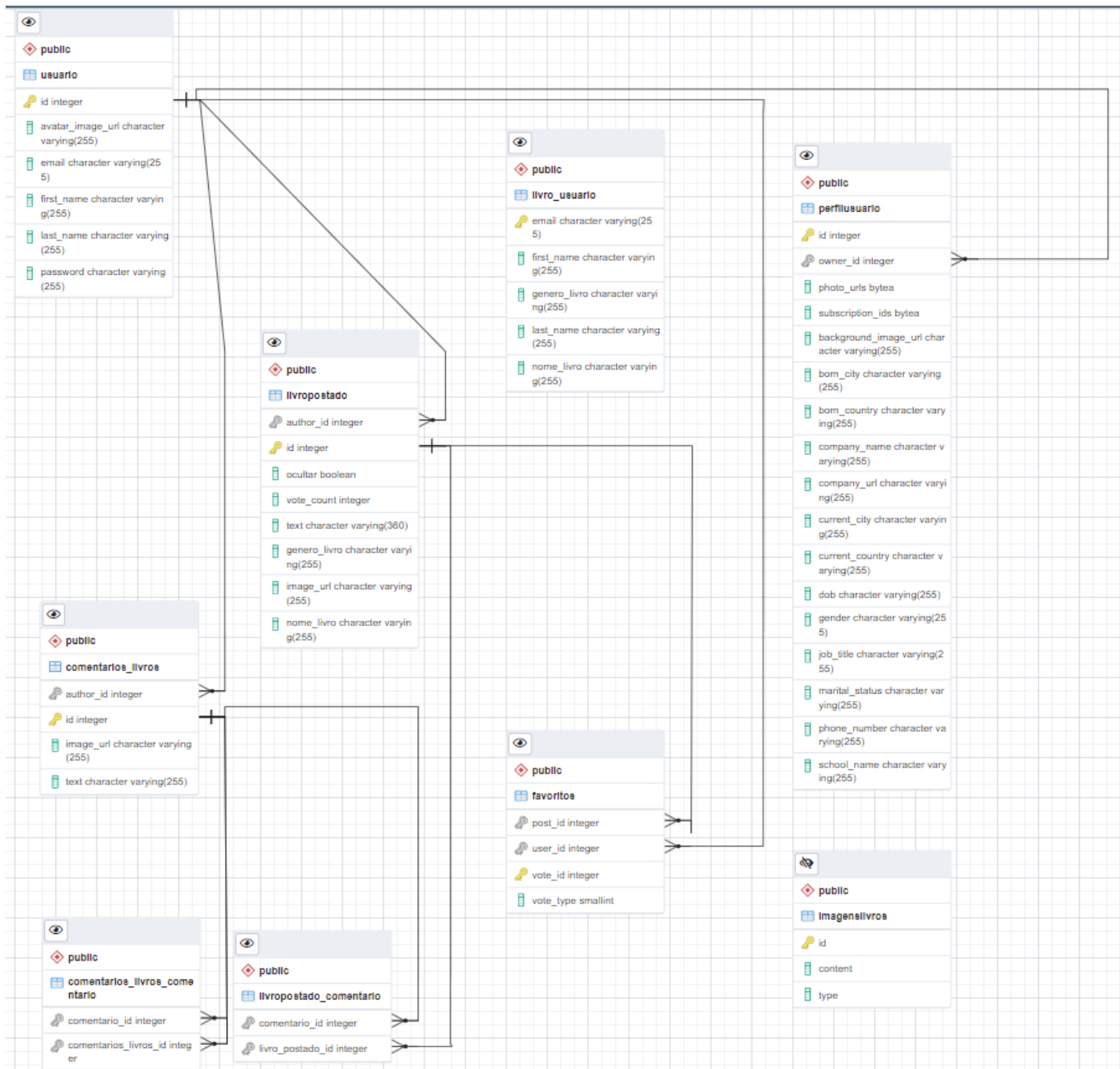
FONTE: Os Autores (2023)

**DIAGRAMA DE SEQUÊNCIA – Deletar Perfil de Usuário.**



FONTE: Os Autores (2023)

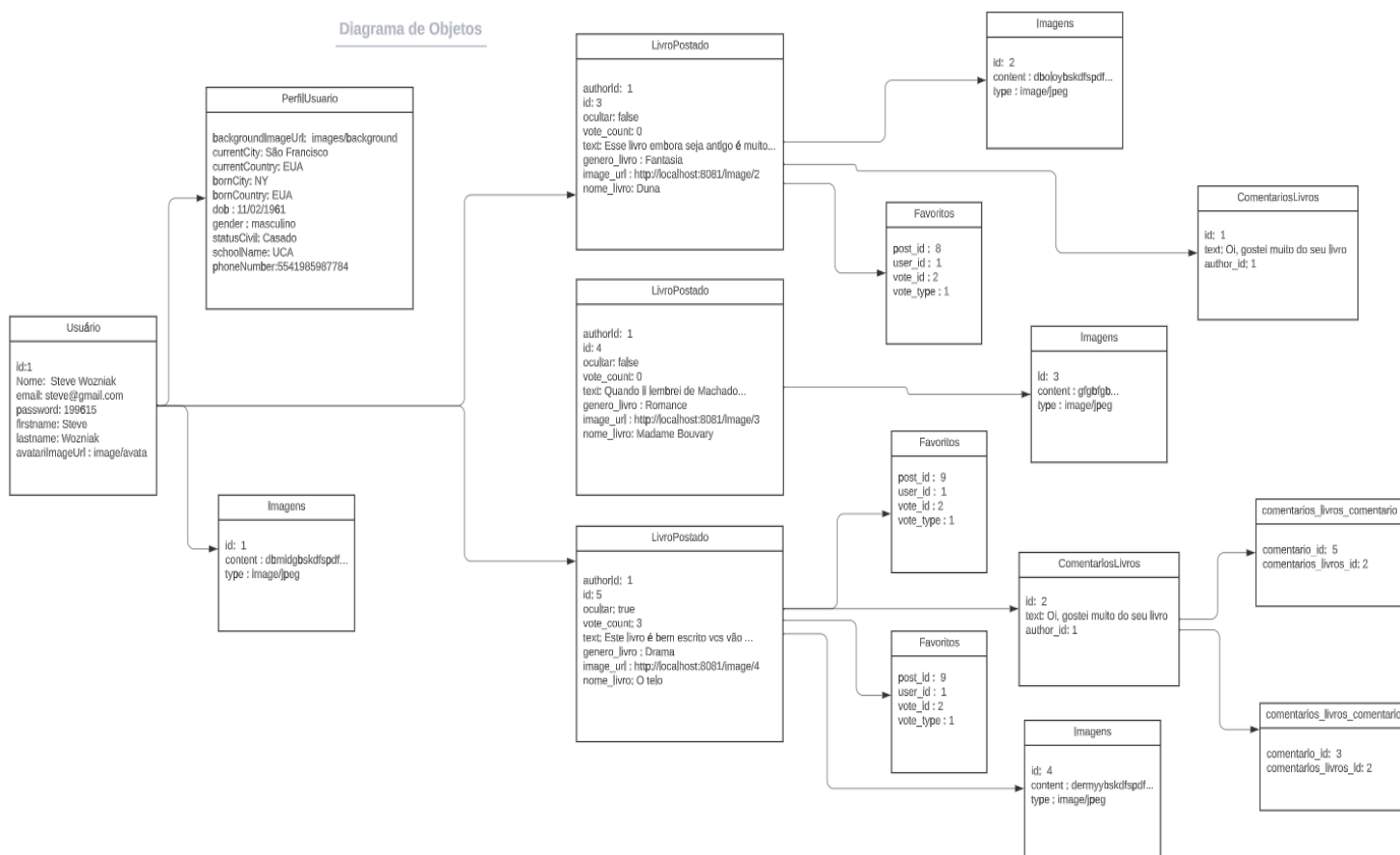
## APÊNDICE G – DIAGRAMA ENTIDADE-RELACIONAMENTO



FONTE: Os Autores (2023)

## APÊNDICE H – DIAGRAMA DE OBJETOS

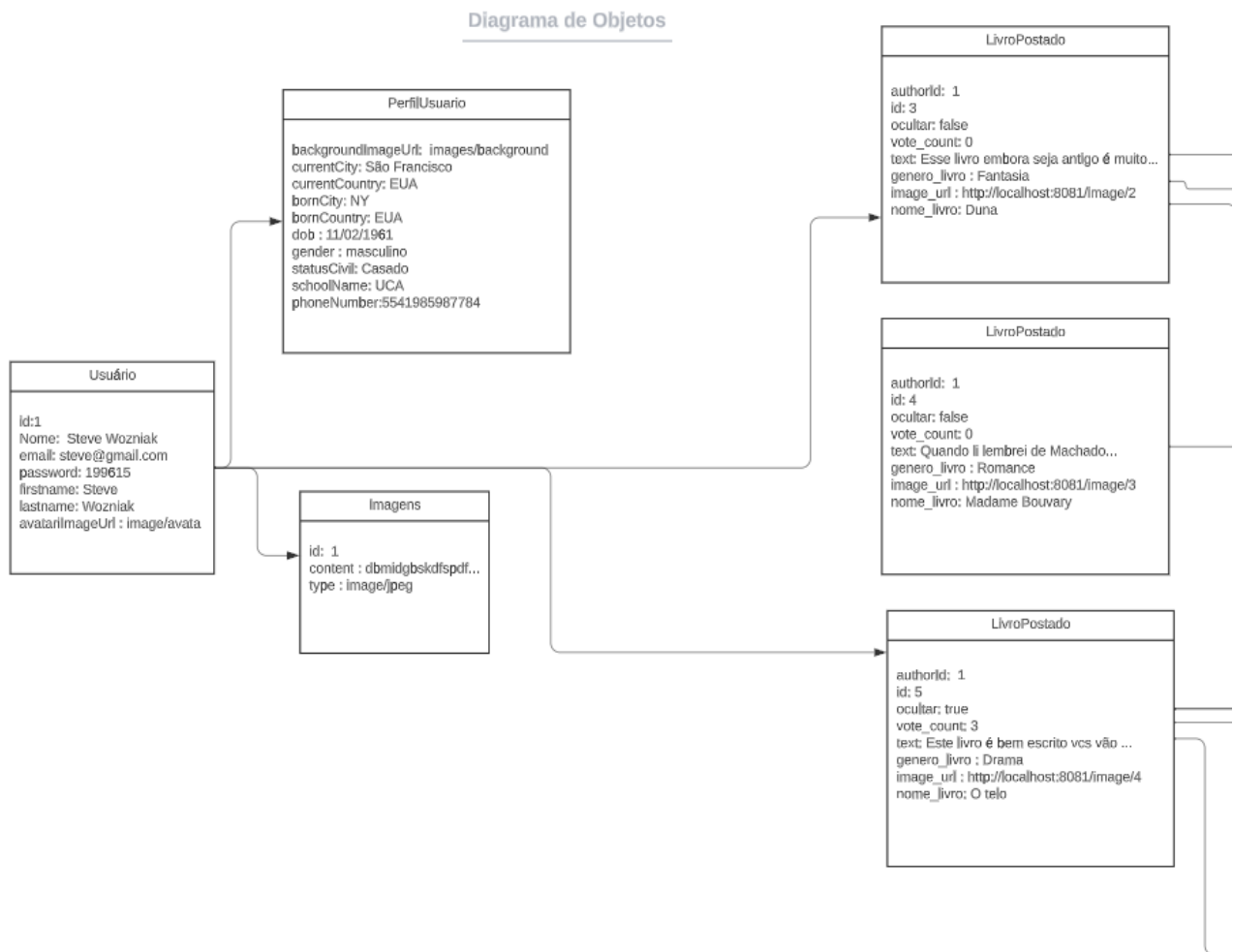
Diagrama de Objetos



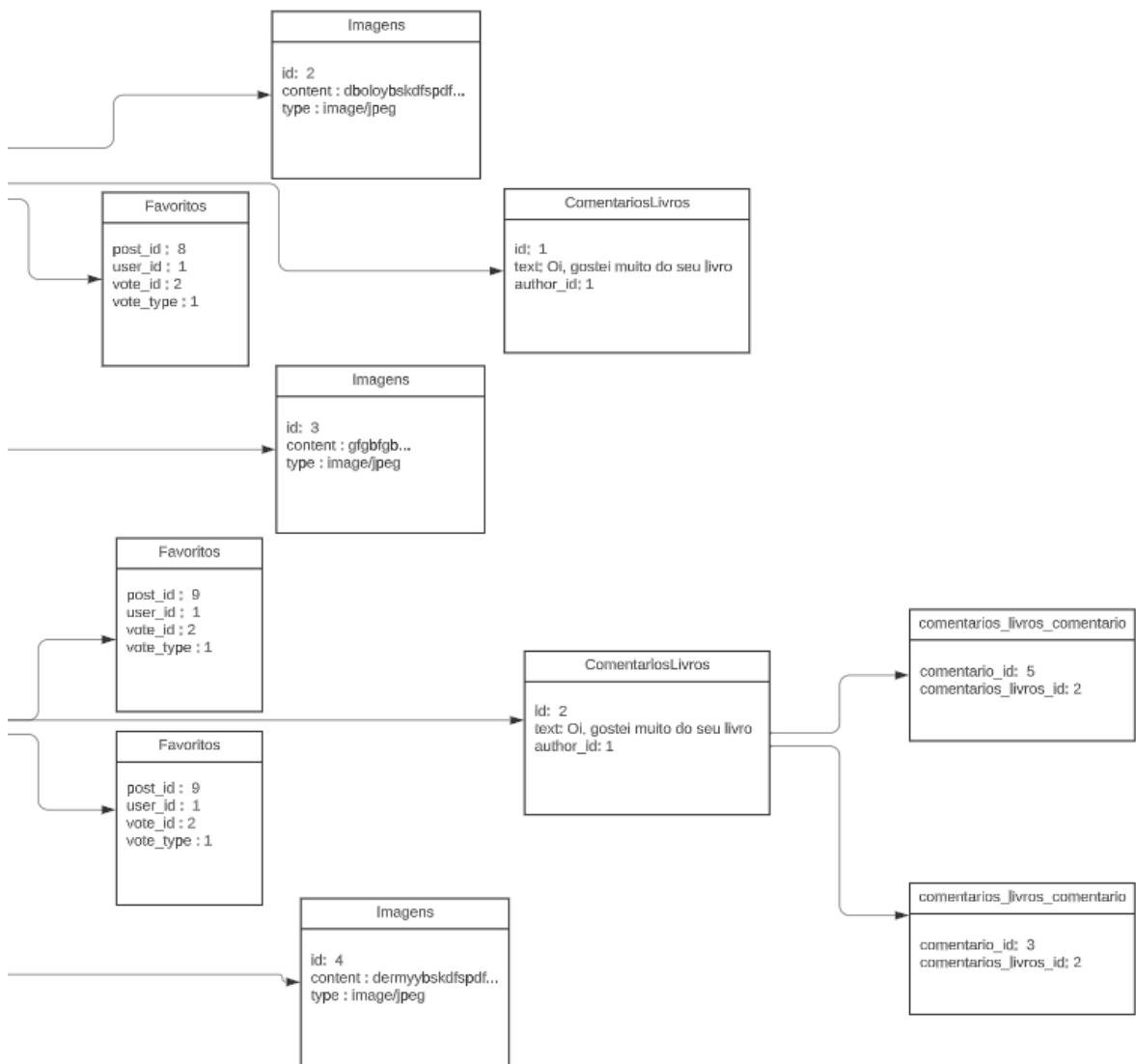
FONTE: Os Autores (2023)

# DIAGRAMA DE OBJETOS

## AMPLIADO



FONTE: Os Autores (2023)



FONTE: Os Autores (2023)