

Lucas Mendonça de Moraes Cavalcante  
Carlos Vinícios de Souza Azeredo

# **Sistema Computacional para o Gerenciamento de Empréstimo de Livros entre Usuários**

Rio das Ostras

2023

Lucas Mendonça de Moraes Cavalcante  
Carlos Vinícios de Souza Azeredo

## **Sistema Computacional para o Gerenciamento de Empréstimo de Livros entre Usuários**

Trabalho de Conclusão de Curso apresentado  
ao curso de graduação em Ciência da Compu-  
tação, da Universidade Federal Fluminense,  
como requisito parcial à obtenção do grau de  
Bacharel em Ciência da Computação

Universidade Federal Fluminense – UFF

Orientador: Dalessandro Soares Vianna

Rio das Ostras

2023

Lucas Mendonça de Moraes Cavalcante

Carlos Vinícios de Souza Azeredo

Sistema Computacional para o Gerenciamento de Empréstimo de Livros entre  
Usuários – Rio das Ostras, 2023

47p. : il. (algumas color.) ; 30 cm.

Orientador: Dalessandro Soares Vianna

Monografia (Graduação) – Universidade Federal Fluminense – UFF, 2023.

1. Sistema de Biblioteca. 2. Empréstimo de Livros. 3. Desenvolvimento de software.  
4. Acervo. I. Dalessandro Soares Vianna. II. Universidade Federal Fluminense – UFF.  
III. Sistema Computacional para o Gerenciamento de Empréstimo de Livros entre  
Usuários.

Lucas Mendonça de Moraes Cavalcante  
Carlos Vinícios de Souza Azeredo

## **Sistema Computacional para o Gerenciamento de Empréstimo de Livros entre Usuários**

Trabalho de Conclusão de Curso apresentado  
ao curso de graduação em Ciência da Compu-  
tação, da Universidade Federal Fluminense,  
como requisito parcial à obtenção do grau de  
Bacharel em Ciência da Computação

Trabalho aprovado. Rio das Ostras, data da aprovação (exemplo: 24 de novembro  
de 2012):

---

**Dalelessandro Soares Vianna** – Orientador  
Universidade Federal Fluminense – UFF

---

**André Renato Villela da Silva**  
Universidade Federal Fluminense – UFF

---

**Marcos Ribeiro Quinet de Andrade**  
Universidade Federal Fluminense – UFF

Rio das Ostras  
2023

*Dedicamos este trabalho primeiramente a Deus e as nossas famílias e professores. Sem eles não teríamos chegado até aqui.*

# Agradecimentos

Em primeiro lugar, a Deus, que fez com que nossos objetivos fossem alcançados, durante todos os anos de estudos.

Aos nossos pais e irmãos, que nos incentivaram nos momentos difíceis e compreenderam a nossa ausência enquanto nos dedicávamos à realização deste trabalho.

Ao professor Dalessandro Vianna, por ter sido nosso orientador e ter desempenhado tal função com dedicação e amizade.

*“Entrega teu caminho ao senhor,  
confia nele, e ele o fará.”  
(Bíblia Sagrada, Salmos 37, 5)*

# Resumo

Os sistemas de bibliotecas permitem o empréstimo de livros de uma instituição para seus usuários. Eles também poderiam gerenciar empréstimos entre seus participantes, pois existem pessoas que querem contribuir com o acervo, mas não conhecem as pessoas que possam se aproveitar das suas contribuições. O sistema computacional desenvolvido neste trabalho tem como objetivo permitir que esses integrantes possam aumentar a quantidade disponível de livros em uma biblioteca, fazendo eles mesmos os empréstimos entre si. Para o alcance deste objetivo, primeiramente foi feita uma busca na literatura por trabalhos semelhantes. Em seguida, foi feita a extração de requisitos, a análise de regras de negócio, a definição de casos de uso, a modelagem das classes do sistema e a modelagem do banco de dados. Após a modelagem realizada, foi realizada a implementação do sistema, seguido de testes e correções para completar a primeira versão contendo todas as funcionalidades descritas nos Casos de Uso. O sistema proposto foi desenvolvido utilizando as tecnologias Java, HTML, CSS e Javascript, SpringBoot e MySQL. A principal contribuição deste trabalho é permitir uma maior disponibilidade de livros aos seus usuários, uma vez que permite o que os próprios usuários também disponibilizem os livros.

**Palavras-chave:** Sistema de biblioteca. Empréstimo de Livros. Desenvolvimento de software. Acervo.



# Abstract

Library systems allow the borrowing of books from an institution to its users. They could also manage loans among their participants, as there are people who want to contribute to the collection but don't know people who can take advantage of their contributions. The computational system developed in this work aims to allow these members to increase the amount of books available in a library, making loans to each other. To achieve this objective, a search was first made in the literature for similar works. Next, requirements were extracted, business rules were analyzed, use cases were defined, system classes were modeled, and the database was modeled. After the modeling was carried out, the system was implemented, followed by tests and corrections to complete the first version containing all the functionalities described in the Use Cases. The proposed system was developed using Java, HTML, CSS and Javascript, SpringBoot and MySQL technologies. The main contribution of this work is to allow a greater availability of books to its users, since it allows the users themselves to also make the books available.

**Keywords:** Library systems. Book borrowing. Software development. Collection.

# Lista de ilustrações

Figura 1 – Página da biblioteca UNIVATES . . . . .	18
Figura 2 – Diagrama de Classes do Sistema . . . . .	34
Figura 3 – Modelo Relacional de Banco de Dados . . . . .	34
Figura 4 – Tela de Login . . . . .	36
Figura 5 – Menu Principal Bibliotecário . . . . .	36
Figura 6 – Tela de Usuários . . . . .	37
Figura 7 – Tela de Cadastro de Usuários no Sistema . . . . .	37
Figura 8 – Tela de Remoção de Usuários . . . . .	38
Figura 9 – Tela de Livros . . . . .	38
Figura 10 – Tela de Cadastro de Livro no Sistema . . . . .	39
Figura 11 – Tela de Remoção de Livros no Sistema . . . . .	39
Figura 12 – Tela de Consulta de Livros no Sistema . . . . .	40
Figura 13 – Tela de Livros Retornados na Busca . . . . .	40
Figura 14 – Tela de Empréstimos . . . . .	41
Figura 15 – Consulta de Livros para Empréstimo . . . . .	41
Figura 16 – Livros Consultados . . . . .	42
Figura 17 – Consulta de Livros para Devolução . . . . .	42
Figura 18 – Menu Principal Usuário . . . . .	43
Figura 19 – Tela de Empréstimos . . . . .	43

# Lista de quadros

Quadro 1 – Regras de Negócio . . . . .	27
Quadro 2 – Requisitos Funcionais . . . . .	28
Quadro 3 – Requisitos Não Funcionais . . . . .	28

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>13</b>
<b>1.1</b>	<b>Contextualização</b>	<b>13</b>
<b>1.2</b>	<b>Objetivos</b>	<b>13</b>
<b>1.3</b>	<b>Justificativa</b>	<b>14</b>
<b>1.4</b>	<b>Delimitação</b>	<b>14</b>
<b>1.5</b>	<b>Organização do Texto</b>	<b>15</b>
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>16</b>
<b>2.1</b>	<b>Sistemas de Biblioteca</b>	<b>16</b>
2.1.1	Histórico	16
2.1.2	Sistema UNIVATES	18
<b>2.2</b>	<b>Tecnologias</b>	<b>19</b>
2.2.1	HTML	19
2.2.2	CSS	19
2.2.3	Javascript	20
2.2.4	Sistema Gerenciador de Banco de Dados MySQL	20
2.2.5	Spring Boot	20
2.2.6	Java	21
2.2.7	Padrão MVC	21
<b>3</b>	<b>METODOLOGIA</b>	<b>22</b>
<b>3.1</b>	<b>Conceitos</b>	<b>22</b>
3.1.1	Extração de Requisitos	22
3.1.2	Análise das Regras de Negócio	22
3.1.3	Definição e Descrição dos Casos de Uso	23
3.1.4	Modelagem de Classes do Sistema	24
3.1.5	Modelagem do Banco de Dados	24
3.1.6	Implementação	25
3.1.7	Testes e Correções	25
<b>3.2</b>	<b>Etapas do desenvolvimento</b>	<b>25</b>
<b>4</b>	<b>SISTEMA COMPUTACIONAL</b>	<b>27</b>
<b>4.1</b>	<b>Análise de Domínio e Regras de Negócio</b>	<b>27</b>
<b>4.2</b>	<b>Análise e Extração De Requisitos</b>	<b>28</b>
<b>4.3</b>	<b>Definição de Casos de Uso</b>	<b>28</b>
<b>4.4</b>	<b>Modelagem das Classes do Sistema</b>	<b>33</b>

<b>4.5</b>	<b>Modelagem do Banco de Dados do Sistema</b>	<b>34</b>
<b>4.6</b>	<b>Implementação</b>	<b>35</b>
<b>4.7</b>	<b>Navegação e Utilização do Sistema</b>	<b>35</b>
4.7.1	Perspectiva do Bibliotecário	35
4.7.2	Perspectiva do Usuário	40
<b>5</b>	<b>CONSIDERAÇÕES FINAIS</b>	<b>44</b>
<b>5.1</b>	<b>Conclusão</b>	<b>44</b>
<b>5.2</b>	<b>Trabalhos futuros</b>	<b>45</b>
	<b>REFERÊNCIAS</b>	<b>46</b>

# 1 Introdução

## 1.1 Contextualização

Em um ambiente acadêmico, ou em qualquer outro ambiente escolar de ensino e pesquisa, se faz presente uma grande demanda por informação, que pode ser solucionada por meio de acesso a internet, mídias digitais, documentos ou livros. Para suprir parte importante dessa demanda surge a biblioteca, com o propósito de aumentar as bases de pesquisa e apoio aos alunos.

A biblioteca tem um grande papel dentro de um ambiente acadêmico ou de estudo, por meio dela é possível complementar as aulas, realizar pesquisas, revisar conteúdos, ou seja, um leque considerável de possibilidades que enriquecem o aprendizado dos alunos, (HUBNER; KUHN, 2017). A biblioteca também apresenta um papel social importante, pois possibilita que alunos com uma renda mais baixa que não possuem recursos financeiros para comprar livros possam ter acesso a livros que os auxiliem durante o aprendizado. Esta é uma questão que ocorre em todas instituições e principalmente em instituições públicas onde parte significativa dos alunos possuem uma renda menor.

No entanto, existem algumas limitações dentro da estrutura da biblioteca que podem ser tratadas gerando melhorias em seus serviços. Geralmente, uma biblioteca acadêmica possui uma quantidade significativa de exemplares e documentos, mas os mesmos acabam se limitando às áreas de conhecimento abordadas pelos cursos da instituição.

Uma outra questão muito importante em uma biblioteca acadêmica é a necessidade de se mapear e configurar um plano para a gestão dos livros e de seus empréstimos aos alunos. Este é um problema complexo que normalmente necessita de um sistema computacional interno que fique à disposição dos funcionários e administradores da biblioteca. O trabalho de Alvarenga (2001) traz maiores detalhes sobre a complexidade existente no processo de gestão de bibliotecas.

Dito isto, é possível desenvolver um sistema eletrônico que solucione a questão da complexidade da gestão dos empréstimos de livros e ao mesmo tempo aumente o escopo do seu acervo o expandindo para mais áreas de conhecimento que não necessariamente se limitam aos objetos de estudo dos cursos da instituição?

## 1.2 Objetivos

O objetivo geral deste trabalho de conclusão é desenvolver um sistema computacional que apoie a biblioteca nos seus serviços de gestão de empréstimos de livros e

que contorne a limitação do seu acervo através de funcionalidades que tornem viável o empréstimo de livros pessoais diretamente entre os usuários, como alunos e professores, por exemplo.

Para que o objetivo geral deste trabalho seja cumprido, é necessário que os seguintes objetivos específicos sejam alcançadas no processo de desenvolvimento do sistema:

- realizar um mapeamento realista do negócio para construir funcionalidades que solucionem as principais demandas de uma biblioteca física;
- implementar a estratégia de compartilhamento de livros entre os frequentadores do ambiente acadêmico de maneira clara e objetiva; e
- implementar uma interface gráfica simplificada e amigável.

### 1.3 Justificativa

Este sistema computacional possibilitará que a biblioteca cadastre os usuários e organize os empréstimos dos livros. Simultaneamente, também tornará possível que outros usuários, como professores e alunos, disponibilizem livros pessoais para serem emprestados entre si.

Com este mecanismo proposto pelo sistema, será possível obter um estímulo maior para a interação social entre alunos e entre professores no ambiente acadêmico, contribuindo para a integração entre alunos e professores e auxiliando no desenvolvimento interpessoal dos alunos. Este é um aspecto muito importante na vida profissional em qualquer área. Além disso, também pode ajudar alunos com problemas de ansiedade ou depressão a se sentirem mais confortáveis e incluídos no ambiente acadêmico; esta é uma questão bem presente nos tempos atuais, podendo ser constatado com os índices de depressão e problemas relacionados ao longo das últimas décadas. No trabalho de [Bolsoni-Silva e Loureiro \(2016\)](#) é abordado o impacto da depressão para as interações sociais de universitários.

Outro ponto muito importante é o fato do sistema possibilitar uma expansão do acervo da biblioteca, gerando o benefício de ter mais material disponível para alunos que não possuem recursos financeiros para comprarem livros novos. Ao mesmo tempo, incentiva alunos e professores a emprestar livros pessoais que muitas vezes ficam subutilizados em suas casas, após serem lidos ou integralmente consultados.

### 1.4 Delimitação

Neste projeto, não será realizado um processo de implantação integral do sistema computacional desenvolvido. Será implementado uma versão web inicial que contemple

todas as funcionalidades propostas. Além disso, o escopo deste trabalho não engloba o desenvolvimento de um aplicativo para dispositivos móveis com as mesmas funcionalidades do sistema proposta. No entanto, é considerada a possibilidade de futura expansão do sistema com base nos requisitos estabelecidos inicialmente.

## 1.5 Organização do Texto

O restante deste trabalho será dividido em 4 capítulos. O Capítulo 2, "Fundamentação Teórica", apresentará conceitos e informações necessárias para o adequado entendimento do sistema proposto. A metodologia utilizada para o desenvolvimento do sistema será descrita no Capítulo 3. O sistema computacional desenvolvido será descrito em detalhes no Capítulo 4 e por fim, no Capítulo 5, conclusões e sugestões para eventuais trabalhos futuros serão apresentados.



## 2 Fundamentação Teórica

Este capítulo apresentará informações sobre a história dos sistemas de biblioteca (Seção 2.1) e também detalhes sobre as tecnologias escolhidas para o desenvolvimento do projeto, assim como as justificativas para o uso destas (Seção 2.2).

### 2.1 Sistemas de Biblioteca

Esta seção irá descrever os detalhes sobre a história das bibliotecas, desde os mais simples armazenamentos de informação até os sistemas eletrônicos que existem atualmente (Seção 2.1.1). Além disso, será abordado um sistema que compartilha similaridades com o projeto desta monografia (Seção 2.1.2)

#### 2.1.1 Histórico

A necessidade de guardar informação foi algo vital para a sobrevivência humana ao longo de sua história, e uma das formas mais eficazes de guardar informação é a partir da escrita. Os primeiros registros de uma escrita sistêmica aparecem na mesopotâmia com os hieróglifos (FISCHER, 2009), dominados apenas pelas mais altas camadas da sociedade, serviam principalmente para marcações em túmulos e templos. E foi também na mesopotâmia onde se encontravam os primeiros acervos feitos de argila em formato de prateleiras, que organizavam registros importantes em um lugar específico, servindo justamente como uma espécie de biblioteca.

A organização desses registros foi evoluindo conforme a sociedade, com a escrita se popularizando devido às demandas da sociedade, e com isso o volume de registros assim como sua organização foram aumentando, resultando na biblioteca de Alexandria no norte do Egito (BARBIER, 2019), a maior quantidade de conhecimento da antiguidade concentrado em um só lugar. A biblioteca também contava com laboratórios, zoológico, observatório e jardim botânico, não sendo somente um acervo de informação, mas um centro cultural que contava muito com a interação dos indivíduos presentes. A biblioteca sofreu vários ataques e perda de material durante sua história, mas sempre foi visto como um marco da sociedade, sendo sua destruição considerada uma das maiores perdas da humanidade, demonstrando sua importância em relação à sociedade, tanto antiga como atualmente.

Mesmo na Idade Média, as bibliotecas se encontravam muito presentes. Com o aumento das obras religiosas, foi necessário novamente a criação de um lugar para mantê-las. Mesmo com uma mudança de paradigma, as características de uma biblioteca

permaneceram semelhantes, pois se mantinham de difícil e limitado acesso. Para tal acesso, era necessário ter prestígio ou posição de poder nas instituições religiosas, até porque eram elas que contralavavam esse tipo de informação.

A partir do Renascimento, houve um ressurgimento do interesse pelas ciências e artes, que antes eram fortemente reguladas pela igreja. Nesse contexto, surgiram novamente bibliotecas focadas em registros não religiosos, com ênfase na disseminação do conhecimento pela sociedade e no renascimento da ciência. O papel do bibliotecário se tornou de suma importância para a disseminação dessas informações, pois ajudou a remover muitas das restrições impostas anteriormente, facilitando o acesso ao conhecimento para um público mais amplo.

Depois do século XVII, as bibliotecas se tornaram muito mais acessíveis, ganhando uma grande importância no setor social e se tornando referência por todos as camadas da sociedade na busca por informação. Passaram a ter também uma grande importância no processo de democratização da leitura e escrita, com acervos de livros gerais, gratuitos e abertos ao público regularmente. Em tese, qualquer um poderia buscar informações e conhecimento.

Com o advento da tecnologia, a informação nunca foi tão valiosa, sendo usada nos mais diversos meios e formas. Surgiram também novas formas de armazenamento que atendem diferentes demandas, como foi o exemplo do CD-ROM que no ambiente das bibliotecas, foi uma nova estratégia para o resgate de informações (OHIRA; PRADO, 2002). As bibliotecas sempre acabam ressurgindo ou se adaptando à situação da sociedade, e atualmente não poderia ser diferente. A adaptação das bibliotecas ao meio virtual se apresentou de duas maneiras, uma adaptação parcial e uma adaptação total. No modelo de adaptação parcial, reservas de livros físicos podem ser gerenciadas ou até mesmo feitas virtualmente; o interessado tendo somente que comparecer no local para pegar o livro. No modelo de adaptação total, os livros são reservados, gerenciados e lidos virtualmente, pois com a popularização de computadores pessoais, telefones e smartphones, qualquer dispositivo pode se tornar um livro completo, existindo até dispositivos especificamente construídos com esse propósito como o *Kindle*<sup>1</sup>.

Alguns sistemas de biblioteca já foram idealizados e montados no ambiente virtual. Como bibliotecas sempre estão mudando e evoluindo com a sociedade, elas também têm que se adaptar a novos paradigmas e tecnologias; esses sistemas são atualizados constantemente. O surgimento da internet e de programas WEB fez com que esses sistemas pudessem ser acessados remotamente não somente por um bibliotecário, mas também por seus usuários introduzindo facilidades que se tornariam padrões atualmente (SMITH; JOHNSON, 2019).

É possível perceber que desde os tempos mais primitivos até o mais atuais, a

---

<sup>1</sup> *Kindle*: dispositivo de leitura de livros eletrônicos.

biblioteca se encontra com um papel importante na sociedade. Talvez tenha perdido um pouco de espaço para formas mais modernas de comunicação, mas ainda é indispensável no meio social. Sua sobrevivência na história foi vital para o avanço da sociedade, e cabe aos indivíduos da sociedade atual continuar o legado das bibliotecas, a preservando e modernizando para que futuras gerações possam usufruir do conhecimento e continuarem progredindo.

### 2.1.2 Sistema UNIVATES

O sistema de biblioteca da UNIVATES (Figura 1), proposto por Bonfadini e Pretto (2016), foi projetado usando as tecnologias: HTML, PHP, CSS, Javascript e banco de dados PostgreSQL. O projeto tinha como objetivo reconfigurar a interface já existente do sistema da biblioteca universitária. A partir de conceitos de Interface Humano Computador e de Engenharia de Software, melhorar a usabilidade do sistema com boas práticas, visto também nesse projeto, com o uso de paradigmas já estabelecidos e bem conceituados na área, sendo um dos pontos vitais da aplicação.

Figura 1 – Página da biblioteca UNIVATES



Fonte: Bonfadini e Pretto (2016).

Diferente do sistema proposto neste projeto, ele mantém a abordagem clássica, tendo a biblioteca como acervo de livros e seus usuários podendo reservar, avaliar, buscar e devolver livros que já estão presentes na biblioteca, mas não há uma maneira de usuários interessados disponibilizarem seus próprios livros para poderem fazer empréstimos entre si, o que o atual projeto se propõe a resolver e adicionar no leque de possibilidades para este tipo de sistema.

A proposta do sistema UNIVATES se concentrava em melhorar a experiência do usuário, substituindo por um período de 20 dias o sistema padrão. A Figura 1 mostra uma

tela usada nesse período. Ao monitorar as ações do sistema foi possível fazer uma coleta de dados, a qual mostrou que a ênfase em boas práticas de interface resultaram em um baixo tempo de resolução, alto índice de aceitação e baixo grau de dificuldade, atingindo seus objetivos iniciais.

## 2.2 Tecnologias

Esta seção apresentará as tecnologias utilizadas, dando uma breve explicação sobre elas e o porquê de serem utilizadas neste projeto.

### 2.2.1 HTML

HTML (Linguagem de Marcação de Hipertexto) é a tecnologia padrão do mercado para construção de páginas WEB, responsável por definir estruturas e significados de seus elementos, como textos, imagens e vídeos. A primeira versão do HTML, criada por Tim Berners-Lee, foi disponibilizada em 1991 e tinha como objetivo inicial interligar instituições de pesquisa próximas e compartilhar documentos com facilidade, visto que a utilização e navegação na internet no fim dos anos 80 estava em um dos seus estados iniciais e necessitava de conhecimento técnico para realizar tal atividade. Com o passar das décadas, o HTML foi evoluindo, ganhando popularidade e se integrando a outras tecnologias, transformando o desenvolvimento WEB até a forma atual para este projeto. Tendo em vista sua popularidade e facilidade de aprendizado, foi a tecnologia escolhida para este projeto ([DUCKETT, 2011](#)).

### 2.2.2 CSS

CSS (Folha de Estilo em Cascata) é uma tecnologia utilizada para estilizar a forma que o conteúdo de uma página WEB é apresentada nos navegadores. Ela permite que sejam definidas várias características dos elementos de uma página WEB, como alinhamento, posicionamento, margem, preenchimento, cor de fundo e transparência. Esta tecnologia também permite que o conceito de responsividade seja implementado e incorporado as páginas, possibilitando que o estilo se adeque de acordo com os diferentes tamanhos que as telas irão exibir, algo muito importante pelo fato de que aplicações WEB executam em diferentes plataformas e dispositivos que diferem entre si em diversos aspectos. A tecnologia CSS foi proposta pela primeira vez em Outubro de 1994 por Hakon Lie, com o objetivo de facilitar a programação de sites, a partir da representação de conteúdo e elementos de uma página WEB. Ao longo do tempo, o CSS evolui e ganhou popularidade, tornando possível o desenvolvimento de interfaces gráficas melhores e mais amigáveis aos usuários, o que contribui bastante para o crescimento de serviços e sistemas WEB, uma vez que facilita sua utilização comunicando melhor ao usuário com suas funcionalidades através de aspectos

gráficos. Atualmente, o CSS é utilizado em grande escala no mercado de desenvolvimento para este projeto, seja de forma nativa ou integrado em frameworks (MEYER, 2018). Por este motivo, utilizou-se CSS na construção das interfaces deste trabalho.

### 2.2.3 Javascript

Javascript é uma linguagem de programação interpretada de script em alto nível com tipagem fraca e dinâmica. Apesar da sua natureza mais funcional, é possível desenvolver aplicações em outros paradigmas utilizando a linguagem. Javascript surgiu como uma tecnologia front-end que executava nos navegadores e tinha como principal objetivo tornar as páginas WEB mais dinâmicas. Com o Javascript, os navegadores passaram a ter a capacidade de responder e validar interações e entradas do usuário, alterar estrutura da página, fazer uma comunicação mais simples com o servidor e outros diversos benefícios. Com o passar do tempo, a linguagem Javascript evolui e passou a ser muito utilizada em diversos sistemas em produção atualmente. De forma conjunta com as tecnologias HTML e CSS, constitui o grupo das principais tecnologias de desenvolvimento na WEB (FLANAGAN, 2012). Atualmente, ele serve a propósitos que não faziam parte da sua proposta inicial, sendo utilizado em diversos setores nas aplicações, como a construção do back-end, suporte para construção de aplicações mobile e desktop em conjunto com outras tecnologias e frameworks. No sistema proposto por este trabalho, a linguagem Javascript é utilizada principalmente na comunicação entre o cliente e servidor.

### 2.2.4 Sistema Gerenciador de Banco de Dados MySQL

MySQL é um sistema gerenciador de banco de dados (SGBD) que utiliza a linguagem SQL (*Structured Query Language*), realizando operações com base na abordagem relacional. O MySQL é um dos mais populares SGBD's no ramo e se comporta muito bem em aplicações que não possuem alta demanda de escalabilidade. Ele é um sistema leve, simplificado, de fácil utilização e configuração em comparação a outros SGBDS relacionais, possuindo ferramentas externas auxiliares que facilitam muito no desenvolvimento do software e gerenciamento do banco de dados. O MySQL se apresenta muitas vezes como a escolha certa em sistemas de pequeno e médio porte, podendo também ser utilizado em grandes projetos em que não há forte demanda por escalabilidade, (MURACH, 2019). No desenvolvimento do sistema proposto neste trabalho, foi utilizado o MySQL por dois motivos principais: baixa demanda por escalabilidade e por ter seu desenvolvimento beneficiado pela leveza e simplicidade sem comprometer a aplicação.

### 2.2.5 Spring Boot

O Spring Boot é um framework utilizado para facilitar as configurações de uma aplicação. Ele é uma estrutura de desenvolvimento back-end utilizado para o desenvol-

vimento de serviços WEB e APIs de forma mais simplificada e ágil, (BOAGLIO, 2017). Apesar de possuir uma forte conexão com Java, é possível utilizá-lo em outras linguagens como Groove e Kotlin. O Spring Boot simplifica a configuração de servidores e a criação de serviços WEB. O Spring Boot possui recursos que facilitam o desenvolvimento do back-end e do banco de dados. Sua integração com frameworks ajuda a realizar o mapeamento objeto relacional entre o paradigma de orientação a objetos e o paradigma relacional de bancos de dados, sendo possível ter o desenvolvimento do banco partindo somente do código. Além desses recursos temos validação de dados, segurança, entre outros. Por estes motivos, o sistema proposto nesse projeto faz o uso do Spring Boot no desenvolvimento dos serviços WEB que constituem sua API e também no desenvolvimento do banco de dados.

### 2.2.6 Java

Java é uma linguagem de programação orientada a objetos de tipagem forte que surgiu nos anos 90 com o objetivo de antecipar tendências tecnológicas futuras. Com o passar do tempo, sua popularidade cresceu devido a alta compatibilidade com diversos dispositivos. Ainda hoje está em alta no mercado, seja no setor de manutenção de sistemas ou no desenvolvimento de back-end com frameworks auxiliares como o Spring Boot. Java é uma linguagem bem definida, padronizada e que possui convenções claras na realização de tarefas, atividades e configurações, tendo pouca flexibilidade, mas por outro lado facilitando o desenvolvimento em equipe e a evolução das aplicações. O sistema proposto neste projeto utiliza a linguagem Java na construção do seu back-end por ser bem integrada com o Spring Boot, o que permitiu ter um desenvolvimento facilitado e boa produtividade sem sacrificar a qualidade da aplicação no sistema (BOAGLIO, 2017).

### 2.2.7 Padrão MVC

MVC, ou *Model-View-Controller*, é um padrão de arquitetura de software utilizado no desenvolvimento de sistemas WEB. Foi criado com o objetivo de separar as responsabilidades e organizar o código de forma mais modular e reutilizável. Com a separação clara de responsabilidades entre o modelo, a visão e o controlador, o padrão MVC facilita a manutenção, a escalabilidade e a reutilização do código. (LEMOS et al., 2013)

## 3 Metodologia

Nesta seção, serão apresentados primeiramente conceitos e boas práticas para desenvolvimento de software na Seção 3.1. Em seguida na Seção 3.2, será apresentada a metodologia conduzida para o desenvolvimento do sistema.

### 3.1 Conceitos

Nesta seção, serão apresentados alguns conceitos de desenvolvimento de software que serão usados na implementação do projeto.

#### 3.1.1 Extração de Requisitos

Extração de requisitos é o processo de identificação e documentação de requisitos, que correspondem às funcionalidades que o sistema deve cumprir para se adequar ao objetivo final do software. Durante essa extração, são consideradas as necessidades e expectativas dos usuários para uma boa usabilidade e entendimento do software, atendendo sempre as demandas exigidas.

O presente trabalho irá tratar de dois tipos de requisitos, sendo eles os funcionais e não funcionais. Requisitos funcionais são os requisitos do sistema relacionados ao código, como funcionalidades e comportamentos, descrevendo que ações o sistema tem de ser capaz de executar, podendo ser definidos antes ou durante o desenvolvimento do software, com versões ou *full releases*<sup>1</sup>. Requisitos não funcionais representam as qualidades do sistema, como desempenho, segurança, usabilidade e confiabilidade, ditando basicamente como o sistema deve se comportar, sem relação direta com a implementação em si. (BRAGA, 2017).

#### 3.1.2 Análise das Regras de Negócio

Regras de negócio, como o próprio nome diz, são padrões seguidos no desenvolvimento de software para que o mesmo esteja de acordo com os requisitos registrados anteriormente, (SCHEMES, 2020). São vitais para a correta aplicação de políticas, procedimentos e lógicas que vão influenciar o funcionamento do software. O sucesso da operação muitas vezes tem grande dependência na aplicação correta das regras de negócio, pois como elas ditam os padrões a serem seguidos no desenvolvimento do sistema, certamente haverá retrabalho no software se não seguirem, ou não entenderem corretamente esses padrões

---

<sup>1</sup> *Full releases* são conhecidos como lançamentos completos; são versões de um software consideradas prontas para o uso público.

pré-estabelecidos, levando a atrasos, desperdício de recursos e queda na produtividade, (JUNG, 2015).

As regras de negócio também lidam com a utilização do software, como na questão da restrição de dados, fluxo de trabalho, validações e cálculos. São áreas importantes para delimitar o escopo e a precisão dos dados, em relação a como eles devem ser inseridos e como restrições devem ser aplicadas para que ele se comporte corretamente com as entradas esperadas. Elas também definem a conformidade com as leis do país e regulamentação de políticas internas em grandes organizações juntamente com sua adaptabilidade as mudanças de requisitos, necessários para o desenvolvimento da aplicação durante sua evolução; é possível perceber esta situação em muitos casos na indústria, ainda mais em métodos ágeis, aonde acontecem constantes mudanças nos requisitos para a adequação do software a necessidade da empresa. (CARVALHO; MELLO, 2012).

### 3.1.3 Definição e Descrição dos Casos de Uso

Casos de uso representam a interação entre os atores de um sistema, descrevendo principalmente como o sistema será utilizado a vista de diferentes cenários possíveis, possibilitando ter uma visão ampla dos fluxos de trabalho e funcionalidades. Cada caso mostra uma sequência de eventos para chegar a um objetivo determinado pelo projeto. Casos de uso são indispensáveis em aplicações de grande porte, pois ajudam muito no entendimento de requisitos, comunicação entre as funcionalidades e validação do sistema. Apresentados como diagramas e utilizando a Linguagem de Modelagem Unificada(UML), os atores são representados por ícones e os casos de uso por elipses, sendo suas relações visualizadas como linhas que ligam atores e casos de uso e os casos de uso entre si, (VINICIUS, 2009).

Cada requisito funcional define uma funcionalidade e cada funcionalidade define um caso de uso. A descrição de casos de uso fornece um detalhamento maior do fluxo de uma funcionalidade, listando as etapas e as interações que ocorrem até que a funcionalidade seja concluída. Os casos de uso são frequentemente utilizados para descrever funcionalidades de um sistema para o usuário comum final, pois possuem uma linguagem de nível mais alto e mais próxima do ser humano do que do computador. Os casos de uso também são aproveitados para registro de documentação do sistema. Um caso de uso pode apresentar mais de um fluxo e existem algumas nomenclaturas que ajudam a contextualizar um caso de uso. Entre as mais comuns temos:

- Fluxo Principal – define o fluxo comum do início ao fim do caso de uso.
- Fluxo Alternativo – define uma forma alternativa ao fluxo comum para a execução do caso de uso.



- Fluxo de Exceção – define um comportamento excepcional quando algo no caso de uso não ocorre como o esperado.
- Atores – entidades que participam da realização do caso de uso.
- Pré-condições – definem as condições em que o sistema deve se encontrar para que o caso de uso possa ser realizado.
- Pós-condições - o que muda no estado do sistema após o fim da execução de um caso de uso.

### 3.1.4 Modelagem de Classes do Sistema

A modelagem de classes de um sistema é uma técnica utilizada no desenvolvimento de software orientado a objetos que tem como objetivo representar a estrutura lógica do sistema por meio de classes e seus relacionamentos. Essa abordagem permite uma compreensão clara e organizada do sistema, facilitando a análise, o projeto e a implementação.

Nesse contexto, uma classe representa uma abstração de um objeto ou entidade do mundo real, contendo características (atributos) e comportamentos (métodos) associados a ela. As relações entre as classes são estabelecidas por meio de associações, heranças, agregações e composições, permitindo representar de forma precisa as interações entre os objetos do sistema. (COSTA, 2001)

Essa técnica é amplamente utilizada em diversas metodologias de desenvolvimento de software, como o *Unified Modeling Language* (UML), auxiliando os profissionais a projetar e criar sistemas robustos e de alta qualidade.

### 3.1.5 Modelagem do Banco de Dados

A modelagem de banco de dados é um processo essencial no desenvolvimento de sistemas, que tem como objetivo organizar e estruturar os dados de forma coerente e eficiente. A normalização é uma técnica utilizada nesse processo, que visa eliminar redundâncias e inconsistências nos dados, garantindo a integridade e a consistência das informações armazenadas.

A normalização consiste em aplicar uma série de regras e formas normais aos esquemas de banco de dados, a fim de alcançar uma estrutura consistente, contribuindo para a eficiência no armazenamento, integridade dos dados e na facilidade de manutenção. Também facilita consultas complexas e atualização dos dados de forma consistente, (FERNANDES, 2019).

### 3.1.6 Implementação

A implementação é uma etapa definida em engenharia de software, onde acontece a transformação dos requisitos definidos anteriormente em um código executável, juntamente com a criação de componentes. É uma das etapas mais importantes no processo de desenvolvimento de software, pois é onde todo o planejamento e visão do sistema se tornam realidade, tomando cuidados com prazo, recursos, boas práticas, controle de versão, trabalho em equipe e gestão. Todos esses componentes são vitais para o sucesso no desenvolvimento de software ainda mais em projetos de grande escala, garantindo sua rastreabilidade e integridade ([FLEURY et al., 2014](#)).

### 3.1.7 Testes e Correções

Após concluída a parte de desenvolvimento, passando por todas as etapas necessárias, tem-se a fase de testes onde será verificado se o sistema está atendendo os requisitos exigidos e funcionando de forma correta com os resultados esperados. São utilizadas várias técnicas para diferentes níveis de teste, como teste de unidade, sistema, integração e aceitação. Eles lidam com partes mais referentes ao código puro, com relação a variáveis, funções e módulos, quanto partes mais referentes a funcionalidade, com um foco maior na entrada e saída somente, não se concentrando tanto em como o código funciona e sim o que ele gera. Assim, garantindo que o produto final esteja de acordo com a visão desejada do projeto, sendo indispensável em qualquer tipo de sistema, desde projetos pequenos com equipes pequenas a projetos gigantes de empresas multinacionais. ([MULLER, 2020](#)).

## 3.2 Etapas do desenvolvimento

A metodologia utilizada no desenvolvimento do sistema proposto neste trabalho foi baseada no processo metodológico descrito em ([SOMMERVILLE, 2007](#)). Os passos efetuados para o alcance dos objetivos deste trabalho foram:

- inicialmente foi realizada a análise de negócio, com o objetivo de mapear as características do domínio do gerenciamento de empréstimo de livros em uma biblioteca;
- em seguida, foi feita a etapa de análise e extração de requisitos funcionais, não funcionais e a modelagem de alguns elementos do sistema a nível de documentação;
- ao fim das etapas anteriores, foi dado início à implementação do sistema, que foi realizada por meio de incrementos onde cada incremento correspondia a uma camada do sistema: (i) inicialmente, foi desenvolvida a camada de modelo (entidades do sistema e seus relacionamentos e comportamentos); (ii) em seguida, foi desenvolvida

a camada de banco de dados; e (iii) por fim, foi desenvolvida a camada de interface gráfica;

- após o fim de cada, etapa foram realizados testes para encontrar possíveis correções no sistema.

## 4 Sistema Computacional

Neste capítulo, serão discutidos os detalhes do processo de desenvolvimento do sistema. Serão apresentadas as etapas de modelagem (com a extração dos requisitos, definição dos casos de uso e das funcionalidades), de desenvolvimento do modelo (implementação das entidades do sistema e regras de negócio), de desenvolvimento do banco de dados (normalização, modelagem, definição das tabelas e relacionamentos), de desenvolvimento da interface gráfica (definição dos layouts, construção e associação das telas as funcionalidades) e de testes e correções (testes de funcionamento do sistema, usabilidade, correções e melhorias).

### 4.1 Análise de Domínio e Regras de Negócio

Nesta etapa, foi analisado o conjunto de serviços básicos que uma biblioteca deve fornecer para o gerenciamento de empréstimo de livros. Também foram analisadas as regras e restrições que devem ser seguidas durante os empréstimos e devoluções de livros pelas partes envolvidas no processo. Ao fim dessa etapa, foi possível evidenciar quais serviços eram mais essenciais e deviam ser tratados, quais serviços poderiam receber melhorias nesta adaptação e foi fechado um escopo aonde seriam obtidos os requisitos e as funcionalidades do sistema.

No sistema em questão, foram obtidos as regras de negócio descritas no Quadro 1.

Quadro 1 – Regras de Negócio

Identificador	Descrição
RN01	Cada usuário só poderá manter uma quantidade determinada de livros simultaneamente, especificada pelo bibliotecário no momento do cadastro do usuário.
RN02	O empréstimo de um livro é válido por uma quantidade determinada de dias. O tempo de empréstimo deve ser especificado pelo bibliotecário no momento do empréstimo.
RN03	Caso a devolução de um livro não seja realizada dentro do prazo determinado, o usuário ficará impossibilitado de realizar novos empréstimos até devolver o livro e justificar o atraso.

Fonte: os autores.

## 4.2 Análise e Extração De Requisitos

Durante a etapa de análise e extração de requisitos, foi desenvolvido um modelo para solucionar as demandas mais comuns para uma biblioteca física, demandas essas que foram evidenciadas na etapa anterior de análise de domínio e regras de negócio. Com base nestas demandas, foi desenvolvido o documento de requisitos funcionais e não-funcionais. Entre os requisitos, se destaca a possibilidade dos usuários emprestarem livros entre si utilizando o sistema como um intermediário.

Os Quadros 2 e 3 apresentam, respectivamente, os requisitos funcionais e não-funcionais do sistema proposto.

Quadro 2 – Requisitos Funcionais

Identificador	Descrição
RF01	Entrar no sistema.
RF02	Manter o cadastro de livros no sistema.
RF03	Remover um livro cadastrado.
RF04	Manter o cadastro de usuários do sistema.
RF05	Remover um usuário cadastrado do sistema
RF06	Registrar o empréstimo do livro.
RF07	Registrar a devolução de um livro.
RF08	Sair do Sistema.

Fonte: os autores.

Quadro 3 – Requisitos Não Funcionais

Identificador	Descrição
RNF01	O sistema deve possuir uma interface de fácil utilização.

Fonte: os autores.

## 4.3 Definição de Casos de Uso

Durante a etapa de definição de funcionalidades e casos de uso, foram definidas cada funcionalidade com sua respectiva descrição. Como foi citado na Seção 3.1.3, cada caso de uso define uma funcionalidade que pode ser descrita na forma de descrição de casos de uso.

Para os requisitos do sistema foram definidos os seguintes casos de uso:

- CSU 01 - Entrar no Sistema;
- CSU 02 - Manter o cadastro de livros no sistema;

- CSU 03 - Remover exemplar de um livro cadastrado;
- CSU 04 - Manter o cadastro de um usuário no sistema;
- CSU 05 - Remover o cadastro de um usuário no sistema;
- CSU 06 - Registrar o empréstimo de um livro;
- CSU 07 - Registrar a devolução de um livro; e
- CSU 08 - Sair do sistema.

Com base em cada um dos casos de uso foram geradas as seguintes descrições:

1. CSU 01 - Entrar no Sistema

a) Atores:

- i. Bibliotecário
- ii. Usuário

b) Fluxo Principal:

- i. O bibliotecário ou usuário deve inserir seu CPF e senha.
- ii. O bibliotecário ou o usuário é redirecionado para o seu menu correspondente e o caso de uso termina.

c) Fluxo de Exceção:

- i. Caso as informações de entrada do bibliotecário ou do usuário sejam incorretas, uma mensagem deve ser mostrada.

2. CSU 02 - Manter o cadastro de livros no sistema

a) Atores:

- i. Bibliotecário.
- ii. Usuário.

b) Pre-condições:

- i. Um bibliotecário ou usuário deve estar logado no sistema.

c) Fluxo Principal:

- i. O bibliotecário ou o usuário escolhe se vai cadastrar ou buscar um livro.
- ii. O bibliotecário ou o usuário deve inserir o nome do livro a ser buscado.
- iii. As informações do livro são mostradas e o caso de uso termina.

d) Fluxo Alternativo:

- i. O bibliotecário ou o usuário deve inserir as informações do livro..
  - ii. O livro é cadastrado e o caso de uso termina.
- e) Fluxo de Exceção:
  - i. Se o bibliotecário ou o usuário digitar informações que não correspondem a algum livro cadastrado, deve ser mostrada uma mensagem.
- f) Pós-Condições:
  - i. Os cadastros de livros são atualizados, se preenchido corretamente as informações de cadastro, e após o caso de uso, o sistema retorna para o seu menu principal de funcionalidades do ator correspondente.

### 3. CSU 03 - Remover Livro no Sistema

- a) Atores:
  - i. Bibliotecário.
  - ii. Usuário.
- b) Pre-condições e Observações:
  - i. O bibliotecário ou usuário deve estar logado no sistema e o livro, referente ao código, cadastrado.
  - ii. O usuário só será capaz de remover um livro que ele disponibilizou para empréstimo, ou seja, um livro dele.
- c) Fluxo Principal:
  - i. O bibliotecário ou o usuário insere o código do livro que será removido.
  - ii. Aparece uma tela de confirmação informando que o livro será removido.
  - iii. O bibliotecário ou o usuário confirma a remoção e o caso de uso termina.
- d) Fluxo de Exceção:
  - i. Se o livro correspondente ao código não estiver cadastrado, uma mensagem deve ser exibida.
- e) Pós-Condições:
  - i. Os cadastros de livros são atualizados, se preenchido corretamente as informações de remoção, e após o caso de uso, o sistema retorna para o seu menu principal de funcionalidades do ator correspondente.

### 4. CSU 04 – Cadastrar usuário no sistema

- a) Atores:
  - i. Bibliotecário.
  - ii. Usuário.

## b) Pre-condições e Observações:

- i. O bibliotecário deve estar logado no sistema.
- ii. O usuário pode se cadastrar e ser cadastrado pelo bibliotecário.

## c) Fluxo Principal:

- i. O bibliotecário deve inserir as informações do usuário ou o usuário deve inserir suas informações
- ii. É criado um novo usuário e o caso de uso termina.

## d) Fluxo de Exceção:

- i. Se forem digitadas informações inválidas, uma mensagem deve ser exibida.

## e) Pós-Condições:

- i. O cadastro de usuários é atualizado.

## 5. CSU 05 - Remover usuário

## a) Atores:

- i. Bibliotecário.

## b) Pre-condições:

- i. O bibliotecário deve estar logado no sistema.
- ii. O usuário não pode ter nenhuma pendência no sistema, ou seja, nem empréstimos e nem livros disponibilizados.

## c) Fluxo Principal:

- i. O bibliotecário insere as informações do usuário que será removido.
- ii. Aparece uma tela de confirmação informando que o usuário será removido.
- iii. O bibliotecário confirma a remoção e o caso de uso termina.

## d) Fluxo de Exceção:

- i. Se o usuário correspondente às informações não estiver cadastrado, uma mensagem deve ser exibida.

## 6. CSU 06 - Registrar o empréstimo de um livro

## a) Atores:

- i. Bibliotecário
- ii. Usuário.

## b) Precondições:

- i. O bibliotecário deve estar logado no sistema e o usuário deve estar cadastrado.



- c) Fluxo Principal:
    - i. O bibliotecário insere informações para buscar o livro.
    - ii. O bibliotecário insere informações do usuário.
    - iii. O empréstimo é realizado e o caso de uso termina.
  - d) Fluxo de Exceção 1 (Fluxo principal: Passo 1):
    - i. Se o livro não estiver cadastrado, deve ser mostrada uma mensagem.
  - e) Fluxo de Exceção 2 (Fluxo principal: Passo 2):
    - i. Se as informações não corresponderem a nenhum usuário, deve ser mostrada uma mensagem.
  - f) Fluxo de Exceção 3 (Fluxo principal: Passo 2):
    - i. Caso o livro buscado já esteja emprestado ele aparecerá na busca como indisponível e a opção de empréstimo estará desabilitada.
  - g) Pós-Condições:
    - i. O empréstimo é registrado e os dados do usuário e dos exemplares disponíveis são atualizados.
7. CSU 07 – Registrar a devolução de um livro
- a) Atores:
    - i. Bibliotecário
    - ii. Usuário.
  - b) Precondições:
    - i. O bibliotecário deve estar logado no sistema, o usuário e o empréstimo do livro devem estar cadastrados.
  - c) Fluxo Principal:
    - i. O bibliotecário insere informações do exemplar.
    - ii. A devolução é realizada e o caso de uso termina.
  - d) Fluxo de Exceção:
    - i. Se os dados do exemplar não corresponderem a nenhum exemplar em empréstimo deve ser mostrada uma mensagem.
  - e) Pós-Condições:
    - i. O empréstimo é finalizado e os dados do usuário e dos exemplares atualizados.
8. CSU 08 – Sair do Sistema

- a) Atores:
  - i. Bibliotecário.
  - ii. Usuário.
- b) Precondições:
  - i. Nenhuma.
- c) Fluxo Principal:
  - i. O bibliotecário ou o usuário seleciona a opção de sair, volta a página inicial e o caso de uso termina.

## 4.4 Modelagem das Classes do Sistema

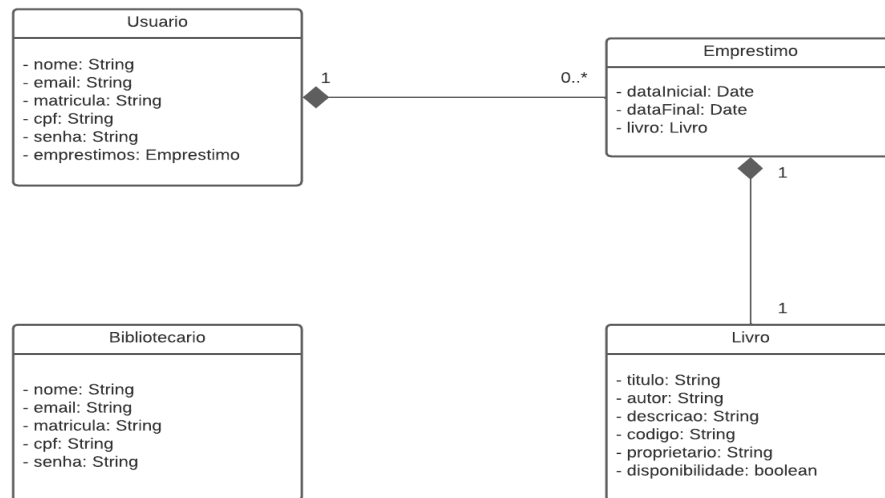
Na etapa de modelagem das classes do sistema, foram definidas as principais classes do sistema, com seus atributos e relacionamentos. Para o sistema em questão foi definido o seguinte modelo de classes apresentado na Figura 2.

O modelo de classes deste projeto é composto por 4 classes fundamentais que representam a estrutura de negócio. A classe 'Usuario' está relacionada com a classe 'Emprestimos', sendo que a classe 'Usuario' possui uma lista de empréstimos associados a ela. Por sua vez, a classe 'Emprestimo' está relacionada com a classe 'Livro', onde cada instância da classe 'Emprestimo' possui uma referência a um livro específico. Vale ressaltar que a classe 'Bibliotecario' não possui relacionamento com nenhuma outra classe no modelo e os livros inseridos são objetos diferentes no sistema, mesmo que eles sejam uma cópia de outro livro cadastrado.

Essa estrutura de classes foi projetada para permitir a adequada representação e manipulação das informações do sistema. O relacionamento entre as classes 'Usuario' e 'Emprestimos' é estabelecido por meio de uma lista de empréstimos na classe 'Usuario', possibilitando a associação de múltiplos empréstimos a um único usuário. Já a classe 'Emprestimo' está diretamente relacionada à classe 'Livro', permitindo que cada empréstimo seja associado a um livro específico.

Dessa forma, o modelo de classes proporciona uma base sólida para o gerenciamento das informações. A estrutura estabelecida permite a integridade dos dados e facilita a realização de consultas relacionadas aos empréstimos de cada usuário. Além disso, o relacionamento entre 'Emprestimo' e 'Livro' possibilita a identificação precisa do livro associado a cada empréstimo registrado no sistema.

Figura 2 – Diagrama de Classes do Sistema

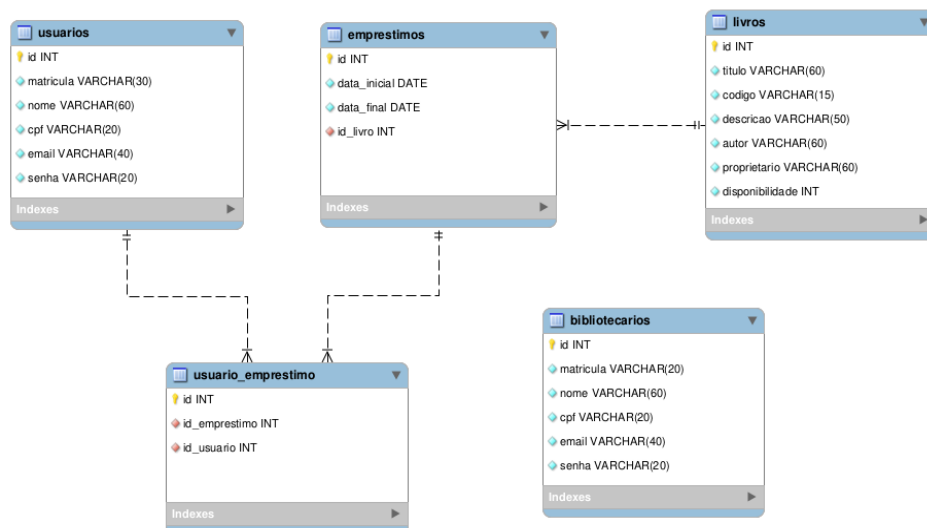


Fonte: os autores.

## 4.5 Modelagem do Banco de Dados do Sistema

Nesta etapa foi projetado o modelo de banco de dados do sistema fazendo o uso da técnica de normalização de dados citado na Seção 4.4. Para o sistema em questão foi definido o modelo relacional de banco de dados descrito na Figura 3.

Figura 3 – Modelo Relacional de Banco de Dados



Fonte: os autores.

## 4.6 Implementação

Nesta etapa, foi realizada a implementação do sistema, seguindo as fases listadas na Seção 3.2.

Inicialmente, foi realizada a implementação do modelo de negócio com base nas classes do sistema. O modelo de negócio foi encapsulado em uma API utilizando as tecnologias Java e Spring Boot que permitiram a construção e disponibilização dos dados do sistema por meio de serviços WEB utilizando o padrão MVC para modularização e reutilização de código, agilizando possíveis manutenções e possíveis aumento do escopo do projeto em relação a sua escalabilidade. Além disso, a utilização do Spring Boot facilitou a integração com o banco de dados.

O banco de dados do sistema foi implementado de forma automática pelo Spring Boot que possui suporte para ORM (mapeamento objeto relacional). Deste modo, foi possível implementar o banco de dados realizando o mapeamento direto nas classes. Sendo assim, ao executar a API, caso o banco de dados já não estivesse criado, este seria criado automaticamente.

A interface gráfica foi desenvolvida utilizando as tecnologias HTML, CSS e Javascript. Nesta etapa, foram desenvolvidas todas as telas do sistema com os dados sendo obtidos por meio de requisições à API que fizeram a integração entre o Front-end e Back-end, trazendo a lógica do programa e funcionalidades para as interação e responsividade das telas.

## 4.7 Navegação e Utilização do Sistema

Nessa seção, será descrita a navegação e a utilização do sistema, com base nas perspectivas do bibliotecário (Seção 4.7.1) e do usuário (Seção 4.7.2).

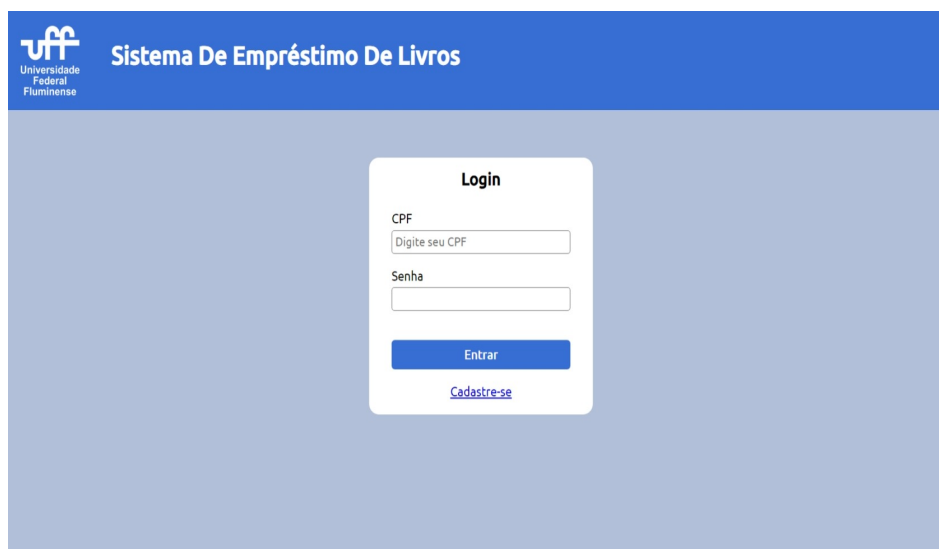
### 4.7.1 Perspectiva do Bibliotecário

O bibliotecário inicia o sistema na página de login (Figura 4), na qual ele deve inserir suas informações de entrada (CPF e senha).

Após realizar o login, o bibliotecário é redirecionado para o menu principal (Figura 5) onde ele pode acessar submenus, cada um relacionado com um item que irá redirecionar para a tela correspondente com algo relacionado ao botão apertado, também podendo sair do sistema e voltar para a página de login.

Ao clicar no submenu **Usuários**, o bibliotecário é redirecionado para a página de menu relacionado ao usuário (Figura 6), onde ele pode cadastrar e remover usuários ou voltar para o menu principal.

Figura 4 – Tela de Login



A screenshot of the login page for the 'Sistema De Empréstimo De Livros' (Book Lending System) at UFF (Universidade Federal Fluminense). The page has a blue header with the UFF logo and the system name. The main content area is light blue and contains a white login box. Inside the box, there is a 'Login' title, a 'CPF' label with a text input field containing the placeholder 'Digite seu CPF', a 'Senha' label with a password input field, a blue 'Entrar' button, and a blue link labeled 'Cadastre-se'.

Fonte: os autores.

Figura 5 – Menu Principal Bibliotecário



A screenshot of the main menu for the 'Sistema De Empréstimo De Livros' (Book Lending System) at UFF. The page has a blue header with the UFF logo and the system name. The main content area is light blue and contains a white box titled 'Menu Principal'. Inside this box, there are four blue buttons stacked vertically, labeled 'Usuários', 'Livros', 'Empréstimos', and 'Sair'.

Fonte: os autores.

Ao clicar na opção **Cadastrar Usuário**, o bibliotecário é redirecionado para a tela de cadastro de usuários (Figura 7), onde ao inserir as informações necessárias, o usuário é cadastrado no sistema. Após cadastrar o usuário, o bibliotecário é redirecionado para o menu principal.

Ao clicar na opção **Remover Usuário**, o bibliotecário é redirecionado para a tela de Remoção de usuários (Figura 8). Ao inserir o CPF do usuário, este é removido. Após a remoção, o bibliotecário é redirecionado a tela do menu principal.

Após clicar no submenu **Livros**, o bibliotecário é redirecionado para a página de

Figura 6 – Tela de Usuários



Fonte: os autores.

Figura 7 – Tela de Cadastro de Usuários no Sistema

Fonte: os autores.

menu relacionado aos livros (Figura 9), onde ele pode cadastrar, remover e consultar livros ou voltar para o menu principal.

Ao clicar na opção **Cadastrar Livro**, o bibliotecário é redirecionado a tela de cadastro de livros (Figura 10), onde ao inserir as informações necessárias, o livro é cadastrado no sistema. Após cadastrar o livro, o bibliotecário é redirecionado para o menu principal.

Ao clicar na opção **Remover Livro**, o bibliotecário é redirecionado a tela de remoção de livros (Figura 11). Ao inserir o código do livro, este é removido. Após a

Figura 8 – Tela de Remoção de Usuários



A interface da tela de remoção de usuários possui um cabeçalho azul com o logo da UFF (Universidade Federal Fluminense) e o título "Sistema De Empréstimo De Livros". O conteúdo principal, em um fundo cinza claro, apresenta o título "Remover Usuário" no topo. Abaixo dele, há um formulário com um campo de entrada de texto rotulado "Digite o CPF" e um botão azul "Remover" ao lado.

Fonte: os autores.

Figura 9 – Tela de Livros



A interface da tela de livros possui um cabeçalho azul com o logo da UFF e o título "Sistema De Empréstimo De Livros". O conteúdo principal, em um fundo cinza claro, apresenta o título "Livros" no topo. Abaixo dele, há quatro botões azuis empilhados verticalmente: "Cadastrar Livro", "Remover Livro", "Consultar Livro" e "Voltar".

Fonte: os autores.

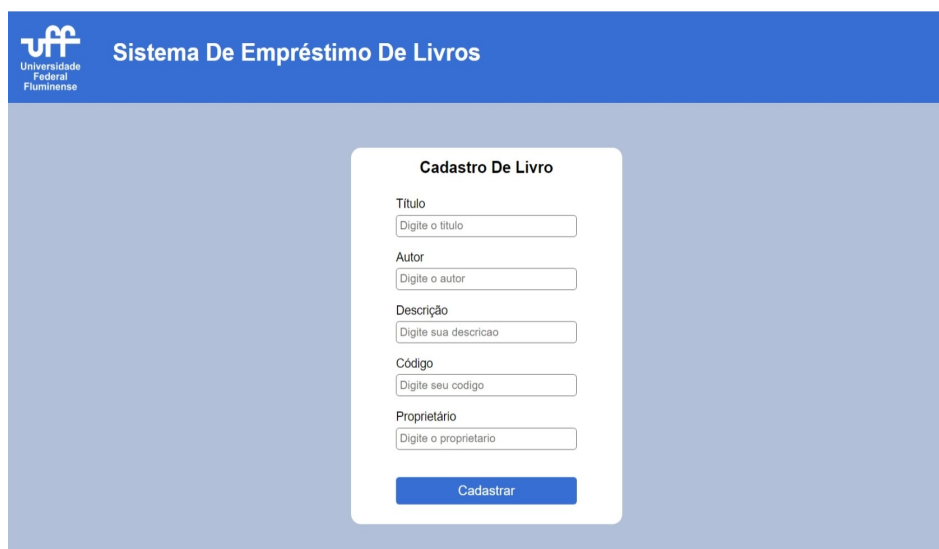
remoção, o bibliotecário é redirecionado a tela do menu principal.

Ao clicar na opção **Consultar Livro**, o bibliotecário é redirecionado a tela de consulta de livros (Figura 12). Ao inserir o código do livro, ele é redirecionado a uma tela que mostrará as opções de livro retornados a partir da busca.

Ao consultar um livro, o bibliotecário será redirecionado a uma tela (Figura 13) onde aparecerão os títulos dos livros retornados na consulta a partir do código do livro no sistema.

Após clicar no submenu **Empréstimos**, o bibliotecário é redirecionado para a pá-

Figura 10 – Tela de Cadastro de Livro no Sistema



A interface de usuário para o cadastro de livros no sistema. No topo, há uma barra azul com o logo da UFF (Universidade Federal Fluminense) e o título "Sistema De Empréstimo De Livros". Abaixo, centralizado, há um formulário branco com o título "Cadastro De Livro". O formulário contém cinco campos de entrada: "Título" (com o placeholder "Digite o título"), "Autor" (com o placeholder "Digite o autor"), "Descrição" (com o placeholder "Digite sua descricao"), "Código" (com o placeholder "Digite seu codigo") e "Proprietário" (com o placeholder "Digite o proprietario"). Abaixo dos campos, há um botão azul com o texto "Cadastrar".

Fonte: os autores.

Figura 11 – Tela de Remoção de Livros no Sistema



A interface de usuário para a remoção de livros no sistema. No topo, há uma barra azul com o logo da UFF (Universidade Federal Fluminense) e o título "Sistema De Empréstimo De Livros". Abaixo, centralizado, há o título "Remover Livro". Abaixo do título, há um campo de entrada com o placeholder "Digite o codigo" e um botão azul com o texto "Remover".

Fonte: os autores.

gina de menu relacionado aos empréstimos (Figura 14), onde ele pode realizar empréstimos, devoluções ou voltar para o menu principal.

Ao clicar em **Realizar Empréstimo**, ele será redirecionado à tela de busca (Figura 15) onde ele irá inserir seu CPF, sua senha e o nome do livro. Clicando em avançar irão aparecer as opções de livros que contém o nome digitado.

Ao clicar em **Realizar Devolução**, o bibliotecário será redirecionado a tela de inserção do código do livro a ser devolvido (Figura 17) onde ele irá inserir o código do livro a ser devolvido. Após a devolução, o bibliotecário é redirecionado ao menu principal.



Figura 12 – Tela de Consulta de Livros no Sistema

The screenshot shows a web interface for a book borrowing system. At the top, there is a blue header with the UFF logo (Universidade Federal Fluminense) and the title 'Sistema De Empréstimo De Livros'. Below the header, the main content area has a light blue background. It features a section titled 'Consultar Livro' with a search input field labeled 'Digite o título' and a blue 'Consultar' button.

Fonte: os autores.

Figura 13 – Tela de Livros Retornados na Busca

The screenshot shows the 'Resultados' section of the book borrowing system. It displays a table of search results for the keyword 'Programação'. The table has three columns: 'TITULO', 'AUTOR', and 'DISPONÍVEL'. The results are as follows:

TITULO	AUTOR	DISPONÍVEL
Programação OO	Fabius Mertens	DISPONÍVEL
Programação Em Java	William Gallagher	INDISPONÍVEL
Programação C e C++	August Andersen	DISPONÍVEL
Programação Funcional	Kyouko Hori	INDISPONÍVEL
Programação Para Web	Kanade Tachibana	DISPONÍVEL

Fonte: os autores.

#### 4.7.2 Perspectiva do Usuário

O usuário inicia o sistema na página de login (Figura 4), onde ele deve inserir suas informações de entrada (CPF e senha).

Após realizar o login, o usuário é redirecionado para o menu principal (Figura 18), onde ele pode acessar submenus, cada um relacionado com um item que corresponde a uma funcionalidade com relação ao que está escrito no botão, também podendo sair do sistema e voltar para a página de login.

Ao clicar na opção **Emprestar livro**, o usuário é redirecionado à tela de Cadastro

Figura 14 – Tela de Empréstimos



A interface da tela de empréstimos apresenta o logo da UFF (Universidade Federal Fluminense) no canto superior esquerdo. O título principal é "Sistema De Empréstimo De Livros". Abaixo, o subtítulo "Empréstimos" introduz três botões de ação: "Realizar Empréstimo", "Realizar Devolução" e "Voltar", todos em um estilo visual consistente.

Fonte: os autores.

Figura 15 – Consulta de Livros para Empréstimo



A interface para consulta de livros para empréstimo mantém o mesmo cabeçalho com o logo da UFF e o título "Sistema De Empréstimo De Livros". O subtítulo "Empréstimo De Livro" precede um formulário com três campos de entrada: "CPF do Usuário" (contendo o valor 10120230300), "Senha Do Usuário" (com caracteres ocultos por pontos) e "Livro" (contendo o valor "Programação"). Um botão "Avançar" está posicionado na base do formulário.

Fonte: os autores.

de livro (Figura 10), onde ao preencher os dados, tem seu livro disponibilizado para empréstimo. Ao finalizar a operação, ele é redirecionado ao menu principal.

Ao clicar na opção **Remover Livro**, o usuário é redirecionado à tela de remoção de livros (Figura 11). Inserindo o código do livro este é removido. Após a remoção o usuário ele é redirecionado a tela do menu principal.

Ao clicar na opção **Consultar Livro**, o usuário é redirecionado à tela de consulta de livros (Figura 12). Inserindo o título do livro ou parte, ele é redirecionado a uma tela que mostrará as opções de livro retornados a partir da busca.

Figura 16 – Livros Consultados

The screenshot shows the 'Sistema De Empréstimo De Livros' interface. At the top, there is a header with the UFF logo and the title 'Sistema De Empréstimo De Livros'. Below the header, there is a search form with two input fields: 'Senha Do Usuário' (password) and 'Livro' (book title). The 'Livro' field contains the text 'Programação'. Below the input fields is a blue button labeled 'Avançar'. Below the search form, there is a section titled 'Resultados' which contains a table of search results.

TITULO	AUTOR	PROPRIETÁRIO	STATUS	AÇÃO
Programação Java	James Gosling	biblioteca	DISPONÍVEL	Empréstimo
Programação Python	Kyouko Horii	biblioteca	INDISPONÍVEL	
Programação Web	Robert Andersen	biblioteca	DISPONÍVEL	Empréstimo
Programação Para Internet	Walter Lima	pedro lima	DISPONÍVEL	CONTATO: pedrolima@email.com
Programação com C++	Alan Turing Da Silva	pedro lima	DISPONÍVEL	CONTATO: pedrolima@email.com
Programação Ruby	José Da Silva	pedro lima	DISPONÍVEL	CONTATO: pedrolima@email.com
Programação C#	Robert Andersen	pedro lima	DISPONÍVEL	CONTATO: pedrolima@email.com

Fonte: os autores.

Figura 17 – Consulta de Livros para Devolução

The screenshot shows the 'Sistema De Empréstimo De Livros' interface. At the top, there is a header with the UFF logo and the title 'Sistema De Empréstimo De Livros'. Below the header, there is a section titled 'Devolução De Livro'. This section contains a form with a single input field labeled 'Digite o código' and a blue button labeled 'Devolver'.

Fonte: os autores.

Após finalizar a consulta, o usuário é redirecionado ao menu principal.

Ao clicar na opção **Meus Empréstimos**, o usuário é redirecionado para uma tela de empréstimos (Figura 19) onde ele pode visualizar os seus empréstimos que estão em andamento.

Figura 18 – Menu Principal Usuário



Fonte: os autores.

Figura 19 – Tela de Empréstimos



Fonte: os autores.

## 5 Considerações Finais

Neste capítulo, serão apresentadas as considerações finais do trabalho, destacando as conclusões (Seção 5.1) e sugestões de trabalhos futuros (Seção 5.2).

### 5.1 Conclusão

Este trabalho teve como objetivo desenvolver um sistema de biblioteca que além de disponibilizar empréstimos de livros para seus usuários, tem o diferencial de poder gerenciar também o empréstimo entre usuários, aumentando seu acervo e promovendo uma socialização maior e um senso de comunidade maior aos que utilizam seus serviços.

Para alcançar o objetivo geral deste trabalho, foi estabelecido como um dos objetivos específicos realizar um mapeamento realista do negócio. Inicialmente, uma busca na literatura foi realizada para encontrar trabalhos semelhantes, o que permitiu obter *insights* valiosos para o desenvolvimento do sistema. Com base nessa pesquisa, a decisão inicial era desenvolver o sistema localmente, por meio de um executável. No entanto, após conversas com o orientador, optou-se por migrar a aplicação para dispositivos web, uma abordagem mais atual e adequada para o problema em questão. Após definir as plataformas e dispositivos a serem utilizados, foram extraídos os requisitos e analisadas as regras de negócio. Isso possibilitou o planejamento inicial das funcionalidades do sistema e o comportamento esperado durante o uso. Por fim, os Casos de Uso foram definidos para representar a interação entre o sistema e seus atores.

Para implementar a estratégia de compartilhamento de livros entre os usuários da biblioteca, foram seguidos alguns passos essenciais. Inicialmente, a modelagem das classes no sistema e do banco de dados foi realizada, representando a estrutura lógica dos objetos e dos dados que seriam armazenados e utilizados pelo sistema.

Após a preparação da modelagem, o sistema foi implementado utilizando diversas tecnologias. O back-end foi programado em Java com o auxílio do framework Spring Boot e gerenciado por um banco de dados MySQL. Para o front-end, foram utilizadas linguagens como HTML, CSS e Javascript, permitindo a criação de uma interface gráfica simplificada e intuitiva. A integração entre o front-end e o back-end também foi cuidadosamente realizada.

Uma vez que o sistema foi implementado, foram realizados testes para garantir seu funcionamento adequado. Durante essa fase de testes, eventuais erros foram identificados e corrigidos, assegurando a qualidade final do sistema.

Considera-se então que o sistema desenvolvido atingiu os objetivos iniciais propostos,

permitindo assim que um maior acervo de livros seja disponibilizado aos usuários de uma biblioteca.

## 5.2 Trabalhos futuros

Devidos a limitações de tempo e recursos, o sistema não foi criado de ponta a ponta, da especificação até a implantação e disponibilização.

Como trabalhos futuros a partir deste projeto, seguem as seguintes sugestões:

- Implantação e disponibilização do sistema.
- Desenvolvimento de um aplicativo para dispositivos móveis.
- Ampliação do escopo do sistema.
- Adição de soluções de segurança.
- Remodelagem das telas com foco em exploração de design minimalista e amigável aos olhos.

## Referências

- ALVARENGA, L. A teoria do conceito revisitada em conexão com ontologias e metadados no contexto das bibliotecas tradicionais e digitais. *DataGramaZero - Revista de Ciência da Informação*, v. 2, n. 6, 2001. Citado na página 13.
- BARBIER, F. *História das Bibliotecas: de Alexandria às Bibliotecas Virtuais*. [S.l.]: Edusp, 2019. 400 p. Citado na página 16.
- BOAGLIO, F. *Spring Boot: Acelere o desenvolvimento de microserviços*. [S.l.]: Casa do Código, 2017. 251 p. Citado na página 21.
- BOLSONI-SILVA, A. T.; LOUREIRO, S. R. O impacto das habilidades sociais para a depressão em estudantes universitários. *Psic.: Teor. e Pesq.*, v. 32, n. 4, 2016. Citado na página 14.
- BONFADINI, E.; PRETTO, F. Usabilidade em aplicações web: Proposta para o sistema de biblioteca da univates. *Destaques Acadêmicos*, v. 8, n. 4, 2016. Citado na página 18.
- BRAGA, R. T. V. *Requisitos de Software*. 2017. <[https://edisciplinas.usp.br/pluginfile.php/3142953/mod\\_resource/content/2/Aula09-Requisitos.pdf](https://edisciplinas.usp.br/pluginfile.php/3142953/mod_resource/content/2/Aula09-Requisitos.pdf)>. Data de acesso: 05 de junho de 2023. Citado na página 22.
- CARVALHO, B. V. de; MELLO, C. H. P. Aplicação do método ágil scrum no desenvolvimento de produtos de software em uma pequena empresa de base tecnológicas. *Gestão e Produção*, v. 19, n. 3, 2012. Citado na página 23.
- COSTA, C. A. A aplicação da linguagem de modelagem unificada (uml) para o suporte ao projeto de sistemas computacionais dentro de um modelo de referência. *Gest. Prod.*, v. 8, n. 1, 2001. Citado na página 24.
- DUCKETT, J. *HTML and CSS: Design and Build Websites*. [S.l.]: Wiley, 2011. 512 p. Citado na página 19.
- FERNANDES, F. G. *Banco de Dados: Conceitos sobre Normalização de Dados*. [S.l.]: Novas Edições Acadêmicas, 2019. 52 p. Citado na página 24.
- FISCHER, S. R. *História da escrita*. [S.l.]: Unesp, 2009. 293 p. Citado na página 16.
- FLANAGAN, D. *JavaScript: O Guia Definitivo*. [S.l.]: Bookman, 2012. 1080 p. Citado na página 20.
- FLEURY, A. L. et al. Alinhando objetivos estratégicos e processo de desenvolvimento em empresas de software. *Prod.*, v. 24, n. 2, 2014. Citado na página 25.
- HUBNER, M. L. F.; KUHN, A. C. A. Bibliotecas universitárias como espaços de aprendizagem. *Revista do Instituto de Ciências Humanas e da Informação*, v. 31, n. 1, 2017. Citado na página 13.
- JUNG, P. L. *Identificação do custo de retrabalho no desenvolvimento de software: estudo de caso*. Trabalho de Conclusão de Curso — Universidade Federal de Santa Maria, 2015. Curso de Engenharia de Produção. Citado na página 23.

- LEMOS, M. F. de et al. Aplicabilidade da arquitetura mvc em uma aplicação web (webapps). *RE3C - Revista Eletrônica Científica de Ciência da Computação*, v. 8, n. 1, 2013. Citado na página 21.
- MEYER, E. *CSS Pocket Reference: Visual Presentation for the Web*. [S.l.]: O'Reilly Media, 2018. 204 p. Citado na página 20.
- MULLER, G. W. *O que é Teste de Software? Por que é necessário?* 2020. <<https://cwi.com.br/blog/o-que-e-teste-de-software-por-que-e-necessario/>>. Data de acesso: 06 de junho de 2023. Citado na página 25.
- MURACH, J. *Murach's MySQL*. [S.l.]: Mike Murach and Associates, 2019. 628 p. Citado na página 20.
- OHIRA, M. L. B.; PRADO, N. S. Bibliotecas virtuais e digitais: análise de artigos de periódicos brasileiros (1995/2000). *Ci. Inf., Brasília*, v. 31, n. 1, 2002. Citado na página 17.
- SCHEMES, T. *Regras de negócio em TI: descubra esse importante conceito e entenda como aplicá-lo no seu negócio*. 2020. <<https://conteudo.movidesk.com/regras-de-negocio-ti/>>. Data de acesso: 05 de junho de 2023. Citado na página 22.
- SMITH, J.; JOHNSON, S. Designing web-based library management system. *International Journal of Engineering and Technical Research*, v. 59, n. 3, 2019. Citado na página 17.
- SOMMERVILLE, I. *Engenharia de Software*. [S.l.]: Pearson Universidades, 2007. v. 8. 568 p. Citado na página 25.
- VINICIUS. *Desenvolvimento de Software Dirigido por Caso de Uso*. 2009. <<https://www.devmedia.com.br/desenvolvimento-de-software-dirigido-por-caso-de-uso/9148>>. Data de acesso: 06 de junho de 2023. Citado na página 23.