

CENTRO UNIVERSITÁRIO RUY BARBOSA WYDEN

**ANALÍSE E DESENVOLVIMENTO DE SISTEMAS**

**GUSTAVO CERQUEIRA BONFIM OLIVEIRA**

Sistema de Controle de Biblioteca

**SALVADOR**

**2024**

**GUSTAVO CERQUEIRA BONFIM OLIVEIRA**

Sistema de Controle de Biblioteca

Trabalho apresentado à disciplina de Engenharia de Software do Curso de Analise e Desenvolvimento de Sistemas da UNIRUY como requisito básico para avaliação.

**SALVADOR**

**2024**

**Sumário**

[**1 INTRODUÇÃO** 4](#_Toc38378982)

[**2 OBJETIVO DA APLICAÇÃO** 5](#_Toc38378983)

[**3 TÉCNICAS E FERRAMENTAS UTILIZADAS** 6](#_Toc38378984)

[**3.1 IMPLEMENTAÇÃO** 6](#_Toc38378985)

[**3.2 BANCO DE DADOS** 6](#_Toc38378986)

[**3.3 DIAGRAMAS UML** 6](#_Toc38378987)

[**3.4 PROTOTIPAÇÃO** 7](#_Toc38378988)

[**4 DESENVOLVIMENTO DO SISTEMA** 8](#_Toc38378989)

[**4.1 ANÁLISE DE REQUISITOS** 8](#_Toc38378990)

[**4.1.1 Requisitos Funcionais** 8](#_Toc38378991)

[**4.1.2 Requisitos Não Funcionais** 9](#_Toc38378992)

[**4.2 DIAGRAMAS DA LINGUAGEM DE MODELAGEM UNIFICADA (UML)** 10](#_Toc38378993)

[**4.2.1 Diagrama de Caso de Uso** 10](#_Toc38378994)

[**4.2.1. Especificação dos Casos de Uso** 11](#_Toc38378995)

[**4.2.1.1 Especificação dos Casos de Uso** 1](#_Toc38378995)2

[**4.2.2 Diagrama de Classes** 13](#_Toc38378996)

[**4.2.2.1 Especificação das classes** 13](#_Toc38378997)

[**4.3 Diagrama de Conteto** 1](#_Toc38378997)4

[**4.3.1 Diagrama de Fluxo Zero** 1](#_Toc38378997)4

[**5 Entrevista/Questionário**](#_Toc38378998) 15

[**6 Dicionário de Dados**](#_Toc38378999) 16

[**7 Política de Testes**](#_Toc38379000) 17

[**8 IMPLEMENTAÇÃO: HOMOLOGAÇÃO, INSTALAÇÃO E TREINAMENTO**](#_Toc38379000) 18

[**8.1 REGRAS DE NEGÓCIO**](#_Toc38379000)  18

9. P[**ROTOTIAÃO DO SOFTWARE: TELAS** 19](#_Toc38379001)

[**9.1 TELA INICIAL** 1](#_Toc38379002)9

[**9.2 TELA CADASTRO USUÁRIO** 1](#_Toc38379003)9

[**9.3 TELA CADASTRO LIVRO** 19](#_Toc38379004)

[**9.4 TELA REGISTRO DE EMPRÉSTIMO**](#_Toc38379005) 20

[**9.5 TELA REGISTRO DE DEVOLUÇÃO**](#_Toc38379006) 20

[**9.6 TELA CONSULTA EMPRÉSTIMOS E DEVOLUÇÃO** 20](#_Toc38379007)

[**CONCLUSÃO** 2](#_Toc38379009)1

[**REFERÊNCIAS**](#_Toc38379010) 22

# **1 INTRODUÇÃO**

Este projeto tem como objetivo desenvolver uma aplicação protótipo de Controle de Biblioteca, utilizando os conceitos e fundamentos da Engenharia de Software.

A aplicação permitirá o gerenciamento de empréstimos de livros, com funcionalidades como cadastro de usuários, cadastro de livros, controle de empréstimos e devoluções, e geração de relatórios.

A metodologia adotada será baseada em práticas ágeis, especificamente o método Scrum, com ênfase em TDD (Test-Driven Development) e refatoração contínua.

# **2 OBJETIVO DA APLICAÇÃO**

· Facilitar o gerenciamento de empréstimos de livros em uma biblioteca;

· Automatizar o processo de cadastro de usuários e livros;

· Monitorar a disponibilidade de livros;

· Gerar relatórios sobre empréstimos, devoluções e status dos livros.

# **3 TÉCNICAS E FERRAMENTAS UTILIZADAS**

## **3.1 IMPLEMENTAÇÃO**

#### Desenvolvimento Front-End:

* **HTML e CSS:** Utilizados para criar a estrutura e o estilo visual das páginas web.
* **JavaScript:** Adicionado para fornecer interatividade e funcionalidades dinâmicas ao front-end.
* **FontAwesome:** Biblioteca de ícones utilizada para adicionar ícones aos elementos da interface.

#### Framework Back-End:

* **Spring Boot:** Framework Java utilizado para simplificar e acelerar o desenvolvimento de aplicativos Java baseados em Spring.

## **3.2 BANCO DE DADOS**

 **Banco de Dados Relacional:** Utilizado para armazenar e gerenciar os dados da aplicação.

 **MySQL, PostgreSQL, etc.:** Exemplos de sistemas de gerenciamento de banco de dados relacionais que podem ser utilizados.

## **3.3 DIAGRAMAS UML**

 **Diagrama de Casos de Uso:** Utilizado para modelar as interações entre os atores e o sistema, identificando os principais casos de uso da aplicação.

 **Diagrama de Classe:** Utilizado para representar a estrutura estática da aplicação, mostrando as classes, seus atributos e os relacionamentos entre elas.

 **Diagrama de Fluxo de Dados (DFD):** Utilizado para visualizar o fluxo de dados através do sistema, mostrando como os dados são processados e transformados.

 **DER (Diagrama de Entidade-Relacionamento):** Utilizado para modelar as entidades e os relacionamentos entre elas no banco de dados relacional.

## **3.4 PROTOTIPAÇÃO**

Para a realização do protótipo visual da plataforma, com funções interativas que descrevem com maior precisam de detalhes foi usado o Visual Studio Code. Através dele é possível criar telas de um site e criar interatividade entre elas.

# **4 DESENVOLVIMENTO DO SISTEMA**

## **4.1 ANÁLISE DE REQUISITOS**

### **4.1.1 Requisitos Funcionais**

**Fonte:** Autoria própria

|  |  |  |
| --- | --- | --- |
| **Código** | **Nome** | **Detalhe** |
| **RF01** | Cadastro de Usuários | Adicionar, editar, remover e listar usuários. |
| **RF02** | **Cadastro de Livros** | Adicionar, editar, remover e listar livros. |
| **RF03** | **Registro de Empréstimos** | Registrar novos empréstimos de livros. |
| **RF04** | **Registro de Devoluções** | Registrar devoluções de livros. |
| **RF05** | **Notificações** | Enviar lembretes de devolução. |
| **RF06** | **Relatórios** | Gerar relatórios de empréstimos e devoluções. |

### **4.1.2 Requisitos Não Funcionais**

|  |  |  |
| --- | --- | --- |
| **Código** | **Nome** | **Detalhe** |
| RNF01 | Usabilidade e desempenho | Interface intuitiva e fácil de usar, Resposta rápida para operações de cadastro e consulta. |
| RNF02 | **Segurança** | Controle de acesso e proteção dos dados dos usuários. |
| RNF03 | **Manutenibilidade** | Código modular e bem documentado. |

**Fonte:** Autoria própria

## 

## **4.2 DIAGRAMAS DA LINGUAGEM DE MODELAGEM UNIFICADA (UML)**

### **4.2.1 Diagrama de Caso de Uso**

### 

**Fonte:** Autoria própria

#### **4.2.1 Especificação dos Casos de Uso**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nome do Caso de Uso** | Cadastro de Usuários | | |
| **Descrição** | Este caso de uso permite que um bibliotecário cadastre novos usuários no sistema. | | |
| **Ator Envolvidos** | Bibliotecário | | |
| **Interação entre Ator e Sistema** | **Usuário** | | **Sistema** |
| O bibliotecário acessa a funcionalidade de cadastro de usuários. | |  |
|  | | O sistema exibe um formulário de cadastro de usuários. |
| O bibliotecário preenche os campos do formulário com as informações do novo usuário (nome, email, telefone, tipo), envia o formulário. | |  |
|  | | O sistema armazena as informações do novo usuário no banco de dados. |
|  | | O sistema valida os dados do formulário |
|  | | O sistema exibe uma mensagem de sucesso informando que o usuário foi cadastrado com êxito. |
| **Alternativas** | **AL01** | Se algum campo obrigatório não for preenchido corretamente:   * O sistema exibe uma mensagem de erro indicando os campos que precisam ser corrigidos. * O fluxo retorna ao passo 2. | |

**Fonte:** Autoria própria

#### **4.2.1.1 Especificação dos Casos de Uso**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nome do Caso de Uso** | Cadastro de Livros | | |
| **Descrição** | Este caso de uso permite que um bibliotecário cadastre novos livros no sistema. | | |
| **Ator Envolvidos** | Bibliotécario | | |
| **Interação entre Ator e Sistema** | **Usuário** | | **Sistema** |
| O bibliotecário acessa a funcionalidade de cadastro de livros. | |  |
|  | | O sistema exibe um formulário de cadastro de livros. |
| O bibliotecário preenche os campos do formulário com as informações do novo livro (título, autor, ISBN) e envia o formulário. | |  |
|  | | O sistema valida os dados do formulário. |
|  | | O sistema armazena as informações do novo livro no banco de dados. |
|  | | O sistema exibe uma mensagem de sucesso informando que o livro foi cadastrado com êxito. |
| **Alternativas** | **AL01** | Se algum campo obrigatório não for preenchido corretamente:   * O sistema exibe uma mensagem de erro indicando os campos que precisam ser corrigidos. * O fluxo retorna ao passo 2 | |

**Fonte:** Autoria própria

### **4.2.2 Diagrama de Classes**

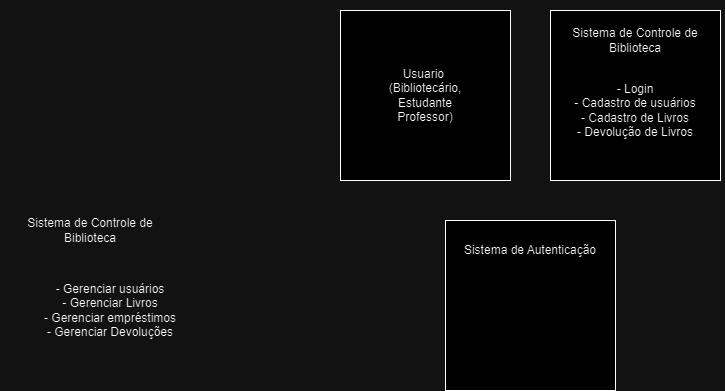


**Fonte:** Autoria própria

#### **4.2.2.1 Especificação das classes**

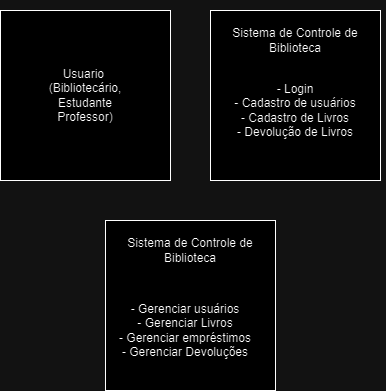
* **Usuário (User)**
  + Atributos: ID, Nome, Email, Telefone, Tipo (Aluno, Professor)
* **Livro (Book)**
  + Atributos: ID, Título, Autor, ISBN, Disponibilidade
* **Empréstimo (Loan)**
  + Atributos: ID, ID\_Usuário, ID\_Livro, Data\_Empréstimo, Data\_Devolução

**4.3 Diagrama de Contexto**



**Fonte:** Autoria própria

**4.3.1 Diagrama de Nível Zero**



**Fonte:** Autoria própria

# **5. ENTREVISTA/QUESTIONÁRIO/**

#### Entrevista / Questionário

* **Público-alvo:** Bibliotecários, usuários frequentes.
* **Perguntas-chave:**
  1. Quais funcionalidades você considera essenciais para um sistema de biblioteca?
  2. Com que frequência você empresta/devolve livros?
  3. Que tipo de relatórios você precisa gerar?

#### Layout de Tela (Protótipo)

* Ferramentas como Figma ou Sketch podem ser usadas para criar protótipos de tela, incluindo:
  + Tela de login.
  + Dashboard principal.
  + Formulário de cadastro de usuários.
  + Formulário de cadastro de livros.
  + Tela de registro de empréstimos e devoluções.
  + Tela de geração de relatórios.

# **6. DICIONÁRIO DE DADOS**

CREATE TABLE Users (

ID INT PRIMARY KEY,

Name VARCHAR(100),

Email VARCHAR(100),

Phone VARCHAR(15),

Type VARCHAR(10)

);

CREATE TABLE Books (

ID INT PRIMARY KEY,

Title VARCHAR(100),

Author VARCHAR(100),

ISBN VARCHAR(20),

Availability BOOLEAN

);

CREATE TABLE Loans (

ID INT PRIMARY KEY,

UserID INT,

BookID INT,

LoanDate DATE,

ReturnDate DATE,

FOREIGN KEY (UserID) REFERENCES Users(ID),

FOREIGN KEY (BookID) REFERENCES Books(ID)

);

# **7. POLÍTICA DE TESTES**

* **Teste de Unidade:** Verificação de funcionalidades individuais.
* **Teste de Integração:** Verificação da interação entre componentes.
* **Teste de Sistema:** Verificação do sistema como um todo.

### Descrever a Implantação

* **Homologação (Validar Aplicação):** Testes em ambiente controlado.
* **Instalação:** Deploy em servidor de produção.
* **Treinamento:** Sessões de treinamento para usuários finais.

# **8 IMPLEMENTAÇÃO: HOMOLOGAÇÃO, INSTALAÇÃO E TREINAMENTO**

O processo de implementação do *software* *para Controle de Biblioteca*, passaram por três fases são elas homologação, instalação e treinamento.

A homologação será realizada em ambiente controlado do desenvolvimento principal com instalação do programa e acesso pelos usuários para os devido testes.

A instalação do sistema se dará por deploy em servidor de produção.

E treinamento para usuários finais.

# **8.1 REGRAS DE NEGÓCIO**

Um usuário não pode emprestar mais de 3 livros simultaneamente.

O prazo máximo para devolução é de 30 dias.

Notificações de atraso são enviadas diariamente após a data de devolução.

# **9 PROTOTIPAÇÃO DO *SOFTWARE*: TELAS**

## **9.1 TELA INICIAL**

Tela inicial do programa, onde será solicitado informações de usuário e senha, e cadastro caso seja o primeiro acesso.

Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente

**Fonte:** Autoria própria

## **9.2 TELA CADASTRO USUÁRIO**

Tela cadastro do usuário, onde constam os principais dados.

Texto

Descrição gerada automaticamente com confiança baixa

**Fonte:** Autoria própria

## **9.3 TELA CADASTRO DE LIVROS**

Tela de cadastro de livros, onde constam as informações referente aos livros.

Texto

Descrição gerada automaticamente com confiança baixa

## **9.4 TELA REGISTRO DE EMPRÉSTIMOS**

Tela de registro de empréstimos com a data do empréstimo e o usuário que o fez.

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

**Fonte:** Autoria própria

## 

## **9.5 TELA REGISTRO DE DEVOLUÇÃO**

Tela de registro de devolução com a data em que foi devolvido e o usuário que devolveu o livro.

Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente

**Fonte:** Autoria própria

## **9.6 TELA CONSULTA DE EMPRÉSTIMO E DEVOLUÇÕES**

Tela de consulta com as informações de empréstimos e devoluções do usuário e a informação do livro.

Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente

**Fonte:** Autoria própria

# **10. CONCLUSÃO**

Esta aplicação tem o intuito de não apenas simplificar o processo de empréstimo e devolução de livros, mas também proporciona uma experiência mais fluida para os usuários, garantindo um acesso fácil e rápido aos serviços da biblioteca. Ao longo do desenvolvimento deste projeto, foram aplicados diversos conceitos e técnicas da Engenharia de Software, desde a modelagem de requisitos até a implementação e testes da aplicação.

Além disso, a aplicação incorporou boas práticas de programação, como o uso do Spring Boot para a construção de uma estrutura robusta de back-end e a implementação de uma interface amigável e simples utilizando HTML, CSS e JavaScript. A integração de um banco de dados relacional possibilitou o armazenamento seguro e eficiente dos dados dos usuários, livros e registros de empréstimos.

Em suma, o projeto de Controle de Biblioteca demonstra não apenas a aplicação de conhecimentos teóricos em Engenharia de Software, mas também a capacidade de desenvolver soluções práticas e funcionais para problemas do mundo real. Este projeto representa um passo importante no aprimoramento dos processos de gestão de bibliotecas, proporcionando benefícios tangíveis tanto para os usuários quanto para os administradores.

# **REFERÊNCIAS**

PRESSMAN, Roger S. Engenharia de Software: uma abordagem profissional. 7. ed. São Paulo: McGraw-Hill, 2011. 920 p

|  |  |
| --- | --- |
|  |  |
|  |  |