**SENAI ETTORE ZANINI**

**TÉCNICO DESENVOLVIMENTO DE SISTEMA**

**Lucas Henrique Soares de Oliveira**

**BiblioLH**Gerenciamento   
de  
Biblioteca

**Sertãozinho  
2024**

LISTA DE ILUSTRAÇÕES SE HOUVER (GERAR AUTOMATICO)

TABELAS SE HOUVER (GERAR AUTOMATICO)

**SUMÁRIO**

**1 INTRODUÇÃO**

A BiblioLH é uma biblioteca inovadora que se destaca por sua missão de promover o conhecimento e a cultura entre a comunidade. Com um acervo diversificado que abrange desde clássicos da literatura até publicações contemporâneas, a BiblioLH se esforça para atender a todos os perfis de leitores. No entanto, à medida que o número de usuários e o volume de empréstimos aumentam, surgem desafios significativos na gestão do acervo.

Problemas como a dificuldade em rastrear a disponibilidade de livros, a organização dos empréstimos e devoluções, e a necessidade de um sistema eficiente de reservas têm impactado a experiência dos usuários e a eficiência da biblioteca. Para resolver essas questões, a BiblioLH busca implementar um sistema integrado que otimize todos os processos relacionados ao empréstimo e à gestão do acervo. Este sistema não apenas melhorará a experiência dos usuários, mas também facilitará a organização interna, permitindo que a BiblioLH continue a ser um espaço vibrante e acolhedor para todos os amantes da leitura.

O capítulo 2 trata sobre Requisitos do Sistema, o capítulo 3 descreve os Diagramas do Sistema, o capítulo 4 é sobre as Rotas da Aplicação do Back-End e o capítulo 5 irá mostrar a Interface e Experiência.

**2 REQUISITOS DO SISTEMA**

“Todo capítulo precisa ter no mínimo três linhas”.

“Todo capítulo inicia em uma página”

2.1 Requisitos funcionais

Descrever sobre o que é os requisitos funcionais (dissertar, nada de pergunta e resposta)

“Toda tabela deve legenda”

|  |  |
| --- | --- |
| Requisito Funcional | |
| RF001 | Cadastrar Livro |
| RF002 | Editar Livro |
| RF003 | Excluir Livro |
| RF004 | Numerar Livros |
| RF005 | Sistemas de Notificações |
| RF006 | Interface do Usuário |
| RF007 | Empréstimo Livro |
| RF008 | Registrar Devolução |
| RF009 | Consultar Empréstimo |

**2.2.1 RF001 – Cadastrar Livro**

Descrição: O sistema deve permitir que o usuário cadastre novos livros no acervo da biblioteca.

Prioridade: Alta

Entrada:

* Título do livro (obrigatório).
* Autor(es) (obrigatório).
* Preencher com os demais campos

Processamento:

* O sistema deve validar os campos obrigatórios.

Saída:

* Mensagem de sucesso ao cadastrar o livro.
* Mensagem de erro em caso de campos obrigatórios não preenchidos ou duplicidade de cadastro.

Pós-condição:

* O livro cadastrado deve ser exibido na lista de acervo disponível.

**2.2.2 RF002 – Editar Livro**

Descrição: O sistema deve permitir que o usuário edite as informações de um livro previamente cadastrado.

Prioridade: Alta

Entrada:

* Identificação do livro a ser editado (ID ou ISBN).
* Campos a serem editados (colocar aqui).

Processamento:

* O sistema deve validar os campos obrigatórios.

Saída:

* Mensagem de sucesso ao salvar as alterações.
* Mensagem de erro caso haja duplicidade ou campos obrigatórios não preenchidos.

Pós-condição:

* As alterações devem ser refletidas na lista de acervo e nos registros de empréstimos associados ao livro.

**2.2.3 RF003 – Excluir Livro**

Descrição: O sistema deve permitir que o usuário exclua um livro do acervo.

Prioridade: Média

Entrada:

* Identificação do livro a ser excluído (ID).

Processamento:

* O sistema deve verificar se há empréstimos ativos ou pendentes associados ao livro.
* Se houver empréstimos associados, o sistema deve impedir a exclusão e informar o usuário.

Saída:

* Mensagem de sucesso ao excluir o livro.
* Mensagem de erro caso existam empréstimos associados impedindo a exclusão.

Pós-condição:

* O livro deve ser removido da lista de acervo e não deve aparecer nas buscas e consultas.

# 2.2.4 RF004 – Relatórios e Estatísticas

Descrição: Gerar relatórios sobre livros mais emprestados e usuários com mais empréstimos

Prioridade: Alta

Entrada: Gerar relatórios

Saída: Fornecer dados sobre a frequência de uso da biblioteca

# 2.2.5 RF005 – Sistemas de Notificações

Descrição: Enviar lembretes sobre devoluções, reservas disponíveis e novas aquisições.

Prioridade: Alta

Entrada: Notificações por E-mail

Saída: Enviar lembretes, reservas, e novas aquisições

# 2.2.6 RF006 – Interface do Usuário

Descrição: Desenvolver uma interface amigável que facilite a navegação e o uso do sistema.

Prioridade: Alta

Entrada: Desenvolver uma interface

Saída: Garantir que o sistema seja acessível a todos os usuários, incluindo aqueles com deficiências.

RF007 – Empréstimo de Livro

Descrição: O sistema deve permitir que o usuário registre o empréstimo de um ou mais livros para um aluno cadastrado.

Prioridade: Alta

Entrada:

* Identificação do aluno (número de matrícula).
* Identificação do(s) livro(s) a ser(em) emprestado(s) (ID do livro).
* Data de início do empréstimo (obrigatória).
* Data prevista de devolução.

Processamento:

* O sistema deve verificar a disponibilidade dos livros no acervo.
* O sistema deve reduzir a quantidade disponível do livro emprestado.
* O sistema deve associar o empréstimo ao aluno e ao(s) livro(s) selecionado(s).

Saída:

* Mensagem de sucesso ao registrar o empréstimo.
* Mensagem de erro caso algum dos livros não esteja disponível.

Pós-condição:

* O empréstimo deve ser registrado com status "Ativo".
* A quantidade disponível do livro deve ser atualizada no sistema.

3 DIAGRAMAS DO SISTEMA

Elas são ferramentas fundamentais para o desenvolvimento de sistemas e para os programadores, pois oferecem diversas vantagens como: visualização; comunicação; documentação; planejamento, identificação de problemas e manutenção. Os diagramas melhoram a eficiência, a comunicação e a qualidade do desenvolvimento do software

3.1 Diagrama de Classe

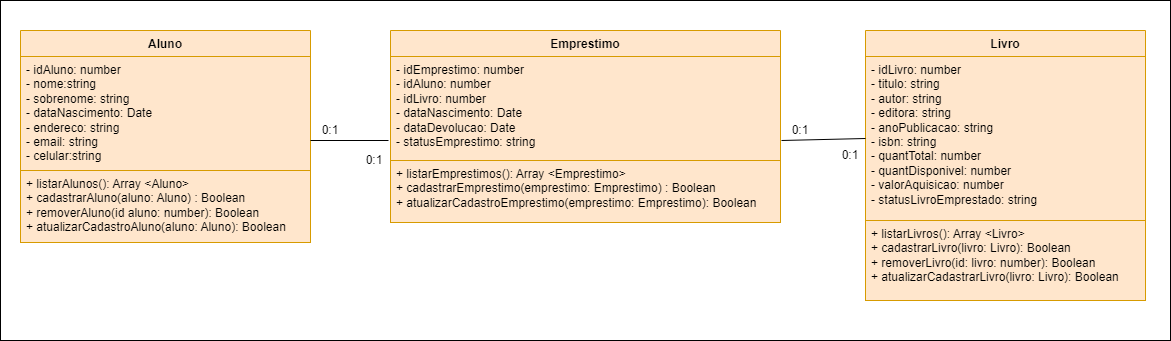
A escrita deve ser direcionada para a importância do diagrama de classe para o sistema/ programador e inserir a imagem

Finalidade do Diagrama de Classes:

Para o programador: Estruturação; planejamento; implementação; interações; e documentação

Para o sistema: Visualização; Análise de requisitos; facilidade de manutenção e consistência

O diagrama de classes é fundamental tanto para o programador backend quanto para a compreensão e manutenção do sistema como um todo



3.2 Banco de Dados

Nesse capítulo deve dissertar sobre o banco utilizado, e suas características.

Deve citar o DER do sistema, qual a importância dele e inserir a imagem.

Deve citar sobre os scripts e relatar a função de cada um e inserir ele na documentação.

SGBD PostgreSQL

Características:

Open Source: O Postgre é um sistema de gerenciamento de banco de dados relacional de código aberto, ( é gratuito e pode ser modificado )

Extensibilidade: Suporta extensões que permitem adicionar novos tipos de dados, operadores e funções, proporcionando flexibilidade para atender a necessidades específicas

Suporte a SQL e Procedural Languages: Oferece suporte completo a SQL e várias linguagens procedurais, como PL/pgSQL, PL/Perl e PL/Python, permitindo a criação de funções e procedimentos armazenados

Conformidade com ACID: Garante atomicidade, consistência, isolamento e durabilidade nas transações, assegurando a integridade dos dados

**Suporte a dados não estruturados**: Além de dados relacionais, suporta tipos de dados não estruturados, como JSON, XML e arrays

**Escalabilidade e Desempenho**: É projetado para gerenciar grandes volumes de dados e pode ser escalado horizontalmente com sharding

**Replication and Clustering**: Oferece suporte a replicação e clustering para alta disponibilidade e recuperação de desastres

Modelo

Ele é baseado em um modelo relacional, onde os dados são organizados em tabelas compostas por linhas e colunas. As principais características do modelo incluem:

**Tabelas**: Estruturas fundamentais para armazenamento de dados

**Chaves Primárias e Estrangeiras**: Permitem relacionar tabelas e garantir a integridade referencial

**Views**: Permitem a criação de representações virtuais de consultas, simplificando o acesso a dados complexos

**Indices**: Melhoram a performance de consultas ao permitir acesso mais rápido aos dados

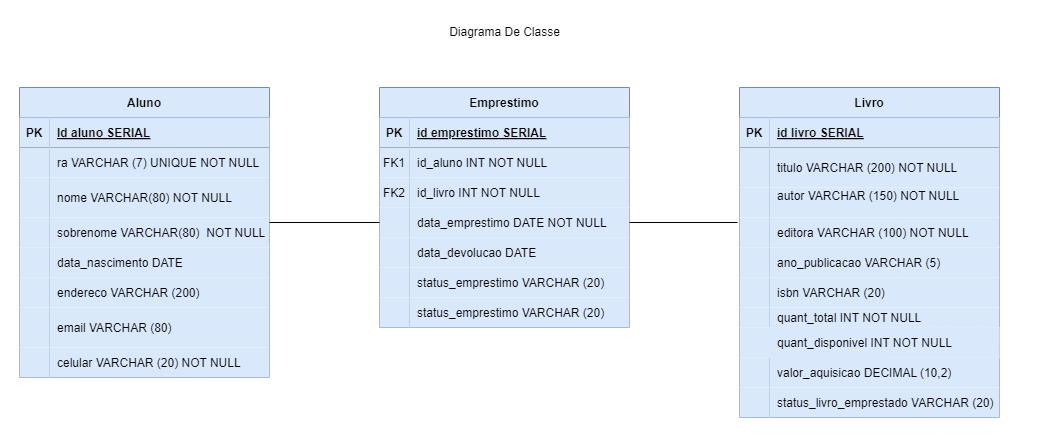
Interface

**psql**: Uma interface de linha de comando poderosa que permite executar comandos SQL diretamente no banco de dados

**PgAdmin**: Uma interface gráfica que facilita a administração e visualização de bancos de dados, oferecendo ferramentas para gerenciamento, execução de consultas e visualização de dados

**Drivers e APIs**: Suporte a diversas linguagens de programação (como Python, Java, Node.js) através de drivers e bibliotecas, facilitando a integração com aplicações

O PostgreSQL é um SGBD robusto e flexível com um modelo relacional bem estruturado e múltiplas interfaces para facilitar a interação



**4 ROTAS DA APLICAÇÃO – BACK-END**

Explica o que é rota , lembrando que nada nesse documento deve ser tratado como pergunta e resposta, cabe a contextualização ou o termo dissertação.

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

5 INTERFACE E EXPERIÊNCIA DO USUÁRIO

Escreva o que é uma interface e o objetivo dela

5.2 Paleta de Cores

Escreva sobre a paleta de cores e insira a imagem delas

5.3 Mockup

Texto antes da imagem, relatando a qual entidade aquela interface pertence

BIBLIOGRAFIA

Link das documentações utilizadas e livros consultados