**SENAI ETTORE ZANINI**

**TÉCNICO DESENVOLVIMENTO DE SISTEMAS**

**Dandara Gabrielli Santos Domingos**

**Nuvem de Livros**Gerenciamento   
de  
Biblioteca

**Sertãozinho  
2024**

LISTA DE ILUSTRAÇÕES SE HOUVER (GERAR AUTOMATICO)

TABELAS SE HOUVER (GERAR AUTOMATICO)

**SUMÁRIO**

**1 INTRODUÇÃO**

Uma biblioteca é onde os alunos podem explorar livremente novos universos literários e adquirir conhecimento a cada leitura. Porém gerenciar todo esse acervo pode ser uma tarefa desafiadora. Controlar empréstimos, localizar obras específicas e acompanhar a disponibilidade de livros são processos que, sem uma organização eficiente, podem consumir tempo e energia tanto dos alunos quanto dos bibliotecários.

É aí que entra a importância de um sistema digital, com ele os estudantes podem de onde estiverem verificar se o livro que desejam está disponível, também fazer reservas e acompanhar datas de devolução. Já para os bibliotecários o sistema facilita o gerenciamento do acervo, registrando movimentações de empréstimos e devoluções além de gerar relatórios úteis para a organização do espaço. Assim a Nuvem de Livros não é apenas uma biblioteca, mas um ambiente dinâmico e prático, onde a tecnologia aprimora a experiência de leitura, tornando-a mais acessível e agradável para todo.

**2 REQUISITOS DO SISTEMA**

2.1 Requisitos funcionais

Os requisitos de sistema funcionais são especificações que definem o comportamento e as funcionalidades de um sistema ou software, detalhando o que ele deve fazer para atender às necessidades do usuário. Portanto são fundamentais para guiar o desenvolvimento, garantindo que o produto final cumpra os objetivos estabelecidos.

|  |  |
| --- | --- |
| Requisito Funcional | |
| RF001 | Cadastrar Livro |
| RF002 | Editar Livro |
| RF003 | Excluir Livro |
| RF004 | Cadastrar Aluno |
| RF005 | Editar Aluno |
| RF006 | Excluir Aluno |
| 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).
* Editora (obrigatório)
* Ano de publicação
* ISBN
* Quantidade total (obrigatório)
* Quantidade disponível (obrigatório)
* Valor de aquisição
* Status do livro emprestado

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 – Cadastrar Aluno**

Descrição: O sistema deve permitir que o usuário cadastre novos alunos.

Prioridade: Alta

Entrada:

* RA do aluno (obrigatório).
* Nome (obrigatório).
* Sobrenome (obrigatório).
* Data de nascimento.
* Endereço do Aluno.
* Email.
* Celular (obrigatório).

Processamento:

* O sistema deve validar os campos obrigatórios.

Saída:

* Mensagem de sucesso ao cadastrar do aluno.
* Mensagem de erro em caso de campos obrigatórios não preenchidos.

Pós-condição:

* O aluno cadastrado deve ser exibido na lista de alunos registrados.

**2.2.5 RF005 – Editar Aluno**

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

Prioridade: Média

Entrada:

* RA do aluno (obrigatório)
* Nome
* Sobrenome
* Data de nascimento
* Endereço do Aluno
* Email
* Celular (obrigatório)

Processamento:

* O sistema deve validar o RA do aluno para garantir que o aluno existe.
* O sistema deve validar o campo obrigatório "Celular".

Saída:

* Mensagem de sucesso ao atualizar as informações do aluno.
* Mensagem de erro caso o RA não seja encontrado ou campos obrigatórios não estejam preenchidos.

Pós-condição:

* As informações atualizadas do aluno devem ser refletidas na lista de alunos registrados.

**2.2.6 RF006 – Excluir Aluno**

Descrição: O sistema deve permitir que o usuário exclua alunos previamente cadastrados no sistema.

Prioridade: Alta

Entrada:

* RA do aluno (obrigatório).

Processamento:

* Sistema deve garantir que o aluno seja removido.
* O sistema deve validar se o RA informado corresponde a um aluno cadastrado.

Saída:

* Mensagem de sucesso ao excluir o aluno.
* Mensagem de erro caso o RA não esteja registrado no sistema.

Pós-condição:

* O aluno excluído não deve mais aparecer na lista de alunos registrados.

**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.

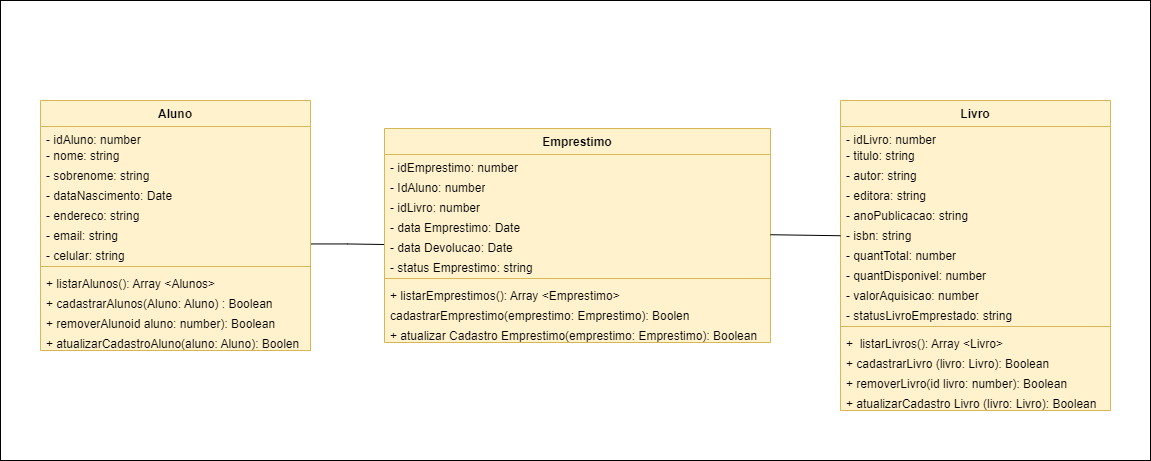
**SPRINT 2**

**3 DIAGRAMAS DO SISTEMA**

Os diagramas são ferramentas essenciais para programadores e o desenvolvimento de sistemas, pois oferecem uma visão clara e organizada da arquitetura dos fluxos de dados e das interações entre componentes. Eles facilitam a comunicação entre equipes técnicas e não técnicas permitindo que todos compreendam melhor o funcionamento do sistema, além disso ajudam a identificar problemas e otimizar processos tornando o desenvolvimento mais eficiente e alinhado aos requisitos estabelecidos.

**3.1 Diagrama de classe**

O diagrama de classe é importante para o programador backend porque define a estrutura e os relacionamentos entre as classes que compõem o sistema. Assim, ele facilita a modelagem da lógica de negócios garantindo que a organização do código siga padrões claros, além de ajudar na implementação e manutenção de um código robusto e escalável.

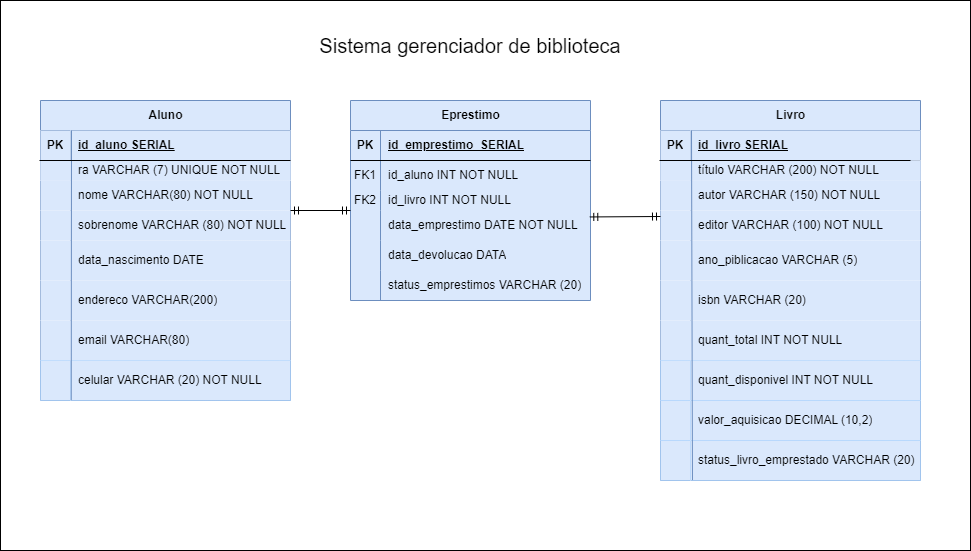


**3.2 Banco de Dados**

Um Sistema de Gerenciamento de Banco de Dados (SGBD) é utilizado para organizar, armazenar e gerenciar grandes volumes de dados de forma eficiente, oferecendo suporte a diferentes modelos de dados, como o relacional (baseado em tabelas), o NoSQL (orientado a documentos, chave-valor ou grafos) dependendo

das necessidades do sistema. Tendo características que incluem o controle de transações, integridade, segurança e mecanismos de recuperação de falhas, portanto que a interface de um SGBD pode ser tanto gráfica facilitando o uso por administradores com ferramentas visuais, quanto baseada em linguagens como SQL permitindo consultas e manipulações diretas pelos programadores.

O Diagrama DER é uma representação visual dos dados e seus relacionamentos dentro de um sistema que é essencial para o design de bancos de dados, ele permite que programadores e analistas compreendam claramente como as entidades se conectam, assim ajudando a garantir a integridade dos dados e a eficiência na implementação da estrutura de dados da aplicação.



**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