

Documentação

Desafio Livraria Virtual - SQUAD 1

Autores

Bruno Gomes
Danielle Sanches
Fábio Macedo
Yasmin Lima
Higor

2024

1. Visão geral

O objetivo deste documento é trazer informações a respeito do projeto Livraria Virtual, e é destinado aos usuários da aplicação para fornecer um melhor entendimento de seu funcionamento e sanar quaisquer dúvidas provenientes ao seu uso.

A livraria virtual é um projeto que consiste em desenvolver um sistema de gerenciamento de uma livraria virtual, onde é possível o cadastro, a compra e a venda de livros, tanto eletrônicos quanto impressos. Esse desafio foi proposto pela Solutis School Dev Trail - Nivelamento para a turma colocar em prática os conhecimentos adquiridos acerca de conceitos como classes, classes abstratas, composição, herança e polimorfismo utilizando a linguagem de programação Java.

O objetivo do projeto é construir uma aplicação em Backend para cadastro listagem e venda de livros de uma biblioteca, bem como sua conexão ao banco de dados MySql

2. Tecnologias utilizadas

- **Linguagem de programação**

A linguagem de programação utilizada para desenvolver esse projeto foi o Java

- **IDEs**

As IDEs utilizadas e recomendadas para o desenvolvimento do projeto foram:

- a) Eclipse;
- b) Netbeans;
- c) IntelliJ IDEA.

- **Banco de dados**

O banco de dados MySQL foi utilizado para armazenar as informações do sistema.

- **Controle de versão**

Foi utilizado o Git/GitHub para gerenciamento de versão e colaboração em equipe.

3. Funcionamento do sistema

1. Menu: A página inicial do sistema é um menu com as seguintes opções:

- Cadastrar livro;
- Realizar uma venda;
- Listar livros;
- Listar vendas;
- Sair.

2. Cadastrar livro: Nesta seção, o usuário escolhe qual tipo de livro deseja cadastrar através das seguintes opções:

- Impresso:
 - Título do livro;
 - Autores;
 - Editora;
 - Frete;
 - Estoque;
 - Preço.
- Eletrônico:
 - Título do livro;
 - Autores;
 - Editora;
 - Tamanho do arquivo;
 - Preço.
- Ambos.
 - Neste caso, o usuário cadastrará o mesmo livro em ambos os formatos, em que preencherá os dados referentes a ambos, como os atributos específicos a cada tipo

Com o cadastro devidamente realizado, o livro cadastrado será persistido no banco de dados.

3. Realizar uma venda:

Nesta seção, o usuário preenche suas informações e escolhe o(s) livro(s) que irá comprar, através de uma lista dos livros cadastrados no banco de dados, por meio das seguintes opções:

- Insira o nome do cliente;
- Quantos livros serão comprados?
- Qual tipo de livro?

- Selecione o livro.

Após preencher essas informações, o sistema exibe o valor total da venda e solicita a confirmação do usuário. Quando o usuário confirma as informações, ele informa que a venda foi realizada com sucesso e, neste caso, o estoque do livro é atualizado. Caso o usuário não confirme as informações, o sistema informa que a venda foi cancelada.

4. Listar livros: Nesta seção, o usuário pode escolher para que exiba os livros cadastrados previamente e que estão no banco de dados:

- Listar livros impressos: Exibe as informações sobre os livros impressos que foram cadastrados no sistema;
- Listar livros eletrônicos: Exibe as informações sobre os livros eletrônicos que foram cadastrados no sistema;
- Listar ambos: Exibe as informações sobre ambos os tipos de livros cadastrados no sistema;
- Retornar ao menu principal.

5. Listar vendas: Nesta seção são exibidas na tela as vendas realizadas, juntamente com o nome do cliente e o valor da compra.

6. Sair: Essa opção encerra a execução do programa.

4. Banco de Dados

O banco de dados para o projeto da livraria virtual foi criado utilizando o MySQL Workbench 8.0.39, juntamente com o plugin JDBC 8.30 para conectar a aplicação Java ao banco de dados. A seguir, explicamos como esse banco de dados foi estruturado e como as classes DAO (Data Access Object) interagem com ele.

1. Estrutura do Banco de Dados

O banco de dados foi criado no MySQL e possui uma tabela principal chamada Livro, que armazena informações sobre os livros cadastrados. Além disso, há uma tabela chamada Venda, que registra as vendas realizadas, e uma tabela LivrosVendidos, que estabelece a relação entre uma venda específica e os livros vendidos.

2. Conexão com o Banco de Dados

A classe DAO que está na pasta model é responsável por estabelecer a conexão com o banco de dados. Ela utiliza o driver JDBC (com.mysql.cj.jdbc.Driver) e os parâmetros necessários, como a URL do banco de dados, o usuário, e a senha, para criar a conexão:

```
15         public Connection conectar() {
16             try {
17                 Class.forName(driver);
18                 con = DriverManager.getConnection(url,user,password);
19                 return con;
20
21             } catch (Exception e) {
22                 System.out.println(e);
23                 return null;
24             }
25         }
```

3. Manipulação de Dados com a Classe LivroDAO

A classe LivroDAO é utilizada para realizar operações relacionadas aos livros, como cadastrar, listar, e buscar livros no banco de dados. Algumas operações específicas incluem:

- Cadastro de Livros Impressos e Eletrônicos:
 - Os métodos cadastrarLivroImpresso e cadastrarLivroEletronico inserem dados na tabela Livro, diferenciando entre livros impressos e eletrônicos com base no tipo (tipo).

```
String sql = "INSERT INTO Livro (titulo, autores, editora, preco, tipo, estoque, tamanho, frete) VALUES (?, ?, ?, ?, ?, ?, ?, ?)";
```

- As informações específicas de cada tipo de livro são preenchidas nas colunas apropriadas, como frete para livros impressos e tamanho para livros eletrônicos.
- Listagem de Livros:
 - O método listarLivros permite listar livros de acordo com o tipo (impresso ou eletronico), utilizando uma consulta SQL para filtrar os resultados.

```
String sql = "SELECT * FROM Livro WHERE tipo = ?";
```

4. Manipulação de Vendas com a Classe VendaDAO

A classe VendaDAO gerencia as operações relacionadas às vendas, incluindo:

- Cadastro de Vendas:
 - O método adicionarVenda insere uma nova venda na tabela Venda, utilizando o retorno da chave gerada automaticamente (RETURN_GENERATED_KEYS) para associar os livros vendidos a essa venda específica.

```
String sqlVenda = "INSERT INTO Venda (cliente, valor) VALUES (?, ?)";
```

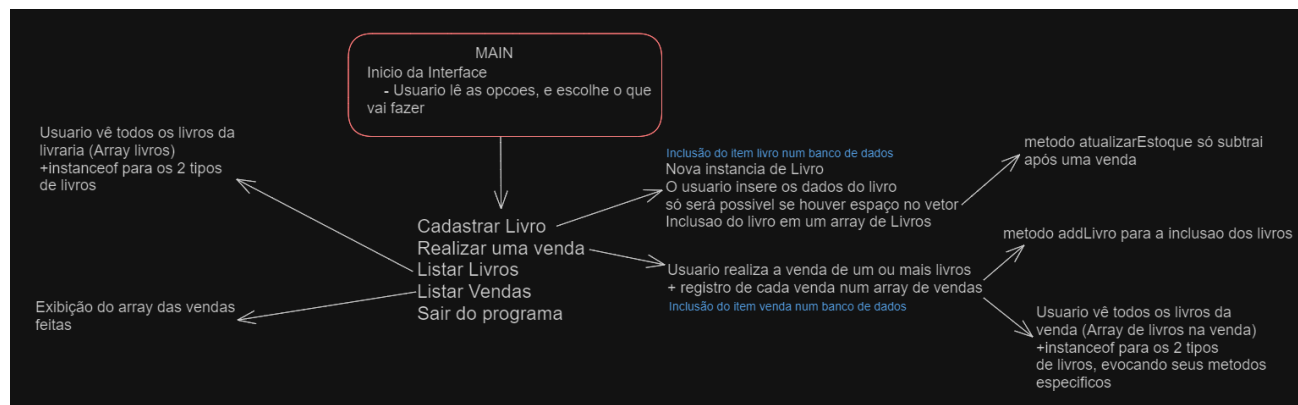
- Associação de Livros Vendidos:
 - O método adicionarLivroVendido insere os IDs dos livros vendidos e o ID da venda na tabela LivrosVendidos, criando uma relação entre a venda e os livros.

5. Conclusão

Essa aplicação Java, juntamente com toda a devida estrutura de Banco de Dados MySQL, é capaz de lidar com uma gama considerável de situações envolvendo o uso para a qual foi destinada.

6. Conteúdo Bônus

Para encerrar, este foi o rascunho inicial para elucidar o pensamento lógico que deu o pontapé do projeto:



Todo o projeto foi feito com fins didáticos, com muito esforço, trabalho em equipe e dedicação. Caso queira acompanhar o repositório individual dos participantes da equipe, abaixo os links:

Bruno: <https://github.com/devbrunogomes>

Danniele: <https://github.com/DanySanches>

Fábio: <https://github.com/fabio-macedo>

Higor: <https://github.com/HigorAzvedo>

Yasmin: <https://github.com/yasmim-lima>