

Table of Contents

Biblioteca 2

Biblioteca



Login e Acesso

- **Admin:** cadastra livros, usuários e vê relatórios.
- **Usuário comum:** realiza e devolve empréstimos.



Funcionalidades



Cadastro de Livros

- Campos: título, autor, ano, editora, ISBN, quantidade, categoria.
- Edição e exclusão diretas na tabela.



Gerenciamento de Usuários

- Cadastro de nome, curso e turma.
- Visualização e edição de dados.



Login e Sessões

- Login com verificação de e-mail e senha.
- Diferencia admin e usuário comum por perfil.



Empréstimos de Livros

- Seleção de usuário + livro.
- Verifica disponibilidade (quantidade > 0).
- Marca devoluções e registra data.



Relatórios e Filtros



Top Livros Emprestados (com gráfico)

- Usa `BarChart` com CSS personalizado.
- Mostra os livros mais emprestados com destaque.



Histórico por Período

- `DatePicker` permite filtrar por:
 - Mês, Ano ou Intervalo
- Disponível para:
 - Histórico individual do aluno
 - Histórico geral de devoluções



Tratamento de Erros

Situação	Solução
Campo vazio	Validação via alertas
<code>NullPointerException</code>	Verificação de dados retornados (ex: <code>getUser()</code> nulo)
CSS não aplicado	Verifique caminho correto: <code>barChart.getStylesheets().add</code>

Conclusão

Sistema de Biblioteca em JavaFX

Este sistema de biblioteca foi desenvolvido utilizando JavaFX e PostgreSQL, com o objetivo de gerenciar livros, usuários e empréstimos,

incluindo funcionalidades específicas como login de administrador, controle de estoque de livros, visualização de histórico de empréstimos e gráficos analíticos.

Estrutura do Projeto

```
src/  
└─ main/  
└─ java/com/example/library/  
    │ └─ controle/           # Controllers (JavaFX)  
    │ └─ dao/               # Data Access Objects  
    │ └─ model/             # Entidades (Book, Loan, User...)  
    │ └─ util/              # Classes utilitárias (ex: Database.java)  
    └─ resources/View/      # Arquivos FXML e CSS
```

Funcionalidades Principais

1. Login e Controle de Acesso

Login baseado em email e senha.

Distinção entre usuários e administradores.

Acesso restrito a algumas telas para administradores.

2. Cadastro e Gerenciamento de Livros

Adição, edição e remoção de livros.

Registro de quantidade de exemplares.

Controle automático de disponibilidade com base em empréstimos ativos.

3. Cadastro e Gerenciamento de Usuários

Registro de alunos com nome, curso e turma.

Busca e edição de dados.

4. Empréstimos

Tela para registrar novo empréstimo.

Verificação de disponibilidade do livro.

Prevenção de empréstimos duplicados para o mesmo livro.

5. Devoluções

Tela dedicada para marcar devoluções.

Atualização automática do status e da quantidade de exemplares disponíveis.

6. Histórico por Usuário

Lista completa de empréstimos por aluno.

Filtros por período (início e fim).

Indicação de status ("Em Aberto" ou "Devolvido").

7. Histórico Geral de Devoluções

Tela para visualizar todos os livros devolvidos.

Filtro por intervalo de datas.

8. Gráfico de Livros Mais Empréstados

BarChart mostrando os 15 livros mais emprestados.

Estilizado com chart-style.css (cores distintas).

Rótulos com contagem exibidos acima das barras.

DAO (Data Access Object)

Principais DAOs:

BookDAO: Insere, lista, edita e deleta livros.

LoanDAO: Gerencia empréstimos e devoluções.

LibraryUserDAO: CRUD completo para usuários.

CategoryDAO: Gerencia categorias dos livros.

Utilizam conexões via `Database.getConnection()`.

Interface JavaFX

Componentes utilizados:

FXML: Interface declarativa (FXML + SceneBuilder).

TableView, TextField, DatePicker, ComboBox, BarChart.

FXMLLoader com controle modularizado por tela.

Telas principais:

login.fxml

main.fxml

add_book.fxml

loan_management.fxml

returned_loans.fxml

user_history.fxml

top_books.fxml

Validações e Tratamentos

Erros SQL e entradas inválidas são tratados com Alert.

Quantidade e ano validados como inteiros.

Combos obrigatórios checados antes do envio.

Estilização e Gráficos

Arquivo chart-style.css define cores para até 15 barras distintas.

Estilo aplicado via:

```
barChart.getStylesheets().add(getClass().getResource("/View/chart-style.css").toExternalForm());
```

Rótulos com valor adicionados dinamicamente:

```
Label label = new Label(String.valueOf(entry.getValue()));  
label.setTranslateY(-20);  
((StackPane) data.getNode()).getChildren().add(label);
```

Como Rodar o Projeto (Linux e Windows)

Linux

```
cd target  
java -jar library-fx-1.0-SNAPSHOT.jar
```

Ou use o script:

```
#!/bin/bash  
java -jar target/library-fx-1.0-SNAPSHOT.jar
```

Windows

Compile no IntelliJ ou Maven.

Navegue até target.

Execute com:

double-click em library-fx-1.0-SNAPSHOT.jar

Requisitos

Java 17+

JavaFX SDK 17+ (controls, fxml)

PostgreSQL

IDE (recomendado: IntelliJ com JavaFX Plugin)

Tabela do Banco de Dados

Tabela loans

Campo

Tipo

Descrição

id

serial

ID do empréstimo

user_id

int

Aluno que pegou o livro

book_id

int

Livro emprestado

loan_date

date

Data de empréstimo

return_date

date

Data de devolução (se houver)

returned

boolean

Se foi devolvido ou não

✅ Conclusão

Este projeto oferece um sistema robusto para bibliotecas com controle completo de usuários, livros, empréstimos, histórico e estatísticas visuais. Modularizado, com boas práticas de MVC e pronto para melhorias futuras (ex: relatórios PDF, envio por email, backup automático etc).

Sinta-se à vontade para adaptar este README para o Writerside ou documentação interna.

 Execução

 Linux

```bash

java -jar target/library-fx.jar