

Documentação do Projeto — Sistema de Biblioteca

Autor: Gabriel Vinícius

Resumo / Objetivo

Este projeto é um sistema de gerenciamento de biblioteca implementado em C. Seu objetivo é controlar empréstimos e reservas de recursos físicos da instituição (livros, calculadoras, fones de ouvido e salas), fornecendo persistência por meio de arquivos texto simples (`src/bd/`). O sistema foi desenvolvido para uso em linha de comando, com níveis de acesso para administrador e usuário comum.

Estrutura da Aplicação

Arquivos e pastas principais:

- `main.c` — Ponto de entrada do programa, apresenta o menu inicial e delega para as interfaces.
 - `compile.sh` / `compile.bat` — Scripts de compilação (Linux / Windows).
 - `run.sh` / `run.bat` — Scripts de execução (Linux / Windows).
 - `src/` — Código-fonte organizado por responsabilidade:
 - `src/bd/` — Arquivos de dados (`txt`) usados para persistência: `livros.txt`, `calculadoras.txt`, `fones_ouvido.txt`, `salas.txt`, `emprestimos.txt`, `reservas.txt`.
 - `src/datas/` — Manipulação e validação de datas (`datas.c`, `datas.h`).
 - `src/input/` — Funções de entrada do usuário (`input.c`, `input.h`).
 - `src/interfaces/` — Menus e fluxos de interação no terminal (`interfaces.c`, `interfaces.h`).
 - `src/manipulacao/` — Leitura e escrita em arquivos, persistência (`arquivo.c`, `arquivo.h`).
 - `src/structs/` — Definições de estruturas de dados (`structs.h`).
 - `src/themes/` — Definições de cores/temas para saída no terminal (`theme.h`).
 - `src/utils/` — Funções auxiliares usadas pelo sistema (`utils.c`, `utils.h`).
-

Modelos de Dados (Structs)

Principais structs usadas no sistema (resumido):

- `Livro` — `id`, `nome`, `autor`, `categoria`, `disponivel`
- `Calculadora` — `id`, `modelo`, `marca`, `disponivel`
- `Fone_Ouvido` — `id`, `modelo`, `marca`, `disponivel`

- **Sala** — id, sala, max_pessoas, disponivel
 - **Emprestimo** — id, id_recurso, tipo_recurso, nome_recurso, data_emprestimo, data_devolucao, concluido, atrasado
 - **Reserva_Sala** — id, sala, data_reserva, duracao, qntd_pessoas, concluido, atrasado
 - **Data** — dia, mes, ano, hora, minuto
-

Funcionalidades Implementadas

- Listagem de recursos: livros, calculadoras, fones de ouvido e salas.
 - Empréstimos de livros, calculadoras e fones de ouvido: registro do empréstimo, controle de disponibilidade e devolução.
 - Reservas de salas: reserva informando data, horário, duração e quantidade de pessoas.
 - Listagem de empréstimos e reservas atuais.
 - Marcar empréstimos e reservas como concluídos (devolução / fim da reserva).
 - Validação de disponibilidade: o sistema não permite emprestar ou reservar itens indisponíveis.
 - Perfis de usuário: Administrador (opções extras para gestão) e Usuário Comum.
 - Administrador: registro, exclusão e edição de recursos.
-

Funcionalidades Não Implementadas / Pendências

- Filtros avançados na listagem de recursos (por autor, categoria, marca, etc.).
 - Aplicação de multa em empréstimos atrasados.
 - Bloqueio automático de usuários que tenham atrasos ou multas.
 - Sistema de pagamento de multas.
-

Como Compilar e Executar

Linux

1. Compilar:

```
./compile.sh
```

ou

```
bash compile.sh
```

2. Executar:

```
./run.sh
```

ou

```
bash run.sh
```

Observação importante: o executável gerado (**a.out** por padrão) pode precisar de permissão de execução. Se receber erro de permissão, execute:

```
chmod +x a.out  
# e então  
./a.out
```

Se o seu **run.sh** invocar **./a.out**, verifique também se **run.sh** tem permissão de execução:

```
chmod +x run.sh  
./run.sh
```

Windows

Use **compile.bat** e **run.bat** no Prompt de Comando.

Local dos Dados

Os arquivos de dados ficam em **src/bd/**. Para popular o sistema, veja o arquivo **BANCO_DADOS.txt** com exemplos iniciais.

Observações Finais

- O sistema é voltado para uso acadêmico e demonstra conceitos de manipulação de arquivos, estruturas em C, alocação dinâmica de memória e organização de um pequeno projeto CLI.
 - Para detalhes de implementação (funções, assinaturas, fluxos) consulte **DOCUMENTACAO_BIBLIOTECA.md** e os arquivos fonte em **src/**.
-

Desenvolvido por Gabriel Vinícius — Disciplina: Linguagem de Programação.

Arquivo gerado automaticamente a partir da estrutura e documentação do projeto.
