

# Relatório Técnico: Sistema de Gerenciamento de Biblioteca

## 1. Introdução

Este relatório técnico descreve o Sistema de Gerenciamento de Biblioteca desenvolvido em C++. O sistema foi projetado para facilitar a administração de usuários, livros e empréstimos em uma biblioteca. Ele inclui funcionalidades para adicionar, remover e consultar usuários e livros, além de registrar e gerenciar empréstimos, aplicando multas.

## 2. Objetivo

O objetivo deste sistema é fornecer uma interface simples e eficiente para a gestão de uma biblioteca. As principais funcionalidades incluem:

- Adicionar, remover e consultar usuários.
- Adicionar, remover e consultar livros.
- Registrar e gerenciar empréstimos de livros.
- Verificar a disponibilidade de livros.
- Aplicar multas e controlar prazos de devolução.
- Gerar relatórios sobre usuários, livros e empréstimos.

## 3. Arquitetura do Sistema

O sistema é organizado em várias classes e funções principais:

### 3.1. Classes

#### 1. GerenciamentoUsuario

- **Responsabilidades:** Gerencia a adição, remoção e consulta de usuários.

- **Métodos Principais:**

- `adicionarUsuario(Usuario usuario):` Adiciona um novo usuário.
- `removerUsuario(int id):` Remove um usuário pelo ID.
- `consultarUsuario(int id):` Consulta um usuário pelo ID.
- `exibirRelatorioUsuarios():` Exibe um relatório com todos os usuários.

## 2. GerenciamentoLivros

- **Responsabilidades:** Gerencia a adição, remoção e consulta de livros.

- **Métodos Principais:**

- `adicionarLivro(Livro livro):` Adiciona um novo livro.
- `removerLivro(int id):` Remove um livro pelo ID.
- `consultarLivro(int id):` Consulta um livro pelo ID.
- `verificarDisponibilidade(int id):` Verifica a disponibilidade de um livro.
- `exibirRelatorioLivros():` Exibe um relatório com todos os livros.

## 3. EmprestimoLivros

- **Responsabilidades:** Gerencia o registro de empréstimos, aplicação de multas e controle de prazos.

- **Métodos Principais:**

- `registrarEmprestimo(Emprestimo emprestimo):` Registra um novo empréstimo.
- `aplicarMulta(int id, double multa):` Aplica uma multa a um empréstimo.
- `controlarPrazoDevolucao(int id, string novaData):` Atualiza a data de devolução de um empréstimo.
- `exibirRelatorioEmprestimos():` Exibe um relatório com todos os empréstimos.

## 4. Funções Auxiliares

- `exibirMenu():` Exibe o menu principal do sistema.
- `isStringValid(const string& str):` Verifica a validade de uma string.

- o `readInt()`: Lê um número inteiro da entrada do usuário.
- o `readDouble()`: Lê um número decimal da entrada do usuário.
- o `readString()`: Lê uma linha de texto da entrada do usuário.

## 4. Funcionamento do Sistema

### 4.1. Menu Principal

```
-----  
Sistema de Gerenciamento de Biblioteca  
-----  
1. Adicionar Usuario  
2. Remover Usuario  
3. Consultar Usuario  
4. Adicionar Livro  
5. Remover Livro  
6. Consultar Livro  
7. Registrar Emprestimo  
8. Verificar Disponibilidade de Livro  
9. Aplicar Multa  
10. Controlar Prazo de Devolucao  
11. Relatorio de Usuarios  
12. Relatorio de Livros  
13. Relatorio de Emprestimos  
0. Sair  
  
Escolha uma opcao: █
```

O menu principal oferece as seguintes opções para o usuário:

1. **Adicionar Usuário:** Permite a inserção de um novo usuário na biblioteca.
2. **Remover Usuário:** Permite a remoção de um usuário existente.
3. **Consultar Usuário:** Permite consultar as informações de um usuário específico.
4. **Adicionar Livro:** Permite a inserção de um novo livro na biblioteca.
5. **Remover Livro:** Permite a remoção de um livro existente.
6. **Consultar Livro:** Permite consultar as informações de um livro específico.
7. **Registrar Empréstimo:** Permite registrar um novo empréstimo de livro.
8. **Verificar Disponibilidade de Livro:** Permite verificar se um livro está disponível para empréstimo.

9. **Aplicar Multa:** Permite aplicar uma multa a um empréstimo existente.
10. **Controlar Prazo de Devolução:** Permite atualizar a data de devolução de um empréstimo.
11. **Relatório de Usuários:** Exibe um relatório com todos os usuários cadastrados.
12. **Relatório de Livros:** Exibe um relatório com todos os livros cadastrados.
13. **Relatório de Empréstimos:** Exibe um relatório com todos os empréstimos registrados.
14. **Sair:** Encerra o sistema.

## 4.2. Operações

- **Entrada de Dados:** O sistema usa funções de leitura e validação para garantir que os dados inseridos sejam válidos. Mensagens de erro são exibidas se os dados não forem válidos.
- **Relatórios:** Relatórios são gerados para listar todos os usuários, livros e empréstimos registrados, facilitando a visualização e o controle das informações.

## 5. Considerações Finais

O sistema de gerenciamento de biblioteca oferece uma solução completa para a administração de uma biblioteca. Ele é projetado para ser fácil de usar, com uma interface de menu intuitiva e funcionalidades bem definidas. A arquitetura orientada a objetos permite uma organização clara e uma manutenção eficiente do código.

Vale ressaltar que ele é bem passível de melhorias, coisas como um “banco de dados” em .txt para guardar os usuários, livros e empréstimos seria um próximo passo para implementação, também faltou explorar um pouco de Exceções, para isso criamos algumas funções de restrições para bloquear determinadas ações do usuário e que funcionam bem. O sistema apensar de simples é bem funcional e cumpre com o prometido.

## 6. Requisitos do Sistema

Para executar o sistema, o ambiente deve atender aos seguintes requisitos:

- **Compilador:** Compatível com C++.
- **Bibliotecas:** Nenhuma biblioteca externa é necessária além das padrão de C++.

## 7. Testes

Os testes foram realizados para verificar as principais funcionalidades do sistema, incluindo:

- Adição e remoção de usuários e livros.
- Registro e gestão de empréstimos.
- Geração de relatórios.

Todos os testes foram bem-sucedidos, confirmando que o sistema funciona conforme esperado.