

UNIVERSIDADE FEDERAL DE JUIZ DE FORA
CAMPUS JUIZ DE FORA
CURSO DE ORIENTAÇÃO A OBJETOS (DCC025)

GABRIEL FRASSON COSTA
ISADORA GONÇALVES FERREIRA
NÉLIO ALVES GOUVÊA

DOCUMENTAÇÃO PARCIAL - PARTE 3

UNIVERSIDADE FEDERAL DE JUIZ DE FORA
2021.3

1. INTEGRANTES DO GRUPO

Gabriel Frasson Costa - 202035001;

Isadora Gonçalves Ferreira - 202035017;

Nélio Alves Gouvêa - 201935037.

2. DOMÍNIO DO PROJETO

Sistema de Administração de uma Biblioteca desenvolvido em Java. Possuirá as funcionalidades descritas no tópico 4 deste documento.

3. ESCOPO DO PROJETO

3.1. Classe Livro: Título, Autor, Ano de Publicação, Editora, Gênero e Disponibilidade de um Livro. Possui métodos que devolvem as informações do livro.

3.1.1. Enum Gênero: Contém os gêneros que um Livro pode possuir.

3.2. Classe Usuário: Classe que possui como atributos o nome, e-mail, telefone e senha dos usuários.

3.2.1. Classe de Usuário da Biblioteca: Usuário comum, que terá acesso apenas à tela com a lista de livros e seus próprios empréstimos. Pode pegar um livro emprestado.

3.2.2. Classe de Funcionário: Possui acesso às funcionalidades de um Usuário Comum e pode cadastrar e deletar livros.

3.2.3. Classe de Gerente: Possui acesso às funcionalidade de um Funcionário e pode também criar novos Funcionários e Gerentes.

3.5. Classe Empréstimo: Possui como atributos o Livro, Usuário, data de empréstimo, data de devolução e se o empréstimo acabou ou não.

3.6. Pacotes de terceiros utilizados:

3.6.1. *java.util*.*: Pacote que contém classes utilitárias diversas, recursos de data e hora (*Date*) e listas (*List* e *ArrayList*).

3.6.2. *javax.swing*.*: Elementos de Interface Gráfica.

3.6.3. *java.awt*.*: Elementos de Interface Gráfica e utilitários.

3.6.4. *java.io*.*: Elementos de Leitura e Escrita de Arquivos.

DOCUMENTAÇÃO PARCIAL - PARTE 3

3.6.5. *com.google.gson.:** Biblioteca para administrar, ler e escrever em arquivos *json*.

4. REQUISITOS DO SISTEMA

4.1. Sistema de Login e Registro: Login usando o e-mail de usuário e senha.

Registro solicita o tipo de usuário, nome e senha.

4.2. Usuários com direitos diferentes: O Funcionário pode cadastrar e excluir livros, além de ver empréstimos dentro do Sistema da Biblioteca. O Gerente pode cadastrar Funcionários e Gerentes, além de administrar os livros. O Usuário Comum pode apenas ver os livros disponíveis e pegá-los emprestados.

4.3. Listagem de Livros: Retorna a lista dos livros cadastrados na biblioteca.

4.4. Relatório de Usuários que emprestaram um livro: Retorna um relatório dos Usuários que emprestaram determinado livro, assim como as datas correspondentes.

4.5. Empréstimo e Devolução de Livros: Empréstimo de um livro para um Usuário, que define a data de empréstimo e a data estimada para devolução. Registro da devolução do livro para a Biblioteca.

4.6. Cadastro e exclusão de Livros: Um Funcionário ou Gerente pode adicionar e excluir livros da Biblioteca.

5. DETALHES DA IMPLEMENTAÇÃO

5.1. Pacotes e Classes: O sistema foi implementado seguindo o padrão *MVC*, e possui cinco pacotes: *Models*, *Views*, *Controllers*, *Data* e *Utils*. No primeiro pacote, estão os modelos de classe que fazem parte do sistema, como os diferentes Usuários (*User*, *LibraryUser*, *Employee*, *Manager*), Livro (*Book*) e Empréstimos (*Loan*), há também o enum *Genre*. Já em *Views*, estão as telas que são apresentadas para os usuários, assim como uma classe-mãe *Screen* da qual todas as outras telas herdam. Em *Controllers* há as classes que realmente controlam, criam e administram Objetos das classes das *Models*, que são chamadas pelas classes de *Views*, completando o modelo *MVC*, há também as classes que controlam *Login* e *WindowEvents*, que controla os eventos de abertura e fechamento da Janela. Além dos pacotes desse modelo, há o pacote *Data*, em que são armazenados os arquivos lidos e escritos, de extensão *.json*, *bookdata*, *employeedata*, *loandata*, *managerdata* e *userdata*. Há também o pacote *Utils*, em que

DOCUMENTAÇÃO PARCIAL - PARTE 3

está uma classe abstrata *Constants*, a qual possui todas as *strings* e valores fixos, a classe *JSON*, que manipula os arquivos *.json* e a classe *FileIO*, que controla a entrada e saída em arquivos usando *gson*.

5.2. Herança e Polimorfismo: A herança é utilizada para todas as telas, que herdam da classe *Screen*, a qual possui as bases de uma janela. Também é usada para os usuários, e *User* é a classe-mãe de *LibraryUser*, *Employee* e *Manager*. O polimorfismo é usado nessas classes para alterar o comportamento das classes filhas.

5.3. Classe Abstrata e Interface: É usada uma classe abstrata *Constants* que armazena *Strings* e valores constantes utilizados por todo o sistema. São implementadas *Interfaces* padrões em *WindowEvents* (*WindowListener*), *ActionListener*, e é criada uma interface *Drawer*, que determina o método de desenho de telas.

5.4. Coleções e Tratamento de Exceções: São utilizadas múltiplas listas da interface *List* e sua implementação *ArrayList* para armazenar informações pertinentes às *models*. A leitura e escrita de arquivos é feita com tratamentos de exceção, assim como a inserção de dados em listas e leitura de dados de formulários.

5.5. Interface Gráfica e Arquivos: A interface gráfica é feita utilizando as classes padrão do *Swing*, do próprio *Java*. A entrada e saída em arquivos *.json* é feita pela biblioteca do [Google, gson](#), usando uma classe *JSON* e arquivos *FileIO*.

6. INSTRUÇÕES DE USO

1. O usuário deve clonar o repositório usando o comando:

```
git clone https://github.com/isa56/Trab - 00 - 21.3 - Alexandria - Library
```

2. O usuário deve abrir um Terminal na pasta

```
Trab - 00 - 21.3 - Alexandria - Library\alexandria - library
```

3. O usuário deve digitar *mvn install* e depois

```
java - jar target/alexandria - library - 1.0 - SNAPSHOT - jar - with - dependencies.jar
```

4. O usuário deve utilizar a interface gráfica para testar o sistema de gerenciamento.

DOCUMENTAÇÃO PARCIAL - PARTE 3

7. REPOSITÓRIO E OUTROS LINKS

Documentação da Primeira Entrega -

<https://docs.google.com/document/d/12Ocq4AJZg6iqNCZt6IzPJvsMbWFAbf9499eRfbHI5hw/edit?usp=sharing>.

Repositório do GitHub - <https://github.com/isa56/Trab-OO-21.3-Alexandria-Library>.

Apresentação de Slides - <https://tinyurl.com/apresentacaoOO>.

Diagrama UML - <https://whimsical.com/library-QDHPYk5Heu1yxXjxdszpeS>.

Vídeo de apresentação - <https://youtu.be/xlqUIzIDYfM>.

Mockup da Interface do Usuário -

<https://www.figma.com/file/vMCnSn9rKDv7W9a2k9zrpG/Alexandria?node-id=0%3A1>.

8. REFERÊNCIAS

Aulas de Orientação a Objetos, Ghiotto, Gleiph. Acesso em: 10/02/2022. Disponível em: <https://classroom.google.com/u/1/c/MzgyNDM5NjU0NTI2>.

UML cheat sheet, Small, C. Acesso em: 29/11/2021. Disponível em: <https://yetanotherchris.dev/cheatsheet/uml-cheat-sheet/>.

Java™ Platform, Standard Edition 7 API Specification, Oracle. Acesso em: 04/01/2021. Disponível em: <https://docs.oracle.com/javase/7/docs/api/overview-summary.html>.