

Especificação Projeto Final - POO

Disciplina: SSC0103-1 - Programação Orientada a Objetos

Professor: Dr. Márcio Delamaro

Alunos PAE: Claudinei Jr e Misael Jr

Grupo 2:

André Luís Mendes Fakhoury - Nº 4482145

Felipe Guilermmo Santuche Moleiro - Nº 10724010

Felipe Moreira Neves de Souza - Nº 10734651

Marcus Vinícius Medeiros Pará - Nº 11031663

1 Aplicação

Área: Ambiente escolar.

Projeto: plataforma para empréstimo e compartilhamento de livros didáticos.

2 Descrição e proposta

O sistema permite com que alunos possam pegar livros emprestados da biblioteca, além de terem a opção de compartilhar seus próprios livros entre si - por meio de emprestar à biblioteca.

Foi desenvolvido para um ambiente escolar, em que existe uma biblioteca (onde ficam armazenados tanto os livros da escola como os dos próprios alunos), que deve gerenciar o compartilhamento e empréstimo de livro entre os estudantes.

3 Objetivos

Este projeto possui como principal objetivo a implementação orientada a objetos, para fixar e aprimorar os conhecimentos obtidos na disciplina SSC0103. Foram desenvolvidas, para isso, diversas classes e interfaces gráficas em *java*. Além da orientação a objetos, foi utilizado no projeto um *banco de dados* (*postgreSQL*), além do versionamento de arquivos com o *git*.

Outro objetivo deste sistema encontra-se no âmbito escolar - o projeto de *compartilhamento de livros entre estudantes* foi desenvolvido de forma simplificada, e pode ser utilizado por escolas.

4 Funcionalidades

As funcionalidades do projeto estão separadas em dois: funcionalidades específicas do bibliotecário (usuário administrador) e funcionalidades comuns aos usuários (no caso, alunos). De modo geral, a *biblioteca* é o ponto de encontro de livros e usuários - ou seja, para emprestar um livro para outra pessoa, deve-se entregá-lo previamente à biblioteca, e a outra pessoa se dirige à biblioteca para pegá-lo.

4.1 Alunos

O aluno pode fazer o *login* em sua conta. Ao fazer isso, é aberta a *página do usuário*, que contém informações pessoais, livros emprestados da biblioteca e livros pessoais emprestados à biblioteca. Nesta página, também possui um botão para a opção de *reservar um livro* - é aberta a janela de busca por um livro e, se o livro requisitado estiver disponível, é feita a solicitação de empréstimo, que deve ser concluída na biblioteca.

Demais funções, como criação de conta, empréstimo de livro pessoal para a biblioteca e cancelamento de empréstimo são exclusivos ao administrador.

4.2 Administrador

O administrador possui todas as funções de um usuário comum (se quiser, pode inclusive reservar livros) - isso foi feito pois o administrador pode não ser exclusivamente o bibliotecário, mas também um representante ou organizador da biblioteca.

Na *página do usuário* o administrador conta com um botão exclusivo, que abre a *página do administrador*. Nela, é possível incluir/editar usuários, livros, empréstimos de livros pessoais à biblioteca, e empréstimos de livros da biblioteca a usuários comuns.

5 Utilização e telas do sistema

Assim que o programa é inicializado, aparece a tela de *login*.

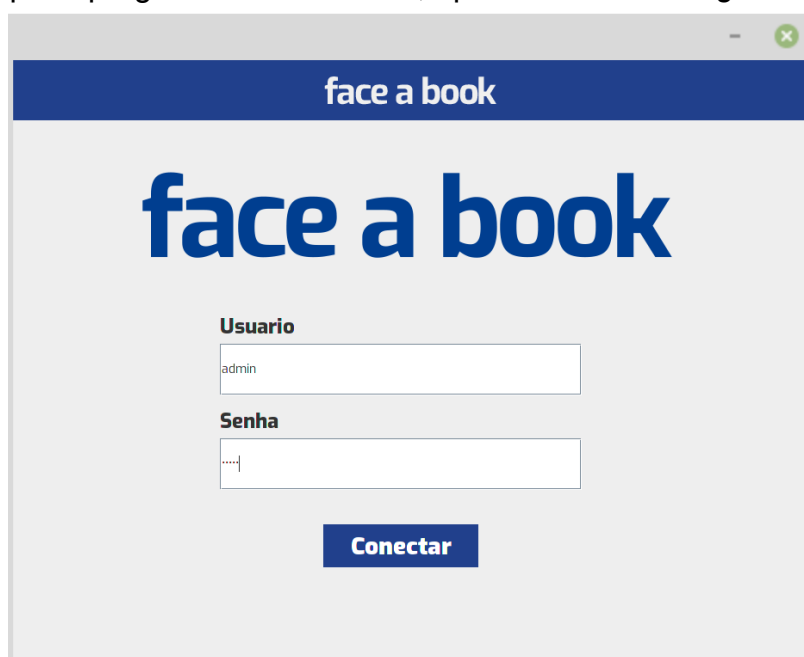
A imagem mostra uma janela de aplicativo com o título 'face a book' em uma barra azul superior. No centro, o nome 'face a book' aparece em uma fonte azul grande e estilizada. Abaixo, há dois campos de entrada: 'Usuario' com o texto 'admin' e 'Senha' com caracteres ocultos por pontos. Um botão azul com o texto 'Conectar' está posicionado abaixo dos campos. A janela possui uma barra de título padrão com botões de minimizar, maximizar e fechar.

Figura 1 - Tela de *login*

A tela de *login* é comum aos dois tipos de usuários (administrador ou comum). A princípio o programa vem com um usuário administrador padrão (username: *admin*, senha: *admin*), que poderá ser atualizado no futuro.

Feito o login, aparecerá a *página do usuário*, que possui algumas distinções entre as versões de administrador e usuário comum.

5.1 Utilização pelo usuário comum

Após o login, essa tela será apresentada:



Figura 2 - Página do usuário comum

Essa tela possui as informações básicas do usuário (como foto de perfil, nome e *username*). Além disso, também mostra todos os livros que o usuário pegou emprestado da biblioteca, e todos os livros que ele emprestou à biblioteca. Os livros emprestados à biblioteca também mostram um inteiro - *status* - que representa se o livro está com alguém no momento ou não. A opção de *Editar perfil* mostra outra janela, em que é possível alterar a senha ou foto de perfil.

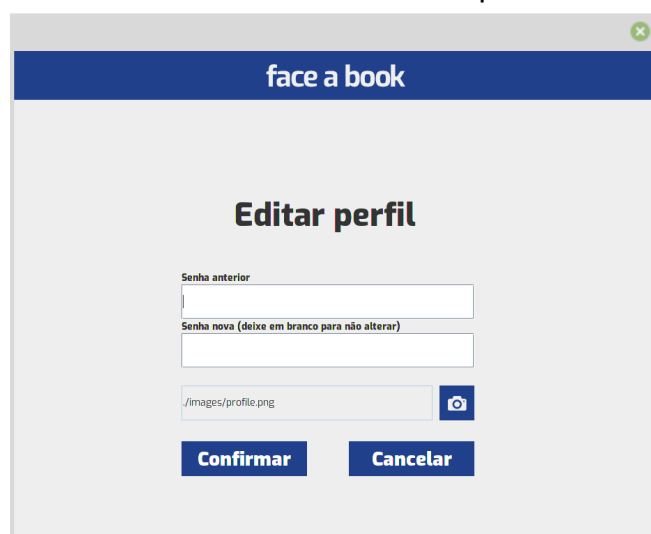


Figura 3 - Editar perfil

Também é possível fazer a consulta por livros, e a busca é realizada por nome, categoria ou autor.



Figura 4 - Consultar livros

Nessa busca, são retornados todos os livros já cadastrados no banco de dados. Após clicar em *reservar*, o usuário saberá se conseguiu reservar o livro, ou se ele está indisponível no momento. Se estiver disponível, o usuário deve ir até a biblioteca pegar o livro, e o bibliotecário deverá informar que o livro já foi pego.

5.2 Utilização pela biblioteca

A *página de usuário* da biblioteca conta com uma opção a mais, que leva à página de administrador.



Figura 5 - página de usuário de um administrador

Ao clicar no botão exclusivo, abre-se a página de administrador.




Figura 6 - Página do administrador

Nesta página, o administrador tem poderes de inserir/editar usuários, livros, categorias de livros, empréstimos à biblioteca e empréstimos da biblioteca. Em um primeiro momento, a biblioteca que fosse aderir ao uso desse sistema deveria cadastrar os livros no aplicativo.

A imagem mostra a interface de edição de livro no sistema "face a book". O título "Editar livro" está centralizado no topo. Abaixo dele, há um formulário com os seguintes campos: um menu suspenso com o texto "(1) Harry Potter - JK Rowling - (1) Harry Potter" e um botão "Editar" azul; um campo de texto com o número "1"; dois campos de texto para o nome do livro ("Harry Potter") e o autor ("JK Rowling"); um campo de texto para a imagem com o caminho "/images/unknown.jpeg" e um ícone de câmera; e um menu suspenso com o texto "1. Aventura" e um botão "Nova categoria" azul. Na base do formulário, há três botões azuis grandes: "Confirmar", "Limpar" e "Cancelar".

Figura 7 - Edição de livro

Na parte de edição de livros, pode-se tanto editar um existente como criar um novo. Vale notar que um dos campos a serem inseridos no livro é a categoria - nisso, vemos mais uma vez a abstração da orientação a objetos em ação.



The screenshot shows a web application window titled "face a book". Inside, the main heading is "Editar categoria". Below the heading, there is a dropdown menu labeled "-- Adicionar --" with a blue arrow icon, followed by a blue "Editar" button. Below these are two text input fields: "Nome da categoria" and "ID da Categoria". At the bottom of the form, there are three stacked blue buttons: "Confirmar", "Limpar", and "Cancelar".

Figura 8 - Edição de categoria

Além de cadastrar os livros (e categorias), é necessário cadastrar os usuários. A única forma de adicionar um usuário é na página de administrador, o que significa que cadastros só podem ser feitos pela biblioteca. Isso implica que usuários deverão se registrar na biblioteca, limitando a criação de cadastros à pessoas que frequentam a biblioteca e também a uma conta por pessoa (esse controle é feito pelo *username*).



The screenshot shows a web application window titled "face a book". Inside, the main heading is "Editar usuario". Below the heading, there is a dropdown menu labeled "-- Adicionar --" with a blue arrow icon, followed by a blue "Editar" button. Below these are five text input fields: "ID do usuario", "Nome do usuario", "Username do usuario", "Senha do usuario", and "Escolha nova foto". To the right of the "Escolha nova foto" field is a blue camera icon. Below the input fields is a checkbox labeled "Admin". At the bottom of the form, there are three stacked blue buttons: "Confirmar", "Limpar", and "Cancelar".

Figura 9 - Edição de usuários

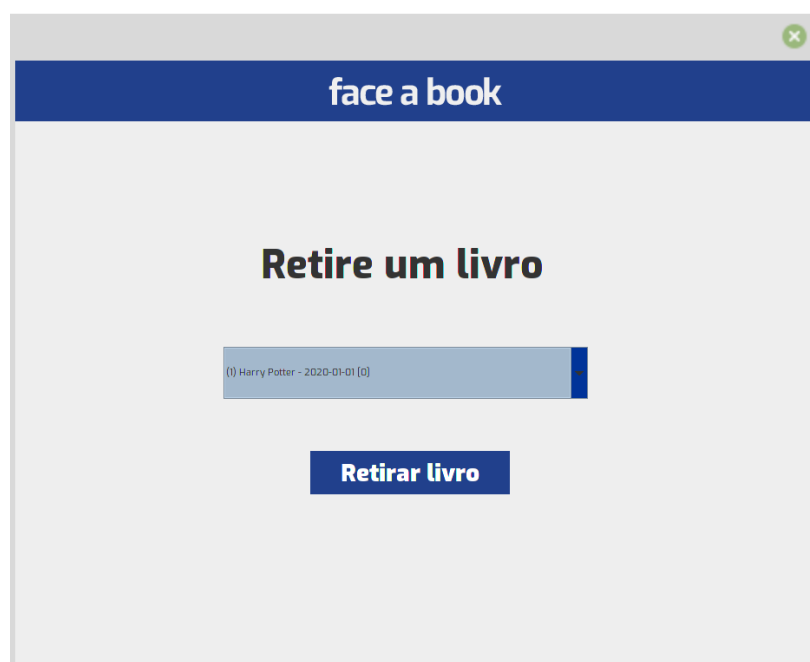
Além do controle dos livros da biblioteca, o administrador também é responsável por controlar os empréstimos de livros entre alunos. Quando um aluno tem algum livro que ele deseja emprestar, ele irá até a biblioteca com este livro, onde o bibliotecário irá cadastrá-lo no sistema com suas informações (além de manter informações de qual usuário emprestou o livro, e em que data ele pretende pegar o livro de volta). A partir desse momento o livro do usuário será visto como um livro da biblioteca, até a data estipulada no cadastro. O usuário pode, a qualquer momento, cancelar esse empréstimo do livro - desde que ele não esteja com ninguém no momento.



The screenshot shows a web application window titled "face a book". The main heading is "Editar emprestimo". Below the heading, there is a form with the following elements:

- A dropdown menu labeled "-- Adicionar --" with a blue "Editar" button next to it.
- A text input field labeled "ID do emprestimo".
- A text input field labeled "Data do emprestimo".
- A dropdown menu showing "(1) Harry Potter - JK Rowling - (1) Harry Potter" with a blue "Novo livro" button next to it.
- A dropdown menu showing "(1) Administrador - admin" with a blue "Novo usuario" button next to it.
- At the bottom, there are three blue buttons: "Confirmar", "Limpar", and "Cancelar".

Figura 10 - Novo empréstimo de livro para a biblioteca



The screenshot shows a web application window titled "face a book". The main heading is "Retire um livro". Below the heading, there is a form with the following elements:

- A dropdown menu showing "(1) Harry Potter - 2020-01-01 (0)" with a blue button next to it.
- Below the dropdown, there is a blue button labeled "Retirar livro".

Figura 11 - Retirar livro emprestado à biblioteca

6 Implementação

A implementação foi feita em Java utilizando programação orientada a objetos e banco de dados para armazenar os dados necessários.

Das telas já mostradas, podem ser destacados alguns detalhes de implementação: a senha dos usuários é salva em *MD5*, não podem ser criados dois usuários com o mesmo *username*, a *foto* salva no banco de dados é, na verdade, o caminho do arquivo da foto (absoluto, salvo por padrão na pasta *images*, e o nome do arquivo inclui um valor aleatório no início, para tentar evitar - de forma ingênua - conflitos de mesmo nome).

A maior parte do desenvolvimento foi feito no *IntelliJ Idea*, porém algumas telas da interface gráfica foram desenvolvidas no *NetBeans*.

6.1 Classes utilizadas

Utilizando os conceitos de POO, várias classes foram criadas para representar cada objeto. O conceito de classes e abstração foi muito útil, levando em consideração que várias partes do projeto utilizam as mesmas classes e trechos de código. Nesta parte da documentação iremos passar pela ideia geral de cada uma delas explicando seu funcionamento.

Livro

A classe livro é bem simples, ela contém os dados de um livro com nome, autor, foto da capa e categoria (Outra classe que será abordada posteriormente), além de um id para representar esse livro no sistema, que será referenciado em busca e para organizar cada livro com uma chave única.

Categoria

A classe categoria contém as informações de uma categoria de livros, que é basicamente o nome dessa categoria, e uma chave única inteira para ser associada a essa categoria que será usada para referenciar e buscar categorias.

Usuário

Uma classe que representa um usuário. Ela contém os campos de usuário como nome, foto de perfil e nome de usuário. Além das informações sobre usuário, também é associada uma chave inteira para cada usuário, e uma flag que indica se o usuário tem poder de administrador ou não.

Empréstimo

Uma classe para representar um empréstimo feito. Esta classe contém um id para o empréstimo, e o status daquele empréstimo. Além disso ela salva qual objeto livro foi emprestado e até que data ele foi emprestado.

Disponibilidade

Uma classe que representa a disponibilidade de um livro. Esta classe contém um id único e um status da disponibilidade. Além disso ela também salva qual objeto livro foi disponibilizado, e até quando ele foi disponibilizado na biblioteca.

Conexão Banco

Uma classe para representar o banco de dados. Através dela que é possível se comunicar com o banco de dados. Ela guarda todas as informações para estabelecer uma conexão com banco de dados, e a partir das funções dela você pode buscar, atualizar as tabelas.

Interfaces Gráficas

Foram desenvolvidas várias interfaces gráficas no projeto (vistas na package *GUI*). Foram desenvolvidas utilizando-se *java swing*. Mensagens de erros e informativas são mostradas ao usuário por *JOptionPane*.

6.2 Banco de dados

Como já dito anteriormente, o banco de dados utilizado foi o *PostgreSQL*. Com ele, foram possíveis inserções, updates e buscas no banco de dados de forma rápida e prática.

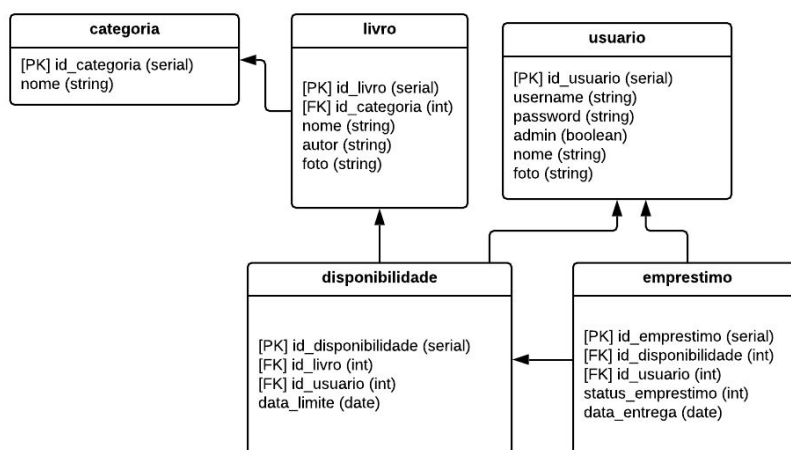


Figura 10 - Diagrama do banco de dados

6.3 Limitações

Este programa foi pensando com um sistema simples de usar e que contém todas as funcionalidades necessárias para uma biblioteca usual e com funções adicionais para usuários poderem emprestar livros entre si. O sistema, porém, não contém todas as funcionalidades possíveis, mas contém tudo que havíamos pensado e nos proposto a fazer no início do projeto.

Se tivéssemos que acrescentar funcionalidades, seriam utilidades como fila de espera para reservar um livro, mostrar e gerenciar melhor as imagens dos livros (que ainda não são utilizadas), mandar e-mail para usuário perto da sua data limite de devolução, entre outras - que não são fundamentais, mas facilitariam a vida do usuário.

7 Como executar

7.1 Pré-requisitos

Para executar o sistema sem erros, é necessário a instalação do banco de dados *PostgreSQL* (foi utilizada a versão 11). Caso o programa seja executado por alguma IDE ou coisa do gênero, é necessária a instalação do *JDBC de postgres*. Caso esta instalação seja feita no *IntelliJ Idea*, basta incluir o arquivo em: *Project Structure - Modules - Dependences - + - Jar or Directories*.

7.2 Configurar o banco de dados

Deve-se criar um *database* denominado *face_a_book*. Após criar o *database*, precisa-se configurar as tabelas e tudo mais do banco de dados.

O arquivo *etc/reset.sql* apresenta os códigos necessários para se configurar o banco de dados (além de poder ser utilizado para voltar às configurações iniciais). Ele também possui a inserção dos usuários padrão do sistema - *admin* e *user*.

Também é necessário ter, no diretório do projeto, um arquivo com nome “DataInfo.txt”, que deve conter as informações de *nome de usuário*, *senha* e *informações do banco de dados*. Caso o banco de dados chame *face_a_book*, basta seguir o exemplo:

```
user
password
jdbc:postgresql://localhost:5432/face_a_book
```

As imagens padrão também precisam ser incluídas no projeto. Para isso, deve-se ter uma pasta “*images*” no mesmo diretório do projeto com os logos e ícones padrão do sistema. Nesta pasta também serão salvas as outras imagens do sistema.

7.3 Execução

Feito isso, basta executar o arquivo da forma que preferir - pelo *jar* (no terminal, basta executar o comando *java -jar <arquivo jar>*) ou diretamente pelo código fonte. Por ser desenvolvido em *Java*, o programa é compatível com praticamente todos os sistemas operacionais mais utilizados. Foi testado previamente no *Windows 10* e *Linux Mint*.