



Universidade Federal de Viçosa – Campus UFV-Florestal  
Ciência da Computação – Programação Orientada à Objetos  
Professor: Philipe de Freitas Melo

Aluno: Arthur Teodoro Borges - Matrícula: 4672

Aluno: Henrique de Souza Campos - Matrícula: 4673

## Trabalho Prático 2

### Introdução:

Este trabalho foi desenvolvido como parte da disciplina de Programação Orientada a Objetos (CCF-313) do curso de Ciência da Computação da Universidade Federal de Viçosa - Campus Florestal. O objetivo principal é projetar e implementar uma versão simplificada de um Cassino Online, onde haveriam tipos de jogos diferentes e com dinâmicas variadas, simulando os atuais modelos de sites de divertimento. O sistema permite que os usuários se cadastrem, façam login, e joguem diferentes tipos de jogos de cassino, como Slots, Bac Bo e Mines, enquanto gerenciam seu saldo virtual.

O projeto foi estruturado de forma modular, com classes bem definidas para cada funcionalidade, como autenticação de usuários, gerenciamento de saldo, e a lógica dos jogos. A aplicação foi projetada para ser intuitiva e fácil de usar, com uma interface gráfica simples que permite aos usuários navegar entre as diferentes telas e realizar ações como depositar dinheiro, fazer apostas e jogar.

A screenshot of a web browser window titled "Cassino Online". The page has a light gray background. In the center, there are two columns of input fields. The left column has a "Nome:" label above a text input field, a "Senha:" label above another text input field, and a "Login" button below. The right column has a "Nome:" label above a text input field, a "Senha:" label above another text input field, and a "Cadastrar" button below. Between the two columns, the word "OU" is centered. The browser window has standard minimize, maximize, and close buttons in the top right corner.

Figura 1 - Página de Login

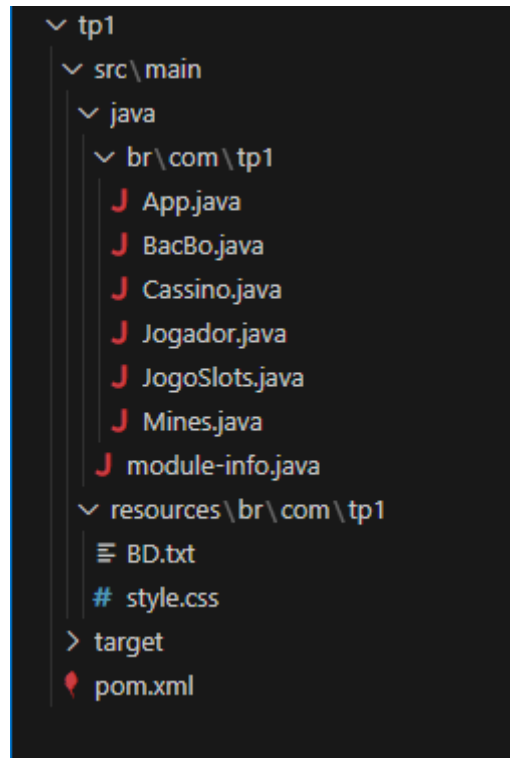
## Classes implementadas

O projeto Cassino Online Simplificado é composto por várias classes que trabalham em conjunto para fornecer uma experiência de jogo completa. A classe principal, `App.java`, é responsável por iniciar a aplicação e gerenciar a navegação entre as diferentes telas, como a tela de login, a tela principal do cassino e as telas dos jogos. Ela utiliza JavaFX para criar a interface gráfica e controla a transição entre as cenas, garantindo que o usuário possa acessar todas as funcionalidades do sistema de forma fluida.

A classe `Jogador.java` é responsável pelo gerenciamento dos dados dos usuários, incluindo cadastro, login e atualização do saldo. Ela interage com um arquivo de texto (`BD.txt`) para armazenar e recuperar informações como nome, senha e saldo dos jogadores. Além disso, essa classe fornece métodos para verificar a existência de usuários, realizar o login e atualizar o saldo no arquivo de texto, garantindo que os dados sejam persistentes entre sessões. A classe `Cassino.java` gerencia a tela principal do cassino, onde o usuário pode visualizar seu saldo, fazer depósitos e escolher entre os jogos disponíveis. Ela também interage com a classe `App` para navegar entre as diferentes telas de jogos, como `JogoSlots`, `BacBo` e `Mines`. A classe `Cassino` é responsável por atualizar o saldo do jogador conforme ele faz apostas e ganha ou perde nos jogos.

A classe `JogoSlots.java` implementa a lógica do jogo de Slots, onde o usuário aposta e gira três slots. Se os três símbolos forem iguais, o usuário ganha um prêmio. A interface gráfica é simples e intuitiva, com botões para confirmar a aposta, girar os slots e voltar ao cassino. O saldo do jogador é atualizado dinamicamente conforme ele joga. A classe `BacBo.java` implementa o jogo de Bac Bo, onde o usuário aposta em qual dos dois lados (esquerda ou direita) terá o maior valor ao lançar dois dados. A interface gráfica exibe os resultados dos dados e permite que o usuário escolha um lado para apostar. O saldo do jogador é atualizado conforme ele ganha ou perde a aposta. Por fim, a classe `Mines.java` implementa o jogo de Mines, onde o usuário deve evitar minas escondidas em uma grade 3x3. O jogador clica nos botões da grade para revelar se há uma bomba ou um local seguro. Se o jogador clicar em uma bomba, ele perde a aposta; caso contrário, ele acumula um prêmio. A interface gráfica permite que o usuário jogue novamente, pare e retire o prêmio acumulado, ou volte ao cassino.

Todas as classes são interconectadas, com a classe `App` atuando como o ponto central de controle, enquanto as outras classes gerenciam suas respectivas funcionalidades. O uso de JavaFX permite uma interface gráfica amigável e responsiva, enquanto a persistência dos dados em arquivos de texto garante que as informações dos usuários sejam mantidas entre sessões. O projeto é modular e pode ser facilmente expandido com a adição de novos jogos ou funcionalidades.



**Figura 2 - Diretório**

## Interface

As interfaces gráficas do projeto foram desenvolvidas utilizando a biblioteca JavaFX, proporcionando uma experiência de usuário moderna e interativa. A tela de login e cadastro, implementada na classe Jogador.java, é a primeira interface que o usuário encontra, permitindo que ele faça login com suas credenciais ou crie uma nova conta. A tela é dividida em duas seções principais: uma para login, com campos para nome e senha, e outra para cadastro, onde o usuário pode se registrar fornecendo um nome de usuário e uma senha. A interface é simples e intuitiva, com mensagens de feedback que informam se o login ou cadastro foi bem-sucedido ou se houve algum erro.



A tela principal do cassino, implementada na classe `Cassino.java`, é onde o usuário pode visualizar seu saldo atual, fazer depósitos e escolher entre os jogos disponíveis. A interface exibe o nome do jogador e seu saldo, além de botões para acessar os jogos de Slots, Bac Bo e Mines. O layout é organizado de forma clara, com um botão de depósito que abre um diálogo para o usuário adicionar fundos ao seu saldo. A navegação entre os jogos é feita de maneira fluida, com botões que redirecionam o usuário para as respectivas telas de cada jogo.

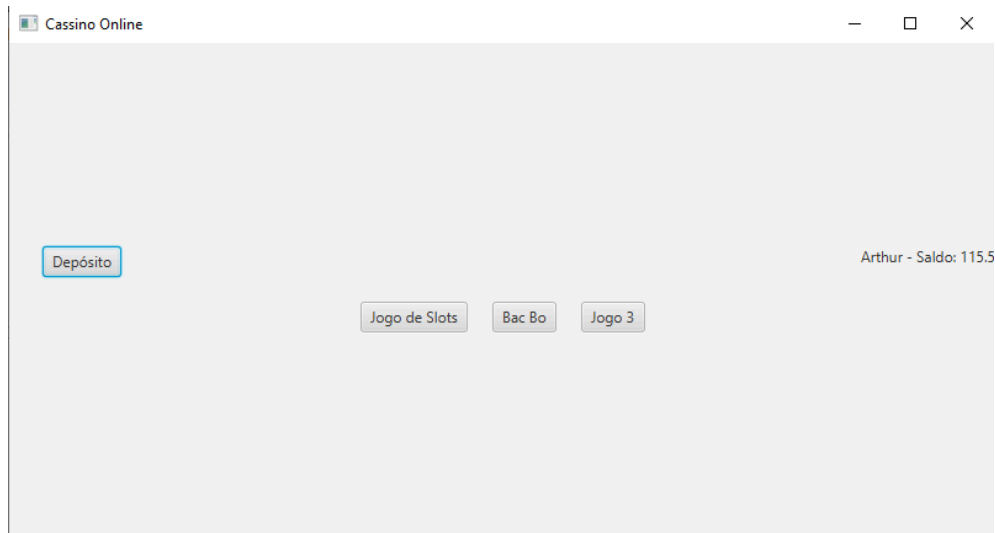


Figura 4- Menu “cru”

A tela do Jogo de Slots, implementada na classe `JogoSlots.java`, permite que o usuário faça apostas e gire três slots para tentar ganhar prêmios. A interface inclui um campo para inserir o valor da aposta, botões para confirmar a aposta e girar os slots, e labels que exibem os símbolos gerados aleatoriamente. O layout é organizado de forma a destacar os slots, com feedback imediato sobre o resultado do jogo e o saldo atualizado do usuário.



Figura 5 - Jogo de Slot

A tela do Bac Bo, implementada na classe BacBo.java, é onde o usuário aposta em qual dos dois lados (esquerda ou direita) terá o maior valor ao lançar dois dados. A interface inclui um campo para a aposta, botões para confirmar a aposta e escolher o lado, e labels que exibem os resultados dos dados. O layout é simples e direto, com feedback imediato sobre o resultado do jogo e o saldo atualizado.



Figura 6 - BacBo

Por fim, a tela do Mines, implementada na classe Mines.java, desafia o usuário a evitar minas escondidas em uma grade 3x3. A interface inclui um campo para a aposta, uma grade de botões que o usuário clica para revelar se há uma bomba ou um local seguro, e botões para jogar novamente, parar e retirar o prêmio acumulado, ou voltar ao cassino. O layout é interativo, com feedback visual imediato sobre cada clique e o saldo atualizado do usuário.

Todas as interfaces foram projetadas para serem intuitivas e de fácil uso, com layouts organizados e elementos visuais claros. O uso de JavaFX permitiu a criação de interfaces modernas e responsivas, garantindo uma experiência de usuário agradável e envolvente. A navegação entre as telas é fluida, e o design modular facilita a expansão do sistema com novos jogos ou funcionalidades no futuro.



Figura 7 - Mines

## Como executar:

Para executar o projeto, será necessário possuir alguma versão do Java(jdk) no seu computador; Após isso, basta seguir os seguintes passos de acordo com o seu sistema operacional:

Para compilar use: `javac br/com/tp1/*.java`

Para executar use: `java --module-path <caminho-para-o-javafx-sdk>/lib  
--add-modules javafx.controls,javafx.fxml br.com.tp1.App`

Ou utilize o `.jar` para executar o programa.

## Conclusão:

Por fim, com esse trabalho foi possível concluir que os conceitos estudados em sala de aula são muito importantes para desenvolver um algoritmo bem estruturado. Os conceitos de Polimorfismo, herança e etc, foram essenciais para a construção deste projeto.