

CLARISSA BONAFÉ GASPAR RUAS
GUILHERME HENRIQUE JARDIM
GUSTAVO SILVA MALVESTITI
LUCAS HENRIQUE SACHETTO

RELATÓRIO DE LINGUAGENS COMERCIAIS DE PROGRAMAÇÃO: Plataforma de Jogos Eletrônicos

1 INTRODUÇÃO

Neste trabalho, buscamos produzir uma plataforma de jogos eletrônicos, contendo os seguintes jogos:

- *Jogo da Memória*: um jogo consistindo de um quadriculado de 3 por 2, com um total de 2 pares de figuras. Elas são distribuídas aleatoriamente no quadriculado, mostradas por 5 segundos e então escondidas, para o jogador identificar os pares.
- *Jogo da Força*: uma palavra aleatória é escolhida, e o jogador tem de inserir letras para adivinhar a palavra. Ele tem um número máximo de 7 tentativas, e pode também tentar adivinhar a palavra toda de uma só vez.
- *Jogo da Vaca Invisível*: uma vaca invisível é colocada em um local aleatório da janela, e, a cada vez que o jogador passa o cursor do *mouse* em cima da vaca, ela muge. O objetivo do jogo é o usuário encontrar e clicar na vaca invisível.
- *Jogo Maior-Menor*: o jogador recebe uma carta, e tem que adivinhar se a carta de um *dealer* (jogado pelo computador) é maior ou menor que a dele.

Para acessar a plataforma, é necessário criar uma conta e fazer *login* nela. Após feito o *login*, um menu é apresentado com todos os jogos disponíveis.

2 METODOLOGIA

2.1 Requisitos do sistema

É necessário ter realizado a instalação da máquina virtual java para a execução do projeto.

2.2 Criação das Interfaces com Swing

A interface gráfica foi criada com componentes gráficos swing e o layout definido através do builder de interface gráfica de usuário (GUI) da IDE Apache NetBeans.

A maioria das classes utilizou o conceito de **subclasse** e **superclasse** ao estender a classe `javax.swing.JFrame`:

```
public class Exemplo extends javax.swing.JFrame { ... }
```

2.3 Classes e métodos

2.3.1 Conta

Classe que representa uma única conta registrada no sistema.

Sintaxe: public class **Conta**

Atributos:

- String **nome**
- String **senha**

Métodos:

- Getters e Setters

2.3.2 Contas

Classe para comunicação e manipulação do banco de dados em arquivo.

Sintaxe: public class **Contas**

Atributos:

- [static] [final] String **PATH** = “contas.txt”
- [static] ArrayList<Conta> arquivo

Métodos:

- [static] Scanner **createFile**
 - Parâmetros: Nenhum
 - Descrição: Criação do arquivo “contas.txt” em **PATH**.
- [static] void **inicializa**
 - Parâmetros: Nenhum
 - Descrição: Preenchimento do ArrayList **arquivo** com as contas registradas em “contas.txt”, ou criação do arquivo caso inexistente.
- [static] boolean **verifica**
 - Parâmetros:
 - String **nome**
 - String **senha**
 - Descrição: Verificação de existência de uma conta com **nome** e **senha** em **arquivo**, com retorno do resultado da busca;
- [static] boolean **adiciona**
 - Parâmetros:
 - String **nome**
 - String **senha**
 - Descrição: Adição de um novo registro de conta em “contas.txt” com verificação de duplicatas de **nome** e retorno do resultado da operação.
- Getters e Setters

2.3.2 Login

Classe para a realização do login no sistema.

Sintaxe: public class **Login** extends javax.swing.JFrame

Atributos:

- [static] String **nome**

Componentes:

- JLabel **lbTitulo**
- JLabel **lbNome**
- JLabel **lbSenha**
- JTextField **tfNome**
- JPasswordField **tfSenha**
- JButton **btnEntrar**
- JButton **btnCriarConta**

Métodos:

- Constructor
 - Parâmetros: Nenhum
 - Descrição: Inicialização dos componentes através do método **initComponents** e de **nome** como null.
- void **initComponents**
 - Parâmetros: Nenhum
 - Descrição: Inicialização e posicionamento dos componentes no frame principal.
- [static] void **main**
 - Parâmetros:
 - String[] **args**
 - Descrição: Criação e exibição do frame principal.
- void **btnEntrarMouseClicked**
 - Parâmetros:
 - MouseEvent **evt**
 - Descrição: Verificação se a conta dos valores de **tfNome** e **tfSenha** existe através do método **verifica** de **Contas**, e caso verdadeiro, preenchimento de **nome**.
- void **btnCriarContaMouseClicked**
 - Parâmetros:
 - MouseEvent **evt**
 - Descrição: Criação e exibição do frame **CriarConta**.
- Getters e Setters

2.3.3 CriarConta

Classe para o registro de contas para o login.

Sintaxe: public class **CriarConta** extends javax.swing.JFrame

Atributos:

- JFrame **login**

Componentes:

- JLabel **lbTitulo**
- JLabel **lbNome**
- JLabel **lbSenha**
- JTextField **tfNome**
- JPasswordField **tfSenha**
- JButton **btnCriar**
- JButton **btnVoltar**

Métodos:

- Constructor
 - Parâmetros:
 - JFrame **login**
 - Descrição: Inicialização dos componentes através do método **initComponents** e de **login** com o frame da classe **Login**.
- void **initComponents**
 - Parâmetros: Nenhum
 - Descrição: Inicialização e posicionamento dos componentes no frame principal.
- [static] void **main**
 - Parâmetros:
 - String[] **args**
 - Descrição: Criação e exibição do frame principal.
- void **btnCriarMouseClicked**
 - Parâmetros:
 - MouseEvent **evt**
 - Descrição: Registro da conta com os valores de **tfNome** e **tfSenha** através do método **adiciona** de **Contas**.
- void **btnVoltarMouseClicked**
 - Parâmetros:
 - MouseEvent **evt**
 - Descrição: Retroceder ao frame **login** da classe **Login**.
- Getters e Setters

2.3.4 Menu

Classe de menu principal para o acesso aos jogos.

Sintaxe: public class **Menu** extends javax.swing.JFrame

Atributos:

- JFrame **login**

Componentes:

- JLabel **lbTitulo**
- JLabel **lbNome**
- JLabel **lbUsuario**
- JButton **btnDesconectar**

- JButton **btnMemoria**
- JButton **btnVaca**
- JButton **btnForca**
- JButton **btnMaiorMenor**

Métodos:

- Constructor
 - Parâmetros:
 - JFrame **login**
 - Descrição: Inicialização dos componentes através do método **initComponents** e de **login** com o frame da classe **Login**.
- void **initComponents**
 - Parâmetros: Nenhum
 - Descrição: Inicialização e posicionamento dos componentes no frame principal.
- [static] void **main**
 - Parâmetros:
 - String[] **args**
 - Descrição: Criação e exibição do frame principal.
- void **btnMemoriaMouseClicked**
 - Parâmetros:
 - MouseEvent **evt**
 - Descrição: Criação e exibição do frame **Memoria**.
- void **btnVacaMouseClicked**
 - Parâmetros:
 - MouseEvent **evt**
 - Descrição: Criação e exibição do frame **Vaca**.
- void **btnForcaMouseClicked**
 - Parâmetros:
 - MouseEvent **evt**
 - Descrição: Criação e exibição do frame **Forca**.
- void **btnMaiorMenorMouseClicked**
 - Parâmetros:
 - MouseEvent **evt**
 - Descrição: Criação e exibição do frame **MaiorMenor**.
- void **btnDesconectarMouseClicked**
 - Parâmetros:
 - MouseEvent **evt**
 - Descrição: Retroceder ao frame **login** da classe **Login**.
- Getters e Setters

2.3.5 Forca

Classe do jogo da Forca.

Sintaxe: public class **Forca** extends javax.swing.JFrame

Atributos:

- JFrame **menu**
- String[] **palavras**
- String **palavra**
- char[] **atual**
- String **letras**
- int **tentativas**

Componentes:

- JLabel **lbTitulo**
- JLabel **lbPalavra**
- JLabel **lbAtual**
- JLabel **lbTituloLetras**
- JLabel **lbLetras**
- JLabel **lbTituloTentativas**
- JLabel **lbTentativas**
- JTextField **tfTentativa**
- JTextField **tfAdivinhar**
- JButton **btnTentativa**
- JButton **btnAdivinhar**
- JButton **btnReset**
- JButton **btnVoltar**

Métodos:

- Constructor
 - Parâmetros:
 - JFrame **menu**
 - Descrição: Inicialização dos componentes através do método **initComponents**, de **menu** com o frame da classe **Menu**, do pool de palavras **palavras** pelo método **iniciaPalavras**, e chamada do método de preparação do jogo **inicializa**.
- void **initComponents**
 - Parâmetros: Nenhum
 - Descrição: Inicialização e posicionamento dos componentes no frame principal.
- [static] void **main**
 - Parâmetros:
 - String[] **args**
 - Descrição: Criação e exibição do frame principal.
- void **inicializaPalavras**
 - Parâmetros: Nenhum
 - Descrição: Inicialização do array/pool de palavras **palavras**.
- int **getNumber**
 - Parâmetros: Nenhum
 - Descrição: Retorno de um número inteiro aleatória definido pelo método **Math.random()** e os limites de máximo e mínimo fixos.

- String **getPalavra**
 - Parâmetros: Nenhum
 - Descrição: Retorno de uma palavra do array **palavras** de índice resultante do método **getNumber**. Ocorre tratamento para não ocorrer a mesma palavra em duas partidas seguidas.
- void **inicializa**
 - Parâmetros: Nenhum
 - Descrição: Preparação do estado inicial do jogo com, os componentes sendo limpos, **tentativas** recebem 7, **palavra** pelo método **getPalavra**, **letras** igual a "" e **atual** para um char array de **palavra**.
- void **btnVoltarMouseClicked**
 - Parâmetros:
 - MouseEvent **evt**
 - Descrição: Retroceder ao frame **menu** da classe **Menu**.
- void **btnResetMouseClicked**
 - Parâmetros:
 - MouseEvent **evt**
 - Descrição: Chamada do método **inicializa** para a preparação de uma nova partida.
- void **testaLetra**
 - Parâmetros:
 - String **str**
 - Descrição: Testa se a letra **str** existe em **palavra**, e atualiza **atual**, **letras** e componentes relacionados.
- void **btnTentativaMouseClicked**
 - Parâmetros:
 - MouseEvent **evt**
 - Descrição: Realiza a tentativa de adivinhar uma letra em **tfTentativa**, com verificação do valor da letra pelo método **testaLetra** e update na quantidade de **tentativas**.
- void **btnAdivinharMouseClicked**
 - Parâmetros:
 - MouseEvent **evt**
 - Descrição: Realiza a tentativa de adivinhar a palavra em **tfAdivinhar** e com exibição da resposta no final.
- Getters e Setters

2.3.6 MaiorMenor

Classe do jogo MaiorMenor.

Sintaxe: public class **MaiorMenor** extends javax.swing.JFrame

Atributos:

- JFrame **menu**
- int **pontuacao**

- int **atual1**
- int **atual2**

Componentes:

- JLabel **lbTitulo**
- JLabel **lbTituloPontuacao**
- JLabel **lbPontuacao**
- JTextField **tfDealer**
- JTextField **tfPlayer**
- JButton **btnMaior**
- JButton **btnMenor**
- JButton **btnReset**
- JButton **btnVoltar**

Métodos:

- Constructor
 - Parâmetros:
 - JFrame **menu**
 - Descrição: Inicialização dos componentes através do método **initComponents**, de **menu** com o frame da classe **Menu** e chamada do método de preparação do jogo **inicializa**.
- void **initComponents**
 - Parâmetros: Nenhum
 - Descrição: Inicialização e posicionamento dos componentes no frame principal.
- [static] void **main**
 - Parâmetros:
 - String[] **args**
 - Descrição: Criação e exibição do frame principal.
- int **getNumber**
 - Parâmetros: Nenhum
 - Descrição: Retorno de um número inteiro aleatório definido pelo método **Math.random()** e os limites de máximo e mínimo fixos.
- void **changePontuacao**
 - Parâmetros: Nenhum
 - Descrição: Update do texto de **lbPontuacao** para **pontuacao**.
- void **inicializa**
 - Parâmetros: Nenhum
 - Descrição: Preparação do estado inicial do jogo com os componentes sendo limpos e os valores de **atual1** e **atual2** serem atualizados para a próxima partida.
- void **decisao**
 - Parâmetros:
 - int **op**
 - Descrição: Verificação se o valor representado por **op**, maior ou menor, é correto e depois realizar a mudança da pontuação e dos

componentes.

- void **next**
 - Parâmetros: Nenhum
 - Descrição: Prepara o próximo estado da partida, alterando os componentes **lbPlayer** e **lbDealer**.
- void **btnResetMouseClicked**
 - Parâmetros:
 - MouseEvent **evt**
 - Descrição: Chamada do método **inicializa** para a preparação de uma nova partida.
- void **btnVoltarMouseClicked**
 - Parâmetros:
 - MouseEvent **evt**
 - Descrição: Retroceder ao frame **menu** da classe **Menu**.
- void **btnMaiorMouseClicked**
 - Parâmetros:
 - MouseEvent **evt**
 - Descrição: Chamada do método **decisao** com **op 1**.
- void **btnMenorMouseClicked**
 - Parâmetros:
 - MouseEvent **evt**
 - Descrição: Chamada do método **decisao** com **op 0**.
- Getters e Setters

2.3.7 Vaca

Classe do jogo da Vaca.

Sintaxe: public class **Vaca** extends javax.swing.JFrame

Atributos: Nenhum

Componentes:

- JButton **jButton1**
- JButton **jButton2**
- JLabel **jLabel2**
- JLabel **jLabel3**
- JLabel **jLabel4**
- JPanel **jPanel1**
- JPanel **jPanel2**

Métodos:

- Constructor
 - Parâmetros:
 - JFrame **menu**
 - Descrição: Inicialização dos componentes através do método **initComponents**, de **menu** com o frame da classe **Menu** e chamada do método de preparação do jogo **inicializa**.

- void **initComponents**
 - Parâmetros: Nenhum
 - Descrição: Inicialização e posicionamento dos componentes no frame principal.
- [static] void **main**
 - Parâmetros:
 - String[] **args**
 - Descrição: Criação e exibição do frame principal.
- void **btnResetMouseClicked**
 - Parâmetros:
 - MouseEvent **evt**
 - Descrição: Chamada do método **inicializar** para a preparação de uma nova partida.
- void **inicializar**
 - Parâmetros: Nenhum
 - Descrição: Preparação do estado inicial do jogo com os componentes sendo limpos e os valores de posição serem atualizados para a próxima partida.
- void **mugido**
 - Parâmetros: Nenhum
 - Descrição: Reprodução do áudio "cow-moo3.wav".
- Getters e Setters

2.3.8 Memoria

Classe do jogo **MaiorMenor**.

Sintaxe: public class **MaiorMenor** extends javax.swing.JFrame

Atributos:

- JFrame **menu**
- ArrayList<Integer> **posicoesLiberadas**
- ArrayList<Integer> **game**
- ArrayList<Integer> **jogadaP**
- Integer **jogadas**
- Timer **timer**

Componentes:

- JLabel **jLabel1**
- JTextField **A0**
- JTextField **A1**
- JTextField **A2**
- JTextField **A3**
- JTextField **A4**
- JTextField **A5**
- JButton **JButton**

Métodos:

- Constructor
 - Parâmetros:
 - JFrame **menu**
 - Descrição: Inicialização dos componentes através do método **initComponents**, de **menu** com o frame da classe **Menu** e chamada do método de preparação do jogo **inicializa**.
- void **initComponents**
 - Parâmetros: Nenhum
 - Descrição: Inicialização e posicionamento dos componentes no frame principal.
- [static] void **main**
 - Parâmetros:
 - String[] **args**
 - Descrição: Criação e exibição do frame principal.
- int **generateGame**
 - Parâmetros: Integer size
 - Descrição: recebe o maior número presente no jogo, ex: 3 terão 6 botões e os valores dos botões vão de 0 a 2.
- void **jogada**
 - Parâmetros: Nenhum
 - Descrição: Caso já tiverem sido escolhidas duas casas, verificar se são as mesmas, incrementar 1 no contador de jogadas, mostrar as casas caso tenha acertado, virar as casas caso tenha errado e verifica se o jogo terminou e caso sim printa o total de jogadas.
- void **viraPressionados**
 - Parâmetros: Nenhum
 - Descrição: Vira todos botões no Array jogadaP, mostrando seus valores reais
- void **desviraPressionados**
 - Parâmetros: Nenhum
 - Descrição: Desvira todos botões no Array jogadaP, mostrando "X"
- Boolean **fimDeJogo**
 - Parâmetros: Nenhum
 - Descrição: Verifica se o jogo terminou vendo a quantidade de casas liberadas X o tamanho do jogo, retorna true ou falso.
- void **inicia**
 - Parâmetros: Nenhum
 - Descrição: Limpa o jogo, chama função para gerar os valores, mostra o tabuleiro e chama o timer para aguardar 5 segundos e desvirar.
- void **AXMouseClicked**
 - Descrição: Verifica se a posição já não foi virada, caso não a vira e chama a função jogada.

Classe **bloqueiaGame**

Descrição: Classe serve para desvirar as peças após um tempo pré determinado de 5 segundos, herda de TimerTask.