**Especificação de Requisitos Orthokon**

**Especificação de Requisitos de Software**

Versão 1.0

17/02/2021

|  |  |  |  |
| --- | --- | --- | --- |
| **Versão** | **Autores)** | **Data** | **Ação** |
| 1 | Gabriel Soares Flores  Pedro Henrique Azevedo  Maiko Ademir Nunes | 17/02/2021 | Estabelecimento de requisitos |

**Conteúdo:**

1. Introdução;

2. Visão geral;

3. Requisitos de software.

1. **Introdução**

**Objetivo**: desenvolvimento de um programa que suporte a disputa de partidas de Orthokon na modalidade usuários contra usuário e usuário contra programa.

**Referências**:

https://www.youtube.com/watch?v=zljaKu2e82g (regras do jogo).

<http://jnsilva.ludicum.org/Obidos/orthokonpdf.pdf> (regras do jogo)

1. **Visão Geral**

**Arquitetura do programa:** programa orientado a objetos, suporte a uso em redes (Distribuído)

**Premissas de desenvolvimento:**

* O programa deve apresentar uma interface gráfica bidimensional;
* O programa deve ser implementado em Java, devendo executar em qualquer plataforma que disponha da máquina virtual Java.
* O programa deve usar o recurso distribuído em redes, chamado NetGames, que é disponibilizado no moodle da disciplina de Engenharia de software.

**3 Requisitos de Software 3.1 Requisitos Funcionais:**

**Requisito funcional 1 – Implementar estratégia**: o programa deve permitir que sejam disputadas partidas entre um usuário e o programa, sendo que a estratégia de jogo implementada deve garantir que o programa jamais proceda a um lance inválido, de acordo com as regras do jogo (ver referência 1).

**Requisito funcional 2 – Iniciar**: o programa deve apresentar a opção de menu “***iniciar***” para o início de uma nova partida,

**Requisito funcional 3 – Estabelecimento de quem inicia**: na interface da aplicação deve ser possível escolher qual dos dois jogadores inicia a partida, quando executada operação iniciar ou reiniciar;

**Requisito funcional 5 – Procedimento de lance**: o programa deve suportar o procedimento de lance dos usuários por meio de *click* de botão esquerdo de *mouse* sobre a posição em que deve ser colocado umas peças, considerando as seguintes restrições:

* Num tabuleiro 4×4 colocam-se quatro peças negras e quatro brancas, como ilustrado abaixo. Cada jogada consiste em movimentar uma peça, cujo movimento é o de uma rainha fugitiva, isto é, o movimento é horizontal, vertical ou diagonal, mas o número de casas tem de ser o máximo, sem saltar outras peças nem abandonar o terreno de jogo. Quando a peça a movimentada atinge o seu destino, todas as peças adversárias que lhe sejam ortogonalmente vizinhas mudam de cor. Perde quem não dispuser de nenhum lance legal (quer por ter ficado sem peças, quer por só dispor de peças circundadas por peças adversárias). (ver referência 1);

**Requisito funcional 6 – Comunicação do estado da partida**: a cada tentativa de procedimento de lance do usuário (bem-sucedida ou não) e a cada procedimento de lance efetuado pelo programa o novo estado da partida deve ser exibido na interface, considerando as seguintes restrições:

* A matriz de posições deve ser atualizada com a presença de umas novas peças, no caso de procedimento de lance bem-sucedido;
* Em caso de lance irregular, deve ser comunicado o tipo de irregularidade: tentativa de colocar peças em posição ocupada, ou com a partida já encerrada.

**3.2Requisitos Não Funcionais**

**Requisito não funcional 1 – Especificação de projeto**: além do código Java, deve ser produzida especificação de projeto baseada em UML, segunda versão;

**Requisito não funcional 2 – Interface gráfica para usuário**: o programa deverá ter interface gráfica única, partilhada pelos usuários, no caso de partida entre usuários;

**Requisito não funcional 3 – Símbolos dos jogadores**: a presença de peças em posição deve ser representada por um “X” dentro da posição para um dos jogadores e “O” para o outro;

**Requisito não funcional 4 – Tecnologia de interface gráfica para usuário**: A interface gráfica deve ser baseada em *Java-Swing*.