

Projeto Neutron

Especificação de Requisitos de Software

Versão 1.5
01/03/2022

Versão	Autores	Autores	Autores	Ação
1.5	Artur Carmezini	Miguel Duarte	Otavio Augusto	Estabelecimen to dos requisitos

Conteúdo:

1. Introdução;
2. Visão geral;
3. Requisitos de Software;

1. Introdução

Objetivo: Planejar, modelar e desenvolver um jogo Neutron. A modelagem será seguindo os padrões da UML 2 e o desenvolvimento será na linguagem de programação Python. O jogo executa localmente, com dois jogadores humanos competindo.

Referências:

<<https://youtube.com/watch?v=6sWilpyOWgc>> (regras do jogo)

2. Visão Geral

Arquitetura do programa: programa orientado a objetos, stand-alone

Premissas de desenvolvimento:

- O programa deve apresentar uma interface gráfica bidimensional
- O programa deve representar um jogo para dois jogadores locais, desenvolvido em Python, utilizando a biblioteca Tkinter para interface e com um sistema minimamente complexo de turnos e regras de jogo para a vitória. Deve ser feito tendo planejado a modelagem UML utilizando o Visual Paradigm.

3. Requisitos de Software

3.1 Requisitos Funcionais

Requisito funcional 1 - Implementar estratégia: O programa deve garantir que jamais proceda um lance inválido, de acordo com as regras do jogo.

Requisito funcional 2 - Iniciar: O programa deve permitir que o usuário possa iniciar o jogo.

Requisito funcional 3 - Estabelecimento de quem inicia: O programa deve permitir que o usuário escolha com qual peça irá jogar.

Requisito funcional 4 - Procedimento de lance: o programa deve suportar o procedimento de lance dos usuários por meio do click de mouse sobre a peça e posteriormente sobre a posição em que deve ser colocado a peça, considerando as seguintes restrições: · O programa deve permitir apenas lances válidos, de acordo com as regras do jogo (ver referência);
· A cada lance válido, o programa deve verificar se o jogador que jogou ganhou a partida;

Requisito funcional 5 - Comunicação de estado da partida: A cada tentativa de procedimento de lance, o novo estado da partida deve ser exibido na interface, considerando o seguinte: · A matriz com as posições deverá ser atualizada conforme o movimento bem-sucedido; · Em caso de lance irregular, deve ser comunicado o tipo da irregularidade.

3.2 Requisitos Não-funcionais

Requisito não-funcional 1 - Símbolo dos jogadores: Um jogador deve ser representado pelo símbolo Próton, enquanto o outro pelo símbolo Elétron. Há o símbolo Neutron também, que deve ser manejado pelos dois jogadores antes de cada jogada com sua peça.

Requisito não-funcional 2 - Especificação de projeto: Além do código Python, deve ser produzida a modelagem de especificação do projeto baseada em UML, segunda versão e utilizando a plataforma GitHub para versionamento.

Requisito não-funcional 3 - Interface Gráfica: O programa deverá ter uma interface gráfica única, compartilhada pelos usuários.

Requisito não-funcional 4 - Tecnologia de interface gráfica: A interface gráfica deverá ser baseada em Python Tkinter.