

Especificação de requisitos

Robotory

Versão: 1.2

Data: 02/03/2022

Versão	Autor(es)	Data	Ação
1.0	Douglas Pereira Luiz, Vitor Katsuziro Egami	23/11/2021	Definição dos requisitos do projeto e apresentação da interface visual do jogador
1.1	Douglas Pereira Luiz, Vitor Katsuziro Egami	07/02/2022	Alteração na apresentação visual da aplicação (Figura 1)
1.2	Douglas Pereira Luiz, Vitor Katsuziro Egami	02/03/2022	Alteração na apresentação visual da aplicação, com adição de botão de finalização de movimento (Figura 1)

Índice

1.Introdução.....	1
1.1.Objetivo do Desenvolvimento.....	1
1.2.Referências.....	2
2.Visão Geral do Sistema.....	2
2.1.Arquitetura do Software.....	2
2.2.Premissa do Desenvolvimento.....	2
2.3.Definições.....	2
2.4.Regras do Jogo.....	2
3.Requisitos da Aplicação.....	3
3.1.Requisitos Funcionais.....	3
[RF1] Conectar.....	3
[RF2] Iniciar partida.....	4
[RF3] Preparar partida.....	4
[RF4] Proceder jogada.....	4
[RF5] Receber jogada.....	4
[RF6] Verificar jogada.....	4
[RF7] Verificar estado da partida.....	4
[RF8] Desconectar.....	4
3.2.Requisitos Não Funcionais.....	5
[RN1] Especificações do projeto e implementação.....	5
[RN2]Portabilidade.....	5
[RN3] Interface gráfica.....	5

1.Introdução

1.1.Objetivo do Desenvolvimento

Desenvolvimento de um programa Java que permita a realização do jogo Robotory entre dois jogadores em rede.

1.2.Referências

<https://www.ludopedia.com.br/jogo/robotory>

<http://www.inf.ufsc.br/~netgames/NetGamesNRT/>

2.Visão Geral do Sistema

2.1.Arquitetura do Software

- **Paradigma de Programação:** Programa Orientado a Objetos;
- **Tipo aplicação:** distribuída, suportando arquitetura cliente-servidor.

2.2.Premissa do Desenvolvimento

- Implementação em Java 8;
- Gerenciamento de projeto utilizando a ferramenta Apache Maven;
- Interface gráfica com Swing para usuário;
- Utilização do NetGamesNRT para partidas online;
- Modelagem do desenvolvimento de software com a linguagem UML no software Visual Paradigm.

2.3.Definições

- **Robô:** Peça circular/esférica. São três: vermelho, preto e branco;
- **Energia:** Peça quadrada/cúbica. São duas: preta e branca;
- **Suprimento pessoal:** conjunto que cada jogador possui de até quatro peças de energia;
- **Suprimento comum:** conjunto de peças de energia do tabuleiro;
- **Posição:** espaço do tabuleiro de forma hexagonal que pode ser ocupado por uma peça;
- **Consumir:** ato que constitui a substituição de uma peça de energia do tabuleiro por uma peça de robô. A peça de energia é descartada.

2.4.Regras do Jogo

- O número de peças de **energia** é limitado. São **quatorze de cada cor**;

- O número de **robôs** é igual à três, sendo **um de cada cor**;
- Peças de **robô preto** só podem **consumir** peças de **energia preta**;
- Peças de **robô branco** só podem **consumir** peças de **energia branca**;
- Peças de **robô vermelho** podem **consumir** ambas peças de **energia**;
- Uma peça de **robô** só pode consumir uma peça de **energia** cuja posição é adjacente à sua própria;
- O **consumo** de uma peça de **energia** leva ao seu descarte;
- Uma peça de **energia** só pode ser posicionada no tabuleiro se for retirada do **suprimento pessoal**;
- Uma peça de **energia** só pode ser posicionada em uma posição desocupada;
- Uma peça só pode ser posicionada no **suprimento pessoal** se retirada do **suprimento comum**;
- O único destino de uma peça do **suprimento comum** é uma posição de um **suprimento pessoal**;
- O primeiro à jogar é decidido de forma **aleatória**, sendo o **lado claro do primeiro a jogar** e o **lado escuro do outro jogador**;
- Cada jogador recebe duas peças de **energia de cada cor** em seu suprimento pessoal. O restante vai para o suprimento comum;
- A posição inicial dos robôs é a vista na [Figura 1](#);
- A cada turno de um jogador é realizada uma ação pelo mesmo. Depois o turno é passado para o outro jogador;
- Existem três tipos de ação:
 - Escolher um único **robô** e realizar uma sequência de **consumos**. 1 a N consumos;
 - Posicionar uma peça de **energia** no tabuleiro;
 - Retirar uma peça de **energia** do **suprimento comum** adicionando-a ao seu **suprimento pessoal**;
- A partida é finalizada no momento que um dos jogadores retirar a última peça de **energia** de uma das cores do **suprimento comum**;
- O vencedor é aquele que possui mais **robôs** em seu lado;

3.Requisitos da Aplicação

3.1.Requisitos Funcionais

[RF1] Conectar

O software deve apresentar em seu menu a opção “Conectar”, para assim estabelecer conexão com o servidor.

[RF2] Iniciar partida

O programa deve apresentar a opção “Iniciar Partida”, que começa uma nova partida somente se a conexão já estiver estabelecida.

[RF3] Preparar partida

O programa deve, após iniciar partida, posicionar os três robôs, distribuir duas peças de energia de cada cor para o suprimento pessoal de cada jogador e posicionar o restante das peças na reserva comum;

[RF4] Proceder jogada

O Software deve apresentar de quem é a vez do turno e permitir que somente esse jogador realize uma das possíveis ações de jogada. São elas:

- Mover um robô para uma nova posição, clicando sobre ele e clicando na nova posição desejada. A ação só é terminada se a posição do robô é diferente da posição escolhida e o movimento é válido (posição escolhida tenha uma sequência de posições adjacentes uma à outra, iniciada pela posição atual do robô, que contenham peças de energia que podem ser consumidas pelo mesmo robô). Caso contrário a ação é cancelada, permitindo a escolha de uma nova ação.
- Mover uma peça de energia do suprimento pessoal para uma nova posição no tabuleiro, clicando sobre ela e clicando na nova posição desejada. A ação só é terminada se a posição escolhida não está ocupada por outra peça, caso contrário a ação é cancelada, permitindo a escolha de uma nova ação.
- Mover uma peça de energia dos suprimentos comuns para o suprimento pessoal clicando sobre a peça de cor desejada no suprimento comum.

[RF5] Receber jogada

O Software deve identificar o término do turno do jogador adversário, atualizando, então, na interface gráfica do jogador atual, as peças no tabuleiro, as peças nos suprimentos pessoais, as peças nos suprimentos comuns e o status do jogo.

[RF6] Verificar jogada

O software deve verificar se as ações tomadas pelos jogadores são válidas. A jogada só é terminada caso seja validada.

[RF7] Verificar estado da partida

A aplicação deve, a cada de turno, verificar se a ultima peça de energia de alguma das cores foi retirada do suprimento comum. Se foi retirada, é verificado que lado possui mais robôs e o vencedor é anunciado imediatamente.

[RF8] Desconectar

O programa deve apresentar em seu menu a opção “Desconectar”, para assim finalizar a conexão com o servidor, podendo finalizar a partida.

3.2.Requisitos Não Funcionais

[RN1] Especificações do projeto e implementação

Projeto codificado em Java 8 e gerenciado pela ferramenta Apache Maven. Modelagem do desenvolvimento do software feita em linguagem baseada em UML 2.

[RN2]Portabilidade

O programa deve funcionar em qualquer plataforma que disponha da máquina virtual Java, versão 8 ou superior;

[RN3] Interface gráfica

Interface gráfica com Swing para apresentação do tabuleiro e peças, menu de configuração de partida, com itens para conectar, desconectar e iniciar partida e interação dos jogadores com o tabuleiro e peças do jogo através do mouse.

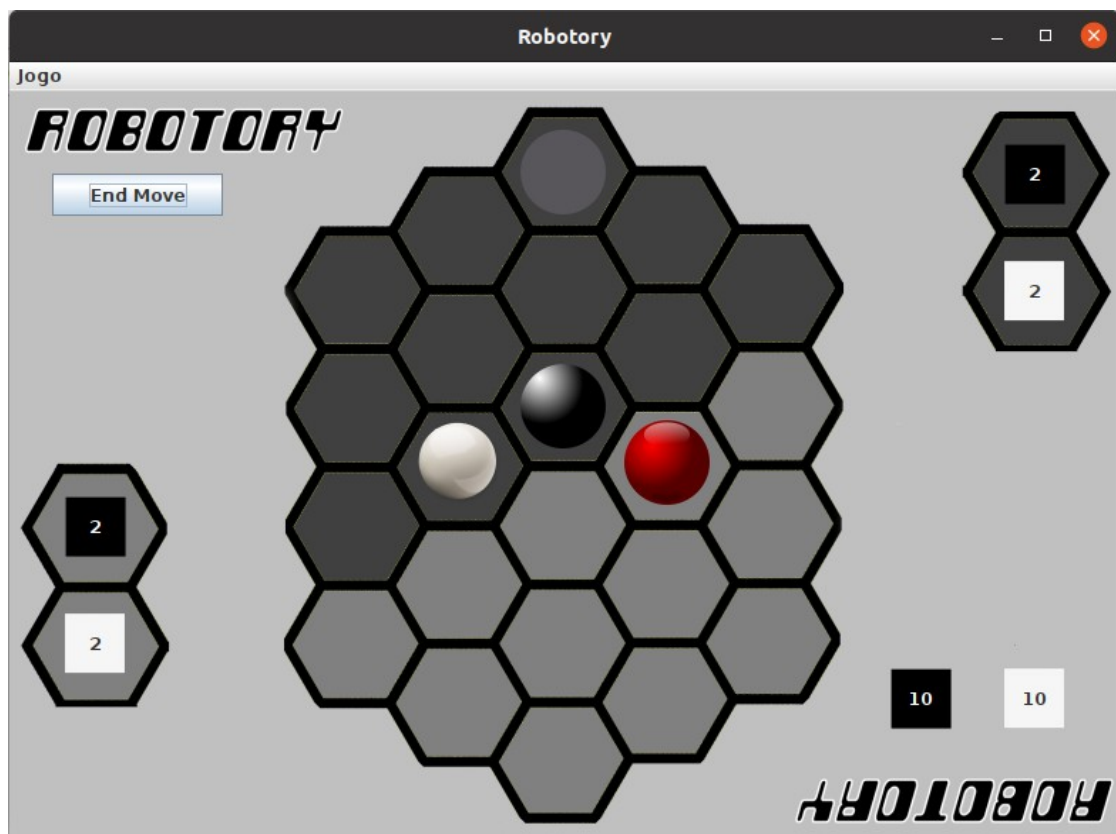


Figura 1: Apresentação visual da aplicação