**Projeto - Shutdown**

**Especificação de Requisitos de Software**

|  |  |  |  |
| --- | --- | --- | --- |
| **Versão** | **Autores** | **Data** | **Ação** |
| 1.0 | Eduardo Rosa  Fernando Mateus  Raoni Moreira da Silva | 10/09/2014 | Estabelecimento dos requisitos |
| 1.1 | Eduardo Rosa  Fernando Mateus  Raoni Moreira da Silva | 11/11/2014 | Modificação dos requisitos e Regras |

**Conteúdo:**

* Introdução;
* Visão Geral;
* Requisitos de Software;
* Apêndice.

1. **Introdução**

**Objetivo:** desenvolvimento de programa que suporte a disputa de partidas do jogo Shutdown na modalidade usuário contra usuário.

**Referências:**

Nosso jogo foi baseado no game K.C.'s Krazy Chase: comercializado no Brasil pela Phillips para o Odyssey. (<http://www.ozyr.com/o2/o2kcskra.html>). O jogo foi modificado para poder comportar partidas entre dois jogadores.

**Regras do Jogo:**

* Cada jogador controla até dois robôs dentro de um tabuleiro;
* O tabuleiro possui dimensões 6x6;
* O jogador dispõe de uma quantidade limitada de movimentos, geradas por dois dados de seis lados;
* Inicia o jogo o jogador que se conectar por ultimo;
* Cada dado gera os movimentos para o “seu” robô;
* Cada robô consome um movimento para ir até o próximo quadrado do tabuleiro;
* O turno somente passa após o jogador realizar todos os movimentos que tem direito;
* Se um robô desativar um robô adversário e ainda tiver movimentos, os mesmos ainda devem ser realizados;
* Um robô somente é desativado por traz (é onde fica o botão de *shutdown*).

1. **Visão Geral**

**Arquitetura do programa:** programa orientado a objetos, observando padrões de desenvolvimento de software estabelecidos no mercado. Além disso, a aplicação será distribuída.

**Premissas de desenvolvimento**:

* O programa apresentará uma interface gráfica bidimensional;
* Desenvolvimento em linguagem Java, voltado para ambientes Windows;
* Uso do framework NetGames NRT para conexão entre jogadores;
* A modelagem UML será desenvolvida na ferramenta Visual Paradigm.

1. **Requisitos de Software**
   1. **Requisitos Funcionais**

**Requisito funcional 1 – Iniciar partida:** o sistema deve apresentar na interface inicial a opção “Iniciar” para permitir o início de uma nova partida, operação em que são identificados os jogadores e qual o jogador que inicia.

**Requisito funcional 2 – Desistir da partida:** o programa deve apresentar na interface de jogo a opção “Desistir”, operação em que a partida é encerrada e o jogador que permaneceu é notificado como vitorioso.

**Requisito funcional 3 – Estabelecimento de turnos:** o programa deve implementar uma estrutura de turnos para a execução de cada partida. A cada turno é obrigatório que apenas um jogador esteja habilitado para realizar um procedimento de lance.

**Requisito funcional 4 – Procedimento de lance:** o programa recebe o procedimento de lance dos usuários por meio dos cliques do mouse, atualizando o tabuleiro observado as seguintes restrições:

* O jogador da vez ainda deve dispor de movimentos não utilizados do total de movimentos recebidos aleatoriamente (mínimo um, máximo seis) no início da rodada, para poder mover seu robô.
* A posição de destino é válida: não está ocupada por outro robô, ou a entrada na posição se dá pela face que contém a chave de desligamento do robô.

**Requisito funcional 5 – Procedimento de shutdown:** o programa deve suportar o procedimento de shutdown, ocorrido quando um robô A, entra no espaço ocupado por outro robô B a partir da face onde se encontra o mecanismo de desligamento do robô B, considerando as restrições a seguir:

* A disputa pelo espaço alocado é entre robôs de jogadores diferentes.
* Ao fim de uma jogada válida de desligamento, o programa deve verificar se há um ganhador para a partida.

**Requisito funcional 6 – Comunicar estado da partida:** a cada tentativa de procedimento de lance no tabuleiro ou de procedimento de shutdown, o novo estado da partida deve ser exibido na interface, dadas as seguintes observações:

* O tabuleiro deve ser atualizado com o número de movimentos restantes para o usuário da vez.
* Qualquer lance irregular deve ser comunicado ao usuário o motivo da irregularidade.
* Os robôs já desligados devem ser removidos do tabuleiro.
  1. **Requisitos Não-Funcionais**

**Requisito não funcional 1 – Tecnologia de interface gráfica:** a interface gráfica deve ser baseada em Java-Swing;

**Requisito não funcional 2 – Interface gráfica do usuário:** o programa deverá apresentar uma interface única para ambos os jogadores.

**Requisito Não Funcional 3 - Avatares:** cada jogador será identificado dentro do software através de seu par robô, distinto do par do seu adversário.

**Apêndice**

**Esboço de Interface**

O esboço abaixo representa a ideia inicial de como a tela do jogo deve ser, contendo todas as informações básicas que permitam a todos os jogadores de uma partida, visualizarem o estado atual da mesma.

