

Projeto Guerra dos Mundos

Especificação de Requisitos de Software

Versão 1.0

09/09/2019

Versão	Autores	Data	Ação
1.0	Gabriel Vitor Garcia do E.S.	09/09/2019	Estabelecimento dos Requisitos

Conteúdo:

1. Introdução;
2. Visão geral;
3. Requisitos de software;
4. Esboço Interface Gráfica;
5. Referências.

1. Introdução:

Objetivo: desenvolvimento de um software distribuído que emule uma disputa de dois jogadores no jogo RPG Guerra dos Mundos (jogo inventado pelo autor do projeto).

1.1. Definições

- **Jogador:** o usuário que responde aos comandos do jogo.
- **Elementos do jogo:** representa todos os elementos do cenário do jogo no qual é possível interagir. São estes: os personagens de ambos jogadores, os escudeiros, os inimigos, as passagens e o dinheiro.
- **Elementos controláveis:** representa os elementos do jogo no qual o jogador consegue controlar. São estes: o personagem e os escudeiros.

- **Casas:** parte jogável do mapa, onde os jogadores poderão interagir com outros elementos do jogo. Cada casa é indicada como uma fração do mapa, chamada de *tile*, nomenclatura padrão de jogos bidimensionais baseados em tiles.
- **Inimigos:** elementos do jogo que disparam eventos de confronto.
- **Passagem:** representa uma casa especial no jogo na qual muda a posição do elemento controlável que nela for.
- **Personagem:** elemento controlável do jogo que representa o jogador (se o personagem tiver seus pontos de vida zerados, o jogador perde a partida).
- **Entidades:** generalização para inimigos, jogadores e escudeiros.
- **Pontos de vida:** propriedade que determina a saúde das entidades. Estes podem ser alterados durante o jogo. Uma entidade sem pontos de vida é eliminada da partida.
- **Itens:** artefatos que garantem alguma vantagem ao jogador comprados com o dinheiro do jogo (não suportado nessa versão do jogo, ideia para uma futura implementação)
- **Eventos:** ações disparadas ao fim do movimento de um jogador, que podem ser de coleta de dinheiro, passagem ou confrontos.
 - **Dinheiro:** elemento do jogo que permite a compra de itens.
 - **Passagem:** casas especiais que alteram a posição dos elementos controláveis que nela pisarem.
 - **Confronto:** quando um jogador se movimenta para uma casa ocupada por um inimigo, acontece um confronto que decidirá quem ocupará a casa em questão. O jogador atual pode decidir entre enfrentar ou conceder a vitória. No primeiro caso, o confronto se repetirá até que uma das partes não tenha pontos de vida restante, ou até que o jogador desista do confronto, concedendo a vitória e regressando uma casa.

2. Visão Geral

2.1. Arquitetura do Software

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

2.2. Premissas de desenvolvimento

- **Desenvolvimento:** o programa deve ser implementado na linguagem de programação Java em sua versão mais recente.

- **Modelagem:** a especificação de projeto será baseada em UML 2 utilizando o software Visual Paradigm versão Community.
- **Conectividade:** a aplicação deve suportar rede através da arquitetura cliente-servidor. Essa funcionalidade deve ser implementada fazendo o uso da ferramenta NetGamesNRT.
- **Interface Gráfica:** o software deve apresentar uma interface bidimensional, dividida em uma matriz onde cada posição será referenciada como um *tile*, com uma paleta de 256 cores, similar aos jogos de 16 bits.

2.3. O jogo

- **História do jogo:** Na história desse jogo, o mundo do fogo e mundo do gelo estão em uma sangrenta guerra. Dessa forma, restou para os príncipes de ambos os mundos e seus fiéis escudeiros tentar acabar com esse mar de sangue. Para isso eles precisam dominar as estátuas sagradas presentes em cada um dos mundos. Porém estas são protegidas por fortes guardiões.

O príncipe que conseguir derrotar o guardião e dominar a estátua sagrada do mundo adversário primeiro sairá vitorioso dessa guerra.

- **Regra do jogo:** ao jogador que iniciar a partida será dado o direito de escolher entre os mundos do fogo ou do gelo, dessa forma, o que não for escolhido pertencerá ao jogador remanescente. Além disso, este ainda será favorecido com a primeira jogada.

Ambos jogadores e seus escudeiros iniciarão nas casas iniciais de seus respectivos mundos. A cada turno os jogadores poderão executar as seguintes ações:

- Rolar dados para determinar o número de casas que o jogador pode movimentar no turno, seja com seu personagem ou com os escudeiros (serão rolados dois dados, podendo assim resultar em qualquer número entre 2 a 12).
- Andar exatamente o número de casas que foi sorteado.
- Decidir interagir com elementos do jogo, como inimigos, passagens ou moedas espalhadas pelo mapa.
- Comprar itens na loja.

Lembrando que a vez do jogador só termina quando este andar o número de casas que foi sorteado. Cada jogador inicia com um personagem e com dois escudeiros controláveis, ambos possuem uma certa quantidade de pontos de vida, os quais podem ser alterados em confrontos. O jogador que tiver seu personagem sem pontos de vida será eliminado da partida conferindo a vitória ao adversário (a morte dos escudeiros não elimina os jogadores, somente a dos personagens).

Vence o primeiro jogador que conseguir derrotar o guardião do mundo adversário e dominar a estátua sagrada desse respectivo mundo (os escudeiros podem ajudar o jogador a derrotar o guardião e outros inimigos do jogo, porém somente o personagem pode dominar a estátua sagrada e vencer a guerra).

3. Requisitos de Software

3.1. Requisitos funcionais

- **Conectar:** como será um software distribuído, este deve apresentar em seu menu a opção “Conectar”, permitindo ao usuário se conectar ao servidor NetGames o qual fará a gestão dos usuários conectados.
- **Desconectar:** o programa deve apresentar em seu menu a opção “Desconectar”, para assim finalizar a conexão com o servidor, encerrando uma possível partida em andamento.
- **Iniciar partida:** uma vez conectado, o programa deve possibilitar a inicialização de uma nova partida através da opção “Iniciar”, onde será indicado o número de jogadores desta partida. O programa deve permitir partidas contendo dois jogadores.
- **Compor o mapa:** o mapa deve ser composto de tiles jogáveis (casas), tiles de cenários e elementos interagíveis. O conjunto de casas servirá como o caminho que deverá ser percorrido pelos jogadores. Os tiles de cenário servem apenas como imagem de fundo, não permitindo que os jogadores caminhem por eles. Já o último são elementos, como monstros, moedas, entre outros, sobre o qual os jogadores conseguem interagir.
- **Eventos disparáveis:** deve ser disparado um evento para o jogador do turno quando o mesmo se movimentar em uma casa ocupada por outro jogador, inimigo, uma casa de passagem ou dinheiro. Os eventos podem alterar os pontos de vida do personagem ou dos escudeiros do jogador, mudar a casa destes ou aumentar o seu dinheiro.

O jogador poderá decidir se realizará o evento ou não. Sendo que a não realização do mesmo fará com que o jogador volte uma casa de seu movimento original e finalize seu turno.
- **Movimento do jogador:** o jogador deve ter a possibilidade de se movimentar no tabuleiro utilizando as teclas direcionais do teclado, sendo que cada movimentação corresponde a um tile e deve respeitar a quantidade restante de movimentações possíveis. Um movimento bem-sucedido segue as seguintes condições:
 1. O jogador deve ter movimentação disponível;
 2. O tile ao qual deseja mover-se deve ser uma casa do tabuleiro.
 3. Ao fim de cada movimento as seguintes ações devem ser executadas:
 - Decrescer a movimentação disponível do jogador.
 - Conferir se um evento foi disparado, em caso positivo decidir se executará ou não o

evento. A opção pela não execução deve fazer com que o jogador regresse uma casa e perca sua vez (mesmo que ainda reste movimentos a serem feitos).

- Conferir se a condição de vitória foi satisfeita e atribuir vitória ao jogador que executou o movimento em caso positivo.
- Se não houverem movimentos restantes e a condição de vitória não foi satisfeita, finalizar o turno do jogador.

3.2. Requisitos não funcionais

- **Especificações do projeto:** A codificação deve ser feita em Java e a especificação de projeto baseada em UML 2;
- **Padronização do código:** Utilizar lowerCamelCase (iniciados por letras minúsculas) na definição de variáveis e métodos, e utilizar UpperCamelCase (iniciados por letras maiúsculas) na definição de Classes.
- **Interface gráfica para o usuário:** O software deve apresentar uma interface gráfica para cada usuário, com as informações dos jogadores atualizadas em tempo de execução.
- **Tecnologia da GUI:** A GUI do jogo deve ser baseada na biblioteca nativa *Java Swing*.
- **Portabilidade:** O programa deverá funcionar em qualquer plataforma que disponha da máquina virtual Java, versão 8 ou superior.
- **Internacionalização:** O jogo deve estar disponível em Português.

4. Esboço da Interface Gráfica

A interface gráfica do jogo (*GUI*), será composta de recursos disponíveis no software RPG Maker MV. O mesmo também foi utilizado na criação do mapa do jogo.



Esboço do mapa do jogo, feito no software RPG Maker MV.

5. Referências

MORGAN, Leonardo. **NetGames**. Disponível em: <<http://www.inf.ufsc.br/~netgames/>>.