

Engenharia de Software I - UFSC

Virtual Tabletop Especificação de Requisitos de Software Versão 1.0

27 de março de 2023

Histórico de revisão

Data	Versão	Autor(es)	Ação
27/03/2023	1.0	Isac, Pedro, Tiago	Especificação dos requisitos

Especificação de Requisitos de Software

1. Introdução

1.1. Objetivo de desenvolvimento

Desenvolvimento de uma plataforma virtual para mesas de RPG.

1.2. Definições e abreviaturas

Mestre - Jogador que cria o combate, arbitra as rodadas e controla os NPC's.

Playable Character (PC) - Personagem controlado por um jogador não mestre.

Non Playable Character (NPC) - Personagem controlado por um jogador-mestre.

Pontos de Vida (PV) - Quantidade de pontos que representa o estado de vida da personagem.

NdX - Qualquer quantidade N de dados de X lados. (Ex: 2d20 = Role 2 dados de 20 lados)

Area of Effect (AoE) - Ação que tem efeito sobre uma área ao invés de apenas um personagem.

1.3. Referências

- Explicação de combate no RPG Dungeons & Dragons

2. Visão geral do sistema

2.1. Arquitetura do Software

Aplicação desktop distribuída em rede com a arquitetura cliente-servidor.

2.2. Premissas de desenvolvimento

- O programa será desenvolvido na linguagem python usando os princípios de programação orientada a objetos
- A interface será desenvolvida usando a biblioteca Tkinter
- A distribuição em rede será implementada utilizando o framework py-netgames
- A modelagem UML dos diagramas será realizado com a ferramenta Visual Paradigm

3. Requisitos da Aplicação

3.1. Regras de negócio

RN01 - O jogo é dividido entre jogadores e mestre. Cada jogador controla um único PC e o mestre tem controle de todos os NPC's.

RN02 - No começo de combate todos os atores rolam um d20+bônus para iniciativa. Isso determina a ordem dos jogadores na rodada.

RN03 - Durante seu turno, a personagem pode mover-se, atacar ou curar.

RN04 - Quando uma personagem é acertada por um ataque, seu PV é reduzido igual ao dano.

RN05 - Caso o PV de uma personagem chegar a zero, ela é eliminada.

RN06 - O combate termina se todos os PC's ou todos os inimigos forem eliminados.

3.2. Requisitos funcionais

RF01 - Iniciar partida: Permitir que o mestre inicie a partida e que os jogadores se conectem a partida.

RF02 - Construir partida: Ao iniciar o jogo, o mestre pode configurar o tamanho, as dimensões do tabuleiro e sua aparência.

RF03 - Construir personagem: Ao se conectar a partida, o jogador e o mestre podem construir seus personagens, adicionando atributos e informações.

RF04 - Listar ação: O jogador ou mestre podem adicionar ações relacionadas a uma personagem, como ataques e curas.

RF05 - Definir iniciativa: Com todos os personagens no jogo, é rolado a iniciativa para determinar a ordem dos turnos. Se trata de uma rolagem de dados para cada personagem, com a adição do bônus de iniciativa deste personagem ao resultado. A ordem é então determinada do número maior ao menor.

RF06 - Mover personagem: No seu turno, uma personagem pode mover seu personagem a distância determinada em seus atributos. Esse movimento pode ocorrer ou não em partes.

RF07 - Ataque corpo-a-corpo: No seu turno, uma personagem pode atacar uma outra personagem em uma posição vizinha, se este tiver uma ação de ataque corpo-a-corpo.

RF08 - Ataque a distância: No seu turno, uma personagem pode atacar uma outra personagem no tabuleiro, se este tiver uma ação de ataque a distância e a distância máxima for respeitada.

RF09 - Ataque AoE: No seu turno, uma personagem pode realizar um ataque em uma posição qualquer, se este tiver uma ação de ataque AoE. Qualquer personagem na área afetada pelo ataque receberá dano.

RF10 - Verificar condição de vitória: Ao final de cada ataque, há a checagem se todos os PC's ou NPC's foram derrotados. Se um dos lados for zerado, o outro ganha a vitória.

3.3. Requisitos não funcionais

RFN01 - Interface: O rascunho da interface, apresentando a aba do jogador e suas ações a esquerda, o tabuleiro de personagem no meio e a direita a aba de iniciativa, indicando a ordem de turnos.



RFN02 - Persistência: O programa deverá permitir ao mestre salvar qualquer personagem do jogo. Além disso, deve permitir que os jogadores salvem seus personagens e os recarreguem em outra partida.