

Especificação de requisitos

Projeto Tower Capture

Versão: 1.1

Data: 22/11/2021

Versão	Autor(es)	Data	Ação
1.0	Leonardo Nunes Matheus Bueno Thiago Comelli	01/11/2021	Definição dos requisitos do projeto
1.1	Leonardo Nunes Matheus Bueno Thiago Comelli	22/11/2021	Implementação do protótipo da interface gráfica utilizando a biblioteca PyGame

Conteúdo:

1. Introdução
2. Visão geral do sistema
3. Requisitos de software

1.Introdução

1.1. Objetivo do desenvolvimento

Desenvolvimento de um programa de computador que implementa um jogo chamado “Tower Capture” entre dois jogadores.

1.2. Referências

O jogo possui a mecânica de combate semelhante ao do jogo “Civilization III” (RPG baseado em turnos) porém sem a ideia de expansão e criação de cidades, mas sim apenas com o objetivo de proteger a sua torre controlando personagens. Neste caso, lembrando a ideia do jogo Kingdom Rush - Tower Defense (RPG de estratégia, disponível para android).

- Link para o jogo Civilization
<https://civilization.com/pt-BR/civilization-3/>
- Link para o jogo Tower Defense
<https://play.google.com/store/apps/details?id=com.ironhidegames.android.kingdomrush>

2. Visão geral do sistema

2.1. Arquitetura do software

- Utiliza o paradigma de orientação a objetos
- Possui arquitetura para ser jogada com dois jogadores em uma mesma interface
- Possui interface gráfica com o usuário

2.2. Premissas de desenvolvimento

- Deverá ser implementado na linguagem de programação Python
- Deverá utilizar a interface gráfica 2D utilizando a biblioteca PyGame
- A apresentação do jogo será para dois jogadores utilizando a mesma interface gráfica
- A modelagem do desenvolvimento deve utilizar a linguagem UML 2.

3. Requisitos da aplicação

3.1. Requisitos funcionais

[RF1] Iniciar partida

What: Corresponde a apresentação da tela para iniciar um novo jogo, bem como os botões para inicializar a partida ou finalizar o programa.

Why: Este requisito é necessário para que, ao final de uma partida, o jogo reinicie e dê as opções para os jogadores de iniciar uma nova partida ou finalizar o programa.

Who: Qualquer um dos dois jogadores pode iniciar uma nova partida.

When: Ao abrir o programa ou ao final de uma partida.

Where: Através dos botões de ("INICIAR") ou ("SAIR").

How: A definição de quem será o jogador com a torre do lado esquerdo e a do lado direito será decidido entre os jogadores antes da inicialização da partida. Tendo isso definido, os jogadores irão alternando a medida que o turno for dado para o jogador da esquerda ou o da direita.

[RF2] Selecionar personagem

What: Consiste na etapa de seleção de qual personagem o jogador vai utilizar.

Why: Esta é uma das ações que cada jogador deve fazer para concluir a sua jogada.

Who: Será realizado por um dos jogadores quando for a sua vez.

When: Caso seja o início da partida, será o jogador da esquerda, ou, caso a partida ainda esteja em andamento, será após a finalização da jogada do jogador anterior

Where: Será na posição no mapa onde está o personagem que o jogador deseja selecionar. Além disso, será apresentado na lateral esquerda, acima do botão de passar o turno, os status daquele personagem selecionado, tal como vida, defesa, movimentação e ataque.

How: Através de um clique do mouse sobre o personagem. Caso o jogador se arrependa de ter selecionado aquele personagem, ao clicar novamente com o mouse nele, o personagem é desselecionado e o jogador deve escolher novamente outro personagem.

[RF3] Selecionar posição

What: Consiste na etapa de seleção de posição que o personagem selecionado pelo jogador vai caminhar ou atacar.

Why: Esta é uma das ações que cada jogador deve fazer para concluir a sua jogada.

Who: Será realizado por um dos jogadores quando for a sua vez.

When: Após selecionado o personagem pelo jogador.

Where: Nas posições disponíveis para aquele personagem

How: Após selecionado o personagem, aparecerá em uma cor clara as posições disponíveis para o personagem andar e em cor avermelhada para o personagem atacar um inimigo.

Através do clique na posição disponível para andar/atacar, o personagem irá realizar o comportamento desejado. O personagem poderá andar e atacar em um mesmo turno, porém, caso ataque antes de andar, este não poderá andar até o próximo turno do jogador.

[RF4] Atacar

What: Consiste no procedimento de um personagem atacar o outro, infligindo dano nele.

Why: A partir da interação de ataques entre os personagens e a torre, será definido quem é o vencedor da partida.

Who: Será realizado por um dos jogadores quando for a sua vez.

When: Após selecionado uma posição ao alcance do personagem onde reside um inimigo no mapa.

Where: Na posição do inimigo selecionado

How: Após selecionado a posição onde reside o inimigo, o personagem irá desferir um ataque que irá reduzir o nível de vida do inimigo na mesma proporção do seu poder de ataque para aquele inimigo.

[RF5] Andar

What: Consiste no procedimento do personagem selecionado andar pelas posições do mapa.

Why: Para que o personagem possa se aproximar dos personagens inimigos e da torre inimiga. Bem como permitir traçar estratégias para o jogador proteger a sua própria torre.

Who: Será realizado por um dos jogadores quando for a sua vez.

When: Após selecionado uma posição ao alcance do personagem.

Where: Em uma posição válida do mapa

How: Após selecionado uma posição válida para andar, o personagem irá caminhar até esta posição.

[RF6] Passar turno

What: Consiste em realizar a ação de passar o turno para que o próximo jogador execute a sua jogada.

Why: Foi pensado na lógica de turnos para que os jogadores pudessem pensar qual jogada poderiam fazer, semelhante a ideia de jogos de tabuleiro.

Who: O jogador que finalizou a sua jogada.

When: Ao final de uma jogada.

Where: Através do botão ("PASSAR TURNO") localizado na lateral esquerda da tela.

How: Ao final de uma jogada, é então clicado para passar o turno para o próximo jogador.

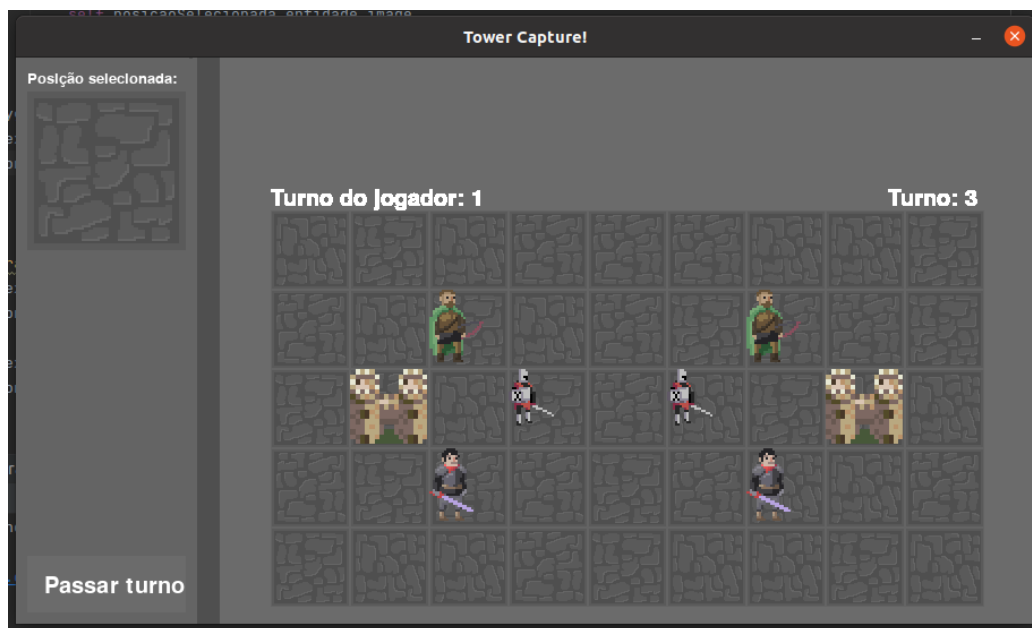
3.2. Requisitos não funcionais

[RN1] Implementação

O jogo será implementado utilizando a versão 3.9 da linguagem de programação Python e utilizando a biblioteca PyGame para a criação da interface gráfica.

[RN2] Interface gráfica

Apresentação visual da implementação do protótipo utilizando a biblioteca python: PyGame.



Tela de início da partida



Seleção de um personagem para iniciar a movimentação

[RN3] Tela simultânea

O programa necessita que dois jogadores interajam utilizando a mesma interface gráfica.