

Especificação Requisitos - Castle Wars

Gabriel Martins Bergmann

Gian Augusto Ortiz

Roberto Henrique Rozario

Novembro de 2021

1 Introdução

Objetivo: desenvolvimento de um programa que suporte as funcionalidades requeridas para jogar uma partida de Castle Wars entre dois usuários. O desenvolvimento será pautado pela descrição das regras a seguir: O objetivo é aumentar o nível de seu castelo a 100 ou reduzir o nível do castelo do seu oponente a zero. O jogo funciona em turnos em que cada jogador pode jogar uma carta ou descartar até 3 cartas, as quais tem diversos efeitos. Para jogar uma carta é preciso pagar seu custo com recursos (espadas, tijolos e cristais) que são produzidos no início de cada turno por trabalhadores (soldados, construtores e magos), a quantidade produzida é igual ao número de trabalhadores que o jogador tem. O baralho utilizado pelos jogadores deve ter no máximo 50 cartas, e não mais que 5 cópias de cada carta para ser considerado válido. A fonte dessas regras se encontra nos seguintes links. <https://www.youtube.com/watch?v=UaxvaV5BoTI>. <https://stom66.github.io/tts-castle-wars/>.

2 Visão Geral

Arquitetura do programa: programa orientado a objetos, *standalone*.

Premissas de desenvolvimento:

- O programa deve apresentar uma interface gráfica bidimensional;
- O programa deve ser implementado em Python;
- O programa será jogado por dois jogadores.

3 Requisitos de Software

3.1 Requisitos Funcionais

Requisito funcional 1 - Seleção do baralho: o programa deve permitir que antes de iniciada uma partida os jogadores possam escolher o baralho que eles utilizarão, terá alguns baralhos padrão disponíveis para a escola. Cada baralho tem no máximo 50 cartas com no máximo 5 cartas do mesmo tipo.

Requisito funcional 2 - Iniciar: o programa deve iniciar a partida assim que os jogadores escolherem seus baralhos. Além disso, o programa deve iniciar cada jogador com 2 magos, 2 construtores e 2 soldados e seu castelo em nível 30. O programa, deve, também, adicionar 8 cartas à mão de cada jogador vindas de seu baralho de forma aleatória.

Requisito funcional 3 - Jogar carta: um jogador na sua rodada pode escolher jogar uma carta que está em sua mão se possuir recursos para isso, o programa deve limitar isto. Se o jogador possuir recursos para jogar a carta, o programa deve realizar a ação que a carta descreve, como exemplo, aumentar em 5 o nível do castelo do jogador.

Requisito funcional 4 - Descartar carta: um jogador pode descartar até três cartas por rodada, ao descartar uma carta ou mais o programa deve obter o mesmo número de cartas do baralho aleatoriamente e retornar as cartas descartadas para o baralho ao fim do turno.

Requisito funcional 5 - Terminar Turno: um jogador pode terminar seu turno a qualquer momento interagindo com um botão definido para essa função e iniciando a rodada do oponente.

Requisito funcional 6 - Comunicação do estado da partida: os jogadores devem conseguir visualizar as informações de cada jogador, estas informações são: o número de magos, o número de construtores, o número de recursos, o nível do castelo, o nível do muro e os buffs de cada castelo.

3.2 Requisitos Não Funcionais

Requisito não funcional 1 - Especificação do projeto: além do código de implementação do programa feito em Python, deve ser desenvolvido a especificação do projeto feito em UML.

Requisito não funcional 2 - Interface gráfica para usuário: o programa deve ter uma interface gráfica única para os dois jogadores, mostrando a mão do jogador que tem a vez e trocando para oponente quando o turno é passado, nenhum jogador pode ver a ordem das cartas de seu baralho ou de seu oponente.

Requisito não funcional 3 - Tecnologia de interface gráfica para usuário:

A interface gráfica deve ser baseada em Pygame.

Requisito não funcional 4 - Cores para os jogadores: cada jogador terá seu castelo e seus recursos, para diferenciar os dois jogadores cada um terá uma cor diferente, ou vermelho ou azul.