**Jogo Cobras e Escadas**

|  |  |
| --- | --- |
| **Nome:** | **Matricula:** |
| Ian Mateus | 433411 |
| Lincoln Oliveira | 375090 |
| Renner | 427609 |

**1. Objetivo**

Este documento visa explicar as decisões do projeto de jogo “CobrasEscadas”, para a disciplina de Projeto e Desenvolvimento de Software, ministrada pela professora Paulyne Matthews Juca.

**2. Background**

O jogo Cobras e Escadas é um antigo jogo de tabuleiro aonde os jogadores, no mínimo 2, tentam chegar ao final do tabuleiro, normalmente um tabuleiro de 100 casas. Durante o percurso cobras podem picar o jogador quando esta cai na casa e fazer ele retornar um número de casas, mas, quando encontra escadas, o jogador pode pular um número de casas, ele não tem decisão sobre quantas casas irá descer ou subir. Um dado de 6 faces é jogador alternando a vez.

**3. Decisões de projeto**

**Padrão Builder**

Responsável pela abstração da criação dos jogadores.

As classes:

* *AbuilderJogador*
* *Diretor*

**Padrão Observer**

Responsável por identificar que alguém ganhou.

As classes:

* *posicaoJogadorMapa*
* *iteradorJogador*

**Padrão Singleton**

Responsável por classes de instancias únicas no sistema, como Mapa e Dado.

As classes:

* *Dado*
* *Tabuleiro*

**Padrão Iterator**

Responsável pela abstração de percorrer a lista de Jogadores e Casas do tabuleiro.

As interfaces:

* *IAgragador*
* *IteradorMapa*
* *IteradorJogador*

As classes são:

* *Mapa*
* *Jogadores*
* *IteradorMapa*
* *IteradorJogador*

**Arquitetura adotada**

Utilizamos a arquitetura MVC (Model-View-Controller), fizemos essas separações para abstrair a lógica do jogo da interface. Facilitando no desenvolvimento em paralelo das duas partes. Nossa arquitetura não é um MVC como normalmente aplicado, mas, uma abstração das responsabilidades para um jogo versão Desktop, já que MVC normalmente não é utilizado para este tipo de aplicação.

**Relação casa jogador**

Utilizamos uma classe PosicaoJogadorMapa na qual possui um HashMap onde o jogador é a chave a casa o valor, assim todo jogador está relacionado a uma casa do tabuleiro.

**Classes de interface**

Organizamos o jogo de fáceis acréscimos de novas funcionalidades e lógica de movimentação do jogador no tabuleiro.