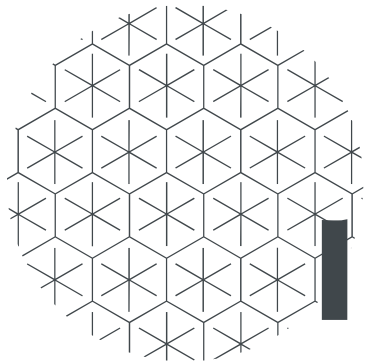


Entrega Final: Projeto Avatar Escape

Feito pelos alunos Beatriz (166531) e Pedro (187044) na disciplina MC322,
durante o 1º semestre de 2021.





Introdução

O que é o Avatar Escape?

Avatar Escape é um jogo com o propósito de brincar com a memória e a tática dos seus usuários.

Tendo a sua temática baseada na animação “Avatar: A Lenda de Aang”, o jogo é dividido em 2 etapas, a primeira funciona como um “campo minado”, enquanto a segunda, como uma batalha de cartas contra seus inimigos.

Nossos diferenciais estão:

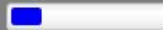
1. Na temática do jogo, que agrada os fãs do seriado e traz uma maior conexão com o história do jogo.
2. Na possibilidade de encontrar personagens do seriado original e ganhar benefícios diversos, o que não aconteceria em um jogo normal de “campo minado”.
3. Nas fases de batalha, que instigam o pensamento tático e estratégico do usuário.
4. Nos gráficos visualmente trabalhados, que remetem ao seriado de modo constante.



Vida - 100%



Estado Avatar - 25%



Vida - 95%



Estado Avatar - 25%



X



Vida - 100%



Força - 25%



Água: 10% de cura
em Vida

1



Terra: 10% de cura
em Estado Avatar

2



Fogo: Dano em 10%
do Estado Avatar

3



Ar: Reduz a Força do
inimigo em 10%

4

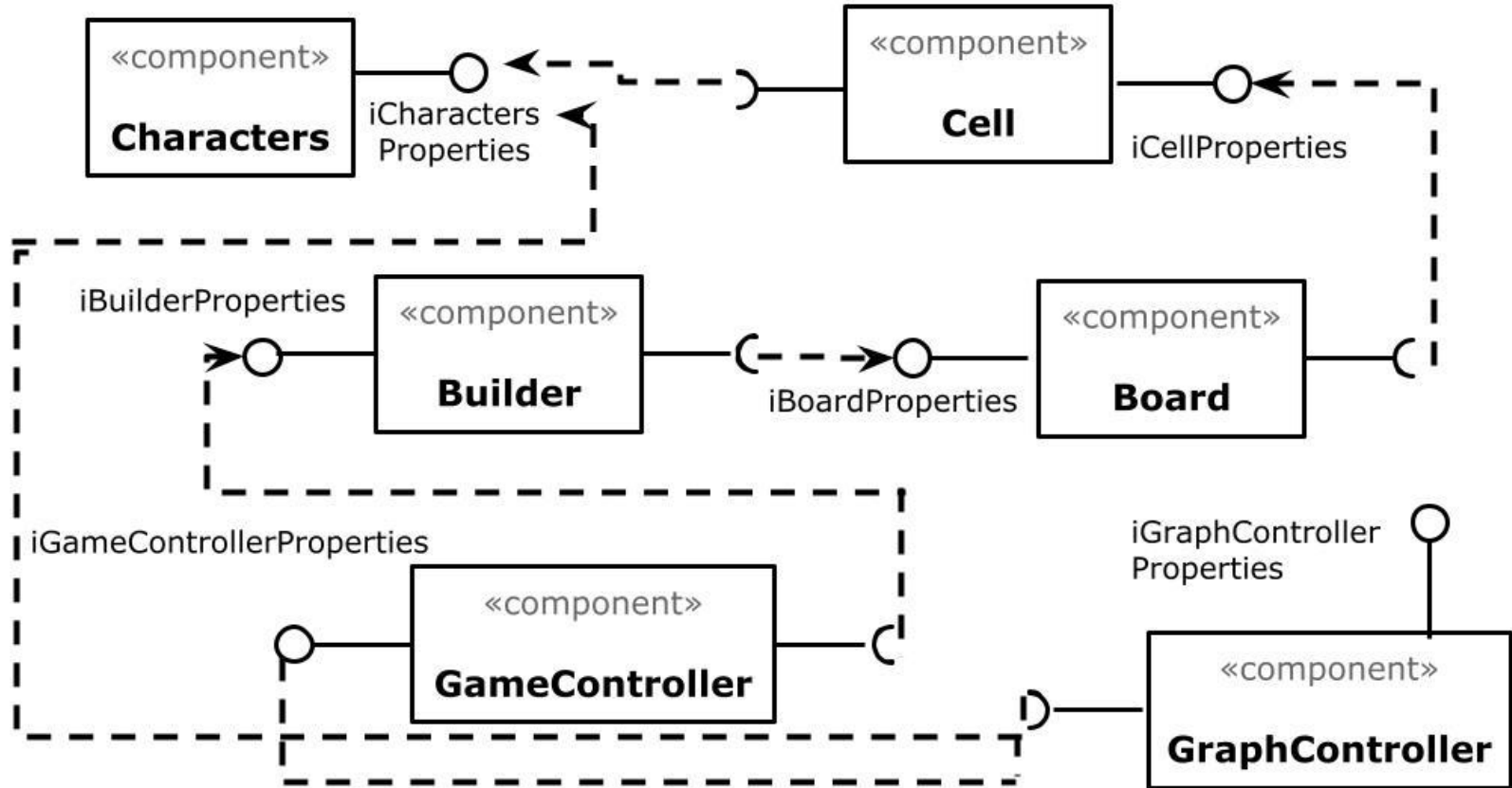


2.

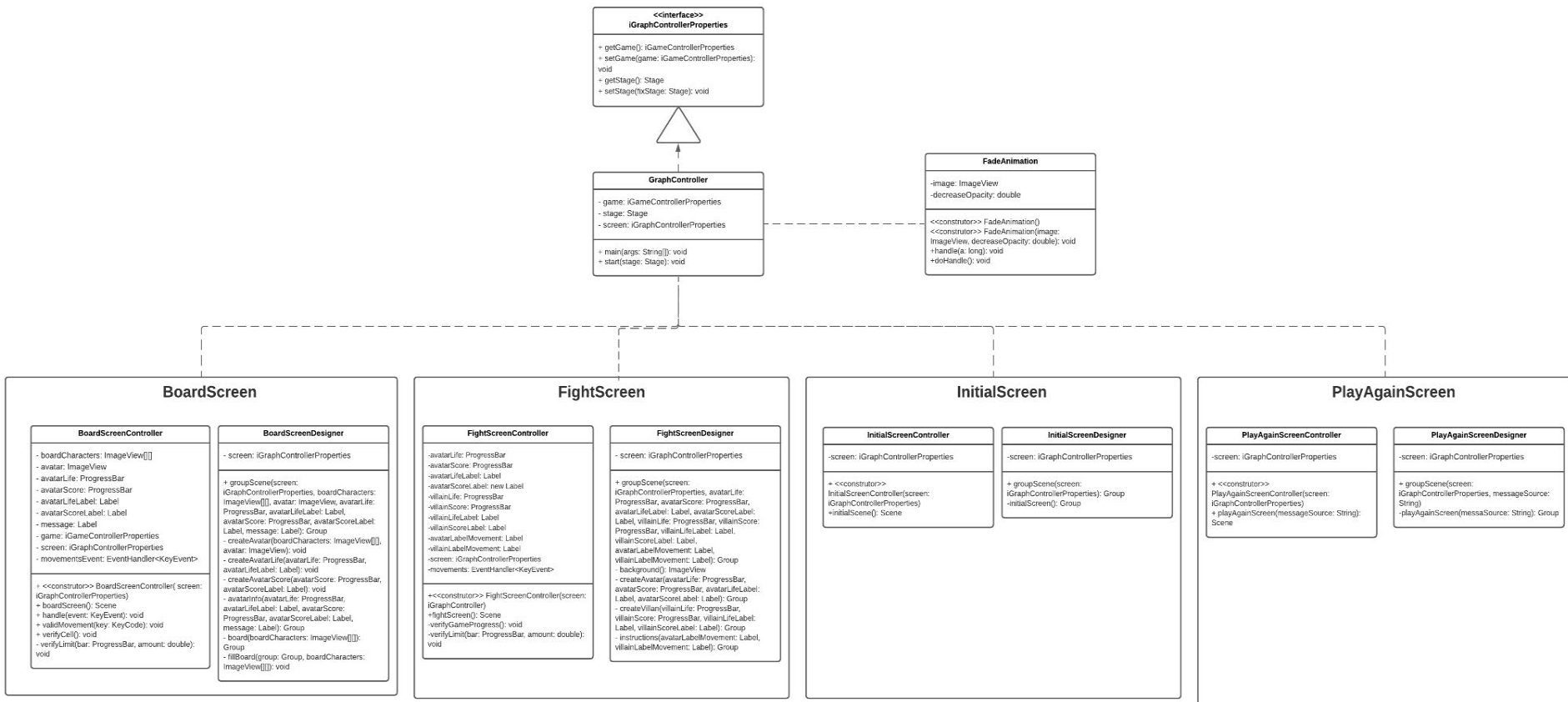
Arquitetura



Nível de Componentes



O componente GameController



The background features several abstract elements: a large, irregular brown shape on the left; a light beige oval in the upper left; a dark grey oval in the lower right containing a white hexagonal pattern; and a black line drawing of a leafy branch in the top right corner.

3.

Design Pattern

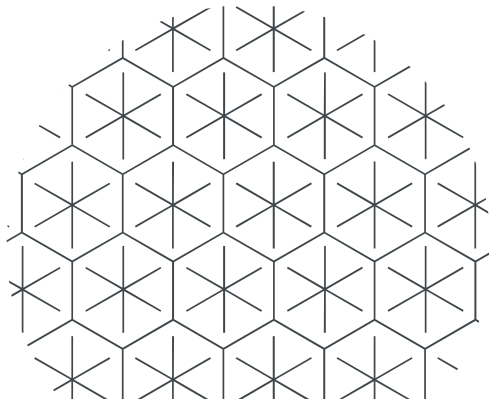
Singleton

Aplicabilidade

Para salvar o estado do player (Aang) durante as mudanças de fases e no próprio controle de criação do tabuleiro, fazemos com que o GameController (assim como acontece com o Stage do JavaFX em que é instanciado uma única vez) seja instanciado apenas uma vez. Caso seja preciso resetar o jogo, temos uma função para isso.

Objetivo

A ideia de utilizar esse Creation Pattern (Singleton) é ter certeza de que uma classe terá uma única instância, provendo acesso global a essa instância.



Singleton

Benefícios

Facilidade na troca de cenário (Scene) sem a necessidade da criação de uma nova tela (Stage). Além disso, isso pode auxiliar em projetos no futuro para salvar o estado do jogo e da janela, como, por exemplo, colocando um tempo para o jogador finalizar uma fase do tabuleiro com a opção de pausar o jogo.

Código

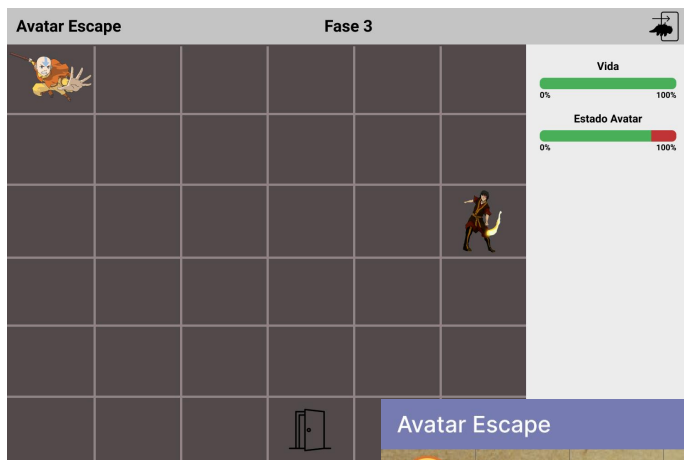
```
public class GraphController extends Application implements iGraphControllerProperties {  
    private static iGameControllerProperties game = new GameController();  
    private static iGraphControllerProperties screen = new GraphController();  
  
    <...>  
  
    public iGameControllerProperties getGame(){  
        return this.game;  
    }  
}
```



4.

Histórico, Aprendizados e o Futuro

1ª Etapa: O protótipo



(1ª versão)

(2ª versão)



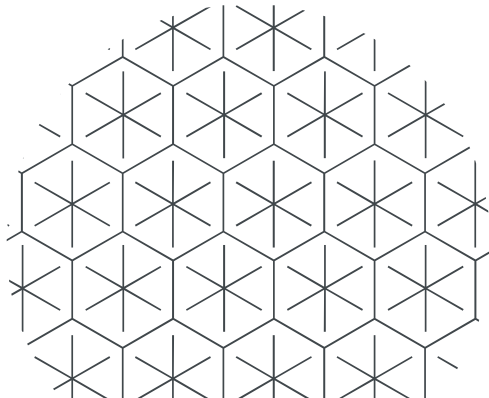
O Figma

Inicialmente nosso foco foi pensar na interface do projeto, por isso, com o auxílio da ferramenta Figma, nós montamos todo o design das telas (que futuramente foi revisado, visto a sua dificuldade)

2ª Etapa: O Começo do Código

Ordem

Optamos por começar programando as classes “menores” (characters, cells, etc), para depois ir para as “maiores” (controllers). Aqui, nós não programamos os gráficos, porque ainda não tínhamos conhecimento de ferramentas para tal aplicação. Nessa etapa do projeto, mudamos alguns pontos da nossa arquitetura e algumas interfaces para melhorar a comunicação dos nossos componentes e o controle do tabuleiro.



3ª Etapa: A Interface Gráfica

O JavaFX

Após programarmos as partes não-gráficas do código, partimos para o desenvolvimento das interfaces gráficas. Entretanto, notamos que não conseguiríamos realizar o controle dos gráficos do modo que havíamos pensado (tudo em um arquivo), por conta da lógica do próprio JavaFX. Para tal, criamos duas classes para cada tipo de tela, de modo que a organização e performance foram beneficiados.

4ª Etapa: Aprendizados

Entre os aprendizados que tivemos: destacam-se 2:

1. JavaFX: O uso da ferramenta JavaFX, que possibilitou a construção da interface gráfica.
2. Gestão de tempo e Priorização de tarefas: Ao longo de projeto, nós tivemos várias ideias. Portanto, foi necessário que nós aprendêssemos a priorizar tarefas, a fim de realizarmos a entrega final no tempo esperado.

5. O Futuro

Mapas Randômicos

A geração de tabuleiros randômica e automática facilitaria nosso trabalho e tornaria nosso jogo mais versátil.

Efeitos Sonoros

O uso de sons em jogos poderia indicar movimentos corretos e incorretos, feitos e não feitos, melhorando a experiência dos usuários.

Animações

Animações mais sofisticadas agregariam maior dinamismo e divertimento ao nosso jogo.

3 Vidas

Algo que nós queríamos realizar, porém não conseguimos por conta do tempo, era que os jogadores tivessem 3 vidas antes de mudar o tabuleiro.

NÓS AGRADECEMOS!

