

Universidade da Beira Interior

Departamento de Informática



**Departamento de
Informática**

Nº 231 - 2022: *ENDLESS RUNNER 2D GAME*

Elaborado por:

Pedro Afonso Cerqueira Jorge Teles Palhinha

Orientador:

Professor Doutor Frutuoso Gomes Mendes da Silva

31 de janeiro de 2023

Agradecimentos

Gostaria de agradecer ao Professor Doutor Frutuoso Gomes Mendes da Silva, pela oportunidade de realizar este projeto e pela sua ajuda ao longo do semestre enquanto meu orientador.

A conclusão deste trabalho, bem como da grande maior parte da minha vida académica não seria possível sem a ajuda dos meus pais e dos meus avós e por isso quero agradecer também a eles por terem esperança e acreditarem em mim mesmo quando eu não conseguia. Por fim quero deixar um pequeno agradecimento a todos os meus amigos que me acompanharam e encorajaram neste longo e árduo percurso.

Conteúdo

Conteúdo	iii
Lista de Figuras	v
1 Introdução	1
1.1 Enquadramento	1
1.2 Motivação	1
1.3 Objetivos	1
1.4 Organização do Documento	2
2 Trabalhos Relacionados	3
2.1 Introdução	3
2.2 <i>Alto's Adventure</i>	3
2.3 <i>Endless Runner 2D</i>	4
2.4 <i>Ovni 2D</i>	4
2.5 Conclusões	5
3 Tecnologias e Ferramentas Utilizadas	7
3.1 Introdução	7
3.2 Tecnologias Utilizadas	7
3.2.1 C#	7
3.3 Ferramentas Utilizadas	7
3.3.1 <i>Unity</i>	7
3.3.2 <i>Unity Hub</i>	8
3.3.3 <i>Visual Studio</i>	8
3.4 Conclusões	8
4 Problemas Encontrados e Implementação	9
4.1 Introdução	9
4.2 Problemas Encontrados	9
4.2.1 <i>Parallax Background</i>	9
4.2.2 <i>Alinhamento dos Botões no Menu da Seleção de Mapas</i> .	10
4.3 Implementação	11

4.3.1	<i>Efeito do Background</i>	11
4.3.2	Menu Inicial	12
4.3.3	Alinhamento dos botões	12
4.3.4	Tratamento da Pontuação do Jogador	12
4.3.5	Sistema de Vidas	13
4.4	Conclusões	14
5	Conclusões e Trabalho Futuro	15
5.1	Conclusões Principais	15
5.2	Trabalho Futuro	15
	Bibliografia	17

Lista de Figuras

2.1	Exemplo do estilo visual de Alto's Adventure	3
2.2	Exemplo do <i>gameplay</i> de <i>Endless Runner 2D</i>	4
2.3	Exemplo do <i>gameplay</i> de <i>Ovni 2D</i>	4
4.1	O <i>background</i> está dependente da câmara de jogo que por sua vez depende do jogador	10
4.2	O resultado da estratégia implementada	10
4.3	Ideia para o menu de seleção dos mapas	10
4.4	Menu de Seleção de Mapas	11
4.5	Menu Inicial	12
4.6	Controlo da pontuação do jogador	13
4.7	Sistema de vidas do jogador	13
4.8	Ecrã de fim de jogo	14

Acrónimos

IDE *Integrated Development Enviroment*

UI *User Interface*

Capítulo

1

Introdução

1.1 Enquadramento

Este documento tem como propósito a explicação do projeto e de como este foi implementado. O projeto referido foi desenvolvido no âmbito da unidade curricular de Projeto lecionada pelo Professor Doutor Nuno Pombo e tendo como orientador o Professor Doutor Frutuoso Gomes Mendes da Silva.

1.2 Motivação

A motivação para a realização deste projeto tem origem no interesse em adquirir conhecimentos e competências na área de jogos digitais, em particular no seu desenvolvimento.

1.3 Objetivos

Os objetivos propostos para este projeto são os seguintes:

- Desenvolver competências no desenvolvimento de jogos usando o motor de jogo Unity
- Criar um jogo digital 2D do tipo *endless scroller* com pelo menos dois níveis jogáveis

1.4 Organização do Documento

De modo a refletir o trabalho que foi feito, este documento encontra-se estruturado da seguinte forma:

1. O primeiro capítulo – **Introdução** – apresenta o projeto, a motivação para a sua escolha, o enquadramento para o mesmo, os seus objetivos e a respetiva organização do documento.
2. O segundo capítulo – **Trabalhos Relacionados** – apresenta alguns jogos com ideias ou implementações semelhantes à definida para este projeto.
3. O terceiro capítulo – **Tecnologias e Ferramentas Utilizadas** – descreve as ferramentas utilizadas para a realização deste projeto, bem como as tecnologias utilizadas durante do desenvolvimento do jogo.
4. O quarto capítulo – **Problemas encontrados e Implementação** – apresenta alguns problemas encontrados no desenvolvimento do jogo e também como foram implementadas as funcionalidades principais do jogo.
5. O quinto capítulo – **Conclusões e Trabalho Futuro** – descreve as conclusões retiradas no fim deste trabalho e apresenta algumas melhorias que se podem fazer ao jogo num possível trabalho futuro.

Capítulo

2

Trabalhos Relacionados

2.1 Introdução

Existem inúmeros jogos digitais 2D do tipo *endless runner* para diversas plataformas. Este capítulo apresenta diversas opções de projetos e iniciativas que visam objetivos semelhantes ao projeto mencionado neste documento.

2.2 *Alto's Adventure*

Alto's Adventure [1] é um *endless runner* desenvolvido para telemóveis, destacando-se pela sua jogabilidade fluida e pelo seu estilo visual. Consiste no jogador chegar o mais longe que conseguir e obter uma pontuação elevada, evitando obstáculos, fazendo truques e apanhando moedas.

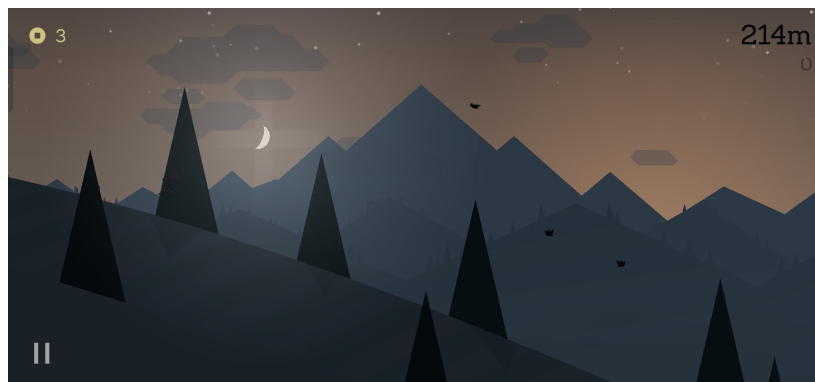


Figura 2.1: Exemplo do estilo visual de Alto's Adventure

2.3 *Endless Runner 2D*

Este é um jogo [2] bastante simples que o próprio nome explica o seu tipo. Tem como objetivo chegar o mais longe possível e desviar de obstáculos, saltando por cima deles ou deslizando por baixo deles.



Figura 2.2: Exemplo do *gameplay* de *Endless Runner 2D*

2.4 *Ovni 2D*

Ovni 2D [3] é um *side scroller* com um conceito muito simples, desviar dos obstáculos movendo a personagem para cima ou para baixo. À medida que os obstáculos avançam o jogador tem de escolher uma de 3 posições para evitar a colisão com os asteróides. Quanto mais tempo conseguir ficar vivo, mais pontuação obtém.

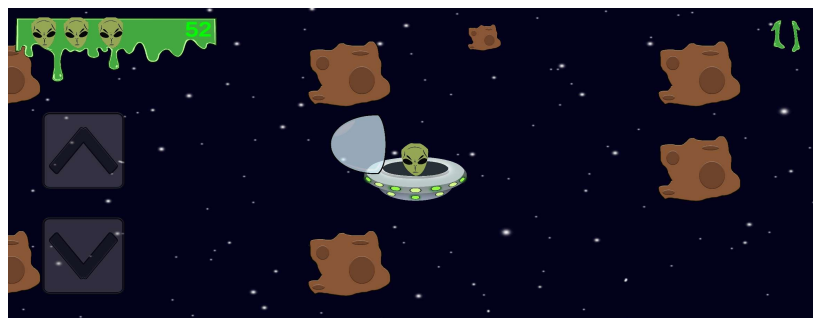


Figura 2.3: Exemplo do *gameplay* de *Ovni 2D*

2.5 Conclusões

Resumidamente são vários os jogos com um conceito semelhante visto que é um tipo de jogo fácil de desenvolver mesmo para alguém esteja a aprender de raiz os conceitos de programação de jogos digitais.

Capítulo

3

Tecnologias e Ferramentas Utilizadas

3.1 Introdução

Este capítulo irá destacar todas as tecnologias e ferramentas utilizadas para a elaboração do projeto.

3.2 Tecnologias Utilizadas

3.2.1 C#

C# [4] é uma linguagem de programação orientada a objetos criada pela Microsoft para desenvolvimento de aplicações para a plataforma .NET. A linguagem é semelhante a outras como C++ e Java, e possui recursos avançados como tratamento de exceções, gestão de memória automática e suporte a orientação a objetos.

3.3 Ferramentas Utilizadas

3.3.1 Unity

Unity [5] é um motor de jogo para criar jogos 2D e 3D para várias plataformas, incluindo dispositivos móveis, consolas e computadores. O Unity tem um editor visual para criar cenas, animações e personagens, bem como uma série de ferramentas para testar e otimizar o desempenho do jogo como *Scene View*, *Unity Remote*, *Unity Test Runner*, etc.

3.3.2 *Unity Hub*

O *Unity Hub* [6] é uma ferramenta da *Unity* que permite gerir de forma fácil as instalações do *Unity* e as configurações de projeto. Ele permite descarregar e gerir várias versões do *Unity*, incluindo versões beta e estáveis, sem precisar de múltiplos instaladores.

3.3.3 *Visual Studio*

O *Visual Studio* [7] é um *Integrated Development Enviroment* (IDE) da *Microsoft*. Suporta várias linguagens de programação, incluindo *C#*, *C++*, *Visual Basic*, *Python*, e *JavaScript* e inclui ferramentas para auxiliar no desenvolvimento de *software*, como um editor de código, depurador, gestor de versões, ferramentas de construção, e suporte a extensões e *plug-ins*.

3.4 Conclusões

Em suma as tecnologias e as ferramentas utilizadas para desenvolver este projeto foram escolhidas por serem as mais apropriadas para o desenvolvimento de um jogo simples, bem como as mais acessíveis devido à imensa variedade de tutoriais disponíveis e à extensa documentação.

Capítulo

4

Problemas Encontrados e Implementação

4.1 Introdução

Neste capítulo irá ser apresentado o desenvolvimento do projeto, alguns problemas encontrados e os métodos utilizados para os resolver e a implementação das mecânicas do jogo.

4.2 Problemas Encontrados

Alguns problemas encontrados no desenvolvimento deste projeto serão abordados neste capítulo bem como a estratégia mais adequada para os resolver tendo em conta a ideia inicial para o jogo.

4.2.1 *Parallax Background*

Inicialmente foi adotada uma estratégia para manter o *background* a repetir-se infinitamente, mas revelou-se falível e ia contra a ideia inicial do *gameplay* pois se o jogador se deslocasse para cima ou para baixo não havia qualquer alteração visual indicando o movimento realizado. Esta estratégia fazia com que o *background* ficasse dependente da câmara de jogo para simular o seu deslocamento infinito, ou seja, no decorrer do jogo seria a câmara a deslocar-se fazendo com que o *background* se repetisse criando a ilusão de que este é infinito. Isto tornou-se inviável pois quando se introduziu o jogador a única maneira de fazer a câmara mover, e consequentemente o *background*, seria

colocar a câmara dependente do jogador, o que causou o problema do movimento vertical não ser visualmente representado.

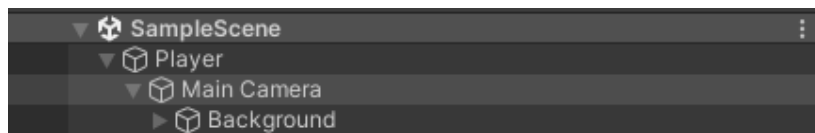


Figura 4.1: O *background* está dependente da câmara de jogo que por sua vez depende do jogador

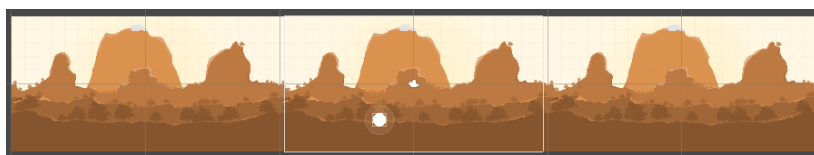


Figura 4.2: O resultado da estratégia implementada

4.2.2 Alinhamento dos Botões no Menu da Seleção de Mapas

No desenvolvimento da *interface* relativa à seleção de mapas surgiu um problema na posição dos botões em relação à imagem de fundo. Este problema surgiu face à ideia inicial de ter o botão relativo a cada mapa, centrado com a sua imagem de fundo representativa, como mostra a figura 4.4

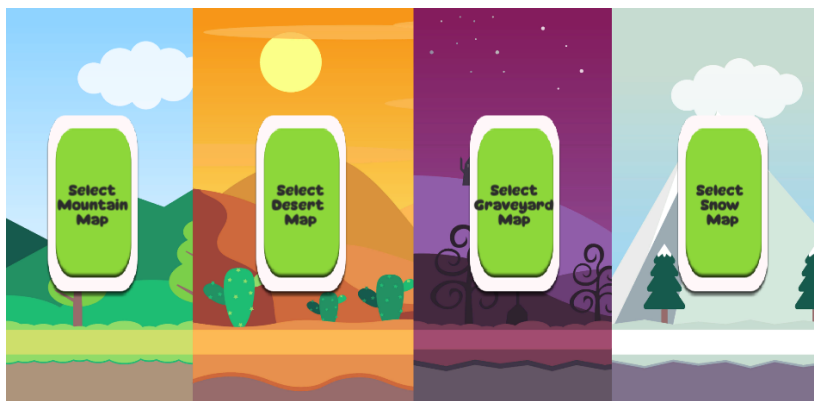


Figura 4.3: Ideia para o menu de seleção dos mapas

Aqui podemos ver que o botão respetivo à seleção do mapa da montanha, por exemplo, se encontra centrado com a imagem de fundo do nível da montanha, tal como os restantes botões. No entanto, ao correr o jogo num ecrã

com maiores dimensões, podemos observar que os botões apenas se encontram centrados no ecrã.



Figura 4.4: Menu de Seleção de Mapas

4.3 Implementação

Nesta secção serão abordadas as estratégias utilizadas para resolver os problemas mencionados na secção 4.2 bem como como foram implementadas as funcionalidades chave do jogo.

4.3.1 *Efeito do Background*

Para resolver o problema em 4.2.1 foi adotada a estratégia de usar um objeto 3D, *quad*, como base para repetir o *background*. Assim usou-se as diferentes camadas do *background* como um material que foi "colado" ao *quad* permitindo assim repetir o *quad* fazendo com que o *background* se repita conforme desejado, pois não fica dependente do movimento da câmara de jogo, visto que ela fica estática, o que permite com que se crie a simulação de deslocamento mesmo que o jogador apenas se mova dentro dos limites da câmara de jogo.

```
void Update ()
{
    bgRenderer.material.mainTextureOffset += new Vector2 (speed *
        Time.deltaTime, 0);
}
```

Excerto de Código 4.1: *script* para a repetição do *background*.

4.3.2 Menu Inicial

O menu inicial consiste numa imagem de fundo, que representa os quatro mapas disponíveis, o nome do jogo, um botão que leva o jogador à seleção do mapa e um botão que fecha o jogo.



Figura 4.5: Menu Inicial

4.3.3 Alinhamento dos botões

Sem sucesso em centrar os botões nas suas imagens de fundo respetivas, foi decidido apenas centrar os botões na tela de jogo através da funcionalidade de ancoragem de elementos de *User Interface* (UI), conforme mostra a figura 4.4.

4.3.4 Tratamento da Pontuação do Jogador

Para o tratamento da pontuação do jogador foi criado um objeto de jogo que fica fora dos limites do jogo a registar a colisão dos balões antes destes se destruírem, incrementando assim a pontuação do jogador, como é mostrado na figura 4.6

Esta abordagem não foi escolhida apenas por ser uma abordagem simples, mas principalmente porque não foi conseguido incrementar a pontuação no momento da destruição de cada balão, pois este é um objeto que não está instanciado, ou seja, não seria possível incrementar a pontuação baseado num objeto que ainda não tenha sido criado.

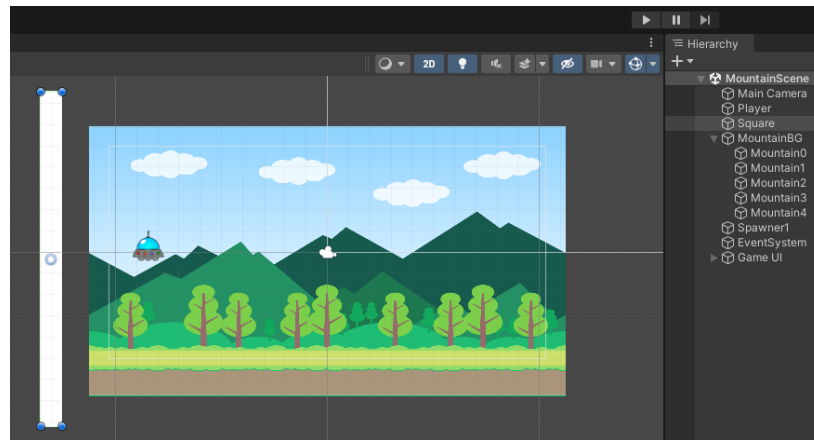


Figura 4.6: Controlo da pontuação do jogador

4.3.5 Sistema de Vidas

O jogador começa sempre o jogo com cinco vidas e no momento de colisão com um balão, ele irá perder uma vida, destruindo também o balão sem atribuir a pontuação relativa à destruição do inimigo.

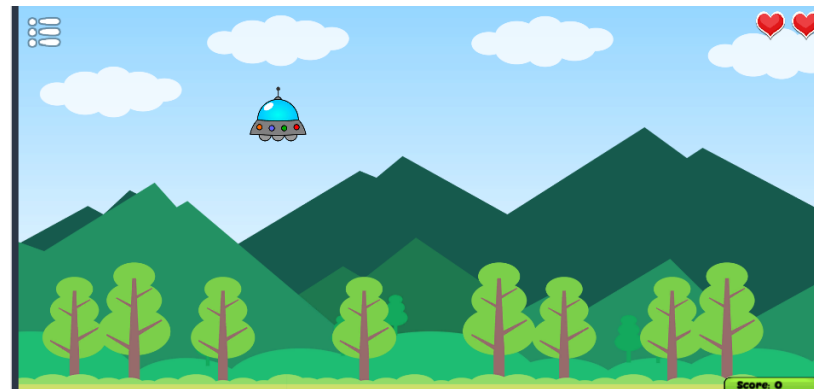


Figura 4.7: Sistema de vidas do jogador

Se o jogador colidir cinco vezes com os balões ele perde, podendo recomeçar o nível de novo ou seleccionar outro mapa e começar a jogar.

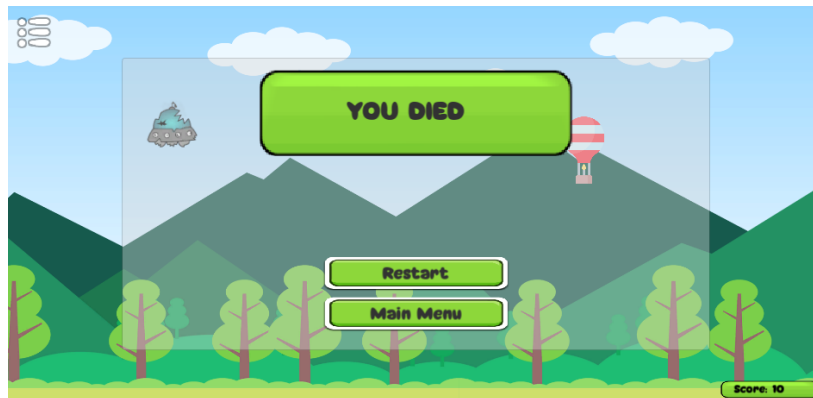


Figura 4.8: Ecrã de fim de jogo

4.4 Conclusões

Em suma, mesmo tendo aparecido alguns problemas na implementação das soluções, estes foram resolvidos de forma ideal com a exceção dos botões do menu da seleção do mapa.

Capítulo

5

Conclusões e Trabalho Futuro

5.1 Conclusões Principais

Com a realização deste projeto consegui aprender os conceitos básicos do desenvolvimento de um jogo 2D, desde a programação dos eventos do jogo, até à escolha dos *assets*.

5.2 Trabalho Futuro

Numa futura abordagem a este projeto existem várias mecânicas que, através do conhecimento adquirido no desenvolvimento deste projeto, podem ser implementadas, como por exemplo:

- Melhorar o sistema de pontuação;
- Diferentes tipos de inimigos, com diferentes formas de atacar o jogador;
- Diferentes modos de dificuldade para os diferentes inimigos;
- Uma UI mais amigável;
- Vários itens que aparecem no decorrer do jogo para oferecer ao jogador algum tipo de recompensa.

Bibliografia

- [1] *Land Sea. Alto's Adventure*, 2020. [Online] <https://www.landandsea.games/altos-adventure>. Último acesso a 31 de Janeiro de 2023.
- [2] *We Sell Apps. Endless Runner 2D*, 2015. [Jogo digital]. *Google Play Store* <https://play.google.com/store/apps/details?id=com.wesellapps.endlessrunner2d>. Último acesso a 31 de Janeiro de 2023.
- [3] Bruno Benedetti. *Ovni 2D*, 2021. [Jogo digital]. *Google Play Store* <https://play.google.com/store/apps/details?id=com.BenedettiGames.Ovni2D&hl=en&gl=US>. Último acesso a 31 de Janeiro de 2023.
- [4] Bill Wagner. *A tour of the C language*, 2022. [Online] <https://learn.microsoft.com/en-us/dotnet/csharp/tour-of-csharp/>. Último acesso a 25 de Janeiro de 2023.
- [5] Lindsay Schardon. *What is Unity? – A Guide for One of the Top Game Engines*, 2023. [Online] <https://gamedevacademy.org/what-is-unity/>. Último acesso a 25 de Janeiro de 2023.
- [6] *Unity Technologies. Install the Unity Hub and Editor*, 2022. [Online]. <https://learn.unity.com/tutorial/install-the-unity-hub-and-editor#>. Último acesso a 31 de Janeiro de 2023.
- [7] *Microsoft. Visual Studio: IDE and Code Editor for Software Developers and Teams*, 2023. [Online] <https://visualstudio.microsoft.com/>. Último acesso a 31 de Janeiro de 2023.