



UNIVERSIDADE FEDERAL DE RORAIMA
CENTRO DE CIÊNCIA E TECNOLOGIA
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO
CURSO DE CIÊNCIA DA COMPUTAÇÃO
COMPUTAÇÃO GRÁFICA
PROFESSOR DOUTOR LUCIANO FERREIRA SILVA

JORGE SIQUEIRA SERRÃO
NATÁLIA RIBEIRO DE ALMADA

PROJETO IV DE COMPUTAÇÃO GRÁFICA
Relatório da Implementação do Projeto Final: Jogo - O Voo de Macau

Boa Vista
Novembro de 2022

1. Introdução

O jogo escolhido para ser desenvolvido neste projeto foi do tipo endless game (conceito em que o jogador é, teoricamente, capaz de marcar pontos infinitamente em uma única fase de jogo. O objetivo é alcançar o máximo de pontos possível, interagindo repetidamente contra o gatilho do jogo, e acaba quando o jogador morre) 2D, onde o intuito é chegar ao Templo de Hórus. Foi utilizada a versão 2020.3.30f1 da plataforma Unity.

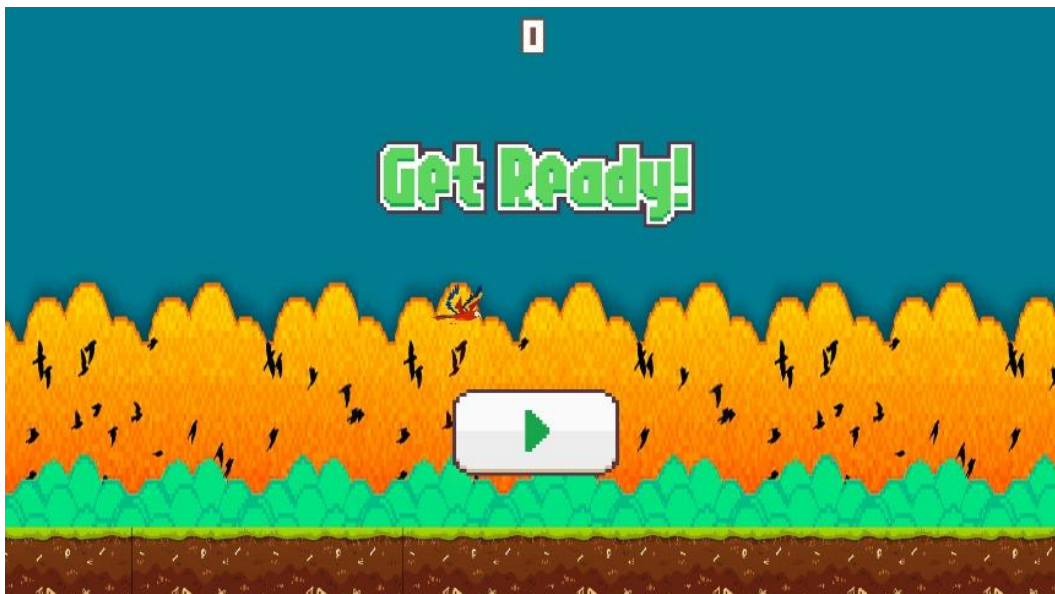
2. Roteiro do Jogo

Nas últimas 30 décadas, grandes áreas florestais foram desmatadas no Brasil e, em 2023, um grupo de caçadores de animais silvestres iniciou uma caçada a aves exóticas, que desencadeou um incêndio catastrófico no Pantanal Matogrossense. Macau, a jovem Arara-Canindé, desesperada por sua vida, para fugir do cruel destino do tráfico, descobre que ao leste, além das fronteiras de seu país, existe um paraíso para todas as aves, regido pelo deus Hórus, e em sua jornada para o Templo de Hórus, encontra como seu obstáculo uma combinação letal de fogo com o que um dia foi sua morada segura. Será que Macal conseguirá cumprir sua jornada e encontrar a paz num templo onde seus ela e seus semelhantes são considerados sagrados? Ou será que ela será mais uma vítima das chamas e da maldade da ganância humana?

3. Detalhes do Jogo

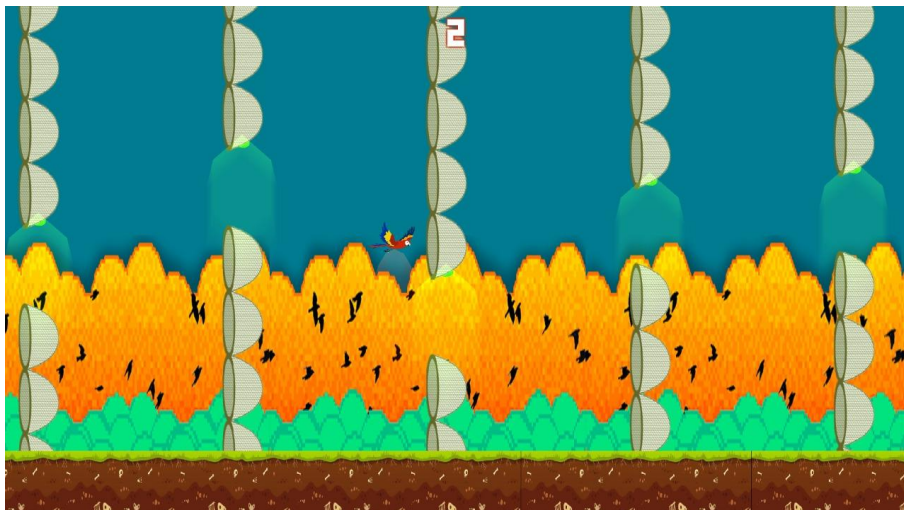
Os elementos principais do jogo são os obstáculos e o jogador, sob o ponto de vista de uma arara. A estratégia é passar pelos obstáculos, que são redes de caça para aves, sem encostar em nenhum deles, se mantendo no ar.

Figura 1: Tela do usuário antes de iniciar o jogo no momento do jogo



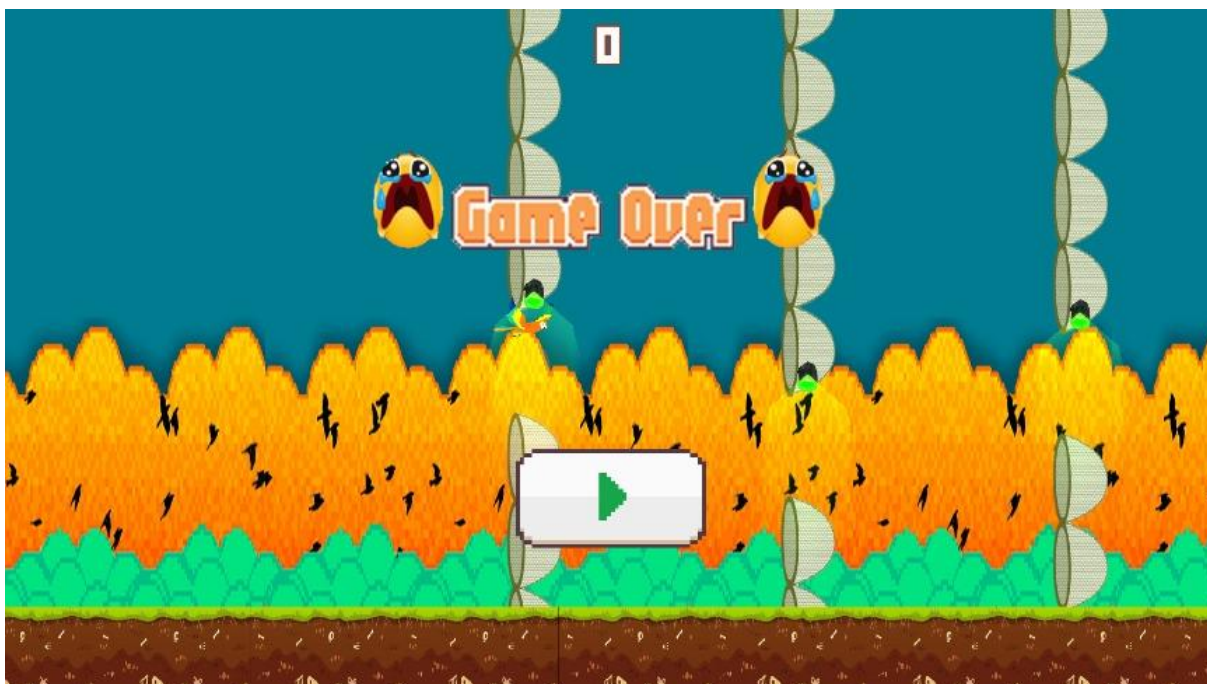
Fonte: Autoria Própria, 2022

Figura 2: Tela do usuário durante a execução do jogo



Fonte: Autoria Própria,2022

Figura 3:Tela quando o personagem principal acerta um obstáculo e morre no jogo.



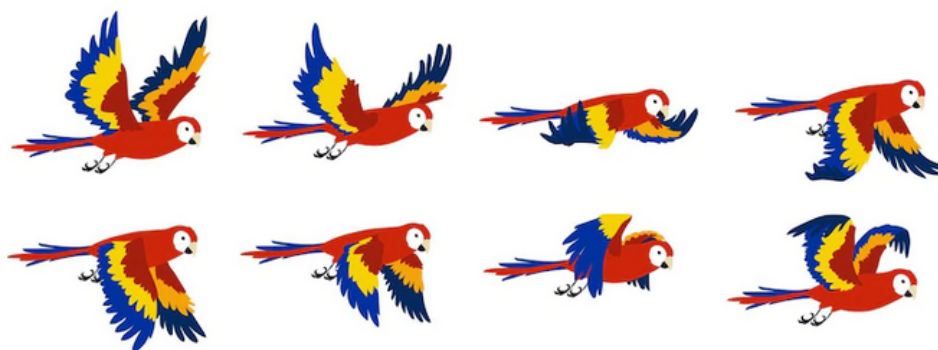
Fonte: Autoria Própria,2022

O jogador, ao perder, vai para uma tela avisando que ele perdeu e com um botão para reiniciar o jogo do início.

Figura 4: Background do jogo



Figura 5: Arara Macau em várias posições



Fonte: Disponívem em: <https://br.freepik.com/vetores-gratis/> ,2022

Figura 6: Rede de captura (obstáculo) rotacionada para demonstração



Fonte: Disponívem em: <https://br.freepik.com/vetores-gratis/> ,2022

4. Ferramentas usadas

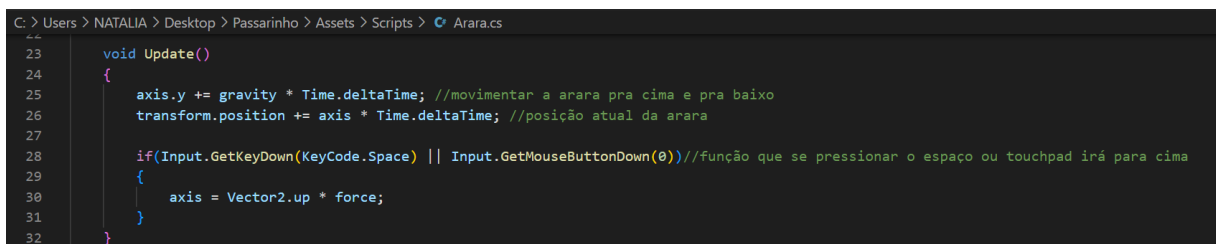
Foi utilizado a plataforma de edição da da Unity na versão 2020.3.30f1.

Os prefabs, sprites e assets foram obtidos através do Site Freepik.com, todos de forma gratuita, e o background de autoria própria.. Para edição dos scripts foi utilizado o Visual Studio Code e a programação toda utilizando a linguagem C#.

5. Principais técnicas de Computação Gráfica aplicadas

As principais técnicas utilizadas foram as de transformação planar, em específico a de translação 2D. Isso foi utilizado para que a ave pudesse se movimentar pelo espaço do jogo. Como pode-se observar no trecho do código da Figura 7, preserva-se o eixo X e a movimentação ocorre no eixo Y:

Figura 7:

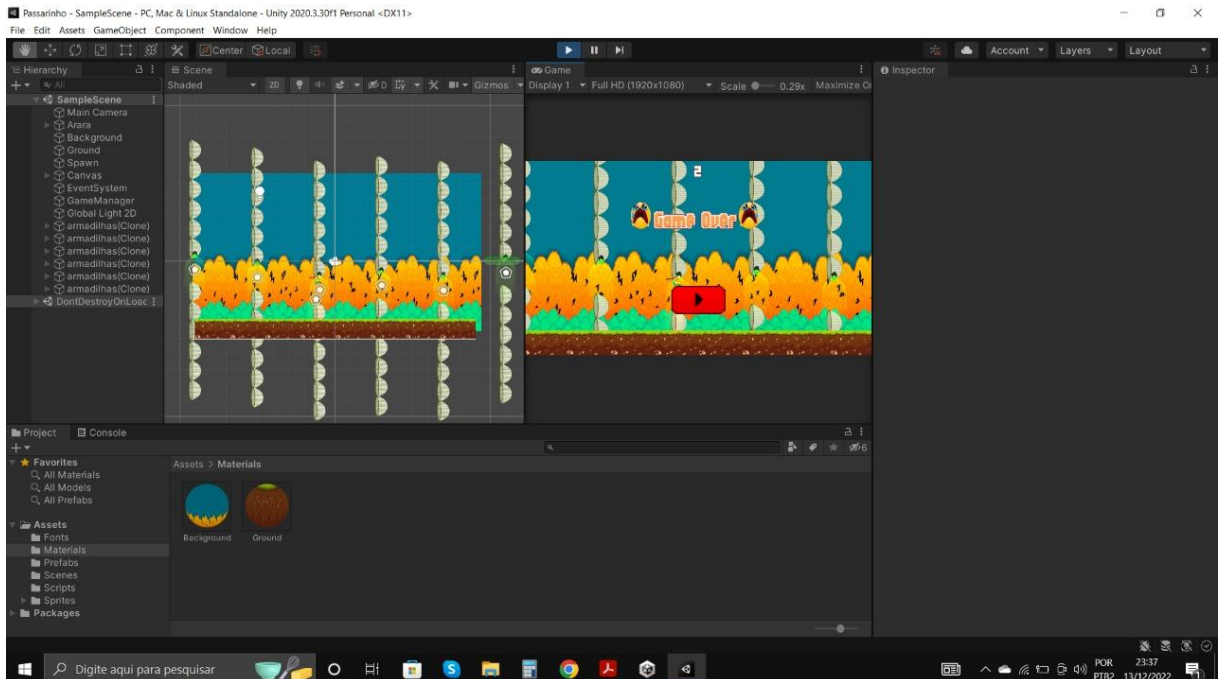


```
C:\Users\NATALIA\Desktop\Passarinho\Assets\Scripts> Arara.cs
23
24 void Update()
25 {
26     axis.y += gravity * Time.deltaTime; //movimentar a arara pra cima e pra baixo
27     transform.position += axis * Time.deltaTime; //posição atual da arara
28
29     if(Input.GetKeyDown(KeyCode.Space) || Input.GetMouseButtonDown(0))//função que se pressionar o espaço ou touchpad irá para cima
30     {
31         axis = Vector2.up * force;
32     }
33 }
```

Também foram utilizados os conceitos de Iluminação Global para todo o cenário através da aplicação de ferramenta nativa da plataforma, e de Iluminação Local na Arara Macau e nos holofotes.

Como pode-se perceber na figura 8, também foi recortada nativamente a figura do obstáculo para caber no viewport com os spawns aleatórios em diferentes posições.

Figura 8: Tela com as imagens posicionadas fora do viewport e saindo apenas o que está dentro do viewport.



6. Conclusões

Apesar do conhecimento em manusear a plataforma Unity e a pouca familiaridade com a linguagem C#, os resultados obtidos foram satisfatórios. É particularmente interessante observar durante a construção de um jogo, como funciona na prática o que é transmitido durante as aulas, mesmo que não sejam aplicados todos os conhecimentos abordados. Ao saber perceber os algoritmos após o estudo de Computação gráfica, é possível inclusive prever possíveis equívocos que podem ser cruciais no desempenho do jogo, e que alguém sem essa base técnica não conseguiria distinguir.

Links:

Os códigos e o jogo estão na mesma pasta no link abaixo:

<https://github.com/nataliaalmada/ProjetoFinalCG-JorgeNat-lia-V-oDeMacau/tree/main>

Para o jogo: Basta baixar executar o arquivo unity de nome “Ararinha.exe” que o jogo irá rodar.