**UNIVERSIDADE ESTADUAL DO OESTE DO PARANÁ - UNIOESTE**

CAMPUS FOZ DO IGUAÇU

CIÊNCIA DA COMPUTAÇÃO

DISCIPLINA DE COMPUTAÇÃO GRÁFICA

DOCENTE: CLAUDIO ROBERTO MARQUETTO MAURICIO

ISABELA PIMENTEL LOEBEL

**TRABALHO 2**

**STUDENT RUNNING**

FOZ DO IGUAÇU

2020

**SUMÁRIO**

[**INTRODUÇÃO**](#_79dti9kylmsq) **3**

[**DESENVOLVIMENTO**](#_c8qercz4cveu) **4**

[**CONCLUSÃO**](#_jxwaf8in33ns) **8**

# INTRODUÇÃO

O jogo trata-se das aventuras de um estudante em busca da formação acadêmica em meio às dificuldades da atualidade, como o COVID-19 e também os obstáculos habituais de um estudante comum, como festas e bebidas.

Esses obstáculos foram representados em jogo através dos ícones de copo de cerveja, globo de festa espelhado e por fim, um ícone de vírus.

Assim como no jogo do dinossauro apresentado quando perde-se a conexão com a Internet no Google Chrome, esse jogo é de modalidade “endless running”, ou seja, o jogo por si não tem fim, o objetivo dele é a maior pontuação obtida pela distância que cada player consegue ir, sendo assim, o foco torna-se ir mais longe sem bater nos obstáculos, já que se bater, a partida é encerrada e o jogo reiniciado.

Desenvolvido, principalmente, na plataforma Unity3D, os scripts responsáveis pelas funcionalidades foram escritos na linguagem C#, já que a própria plataforma tem maior suporte para essa linguagem que outras.

A player principal foi elaborada no aplicativo mobile Autodesk SketchBook, visto que foi feito frame a frame, totalizando 6 frames para representar a animação de corrida, 2 frames parada para a tela inicial, 3 frames pulando e 2 frames caindo para a tela de game over.

# DESENVOLVIMENTO

Inicialmente, foi necessário a criação de 4 objetos “Quad”, cada um para os backgrounds e o chão, sendo 3 backgrounds, o céu, a sombra da cidade e a cidade propriamente. Após criado os objetos, foi preciso a criação do material para aplicar nesses objetos quads, para isso, os materiais criados tiveram que ser configurado com Shader Mobile/Particles/Alpha Blended, de modo que transforme o material para 2D.

Quanto aos materiais, importou-se as imagens que seriam usadas e então foram colocadas na textura do material, por fim, tendo 4 materiais diferentes para dos quads (imagem 1).

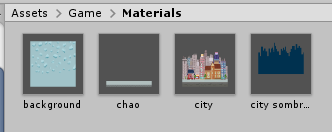


Imagem 1 - Materiais para os backgrounds.

Com o cenário pronto, foi então hora de começar a desenvolver o player principal. Para isso, foi importado os assets referentes ao player, correndo, parado, pulando e caindo. Nele foi adicionado as animações, cada uma separadamente e por fim, foi necessário adicionar o componente de rigibody2D e o box collider 2D, sendo o primeiro para o player possuir massa e gravidade e o segundo para quando encostar nos obstáculos, seja captada sua colisão e para que fique estático junto ao chão.

Após, fez-se necessário a criação de dois objetos vazios, sendo eles um para marcar a posição inicial do player e outro para guardar o script do gerador de obstáculos (imagem 2).

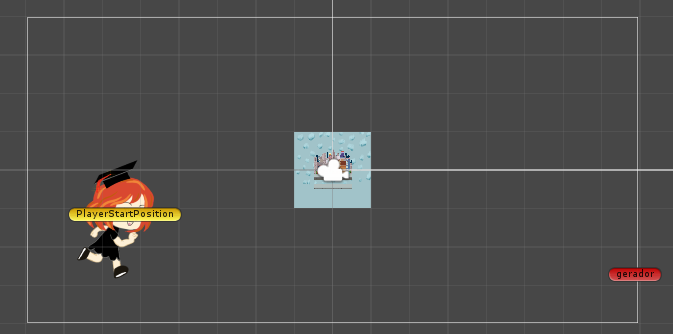


Imagem 2 - Posição inicial em amarelo e gerador em vermelho.

Assim como criado o gerador, foi a vez do coletor, um objeto também vazio mas com box collider 2D, para que houvesse reutilização dos objetos obstáculos, sendo posicionado atrás do player, fora de cena (image 3).

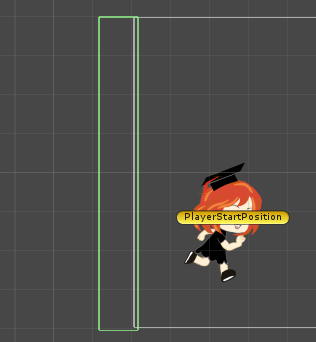


Imagem 3 - Coletor de obstáculos.

Com tantos elementos em tela, para melhor optimização e organização do código foi necessário a criação de um game controller, também sendo um objeto vazio, mas carregado do script responsável da organização dos objetos e agora das futuras telas.

Referente às telas, foram feitas 2 telas, chamadas de GameStart e GameOver, para início e final de jogo, respectivamente. Criadas dentro de um objeto Canvas, foi feito mais dois objetos, mas de diferentes tipos, sendo estes do tipo Text Mesh Pro UGUI, para servirem de marcadores de distância, um nomeado score, onde guarda a contagem da distância e outro scoreLabel, sendo o título “record” (Imagem 4).

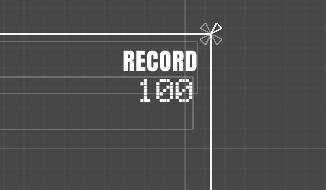


Imagem 4 - Variáveis score e scoreLabel.

Ainda dentro do canvas, criou-se mais dois objetos vazios para as telas anteriormente mencionadas, GameStart e GameOver. Ambas foram adicionadas o component Canvas Group, para poder ser feita a manipulação de ativação e desativação através de scripts, cada um destes objetos têm uma imagem e um botão.

Para os obstáculos foi necessário criar uma pasta chamada Prefabs, já que foram desenvolvidos de forma que sejam prefabs variantes, para que os prefabs filhos possam herdar os componentes e atributos do prefabs pai (imagem 5).

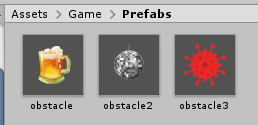


Imagem 5 - Prefabs dos obstáculos.

Foram feitos 10 scripts no total, sendo eles 1 para a escala e ajuste do backgroud, 1 para animação do background, já que é importante ressaltar que nosso objeto player não se move, apenas o cenário, 1 para o game configuration, 1 para o game controller, 1 gameUI, este sendo o responsável pela ativação e desativação das telas GameStart e GameOver quando clicado o botão, 1 para o gerador, o qual continha um vetor de gameOjects nos quais eram atribuídos os prefabs dos obstáculos porém de tamanho triplicado, 1 para configuração de nível, 1 para os obstáculos, 1 para o coletor de obstáculos e por fim, 1 para o player.

A necessidade do tamanho do vetor de obstáculos ser três vezes maior do que a quantidade de obstáculos deve-se à forma que o gerador foi implementado, já que esse cria todos os obstáculos antes do jogo começar e os deixa inativos, aí quando o número randômico é sorteado, o objeto referente àquele número é ativado e mandado para a tela, assim sucessivamente, e caso o gerador escolhesse dois objetos iguais, com o mesmo sprite, não seria possível pois um já estaria em tela e não teria como ser ativado outro.

Por fim, o jogo ficou como na imagem 6, para interagir com o player pode-se apertar a tecla “space” ou a tecla direcional “cima”, ambos trazendo a ação de pulo do personagem como resposta.



Imagem 6 - Tela inicial do jogo.



Imagem 7 - Tela final do jogo.

# CONCLUSÃO

Por fim, esse trabalho foi muito instrutivo, o qual nos apresentou uma nova plataforma, desencadeando a alguns uma nova afinidade e vontade de mais conhecimento.

Quanto ao protótipo do jogo apresentado, os intuitos iniciais o objetivo foi concluído, mas agora com a cabeça mais adentro do projeto, pode-se perceber que há formas de melhorias para esse, como a implementação de novas fases, novos cenários e também a ideia de obstáculos que a player principal não deve apenas fugir mas sim buscar, já por se tratar de uma estudante, a implementação de obstáculos positivos seria uma boa pedida. Caso o trabalho seja aperfeiçoado futuramente, sugere-se o desenvolvimento de obstáculos como livros e notas, para caso ela chegue no final do nível com uma quantidade de livros superior a necessária ela possa prosseguir para próxima fase, caso não, deverá ser refeita a fase atual em busca de mais livros.