



**PODER EXECUTIVO
MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE RORAIMA
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO**

COMPUTAÇÃO GRÁFICA

Trabalho Final

ALUNO:

Jasson Marques Fontoura Júnior – 2019014031

Marcos Vinícius Melo da Silva – 2019017919

**Dezembro de 2023
Boa Vista/Roraima**



**PODER EXECUTIVO
MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE RORAIMA
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO**

COMPUTAÇÃO GRÁFICA

RELATÓRIO DO TRABALHO FINAL

**Dezembro de 2023
Boa Vista/Roraima**

Resumo

Este relatório fala sobre a construção e projeção do jogo “Catch the Orange”, que tem como objetivo servir como projeto final da disciplina de Computação Gráfica.

Conteúdo

1	Introdução	5
2	Roteiro do Jogo	5
3	Detalhes do Jogo	5
3.1	Mecânica	5
3.2	Personagem (Macuxi).....	6
3.3	Laranjas Mágicas	6
3.4	Espinhos	6
3.5	Terreno.....	7
3.6	Checkpoint	7
4	Ferramentas Usadas.....	7
4.1	Scripts.....	8
5	Principais Técnicas de Computação Gráfica Aplicadas	15
6	Conclusões.....	16

1 Introdução

Este relatório descreve a construção do jogo implementado para o trabalho final, mergulhando no universo da criação de jogos em plataforma 2D. Ao longo desse processo, foram enfrentados desafios criativos e técnicos, aprimorando habilidades de programação e design. Este relatório reflete a trajetória envolvente de concepção, implementação e aprendizado ao dar vida a um mundo interativo e envolvente.

2 Roteiro do Jogo

Em "Catch the Orange", os jogadores assumem o papel do destemido Macuxi em uma missão para coletar todas as laranjas mágicas da Amazônia. O cenário de plano de fundo visa lembrar a noite estrelada de uma floresta, apresentando desafios emocionantes e terrenos diversos. Macuxi, dotado de habilidades únicas, precisa enfrentar obstáculos como espinhos sangrentos, plataformas instáveis. Ao longo da jornada, o jogador deve usar a destreza de Macuxi para superar esses desafios e coletar as laranjas para atingir sua meta de obter todas as laranjas possíveis presentes na Amazônia. A narrativa envolvente e os níveis intrincados proporcionam uma experiência cativante, destacando a diversão do jogo.

3 Detalhes do Jogo

3.1 Mecânica

O jogo "Catch the Orange" apresenta uma combinação de mecânicas projetadas para proporcionar a movimentação e controle do personagem. A mobilidade do personagem é ativada por meio das teclas de seta esquerda e direita ou pelas teclas A e D, e para saltar a Barra de Espaço. A movimentação foi implementada de forma fluida e responsiva, permitindo que os jogadores guiem Macuxi pelo mapa com facilidade. As habilidades únicas de Macuxi, como saltos especiais, acrescentam uma dimensão estratégica à movimentação, proporcionando aos jogadores opções criativas para superar obstáculos.

A mecânica central do jogo é a coleta estratégica das laranjas mágicas espalhadas pela floresta. Cada laranja contribui para a pontuação do jogador e pode desbloquear novos níveis, áreas ou habilidades. O sistema de coleta registra a quantidade de laranjas coletadas pelo jogador. Todas as laranjas contêm um sistema de "Hitbox" para identificar a colisão do Macuxi com elas, permitindo a coleta.

Para aumentar o desafio, o jogo incorpora obstáculos desafiadores, como espinhos sangrentos e plataformas instáveis. A detecção de colisão é refinada para garantir respostas imediatas, proporcionando uma sensação de controle ao jogador. O design de níveis é progressivo, introduzindo gradualmente novos desafios para manter a curva de aprendizado equilibrada e a experiência desafiadora ao longo do jogo.

3.2 Personagem (Macuxi)

Macuxi é um aventureiro intrépido, vestido com trajes vibrantes que refletem a riqueza da cultura amazônica. Com sua agilidade e coragem, Macuxi é capaz de saltar e correr de maneiras únicas, e possui uma luz própria para iluminar o caminho. Sua missão é coletar laranjas mágicas para proteger atingir o seu objetivo.



3.3 Laranjas Mágicas

As laranjas mágicas são a fonte de energia vital da Amazônia no jogo. Brilhantes e resplandecentes, cada laranja coletada por Macuxi restaura um pedaço da harmonia na floresta. Os jogadores são desafiados a coletar todas as laranjas, desbloqueando novos níveis.



3.4 Espinhos

Os espinhos representam desafios perigosos que Macuxi precisa evitar. Estrategicamente posicionados ao longo do caminho, testam a habilidade e a precisão dos jogadores. Uma colisão com os espinhos resulta em sua morte, incentivando a cautela e a estratégia para superar essas armadilhas.



3.5 Terreno

Plataformas flutuantes e partes de difícil acesso desafiam os jogadores a explorar e dominar o ambiente. Cada nível apresenta terrenos únicos, proporcionando uma experiência dinâmica e imersiva ao explorar os desafios da floresta tropical.



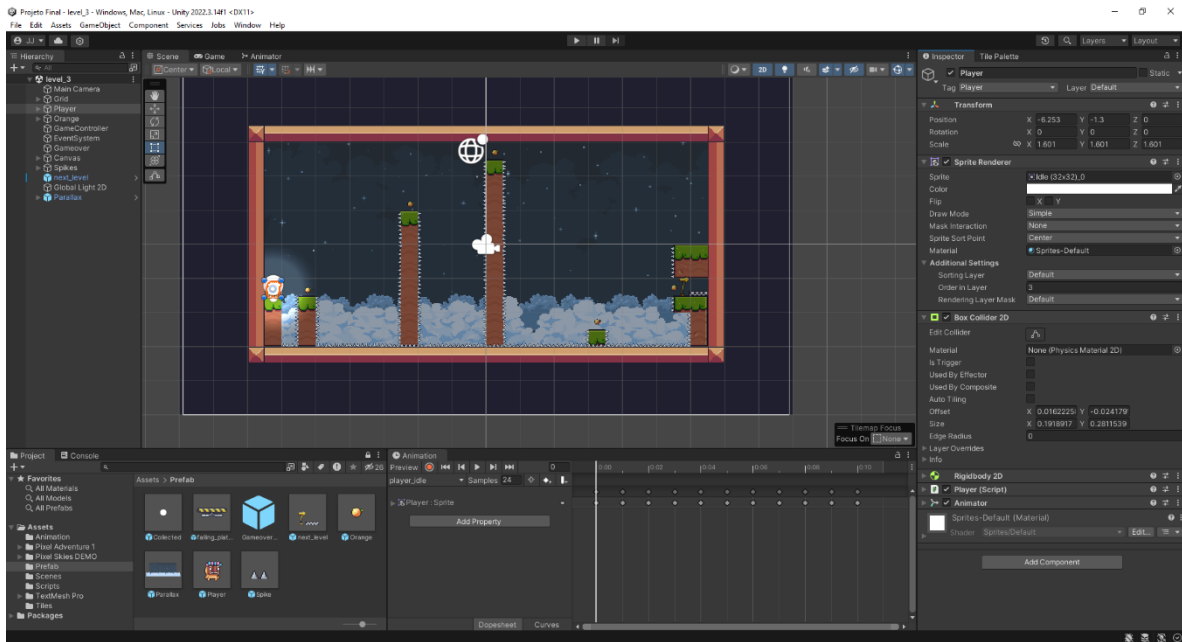
3.6 Checkpoint

Para garantir uma progressão fluida e encorajar a exploração, "Catch the Orange" incorpora um sistema de checkpoint estratégico que permite aos jogadores avançarem para fases subsequentes com maior facilidade. Cada fase do jogo inclui checkpoints automáticos posicionados em locais estratégicos, eles são encontrados ao final de cada nível.



4 Ferramentas Usadas

No desenvolvimento do jogo, foi aproveitado o Unity como principal ferramenta de criação. A versatilidade dessa plataforma facilitou a construção de ambientes interativos, desde os desafios intrincados até os cenários exuberantes da Amazônia. Utilizando a linguagem de programação C#, foram criados scripts que deram vida ao nosso destemido protagonista, Macuxi, conferindo-lhe movimentos fluidos, interações dinâmicas e a capacidade de enfrentar os desafios complexos do jogo.



A arte cativante do jogo foi trazida à vida por meio da técnica de Pixel Animation, adicionando um toque visual único e encantador à jornada de Macuxi. Cada detalhe, desde a vibrante vestimenta do personagem até a luminosidade das laranjas mágicas, foi cuidadosamente animado com pixels, criando uma estética nostálgica e visualmente envolvente para os jogadores.

A combinação dessas ferramentas - Unity para a construção do mundo, C# para a programação inteligente e Pixel Animation para a expressão artística, convergiu para criar uma experiência de jogo coesa e envolvente, onde a magia da Amazônia ganha vida na ponta dos dedos dos jogadores.

4.1 Scripts

Como já falado, criamos scripts em C# para dar vida ao nosso jogo. Aqui, iremos colocar todos eles.

- **Player**

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class Player : MonoBehaviour
{

    public float Speed;
    public float JumpForce;
    public bool isJumping;
```



```

public bool doubleJump;
private Rigidbody2D rig;
private Animator anim;

// Start is called before the first frame update
void Start()
{
    rig = GetComponent<Rigidbody2D>();
    anim = GetComponent<Animator>();
}

// Update is called once per frame
void Update()
{
    Move();
    Jump();
}

void Move()
{
    Vector3 movement = new
Vector3(Input.GetAxis("Horizontal"), 0f, 0f);
    transform.position += movement * Time.deltaTime * Speed;

    if(Input.GetAxis("Horizontal") > 0f)
    {
        anim.SetBool("walk", true);
        transform.eulerAngles = new Vector3(0f, 0f, 0f);
    }

    if(Input.GetAxis("Horizontal") < 0f)
    {
        anim.SetBool("walk", true);
        transform.eulerAngles = new Vector3(0f, 180f, 0f);
    }

    if(Input.GetAxis("Horizontal") == 0f)
    {
        anim.SetBool("walk", false);
    }
}

void Jump()
{
    if(Input.GetButtonDown("Jump"))
    {
        if(!isJumping)

```

```

        {
            rig.AddForce(new Vector2(0f, JumpForce),
ForceMode2D.Impulse);
            doubleJump = true;
            anim.SetBool("jump", true);
        }
        else
        {
            if(doubleJump)
            {
                rig.AddForce(new Vector2(0f, JumpForce),
ForceMode2D.Impulse);
                doubleJump = false;
            }
        }
    }
}

void OnCollisionEnter2D(Collision2D collision)
{
    if(collision.gameObject.layer == 8)
    {
        isJumping = false;
        anim.SetBool("jump", false);
    }

    if(collision.gameObject.tag == "Spike")
    {
        GameController.instance.ShowGameOver();
        Destroy(gameObject);
    }
}

void OnCollisionExit2D(Collision2D collision)
{
    if(collision.gameObject.layer == 8)
    {
        isJumping = true;
    }
}
}

```

- Coleta da Laranja

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;

```

```

public class Orange : MonoBehaviour
{
    private SpriteRenderer sr;
    private CircleCollider2D circle;

    public GameObject collected;
    public int Score;

    // Start is called before the first frame update
    void Start()
    {
        sr = GetComponent<SpriteRenderer>();
        circle = GetComponent<CircleCollider2D>();
    }

    void OnTriggerEnter2D(Collider2D collider)
    {
        if(collider.gameObject.tag == "Player")
        {
            sr.enabled = false;
            circle.enabled = false;
            collected.SetActive(true);

            GameController.instance.totalScore += Score;
            GameController.instance.UpdateScoreText();

            Destroy(gameObject, 0.3f);
        }
    }
}

```

- **GameController** (Ele auxilia no controle de algumas ações)

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;
using UnityEngine.SceneManagement;

public class GameController : MonoBehaviour
{
    public int totalScore;
    public Text scoreText;
}

```

```

public GameObject gameOver;

public static GameController instance;
// Start is called before the first frame update
void Start()
{
    instance = this;
}

public void UpdateScoreText()
{
    scoreText.text = totalScore.ToString();
}

public void ShowGameOver()
{
    gameOver.SetActive(true);
}

public void RestartGame(string lvlname)
{
    SceneManager.LoadScene(lvlname);
}
}

```

- **Queda das Plataformas**

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class FallingPlatform : MonoBehaviour
{
    public float fallingTime;

    private TargetJoint2D target;
    private BoxCollider2D boxColl;
    // Start is called before the first frame update
    void Start()
    {
        target = GetComponent<TargetJoint2D>();
        boxColl = GetComponent<BoxCollider2D>();
    }

    void OnCollisionEnter2D(Collision2D collision)
    {

```

```

    {
        if(collision.gameObject.tag == "Player")
        {
            Invoke("Falling", fallingTime);
        }
    }

    void OnTriggerEnter2D(Collider2D collider)
    {
        if(collider.gameObject.layer == 9)
        {
            Destroy(gameObject);
        }
    }

    void Falling()
    {
        target.enabled = false;
        boxColl.isTrigger = true;
    }
}

```

- **Configuração do CheckPoint**

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

public class next_lvl_point : MonoBehaviour
{
    public string lvlname;
    void OnCollisionEnter2D(Collision2D collision)
    {
        if(collision.gameObject.tag == "Player")
        {
            SceneManager.LoadScene(lvlname);
        }
    }
}

```

- **Parallax Controller**

```

using System.Collections.Generic;
using System.Collections;
using UnityEngine;

```

```

public class ParallaxController : MonoBehaviour
{
    [SerializeField] float moveSpeed;
    [SerializeField] bool scrollLeft;

    float singleTextureWidth;
    // Start is called before the first frame update
    void Start()
    {
        SetupTexture();
        if(scrollLeft) moveSpeed = -moveSpeed;
    }

    void SetupTexture()
    {
        Sprite sprite = GetComponent<SpriteRenderer>().sprite;
        singleTextureWidth = sprite.texture.width /
sprite.pixelsPerUnit;
    }

    void Scroll()
    {
        float delta = moveSpeed * Time.deltaTime;
        transform.position += new Vector3(delta, 0f, 0f);
    }

    void CheckReset()
    {
        if((Mathf.Abs(transform.position.x) - singleTextureWidth)
> 0)
        {
            transform.position = new Vector3(0.0f,
transform.position.y, transform.position.z);
        }
    }

    void Update()
    {
        Scroll();
        CheckReset();
    }
}

```

5 Principais Técnicas de Computação Gráfica Aplicadas

- **Pixel art**

A técnica de Pixel Art foi aplicada para dar vida aos visuais do jogo. Cada elemento, desde o personagem Macuxi até as laranjas mágicas e os desafios do terreno, foi meticulosamente desenhado pixel por pixel. Essa abordagem proporciona uma estética única, repleta de detalhes precisos e uma sensação nostálgica, ressaltando a beleza artesanal da criação.

- **Iluminação**

A iluminação no jogo foi cuidadosamente planejada para criar uma atmosfera noturna e destacar elementos importantes. A utilização de técnicas de iluminação dinâmica e sombras projetadas adiciona profundidade ao ambiente, alterando a aparência dos objetos conforme a fonte de luz se movimenta pois ela vem do nosso personagem, Macuxi, que é controlado pelo jogador. Isso contribui para uma experiência visual mais realista, destacando detalhes e criando uma atmosfera envolvente.

- **Sombreamento**

O sombreamento foi aplicado, por meio de ferramentas do Unity, para aprimorar a representação tridimensional dos objetos no jogo. Técnicas como sombras e mapeamento de sombras foram empregadas para garantir que os elementos interativos, como Macuxi e as laranjas mágicas, projetassem sombras realistas sobre o terreno. Isso adiciona uma camada de autenticidade visual, aprimorando a percepção de profundidade e a coesão estética do mundo do jogo.

- **Projeção de Perspectiva**

A projeção de perspectiva foi fundamental para criar a sensação de profundidade e escala no ambiente. A técnica usada foi a Parallax que garante uma projeção de objetos distantes apareçam menores, enquanto os próximos são representados em escala maior, imitando os princípios visuais da percepção humana. Isso contribui para uma experiência mais realista, permitindo que os jogadores observem o mundo do jogo com uma sensação autêntica de espaço tridimensional.

6 Conclusões

Ao concluir este projeto de desenvolvimento de jogo no Unity, fica evidente que a combinação de ferramentas avançadas, como Unity e C#, junto com técnicas de computação gráfica, como Pixel Art, iluminação, sombreamento e projeção de perspectiva, resultou em uma experiência de jogo envolvente e visualmente cativante. A jornada de Macuxi na Amazônia não apenas proporciona entretenimento, mas também muita experiência para nós que criamos o jogo.

A aplicação cuidadosa de técnicas de computação gráfica, desde a animação pixelada até os efeitos de iluminação dinâmica, contribuiu para a estética única do jogo. A interação fluida entre o personagem, os desafios do terreno e a busca pelas laranjas mágicas cria uma experiência imersiva. O processo de desenvolvimento não apenas aprimorou nossas habilidades técnicas, mas também reforçou a importância da criatividade e da atenção aos detalhes na construção de experiências de jogo memoráveis. Em última análise, esta jornada de criação destacou o potencial emocionante e educativo que os jogos podem oferecer, indo além do entretenimento puro.

7 Fontes e Links

Pixel Adventure 2D:

<https://assetstore.unity.com/packages/2d/characters/pixel-adventure-1-155360>

Pixel Skies 2D:

<https://assetstore.unity.com/packages/2d/environments/pixel-skies-demo-background-pack-226622>

Link para acessar o jogo:

<https://play.unity.com/mg/other/catch-the-orange-demo>