

Politécnico do Porto  
Escola Superior de Media Artes e Design

Grupo 9

9170138 -Carlos Guedes

9170170 – Eloi Leitão

9170243 – Luís Santos

PROJETO: 5 SECONDS RULE

Licenciatura em Tecnologias e Sistemas de Informação Para a Web

## **Sistemas Gráficos**

Orientação: Prof<sup>a</sup>. Doutora Teresa Terroso

Vila do Conde, junho de 2019

## SUMÁRIO

INTRODUÇÃO.....	2
DESENVOLVIMENTO.....	3
CONCEITO E OBJETIVOS .....	3
CODIFICAÇÃO E UTILIZAÇÃO .....	3
REFERÊNCIAS.....	8
CONCLUSÃO .....	9

## INTRODUÇÃO

Este trabalho foi realizado no âmbito da Unidade Curricular de Sistemas Gráficos, onde nos foi solicitado o desenvolvimento de uma aplicação 3D em WebGL ou o Three.js, onde fosse possível avaliar os conhecimentos adquiridos na UC, nomeadamente a interação, visualização e animação. Perante este cenário, consideramos que o melhor a fazer seria dar continuidade ao nosso trabalho já desenvolvido na UC de Modelação e Ambientes 3D e assim criar um jogo original sobre o tema que o mesmo grupo estava a desenvolver na mesma UC.

Desta forma, os objetivos do nosso trabalho foram a exploração de um conceito já existente e o desenvolvimento de um jogo que cumprisse os requisitos pretendidos neste trabalho, respeitando a a ideia até aqui desenvolvida que dá nome ao projeto: “5 Seconds Rule”.

Neste documento vamos relatar o desenvolvimento de todo o projeto assim como explicar toda a nossa ideia, já aqui mencionada. Vamos ainda explicar como funciona a nossa aplicação e revelar quais foram as nossas inspirações.

## DESENVOLVIMENTO

### CONCEITO E OBJETIVOS

Como já foi referido o conceito da nossa aplicação migra da UC de MA3D onde fomos desafiados a criar uma pequena animação, onde teríamos de usar o software e os conteúdos abordados nas aulas. Tendo essa premissa em mãos queríamos desenvolver uma animação com algum conteúdo que dispusesse de algum tipo de narrativa e após algumas ideias, nasceu o “5 Seconds Rule”, baseada na famosa regra dos 5 segundos, que defende que quando deixamos cair algum pedaço de comida no chão temos 5 segundos para a conseguir apanhar antes dos germes e micróbios atacarem. Defendendo assim que estes têm um tempo de reação de 5 segundos. A nossa animação retratou esta teoria. Para esse trabalho tivemos de modelar um jardim/parque e nesse contexto desenvolvemos a nossa ideia através de algumas personagens, alguns pedaços de comida e os nossos germes, objetos esses que fizemos questão de trazer para o nosso jogo.

Tendo em conta a nossa narrativa, tivemos de fazer algumas adaptações para cumprirmos com os objetivos pretendidos neste projeto, assim o nosso protagonista aqui tem de apanhar toda a comida do chão, no caso a bolacha, logo que ela caia, sem tocar nos germes. Sendo então um jogo dinâmico através da deslocação da personagem no cenário controlada pelo utilizador que tem a missão de se proteger dos germes, ao mesmo tempo que tenta apanhar as bolachas.

### CODIFICAÇÃO E UTILIZAÇÃO

Durante o processo de codificação e desenvolvimento da nossa aplicação usamos determinados processos dos quais destacamos alguns e demonstramos aqui, também com o intuito de explicar ao utilizador como deve proceder na utilização da mesma.

Vamos enriquecer estes esclarecimentos com alguns excertos de código, por forma deixar mais claros alguns dos procedimentos por nós utilizados.

Na aplicação, em termos de interação, é possível controlar a posição da personagem com as teclas “A” (a personagem movimenta-se para a esquerda) ou

“D” (a personagem movimenta-se para a direita) e ainda é possível colocar o jogo em pausa usando a tecla “espaço”.

```
function handleKeyDown(event) {  
  
    var char = String.fromCharCode(event.keyCode);  
  
    if (char === "D") {  
        | playerRight = true  
    }  
  
    if (char === "A") {  
        | playerLeft = true  
    }  
  
    if (char === " ") {  
        | paused = !paused  
    }  
}  
  
function handleKeyUp(event) {  
  
    var char = String.fromCharCode(event.keyCode);  
  
    if (char === "D") {  
        | playerRight = false  
    }  
  
    if (char === "A") {  
        | playerLeft = false  
    }  
}
```

Relativamente à criação de todos os objetos do jogo, todos estão a ser construídos em funções separadas e tendo em conta a dimensão e posição do plano. Procurando, desta forma, tornar o código o mais organizado possível. Sendo maior parte deles (player, bolacha, arvores, germes) importados do cinema 4d.

```

function loadFallingObjects() {

    var mtlLoader = new THREE.MTLLoader();
    mtlLoader.load('Elements/Cookie.mtl', function (materials) {
        materials.preload(); // Load a material's resource
        var objLoader = new THREE.OBJLoader();
        objLoader.setMaterials(materials);
        objLoader.load('Elements/Cookie.obj', function (object) { // Load a geometry resource

            // let cookie
            let range = walk.geometry.parameters.height

            cookie = object;
            // cookie.material.color.setRGB(204, 204, 204);
            cookie.position.x = camera.position.x + 500
            cookie.position.y += 500
            cookie.position.z = Math.floor(Math.random() * (range)) + -(range / 2);
            //console.log(cookie.position)

            fallingObjects.push({ object: cookie, type: "good" })
            //console.log(fallingObjects)
            scene.add(cookie)

        });
    });
}

```

Na função “animate()” é feito um update à posição do jogador, da câmara, das vidas, dos scores e dos objetos a cair.

```

if (player.playerMesh) {

    if (playerRight) {

        if (player.playerMesh.position.z <= 400) {
            player.playerMesh.position.z += 10
        }

    }

    if (playerLeft) {

        if (player.playerMesh.position.z > -400) {
            player.playerMesh.position.z -= 10
        }

    }

    camera.position.x += velocity
    player.playerMesh.position.x += velocity

    meshScore.position.x += velocity
    meshLives.position.x += velocity

    if (player.playerMesh.position.x >= 10000) {

        player.playerMesh.position.y -= 10
        //paused = true
        gameOver = true

    }

}

```

Depois são feitas as verificações das colisões que caso seja com uma bolacha( um elemento do array “fallingObjects” de tipo “good”) o score aumenta 200 pontos, ou caso seja com um germe( um elemento do array “fallingObjects” de

tipo “bad”) o score diminui 100 pontos se for superior a 100 ou passa a zero se for inferior a 100 e perde uma vida.

```
function calculateCollisions() {  
  
  for (let i = 0; i < fallingObjects.length; i++) {  
  
    playerBox = new THREE.Box3().setFromObject(player.playerMesh)  
  
    fOBox = new THREE.Box3().setFromObject(fallingObjects[i].object)  
  
    if (fOBox.intersectsBox(playerBox)) {  
  
      var object = fallingObjects[i].object  
  
      if (fallingObjects[i].type === "good") {  
        score += 200  
  
        updateLifeScore()  
  
        fallingObjects.splice(i, 1)  
        scene.remove(object)  
        i--  
        //console.log(score)  
      }  
      else if (fallingObjects[i].type === "bad") {  
        if (score >= 100) { score -= 100 }  
        else {  
          score = 0  
        }  
  
        lifes -= 1  
  
        updateLifeScore()  
        fallingObjects.splice(i, 1)  
        scene.remove(object)  
        i--  
        //console.log(score)  
      }  
    }  
  }  
}
```

Caso as vidas cheguem a zero o jogo acaba e todos os objetos deixam de ser atualizados, parando assim o jogo. O mesmo acontece quando o jogo se encontra em pausa, as posições deixam de ser atualizadas, sendo possível dar continuidade à ação do jogo pressionando novamente na tecla “espaço”.

```
function animate() {

    if (lives == 0) {
        gameOver = true;
    }
    if (gameOver) {
        GameOverText()
    }
    if (!gameOver) {

        frames++

        if (paused) {

            pauseMenu()

        }
    }
}
```

Durante o desenvolvimento do projeto fomos encontrando varias dificuldades mas fomos resolvendo com facilidade através da pesquisa e consultando a professora, no entanto tivemos um problema no final do projeto que não conseguimos resolver, que é a apresentação do “Game Over” quando o jogo acaba e o “Pause” quando o jogo esta em pausa, tentamos resolver este problema de enumeras maneiras mas mesmo assim não conseguimos.

```
function GameOverText() {
    var text = "Game Over"
    var geometry = new THREE.TextGeometry(text, {
        font: fontText,
        size: 20,
        height: 0.5,

        bevelThickness: 1,
        extrudeMaterial: 1
    });
    var material = new THREE.MeshBasicMaterial({ color: 0xffffff })
    gameOverMesh = new THREE.Mesh(geometry, material)
    gameOverMesh.position.x = - (plane.geometry.parameters.width / 2) + 50;
    gameOverMesh.position.y = 20;
    gameOverMesh.position.z = 0;
    //mesh.position.set(-1000,0,-50)
    gameOverMesh.rotateY(-Math.PI / 2)
    scene.add(gameOverMesh)
    console.log(gameOverMesh)
    console.log(fontText)
}
```



## REFERÊNCIAS

Para desenvolver este trabalho, contamos com a ajuda de o Cinema 4D, foi lá que desenhamos as personagens e os objetos que compõem o nosso jogo, e posteriormente foram importados para Three.js. No desenho desses objetos e no restante desenvolvimento deste projeto contamos com algumas inspirações que deixamos aqui referenciadas.

- Modelação de germes

<https://www.youtube.com/watch?v=xbsD6VTFCDY&t=80s>

- Imagem de ajuda na modelação dos Bancos de Jardim

[https://www.outdoor.ie/wp-content/uploads/2018/02/Java\\_Teak\\_Garden\\_Bench\\_A\\_SS\\_1.jpg](https://www.outdoor.ie/wp-content/uploads/2018/02/Java_Teak_Garden_Bench_A_SS_1.jpg)

- Fonte de água

<https://www.youtube.com/watch?v=f3pn94AN9Sw&t=1s>

- Inspiração germes

<https://www.youtube.com/watch?v=92prqeGTmbQ>

- Inspiração plano

<https://www.youtube.com/watch?v=JFH2tp55tzE>

- Inspiração Modelação “Boy”

<https://www.youtube.com/watch?v=MJn4EHHjXi4>

## CONCLUSÃO

Após a finalização deste projeto, somos uma equipa com mais capacidades de visualizar, pensar e executar em ambientes 3D, o que é de facto uma mais valia.

Como é tradição desta licenciatura, este foi mais um projeto exigente que nos permitiu desafiar a nós próprios e que exigiu de nós um raciocínio multidisciplinar em prol dos bons resultados que ambicionamos alcançar.

No âmbito na Unidade Curricular em questão, Sistemas Gráficos, o grande objetivo deste projeto era sermos capazes de criar uma aplicação através do Three.js funcional que permitisse alguma interação por parte do utilizador e esse objetivo foi amplamente alcançado. Ainda assim, há alguns objetivos aos quais nos propusemos trabalhar que não conseguimos alcançar, nomeadamente a nível gráfico e de interface para o utilizador, como já foi descrito no decorrer do presente documento. Isto é, apesar do jogo ter um fim e ser possível fazer pausa no mesmo a qualquer momento, em qualquer das situações existe uma informação no ecrã para o utilizador neste sentido, sendo que em ambos os casos o jogo apenas para. Apesar de termos feito diversas tentativas nunca conseguimos superar esta nossa lacuna.

Em suma, consideramos o resultado positivo, pois para além do já mencionado, a aplicação existe e é com enorme agrado que vemos esta nossa ideia e este nosso conceito ser explorado em mais do que uma frente. Ficam por referir as qualidades que adquirimos em trabalho de grupo, que fomos adquirindo por ser a primeira vez que trabalhamos em conjunto, sendo elas a comunicação e a capacidade de explorar as melhores potencialidades de cada um dos elementos que compõem o nosso grupo.