

Escola de Ciências e Tecnologias  
Departamento de Engenharias  
Licenciatura em Engenharia Informática-  
3º ano, 2º semestre, Turma 3  
Unidade curricular de Computação Gráfica  
Ano letivo 2021/2022

# Relatório do Trabalho Prático

**Discentes:**

Nuno Ferreira al 71473

Ricardo Bessa al 70863

**Docentes:**

Professor Maximino Bessa

Professor Hugo Coelho

## Conteúdo

Resumo.....	3
Introdução à Aplicação gráfica .....	4
Construção de objetos 3D .....	4
Configuração de Câmara .....	5
Configuração de luzes.....	6
Interação com a cena.....	8
Animação .....	9
Auto-Avaliação .....	10
Conclusão: .....	10

## Resumo

Atualmente a área de Computação Gráfica possui um papel importante no ramo da Engenharia Informática. Esta área destina-se principalmente à geração de imagens no geral. Permite-nos a criação de interfaces gráficas para software, como também produzir animações e jogos. Este trabalho prático tem como objetivo a criação de uma aplicação gráfica recorrendo à Biblioteca *WebGL* “three.js”. Os principais requisitos deste trabalho são: a construção de objetos 3D, configuração de câmara, configuração de luzes, interação com a cena, e existência de animação. O *Visual Studio Code* será utilizado como ferramenta chave na elaboração deste projeto. A elaboração deste trabalho irá resultar na aquisição e aprimoramento de competências diversas na área de computação gráfica.

**Palavras-chave:** Computação gráfica, WebGL, Objetos, Animação

## Abstract

Currently an area of Computer Graphics has an important role in the field of Computer Engineering. This area is primarily intended for general imaging. It allows us to create graphical interfaces for software, as well as produce animations and games. This practical work aims to create a graphic application using the WebGL Library “three.js”. The main requirements of this work are: the construction of 3D objects, camera configuration, lighting configuration, interaction with the scene, and existence of animation. Visual Studio Code was used as a construction tool for this project. The elaboration of this work will result in the acquisition and improvement of several capacities in the area of computer graphics.

**Keywords:** Computer Graphics, WebGL, Objects, Animation

## Introdução à Aplicação gráfica

### Construção de objetos 3D

Para a realização deste trabalho pratico, optamos pela realização de um jogo baseado no clássico “*Space shooter - Galaxy attack*”, pois ambos sempre gostamos deste jogo, e também tem alguma complexidade na parte funcional do jogo.

Começando por explicar a abordagem do primeiro requisito (**Construção de objetos 3D**), que era a utilização de primitivas básicas three.js. Para este trabalho utilizamos as primitivas em duas situações, na criação da peça mais importante do cenário que é o ambiente em si com as estrelas (de animação apenas) a virem na nossa direção para dar um efeito de movimento na Galáxia, e também usei na criação da Nave espacial e do asteroide que colide com ela mesmo.

Para a criação do fundo, para gerar as estrelas, recorremos ao uso do THREE.Vector3 para gerar pontos de forma aleatória onde cada ponto representa a posição de uma estrela. Importamos uma textura utilizando o THREE.TextureLoader, para aplicar nas estrelas que se encontram armazenadas no array. São geradas 2000 estrelas de cada vez para o fundo. Foi criada uma função animate para que fosse possível aplicar movimento das estrelas.



*Figura 1- Elemento 3D*

Na criação de meteoros utilizou-se uma primitiva para gerar um objeto com uma forma idêntica a um meteoro, um icosaedro, sendo aplicado de seguida uma textura de meteoro. Aplicamos uma rotação sobre os eixos x e y.



*Figura 2 - Meteoro*

Criamos um objeto complexo que representa saturno, para isso tivemos de recorrer ao uso de primitivas na construção da esfera do planeta, do anel, e da bandeira colocada. Para mover o objeto complexo podemos utilizar as teclas das setas do teclado (comandos no ficheiro README.txt na pasta do projeto).



*Figura 3 - Objeto complexo*

## **Configuração de Câmara**

Para o segundo requisito, que consistia na utilização de uma câmara em perspetiva e uma ortográfica, usamos apenas uma variável “camara” que ia variando à medida que o utilizador pressionasse num botão HTML que implementamos no index.html (alternar camara) e que permitisse ao utilizador ter uma experiência de jogo normal, apos a sua alteração.



*Figura 4 - Câmera em perspectiva*



*Figura 5 - Câmera ortográfica*

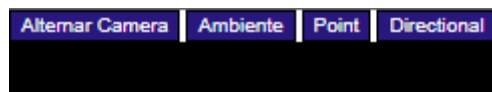
## **Configuração de luzes**

Para a configuração das luzes optamos por uma abordagem semelhante à das câmaras (Fig. 6), em que temos três botões para ligar/desligar as três luzes que implementamos (AmbientLight, PointLight e DirectionalLight). Cada luz que escolhemos usar, fomos buscar ao THREE.JS, e optamos por modificar a cor nos parâmetros da função dada, para assim conseguirmos distinguir qual das luzes está a ser ligada ou desligada.

**AmbientLight** – 0xffffff que corresponde ao **branco**.

**PointLight** – 0xff0000 que corresponde ao **vermelho**.

**DirectionalLight** – 00ff00 que corresponde ao **verde**.



*Figura 6 - Botões camara e luzes*



*Figura 7 - Com todas as luzes ligadas*



*Figura 8 - Sem qualquer luz ligada*



*Figura 9- Ambient Light*



*Figura 10 -Point Light*



*Figura 11-Directional Light*

## **Interação com a cena**

Para interagir com a cena são utilizadas as teclas **A** e **D** para mover a Nave para se desviar dos asteroides que vêm em direção a si, e ainda o **espaço**, para disparar contra os asteroides. Se colocarmos a nave numa posição em que aponte para o asteroide, e de seguida faze-la disparar na direção dele, o asteroide irá desaparecer. Outras funcionalidades para otimizar o nosso jogo foi: fazer com que o tiro desaparecesse quando este atinge uma distância máxima definida; Não é possível disparar um novo tiro enquanto o tiro ainda estiver na cena; O asteroide quando alcança a posição da nave é removido e de seguida é gerado um novo.





*Figura 12- Utilização do espaço (câmara em perspectiva)*



*Figura 14- Asteroide perto da nave*



*Figura 13- Utilização do espaço (câmara ortográfica)*

## Animação

Para completar os requisitos temos 3 animações presentes no jogo.

A primeira, utilizamos um objeto animado, **a nave espacial**, que já vem com uma animação por si e com uma textura a nossa escolha.

O segundo escolhemos um **alien**, que importamos tanto o modelo 3D como as texturas de um site, onde se pode ver ele a mexer os braços como se tivesse sido projetado de um cometa em destruição, mas por problemas de eficiência decidimos não o gerar na cena.

Após terem a animação criada utilizamos o método de importação dado no **tutorial 7** para importar o objeto e fazer com que animação corresse.

Pretendíamos criar uma animação que consistia em ao disparar contra o asteroide, fazer com que este desaparecesse e como consequência fizemos aparecer um novo numa nova posição.

**Após a apresentação do trabalho e do feedback recebido pelo professor conseguimos melhorar e implementar:**

- Detecção da colisão do tiro com o asteroide.
- Existência de um só tiro, quando este atinge uma posição definida por nós(máxima) em relação ao eixo Z, este é removido e será então permitido disparar de novo.
- Redução do nº de estrelas de fundo da cena para melhoria da eficiência.
- Asteroide move-se em direção à nave.
- Quando o asteroide atinge a posição z da nave é removido e é gerado um novo numa posição aleatória.
- É possível utilizar as setas do teclado para interagir com o objeto complexo.

## **Auto-Avaliação**

Após uma reflexão séria, decidimos autoavaliar com 18, pois conseguimos cumprir todos os objetivos propostos no Protocolo do Trabalho Prático, e ao mesmo tempo aplicar uma complexidade na nossa opinião boa, e com uma originalidade média também, pois não se fica apenas por um cenário como muitos trabalhos observados nas aulas, mas sim a criação de um jogo que por norma tem de conter uma complexidade elevada. Foi um trabalho que nos despertou um grande interesse pela área de computação gráfica.

## **Conclusão:**

A fim de desenvolver aptidões e competências fundamentais à área de Computação Gráfica, foi-nos pedido que desenvolvêssemos uma aplicação gráfica.

Com este trabalho aprendemos a usar **JavaScript** para desenvolver um cenário gráfico e também a interagir com este mesmo. Aprendemos a usar a Biblioteca WebGL “**three.js**” e como esta nos pode dar jeito em futuros projetos.