**Animador de cenas 3D**

Pedro Gonçalves 88859 Pedro Silva 89228

***Resumo* – Animador de um Sistema planetário, mais concretamente do Sistema Solar (versão pós 2006), em 3 dimensões, onde se pode observar a rotação e translação de todos os planetas, podendo escolher ver o Sistema Solar à escala real ou com uma escala adaptada para facilitar a visualização de todos os planetas. O utilizador pode comparar o tamanho dos planetas e do Sol com a escala real. Também é possível ver uma representação 3D de um asteroide.**

***Abstract* – Planetary system animator, more specifically the Solar System (after 2006 version), in 3 dimensions, where you can observe the rotation and translation of all the planets, choosing to watch the Solar System in the real scale or with an alternative scale that makes it easier to see all the planets. The use can compare the size of the planets and the Sun with the real scale. It is also possible to see the 3D representation of an asteroid.**

I. Introdução

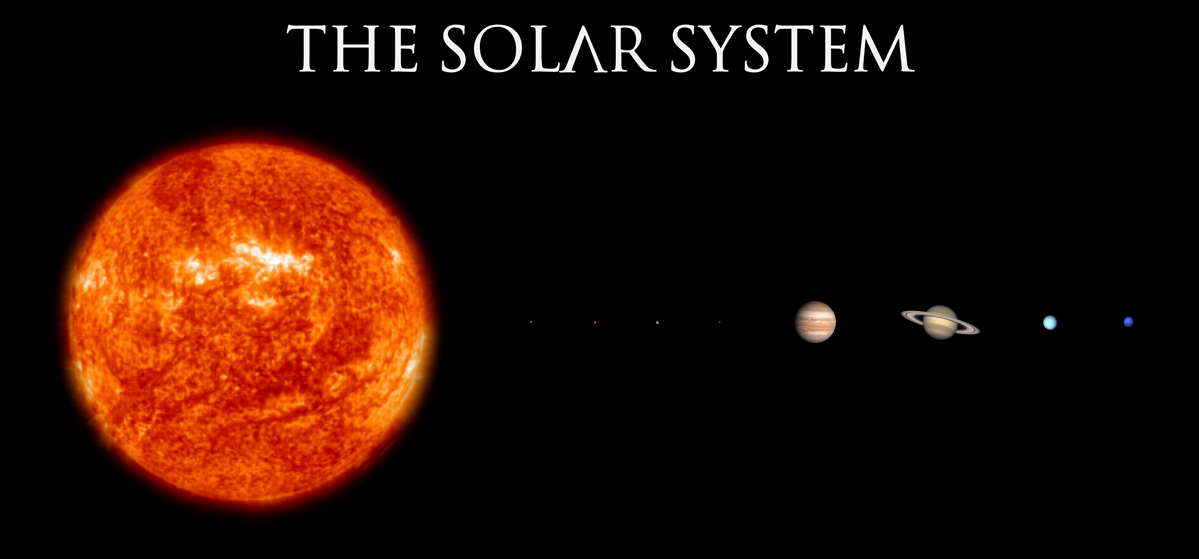
Neste relatório, vamos abordar as várias tecnologias que foram desenvolvidas em WebGL e que foram implementadas neste projeto. Desde a criação dos planetas através de vertices, à sua normalização e aos seus períodos de rotação e translação.

II. Criação do Sol e dos planetas

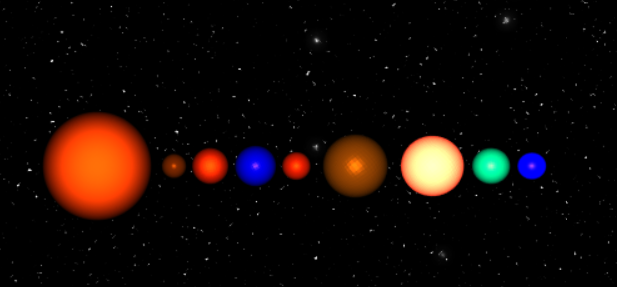
O primeiro objetivo deste projeto era a criação de esferas que se comparassem ao Sol e aos planetas do Sistema Solar.

Para isso adaptamos algum código utilizado nas aulas práticas, onde é possível criar um cubo em 3 dimensões através de um array com a posição dos seus vértices. Após a criação do cubo, desenvolvemos um modelo representando a esfera de raio unitário e centrada na origem deslocando cada um dos seus vértices para a superfície da esfera de raio unitário e centrada na origem usando a função *midPointRefinement( coordsArray, recursionDepth )* que recebe as coordenadas dos vértices de um triângulo e cria uma determinada quantidade de novos triângulos dependendo do seu nível de recursividade, a funçao *normalize( v )* que permite normalizar o vetor v e a função *computeVertexNormals( coordsArray, normalsArray )* que calcula os vetores normais para cada triângulo.

Todos os planetas foram adicionados um a um no ficheiro “sceneModels.js”, no qual definimos os vários atributos, entre estes a sua posição relativa no espaço, a sua escala em relação à janela de *clipping*, os seus ângulos de rotação, controlos de animação, entre outros. Para criar e alinhar todos os planetas utilizamos uma escala irrealista, devido ao facto de o Sol ser muito maior que todos os planetas sólidos, o que iria dificultar a visualização desses mesmos planetas. Todos os planetas ficaram alinhados por ordem, com um espaçamento igual entre si.



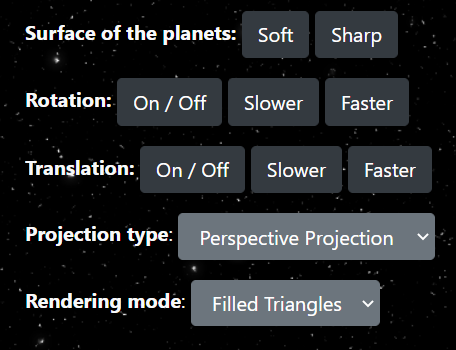
*Figura 1 - Escala real do Sistema Solar*



*Figura 2 – Escala utilizada para a representação do Sistema Solar*

III. Background e interface

Para o fundo da nossa aplicação usámos uma imagem a simular o universo, embutida no body e no *canvas* do documento, desta forma tornando mais realista toda a simulação. Também escolhemos botões e *dropdowns* que ficassem bem com o fundo usado.

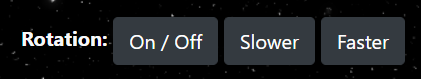
**

*Figura 3 – Botões para manipulação do comportamento dos planetas*

IV. Rotação dos planetas e do Sol

Em relação à rotação dos planetas e do sol, utilizámos as variáveis *isRotZZOn*, que define se a rotação está ou não ativa no planeta definido, *rotZZSpeed* que manipula a velocidade de rotação, visto que todos os planetas têm uma velocidade de rotação diferente e por fim *rotZZDir* que permite escolher a direção da rotação. No caso do Sistema Solar, o único planeta que roda numa direção diferente dos restantes é Vénus, que roda na direção dos ponteiros do relógio.

O utilizador da aplicação pode alterar estes valores através dos botões “On / Off”, para desligar a rotação dos planetas (e do Sol), “Slower” e “Faster” para desacelerar a acelerar a velocidade de rotação dos mesmos, respetivamente.

**

*Figura 4 – Botões de “On / Off”, “Slower” e “Faster”, relativamente à rotação dos planetas*

V. Translação dos planetas

Na parte da translação, simulamos os períodos de translação de todos os planetas em torno do Sol. Nos exemplos das aulas, aplicávamos translação uniforme a toda a cena, já na nossa simulação tivemos que utilizar uma translação com velocidades diferentes para cada planeta. Para esse efeito tivemos que criar um array de ângulos, inicialmente com todos os elementos a 0.0, onde cada um desses ângulos corresponde a cada planeta. Os valores desses ângulos atualizam de forma diferente, de modo a simular a translação com escala real entre todos os planetas, como por exemplo, 1 volta do planeta Marte em torno do Sol equivale a aproximadamente 7,8 voltas do planeta Mercúrio em torno do Sol.

É possível parar e recomeçar a translação com o botão “*On/Off*”, que altera o valor da variável *globalRotationZZ\_ON,* é também possível fazer com que a translação de todos os planetas seja mais rápida, com o botão “*Faster*”, ou que seja mais lenta, com o botão “*Slower*”.



*Figura 5 – Botões de “On / Off”, “Slower” e “Faster”, relativamente à translação dos planetas*

VI. Iluminação dos planetas

Na iluminação dos planetas utilizamos várias *lightsources*. Inicialmente pretendíamos dar uma cor a cada planeta, mas optamos por não lhes dar cor e apenas utilizar a incidência de *lightsources.* Definimos então vários parâmetros para cada planeta, como *kAmbi*, *kDiff*, *kSpec* e *nPhong*, com valores diferenciados para cada planeta de modo a simular cores diferentes. Depois de definidas as cores, tivemos que definir a posição e a intensidade, com os atributos *position*, *intensity* e *ambIntensity.* Definimos uma posição inicial lateral, com a coordenada x negativa, para simular a incidência do Sol nos planetas. Com isto surgiu um problema, que foi como simular a rotação da lightsource nos planetas quando estes efetuam o movimento de translação em torno do Sol.

References

Use the Style “referencia” to the references. Example:

[1] M. T. Rose and K. McCloghrie, "Structure and Identification of Management Information for TCP/IP-based Internets", RFC 1155, DDN Network Information Center, SRI International, May 1990.

This file is available by anonymous FTP in:

greco.inesca.pt: ~/pub/docs/dos/revdetua.doc

$ftp greco.inesca.pt

user: anonymous

passwd: <none>

cd pub/docs/dos

bin

get revdetua.doc

bye

$

Any comments to enhance this format can be send to jlo@inesca.pt.

This document was written in Word for Windows 2.0.

94.6.13 JLO