**ASTEROIDS**

**Grupo 6**

9180120 – Ana Moreira

9180591 – Tiago Areias

**Licenciatura em Tecnologias e Sistemas de Informação para a Web**

Animação Gráfica

Orientação: Prof. Teresa Terroso

***SUMÁRIO***

[INTRODUÇÃO 3](#_Toc31060359)

[DESENVOLVIMENTO DO PROJETO 4](#_Toc31060360)

[DIFICULDADES ENCONTRADAS 9](#_Toc31060361)

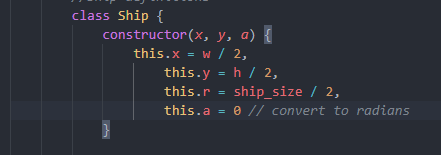
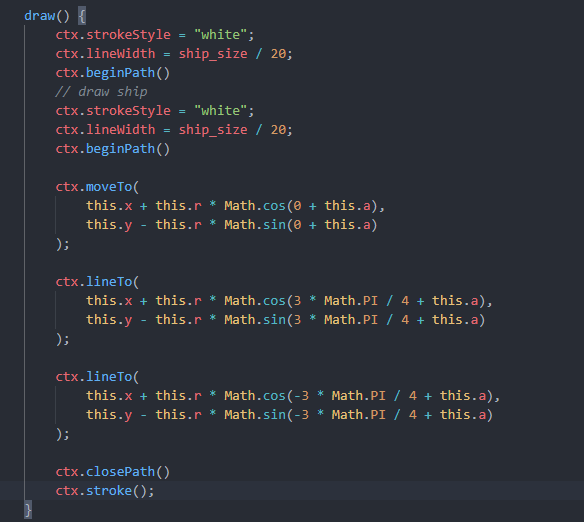
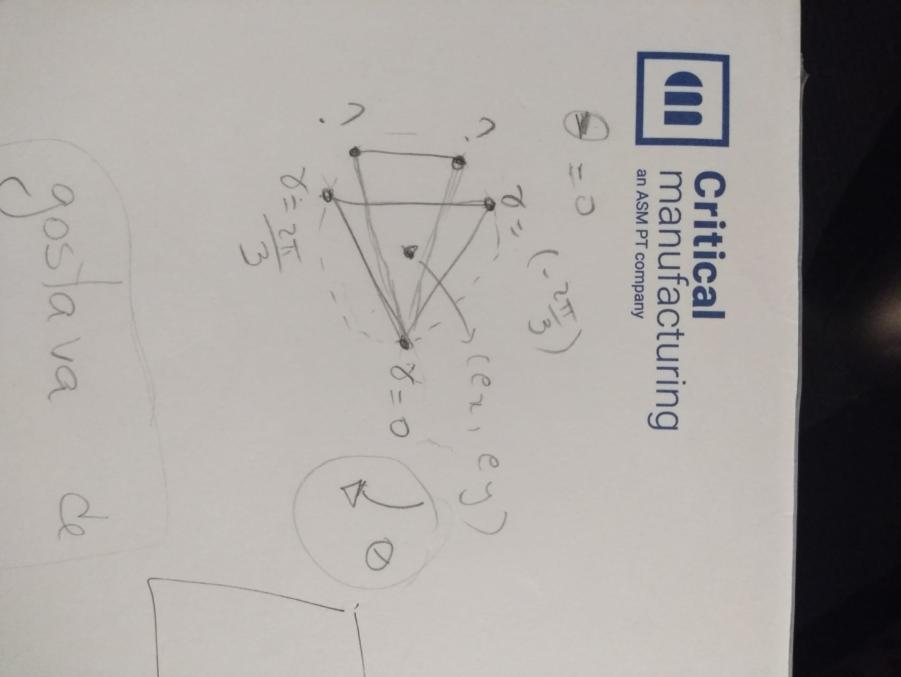
[CONCLUSÃO 10](#_Toc31060362)

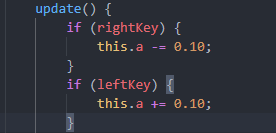
# **INTRODUÇÃO**

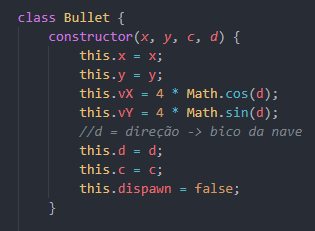
Este trabalho foi desenvolvido no âmbito da UC de Animação Gráfica a fim de colocar em prática os conhecimentos obtidos até à data de elaboração do trabalho. O jogo escolhido para o nosso projeto foi os Asteroids. Decidimos escolher este tema pois ambos achamos que é um jogo desafiante a nível lógico que nos iria obrigar a ultrapassar desafios ao longo do projeto.

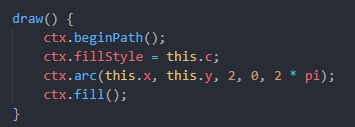
O jogo Asteroids é um jogo que consiste em controlar uma nave com as arrow-keys e evitar o contacto com os asteroids podendo destruí-los com balas (spacebar). Caso o jogador consiga destruir um asteroide, são lhe adicionados pontos. O jogador tem 3 vidas, perdendo-as sempre que é atingido por um asteroide, caso perca todas as vidas é game over e o jogo finaliza.

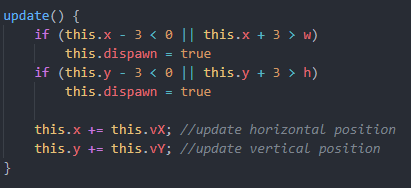
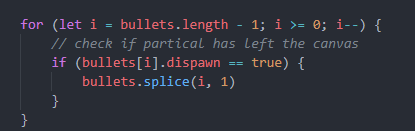
# **DESENVOLVIMENTO DO PROJETO**

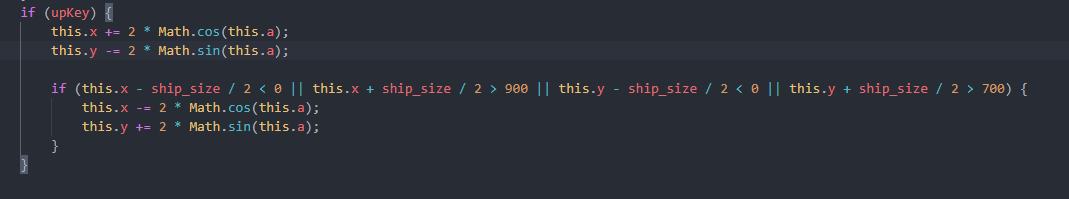
Para o desenvolvimento do jogo começamos por pesquisar várias versões do Asteroids. Após pesquisa, conseguimos chegar a um acordo da estética que queríamos para começarmos o código. Primeiramente criamos uma class ship que tem os seguintes parâmetros: O **X** e o **Y** são as coordenadas, o **R** é o raio e o **A** é o ângulo que controla a rotação da nave. Após a criação da classe atribuímos-lhe uma função para a nave ser desenhada, como não queríamos um triângulo equilátero pois tornaria difícil a perceção do bico da nave, decidimos usar um triângulo isósceles (lógica mostrada na imagem):

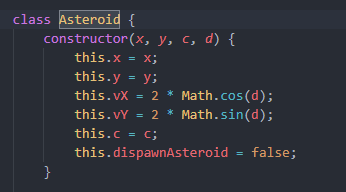
Seguidamente atribuímos uma função update para controlar a rotação da nave:

O próximo passo foi criar uma classe Bullet com os seguintes parâmetros: O **X** e o **Y** são as coordenadas, o **vX** e o **vY** são as velocidades horizontal e vertical, o **D** é a direção (controlada pelo bico da nave) o **C** é a cor e o **dispawn** é o que controla a existência das balas.

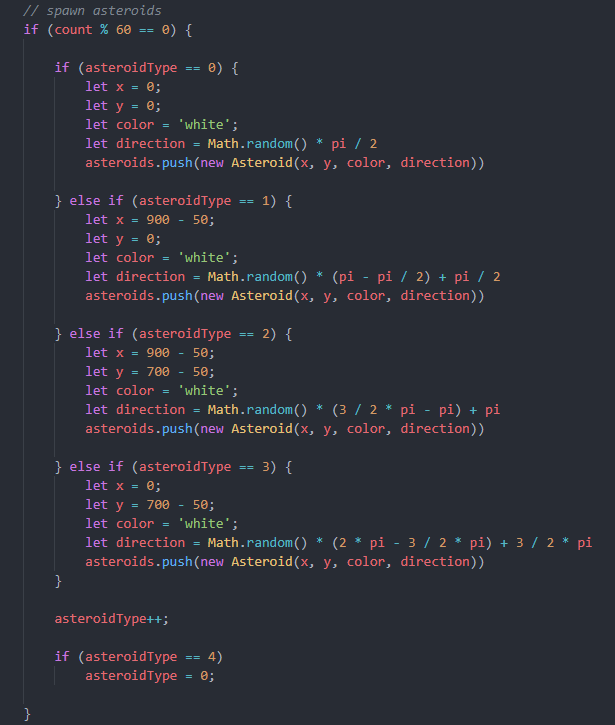
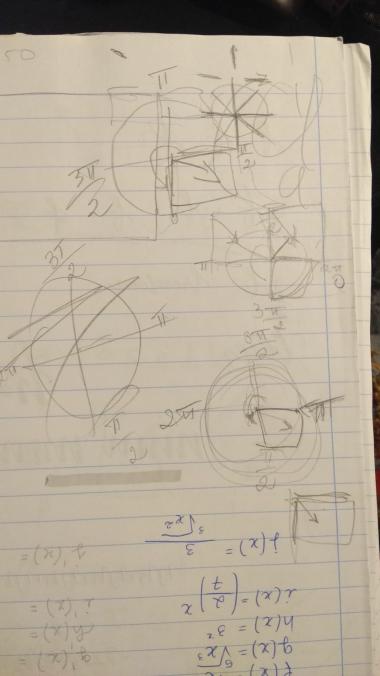
https://i.gyazo.com/dc0bf6bf14636d9af78b842bf716dd67.pngApós a criação da classe atribuímos-lhe uma função **draw()** para as balas serem desenhadas e criamos um array para armazenar as balas disparadas pelo utilizador:

Decidimos seguir para a função **update()** das balas que as remove do array quando estas saem do canvas:

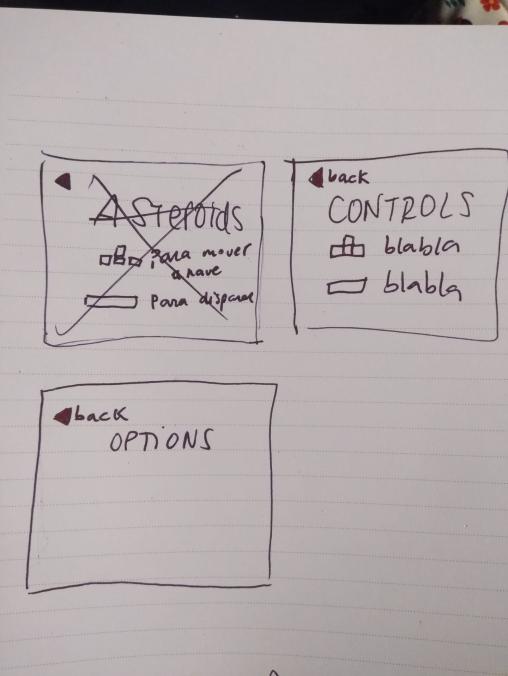
Seguidamente passamos para a movimentação da nave limitando-a para que não saia do canvas:

 Fizemos a classe asteroid com os seguintes parâmetros:

O **X** e o **Y** são as coordenadas, o **vX** e o **vY** são as velocidades horizontal e vertical, o **C** é a cor e o **dispawnAsteroid** é o que controla a existência dos asteroides. Após a criação da classe decidimos fazer as funções **draw()** e **update()** dos asteroides.

Decidimos que os asteroides iam ter quatro pontos de partida sendo estes os quatro cantos do canvas. Aqui está um esboço que fizemos como auxílio para os cálculos da direção dos asteroides seguido do código:

**MENU**

Uma imagem com texto

Descrição gerada automaticamente Inicialmente pesquisamos vários jogos para inspiração de menus e para tentarmos chegar a um acordo de como gostariamos que fosse a estética do nosso menu. Após pesquisas fizemos alguns esboços que nos levaram à ideia principal: ter uma animação de fundo com vários asteroides e ter uma animação no título, inicialmente gostaríamos de ter uma opção de controlos (que foi exequível) e uma opção de definições. Para o título principal tinhamos a ideia de animar cada letra para dar um efeito de cascata com um fade. Decidimos que a cor inicial do título seria transparente, apenas com o contorno da palavra e, no fim da animação, o título estaria todo branco, como mostra na imagem seguinte:

Uma imagem com texto

Descrição gerada automaticamenteApós a animação do título fizemos um background onde usamos asteroides desenhados em playground (<https://editor.method.ac/>) mas manipulamos manualmente as suas coordenadas e animações CSS (blink, opacidade). Decidimos fazer um menu de controlos para informar o jogador quais as teclas do jogo, como mostra na imagem seguinte:

# **DIFICULDADES ENCONTRADAS**

Ao desenvolver o projeto encontramos várias dificuldades, como por exemplo:

- Desenhar a nave – Conseguimos ultrapassar o problema que inicialmente nos ocupou algum tempo. Solucionámo-lo com a ajuda da professora Teresa Terroso numa das aulas de Animação Gráfica;

- Direção balas e ponto inicial das mesmas – Após várias pesquisas e pedidos de ajuda sobre o assunto conseguimos solucionar o problema da direção inicial das balas;

- Movimento da nave – Solucionado com pesquisa e ajuda da professora;

- Colisões (não superado) – Um dos maiores desafios do jogo eram as colisões que, infelizmente, não conseguimos ultrapassar.

# **CONCLUSÃO**

Neste trabalho conseguimos pôr em prática todos os assuntos abordados na UC e concluímos que a animação em Javascript/CSS (jogo/menu) pode ser usada em vários tipos de trabalho e para diferentes objetivos.

Cumprimos grande parte dos objetivos propostos, porém, não conseguimos acabar o jogo (a parte das colisões) pois, infelizmente, percebemos que a primeira entrega do projeto era uma entrega intermédia e não a entrega final dos Asteroids.

Ao longo do projeto não sentimos dificuldade em trabalhar em conjunto e sentimos que nos entre ajudávamos. Todas as semanas tentamos sempre ter evolução no projeto e em todas as aulas tentamos esclarecer várias questões que iriam surgindo.

Em suma, achamos que foi um projeto positivo e gostaríamos de melhorar a questão das colisões pois sentimos que iriamos conseguir ultrapassar os vários obstáculos que iriam aparecer ao longo do tempo.