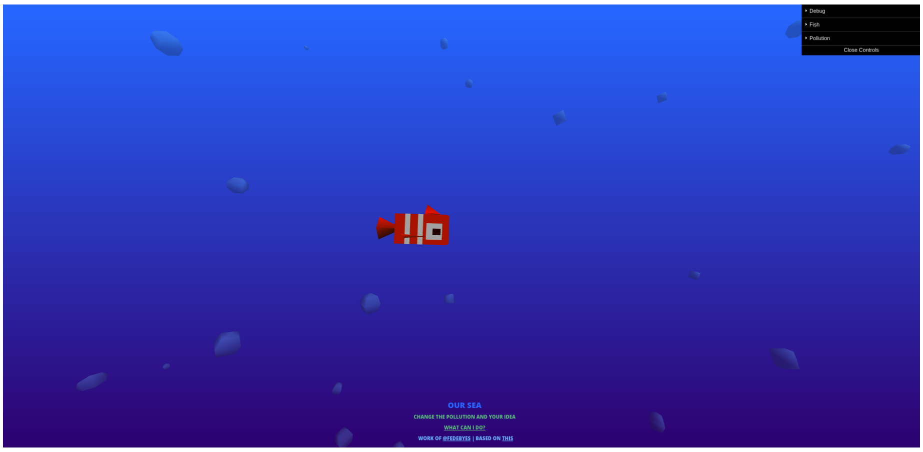
Report Interactive Graphics Project

Teacher: Marco Schaerf
Author: Federico Bacci

26/2/19


• Debug
• Fish
• Pollution
Close Controls
OUR SEA
CHANGE THE POLLUTION AND YOUR IDEA
WHAT CAN I DO?
WORK OF @FEDEBYES | BASED ON THIS

Our Sea
Change the pollution and your idea

1 Introduction

The idea for this project comes from the need to inform and inspire on the Issue of plastic pullution in **Our Sea**. It is pointed out from numerous articles that every year 8 million of metric tons of plastic are thrown awain into the ocean.

The main character is a small fish that is swimming in the ocean, as we change the pollution of the ocean the fish become gradually sicker and sad.

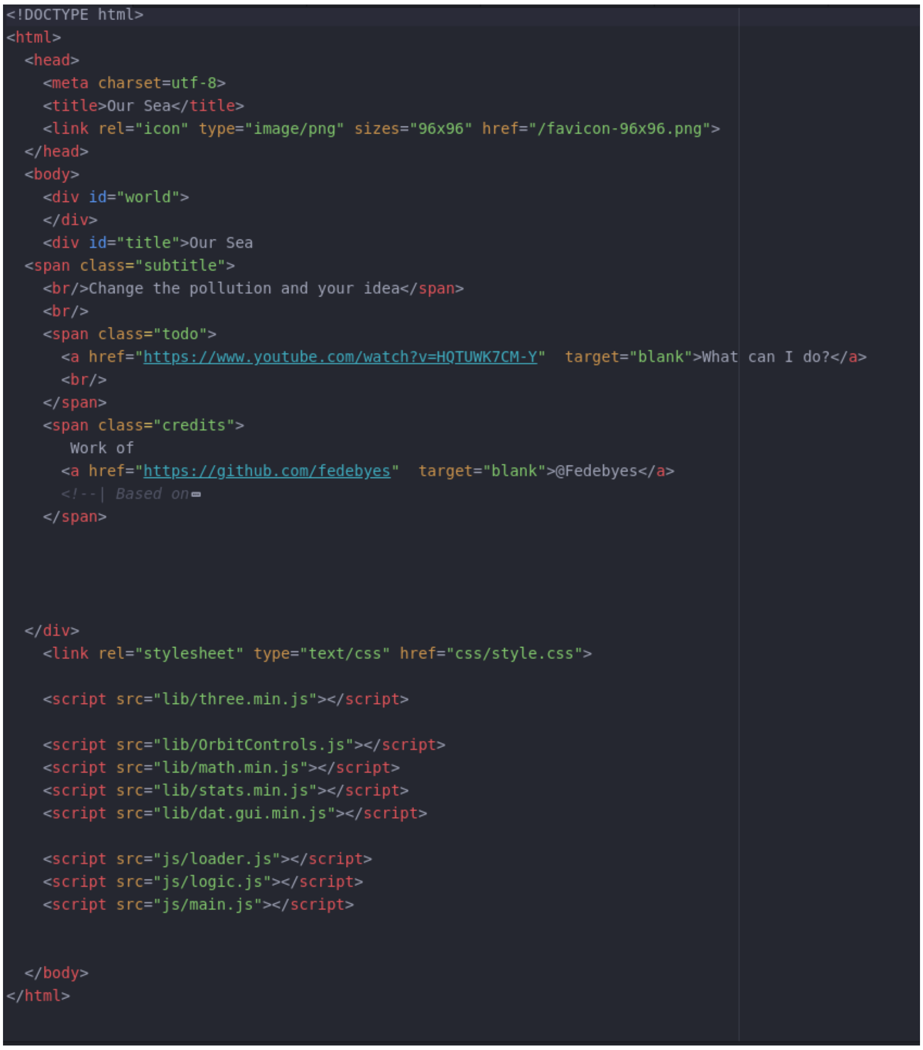
2 Environment

To develop the application I used the **Three.js** library. Preferred over **Web GL** standard library because the large level of abstraction and customization. Also in the project were used different libraries of Javascript here is the complete list:

- **Three.js** to create an easy to use, lightweight, 3D library. The library provides Canvas 2D, SVG, CSS3D and WebGL renderers.
- **OrbitControls.js** part of the Three.js Project to control the camera with the mouse
- **Math.js** for the mathematical computation in the project
- **Stats.js** part of Three.js Project to check the Frames per Second (FPS)
- **Dat.GUI.js** part of Three.js Project to create an easy gui to edit parameters

3 Techical Aspects

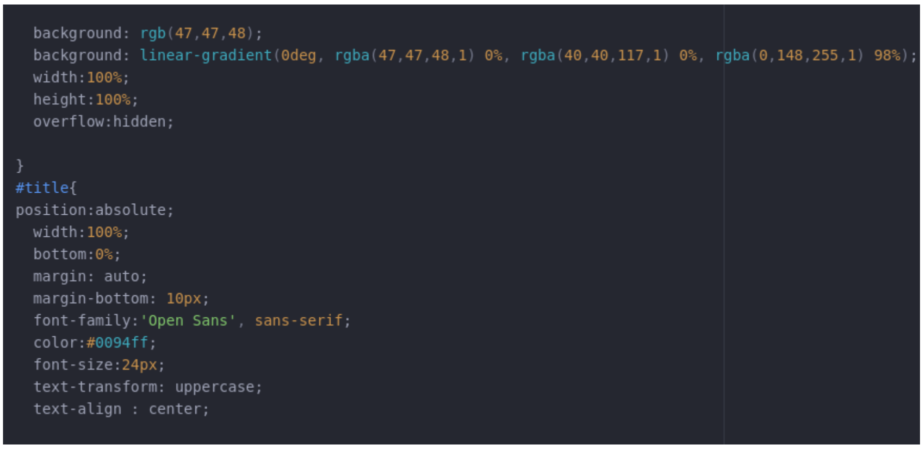
3.1 HTML



```html
<!DOCTYPE html>
<html>
  <head>
    <meta charset=utf-8>
    <title>Our Sea</title>
    <link rel="icon" type="image/png" sizes="96x96" href="/favicon-96x96.png">
  </head>
  <body>
    <div id="world">
    </div>
    <div id="title">Our Sea
  <span class="subtitle">
    <br/>Change the pollution and your idea</span>
    <br/>
    <span class="todo">
      <a href="https://www.youtube.com/watch?v=HQTUWK7CM-Y"  target="blank">What can I do?</a>
      <br/>
    </span>
    <span class="credits">
       Work of
      <a href="https://github.com/fedebyes"  target="blank">@Fedebyes</a>
      <!--| Based on
    </span>




  </div>
    <link rel="stylesheet" type="text/css" href="css/style.css">

    <script src="lib/three.min.js"></script>

    <script src="lib/OrbitControls.js"></script>
    <script src="lib/math.min.js"></script>
    <script src="lib/stats.min.js"></script>
    <script src="lib/dat.gui.min.js"></script>

    <script src="js/loader.js"></script>
    <script src="js/logic.js"></script>
    <script src="js/main.js"></script>


  </body>
</html>
```

The HTML file contains just the title and the Javascript libraries that we have listed plus the files that we have created for the project

3.2 CSS



```
  background: rgb(47,47,48);
  background: linear-gradient(0deg, rgba(47,47,48,1) 0%, rgba(40,40,117,1) 0%, rgba(0,148,255,1) 98%);
  width:100%;
  height:100%;
  overflow:hidden;

}
#title{
position:absolute;
  width:100%;
  bottom:0%;
  margin: auto;
  margin-bottom: 10px;
  font-family:'Open Sans', sans-serif;
  color:#0094ff;
  font-size:24px;
  text-transform: uppercase;
  text-align : center;
```

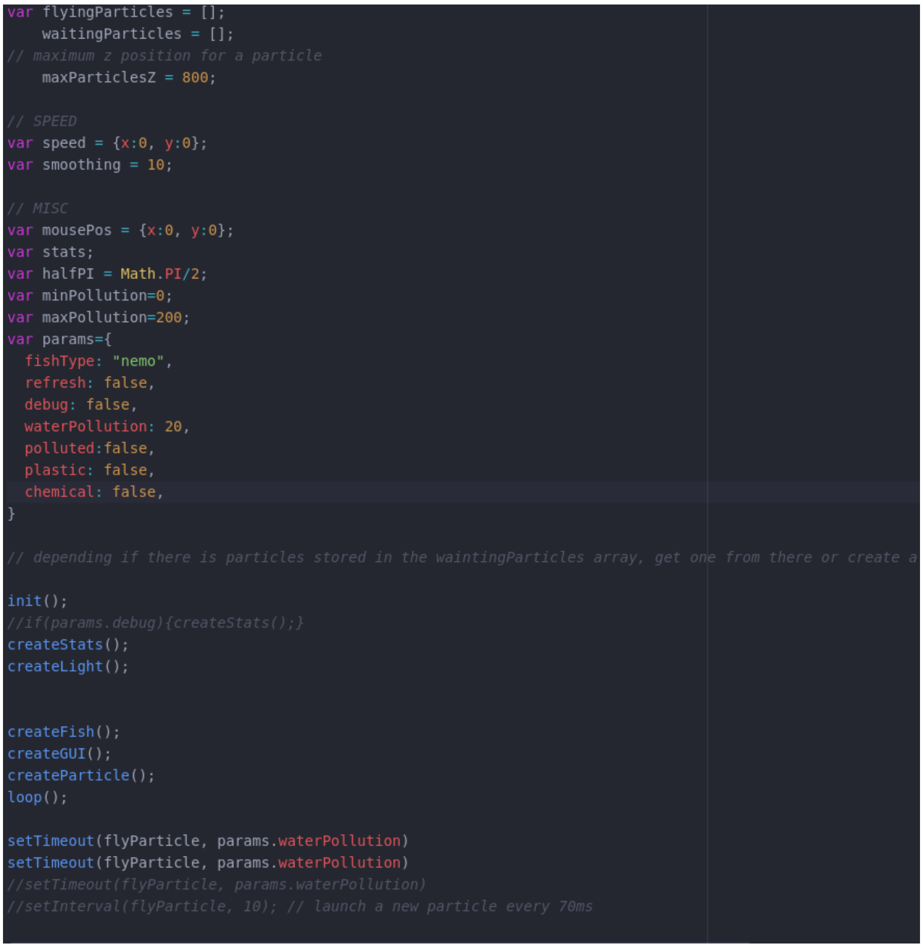
In the file style.css are contained just the information for the background and the title styling

3.3 Javascript

The Javascript code is divided in 3 files

- loader.js
- logic.js
- main.js

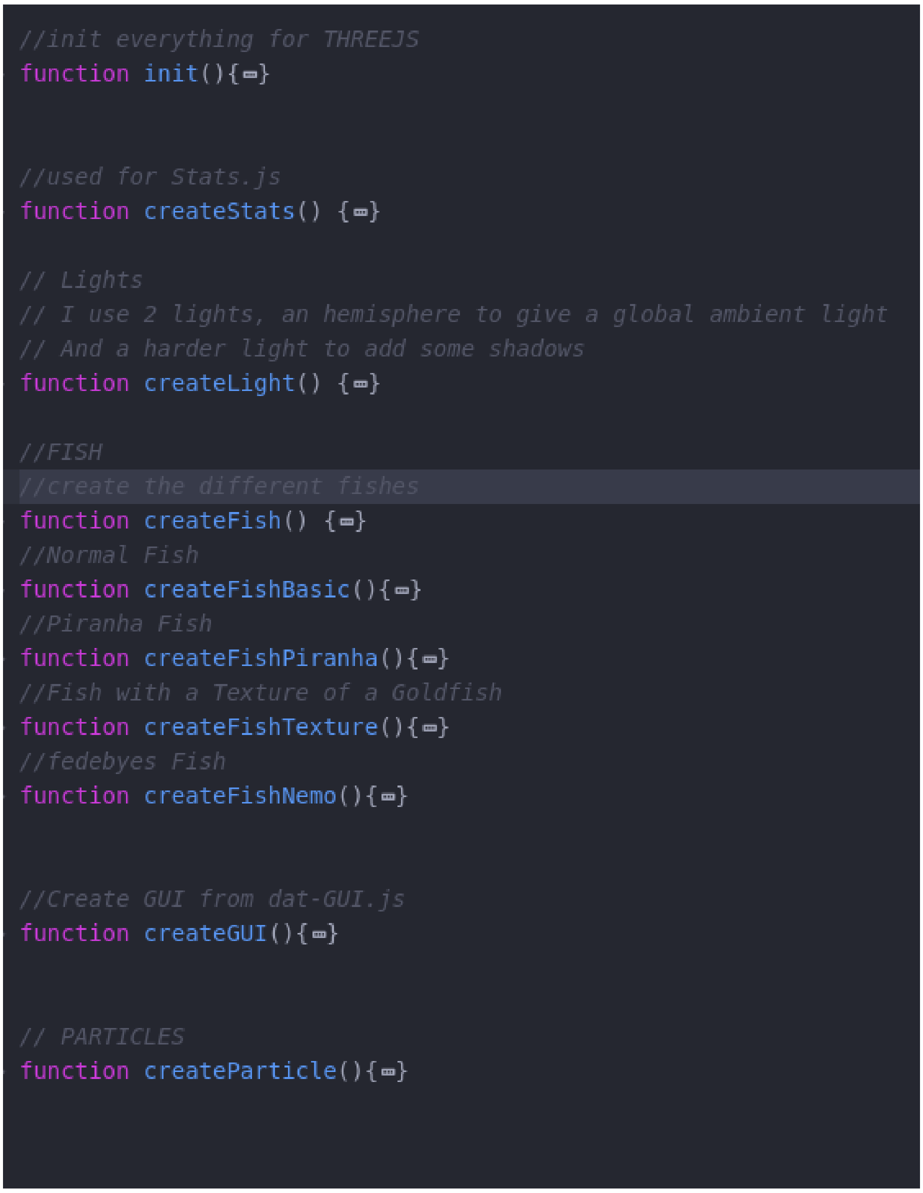
3.3.1 main.js



```
var flyingParticles = [];
    waitingParticles = [];
// maximum z position for a particle
    maxParticlesZ = 800;

// SPEED
var speed = {x:0, y:0};
var smoothing = 10;

// MISC
var mousePos = {x:0, y:0};
var stats;
var halfPI = Math.PI/2;
var minPollution=0;
var maxPollution=200;
var params={
  fishType: "nemo",
  refresh: false,
  debug: false,
  waterPollution: 20,
  polluted:false,
  plastic: false,
  chemical: false,
}

// depending if there is particles stored in the waintingParticles array, get one from there or create a

init();
//if(params.debug){createStats();}
createStats();
createLight();


createFish();
createGUI();
createParticle();
loop();

setTimeout(flyParticle, params.waterPollution)
setTimeout(flyParticle, params.waterPollution)
//setTimeout(flyParticle, params.waterPollution)
//setInterval(flyParticle, 10); // launch a new particle every 70ms
```

In this file all variables are defined, including scene, camera, container and renderer. After that custom variables are defined like colors and speed.

3.3.2 loader.js



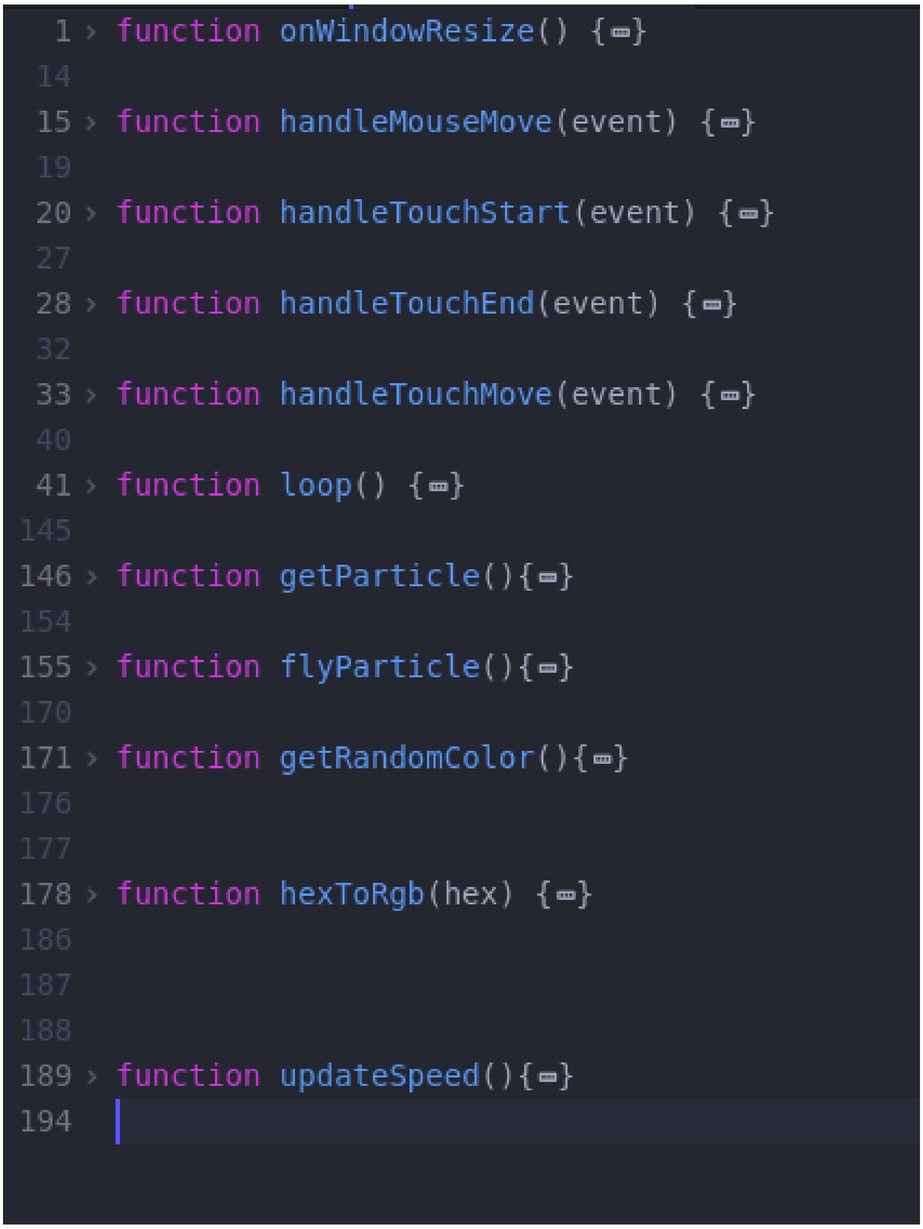
```
//init everything for THREEJS
function init(){⋯}


//used for Stats.js
function createStats() {⋯}

// Lights
// I use 2 lights, an hemisphere to give a global ambient light
// And a harder light to add some shadows
function createLight() {⋯}

//FISH
//create the different fishes
function createFish() {⋯}
//Normal Fish
function createFishBasic(){⋯}
//Piranha Fish
function createFishPiranha(){⋯}
//Fish with a Texture of a Goldfish
function createFishTexture(){⋯}
//fedebyes Fish
function createFishNemo(){⋯}


//Create GUI from dat-GUI.js
function createGUI(){⋯}


// PARTICLES
function createParticle(){⋯}
```

The javascript function of this file are necessary to load the Environment.

- **init()** Creates a scene, the camera, the renderer and add all listeners to the document
- **createStats()** Creates the Stats for Stats.js
- **createLight()** add lights to the scene one HemispereLight and one DirectionalLight
- **createFish()** is in charge of creating the fish, generally composed of a lot of polygons

- **createFishbasic()** create a cube for the body of the fish then a MeshLambertMaterial is added, the tail is a CylinderGeometry, like all the fins, while the eyes are a BoxGeometry as the iris
- **createFishPiranha()** is the same as createFish() just with the adds of a different colour and the teeth of the fish that are another BoxGeometry
- **createFishTexture()** is the same as createFishBasic() just with the add of a Goldfish Texture image
- **createFishNemo()** creates a Red fish with big eyes like the Nemo cartoon, easy to impersonate swimming in the ocean
- **createGUI()** is the function that adds the GUI to the project to change variables, using the library dat.GUI.js
- **createParticle()** Creates different types of particles with random colors, scale and dimensions.
  - The Bubble particles are a Sphere geomety with a transparent MeshLambertMaterial.
  - The Plastic particles are BoxGeometry.
  - The Dirt particles are SphereGeometry.
  - The Chemical particles are TetrahedronGeometry.

3.3.3 logic.js



```
function onWindowResize() {⋯}

function handleMouseMove(event) {⋯}

function handleTouchStart(event) {⋯}

function handleTouchEnd(event) {⋯}

function handleTouchMove(event) {⋯}

function loop() {⋯}

function getParticle(){⋯}

function flyParticle(){⋯}

function getRandomColor(){⋯}


function hexToRgb(hex) {⋯}



function updateSpeed(){⋯}

```

This was the most difficult part of the project with countless of mathematical function to operate all the animations.

- **onWindowResize()** Handles the event for window resizing
- **handleMouseMove(event)** handle the event for touch moving
- **handleTouchStart(event)** handle the event for touch start

- **handleTouchEnd(event)** handle the event for touch end
- **handleTouchMove(event)** handle the event for touch moving so is compatible also with the phone
- **loop()** the most complicated part of the project, this is the also called requestAnimationFrame. Based on the purity of the water it change the color of the fish, while based on the speed it change the scale to let struggle the fish when the mouse is on the right and making a calm pose when the mouse is on the left. Depending on the mouse position the eyes of the fish and the body follow the mouse so we can see it in 3D. This function also let move the fins of the fish with a mathematical function of cosen for the rotation of the tail, leftFin, rightFin and topFin.
- **getParticle()**
- **flyParticle()** let particles move simulating a current
- **getRandomColor()** get a color from the random color that we created
- **hexToRgb(hex)** is used to create a RGB color type from an HEX one
- **updateSpeed()** update the speed according the mouse position

3.4 Components

3.4.1 Scene

The scene is nothing more than a background with two lights added, a HemispereLight and a DirectionalLight.

3.4.2 Particles

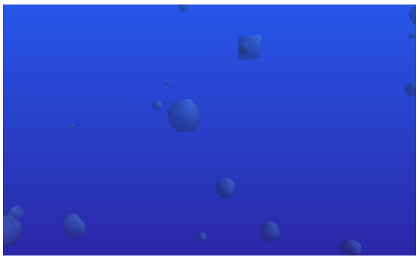


Figure 1: Bubble Particles

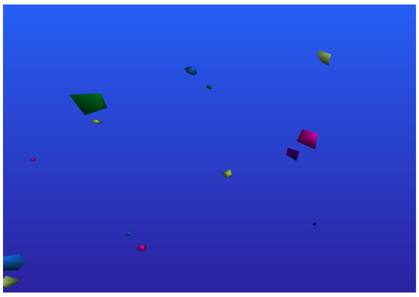


Figure 2: Dirt Particles

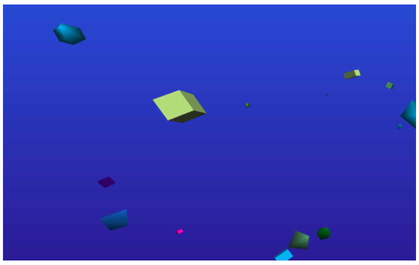


Figure 3: Plastic Particles

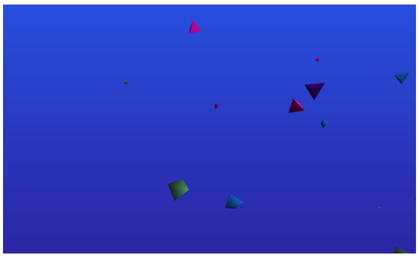


Figure 4: Chemical Particles

These are the different particles that float into the sea, if it is not polluted and the Water Purity is high there are just oxygen bubbles, but if it is polluted the plastic, dirt and chemical partiles start to float around.

3.4.3 Fish

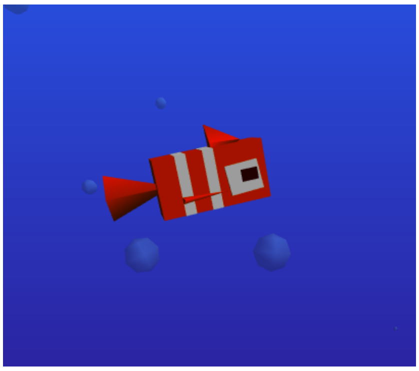


Figure 5: Nemo Fish

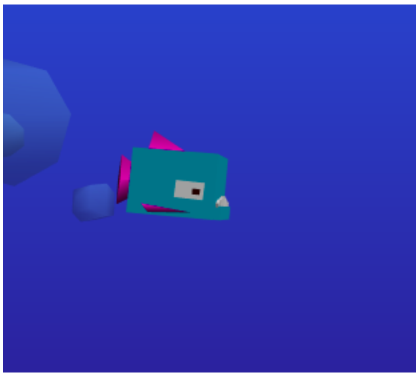


Figure 6: Piranha Fish

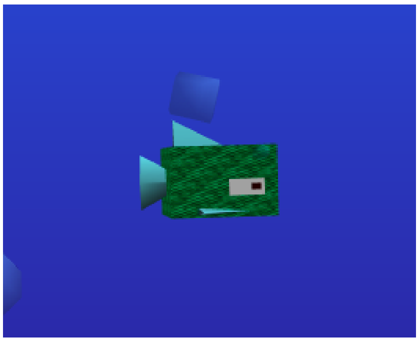


Figure 7: Texture Fish

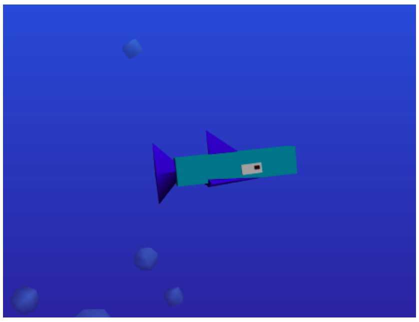
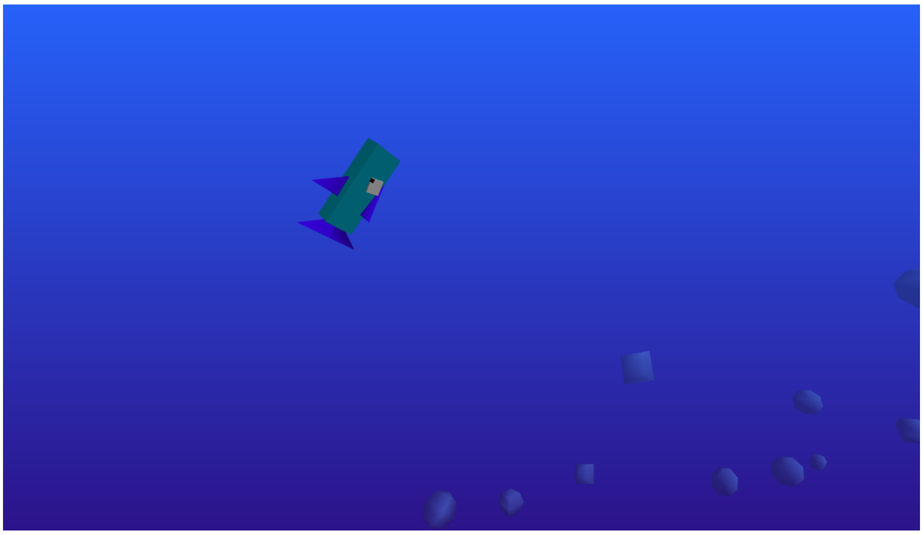


Figure 8: Normal Fish

There are four different types of fish and each one of them interact differently with the colors and the pollution of the water.

Every fish is different but in general they are composed by

- **Body** a BoxGeometry with a MeshLambertMaterial
- **Spot** two BoxGeometry with MeshLambertMaterial with a little distance from the body that is the parent
- **Tail** a CylinderGeometry with a MeshLambertMaterial scaled to resemble better a tail depending from the body
- **Fin** two fin for lef and rigth that are Tails rotated for that
- **Top Fin** another fin very similar to the tail that is scaled and half of it is hidden in the body of the fish
- **Eye** two BoxGeometry that are the Eyes of the fish, their parent is the body
- **Iris** two BoxGeometry with a MeshLambertMaterial



3.5 Interaction

The main part of the project is the interaction, the user can move the mouse (or thouch on the smartphone) on the left or right and the fish will swim slower or faster respectively, scaling the parts and the body to simulate the movement.

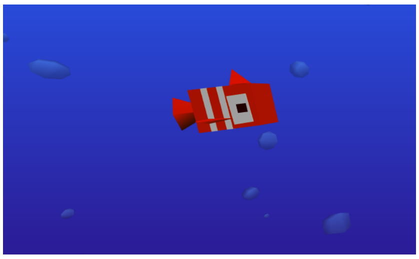


Figure 9: Calm Fish

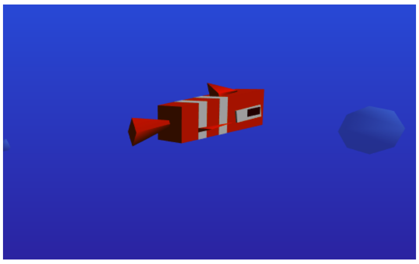


Figure 10: Swimming Fish

Depending on the pollution the fish could be sane and sick, changing color.

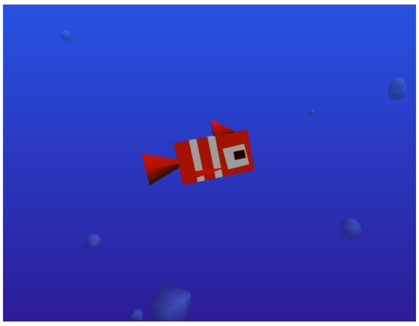


Figure 11: Sane Fish

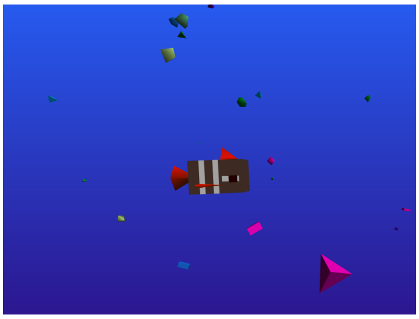


Figure 12: Sick Fish

In the end the fish also can be moved with the mouse and the eyes will follow it using the different functions.

3.6 Conclusion

This project was aimed to understand and use in a real environment the WebGL technologies to create a Graphic that could be Interactive. I think I have learned a lot on the technology creating a project that can be useful for something, the project is currently hosted on fedebyes.github.io/our-sea, I've decided to call it Our Sea to make people aware of the impact of pollution on it.

“We forget that water cycle and
life cycle are one“
-Jaques Yves Cousteau