TGD2151 - Computer Graphics Fundamentals

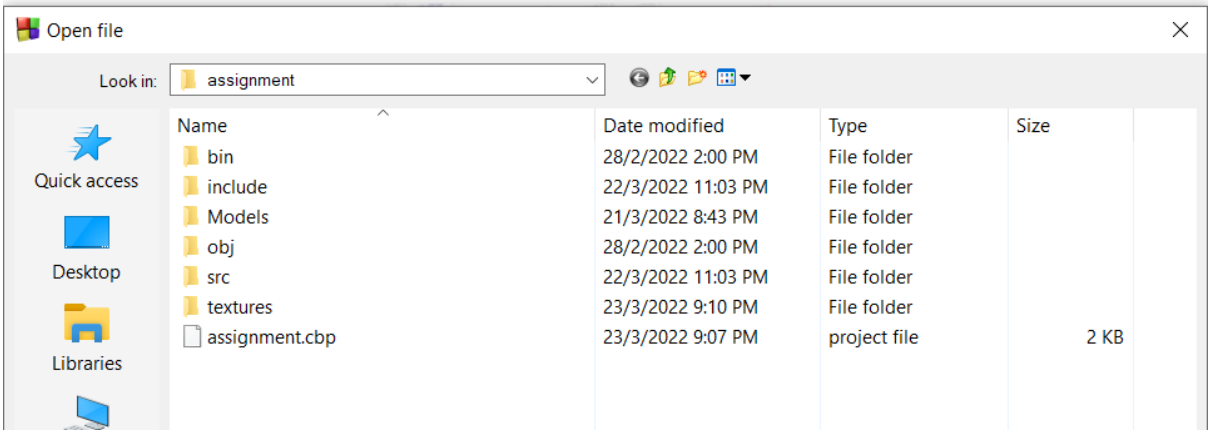
Assignment Report

Group 6

Name	Student ID	Task Distribution	Percentage(%)
Tee Wei How	1191101475	One spongebob house, shark, and turtle, bubbles	33.3%
William Hii Tuan Wei	1191101168	One spongebob house, submarine, and jelly fish	33.3%
Hooi Thing Hong	1191100283	One spongebob house, squid, little fishes, sand	33.4%
		Total	100%

User Manual

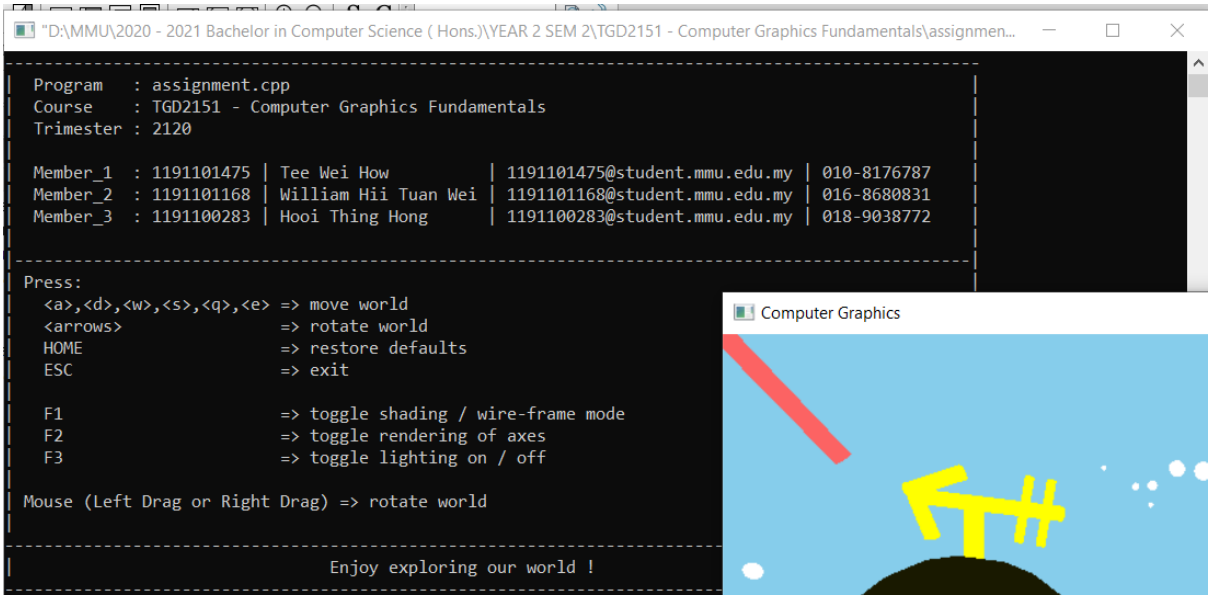
Starting from launching your CodeBlocks IDE, click the “Open the existing project” and select “assignment.cbp”.



And the top navigation bar is all the files that contain in the cbp file which is :

1. main.cpp (main file for running the whole project)
2. header.cpp (header file which extends the main.cpp file)
3. assignment.cpp (contains every models coding and texture mapping)
4. assignment.hpp (header file which use to call the models)
5. BmpLoader.cpp (use to load the picture for texture mapping)
6. BmpLoader.cpp (header file for BmapLoader.cpp)

When the “F9” button is pressed, users can see that 2 new windows will pop out. The first one is our underwater model which contains many models as well as animation, while the second window with black background is our user manual.



As the user manual tab stated , there are a few ways that users can view our assignment models for a better viewing experience. The first one is users can move or zoom in and out

of the angle of view in the assignment model by using a few keys which is “ W , A , S , D , Q , E “ as well as using the mouse to drag the angle.

Table below shows how to control the angle of view and zooming.

Keys	View
W	Zoom out
S	Zoom in
Q	Move up
E	Move down
A	Move right
D	Move left

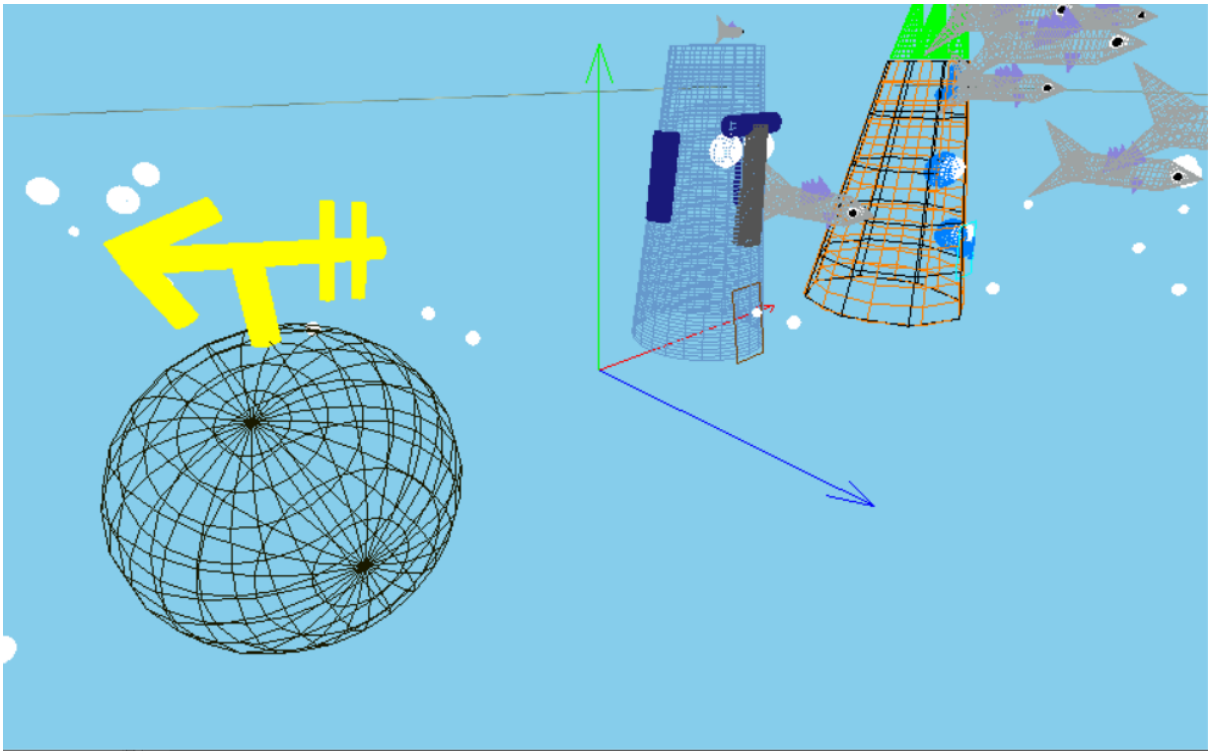
Besides that, there are also some keys which can rotate the angle of view for the users, which is the arrow key.

Arrow Up	Rotate Up
Arrow Down	Rotate Down
Arrow Left	Rotate Left
Arrow Right	Rotate Right

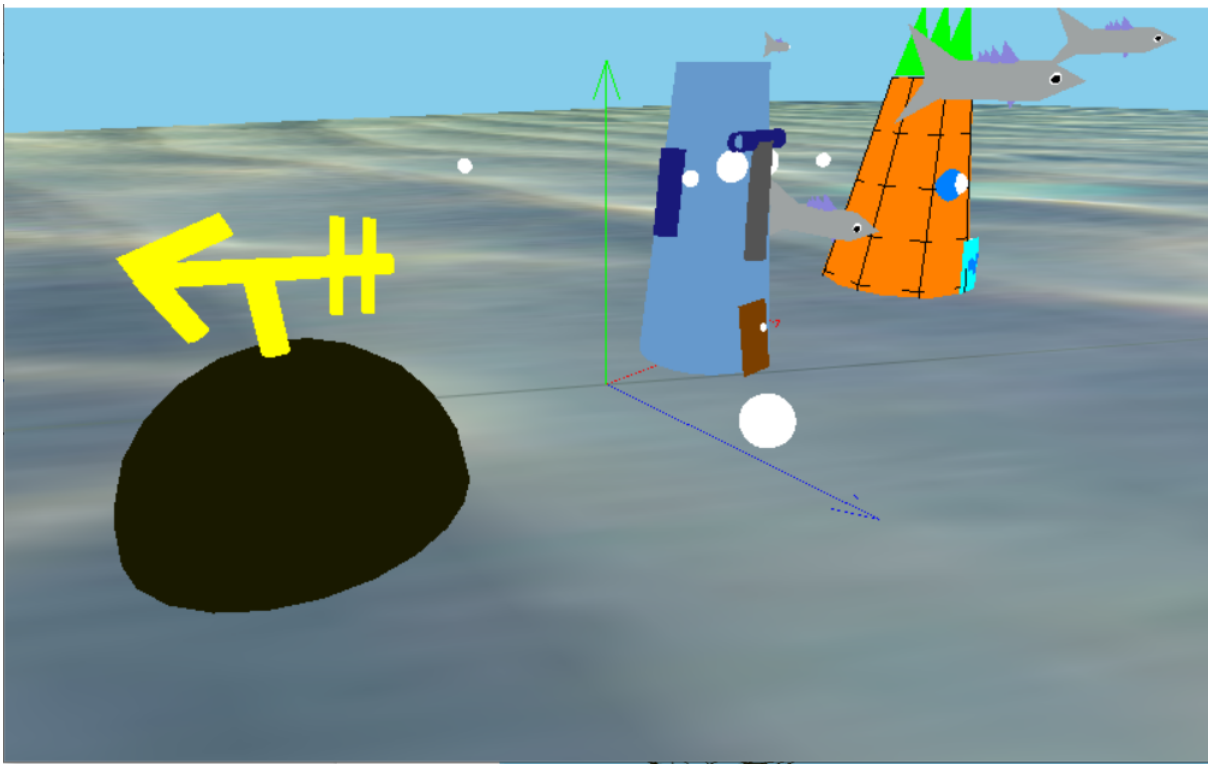
Pressing the “Home” button will reset the angle of view to the default angle when users launch the project, while pressing the “esc” button will terminate the project.

There are certain functions that will change the view of models as well. By pressing the “F1” key, the whole project’s models will switch to shading mode or wire-frame mode, pictures below show the shading and wire-frame mode of models.

Wire-frame mode

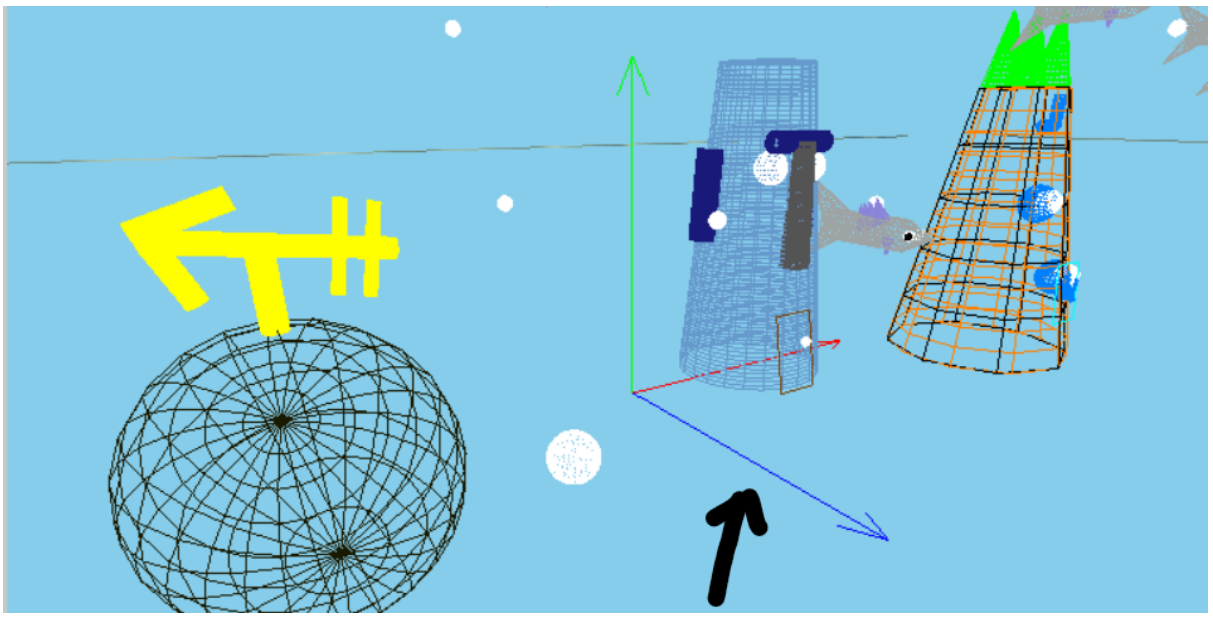


Shading mode

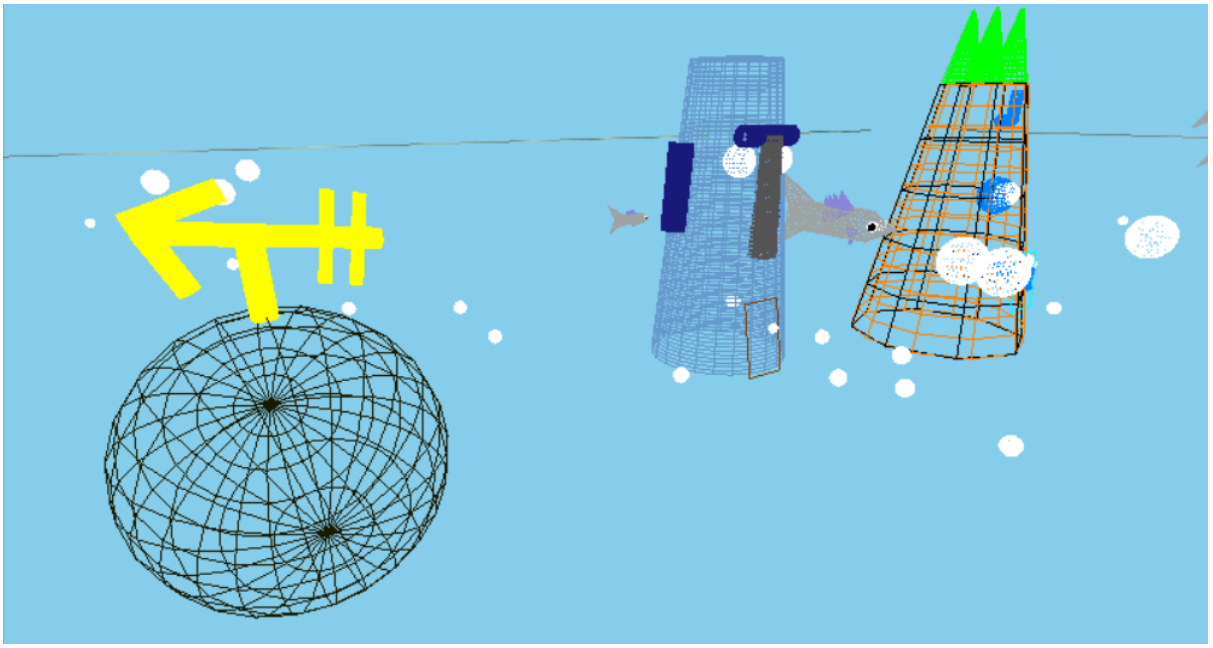


Pressing the “F2” key will disable the default axes that are inside the project, pressing again will enable the axes back.

With axes

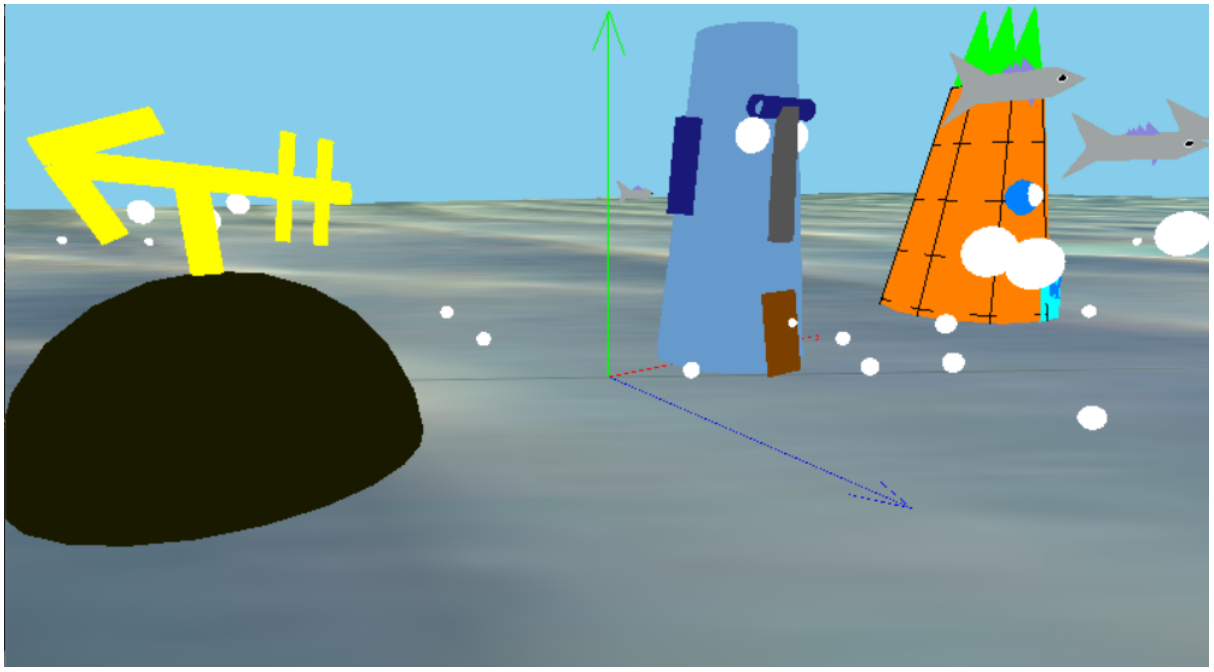


Without axes

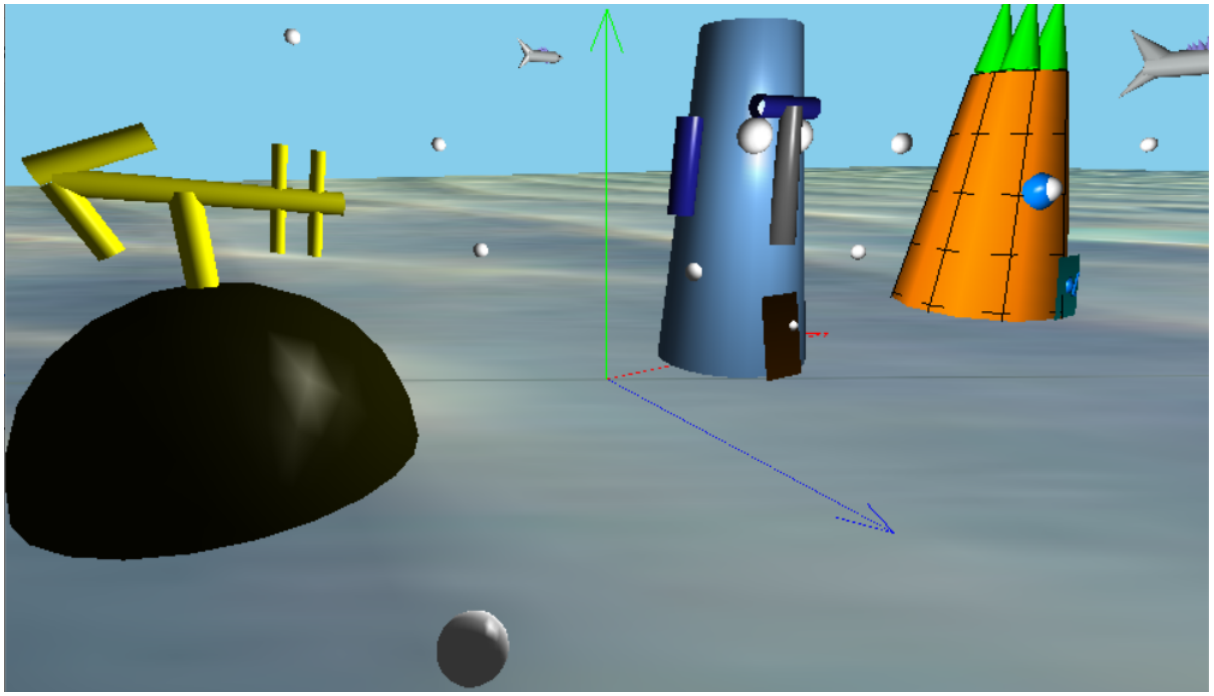


The last function will be toggling the lightning effect. By pressing the “F3” key, users will be able to view more realistic models for the project. Picture below shows the lightning effect on and off.

Lightning effect OFF



Lightning effect ON



Documentation of Virtual Model

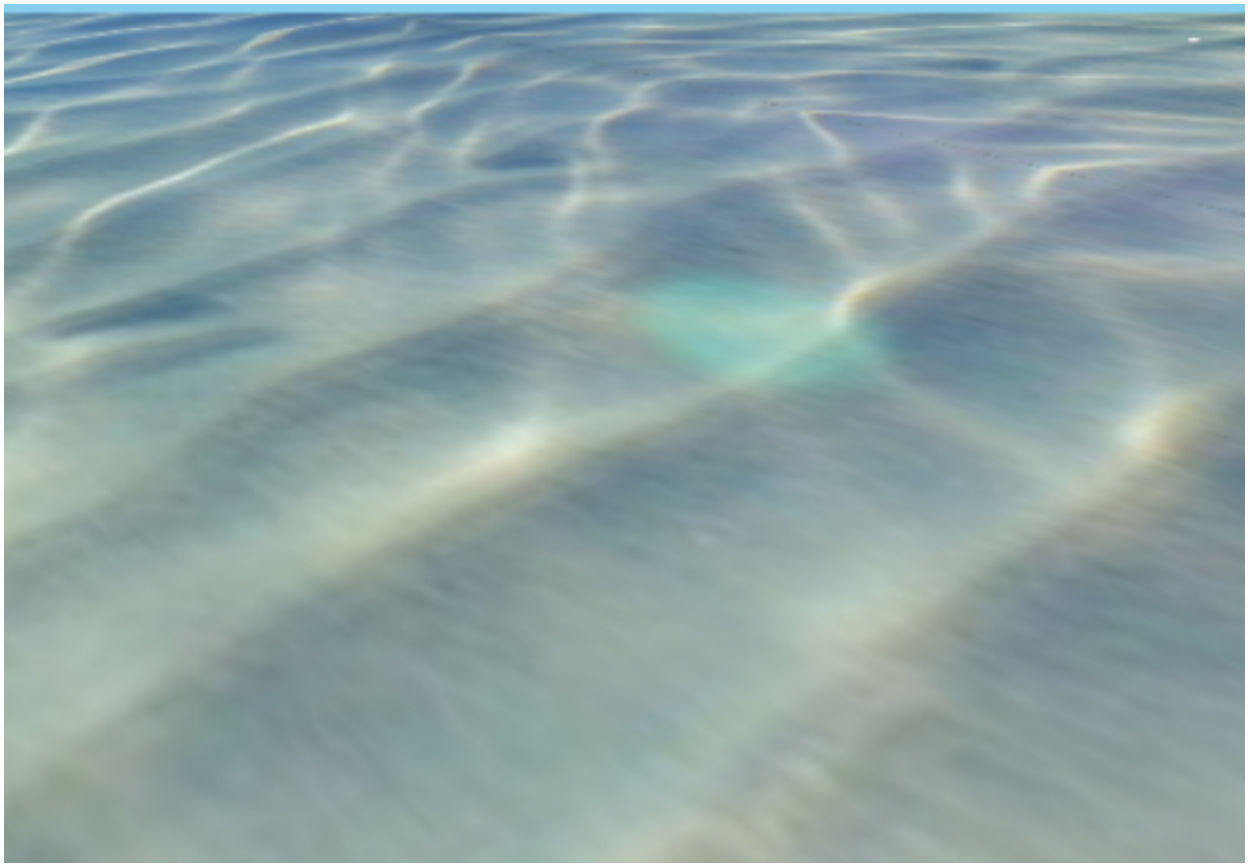
Introduction

The goal of this project is to create an imaginative underwater 3D world using computer graphics. We created the imaginative 3D World by utilising some built-in primitive models and simple openGL animations. The main concept of this virtual world is created as an underwater world with a variety of living and non-living creatures. This 3D World is made up of a portion of the ocean floor, an ocean ecosystem, and imaginative, livable creatures under the water while using various rendering effects like lighting, shading, and texturing.

Models

Sand

In order to create a more realistic underwater world, we decided to create a flat floor which has a rgb of the sand colour. We then applied a caustic texture on that floor by texture mapping the floor with a caustic effect image. Therefore, a realistic underwater sand floor is created. Image below is the sand that we created for our virtual underwater world.

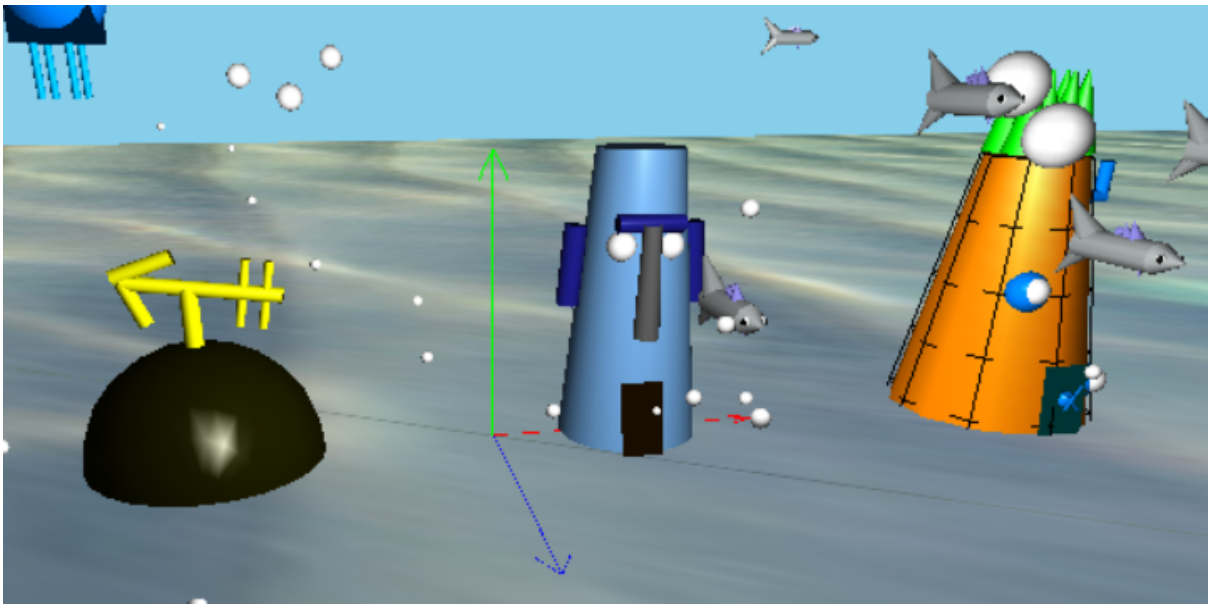


Bubbles

An underwater ocean world always has bubbles that pop up from beneath the sand because of oxygen. Therefore, we decided to make the bubble effect by creating many small white sphere objects in random positions in the underwater virtual world that will flow upward towards the surface of the ocean. This will also improve the virtual world's realism. The small white sphere objects from the image below are the bubbles that we created.

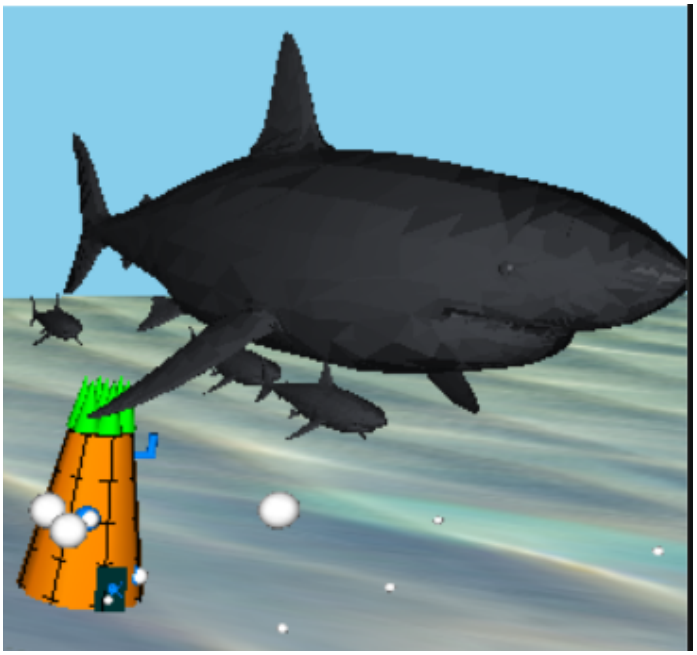
Spongebob Houses

In order to make the ocean floor more interesting, we wanted to add something to it. We came out with the idea from our childhood memories of spongebob houses, which are Patrick's house, Squidward's house, and Spongebob's house. For these houses, we drew it using QUADRICS that we learnt during lab 5. Image below are the three houses.



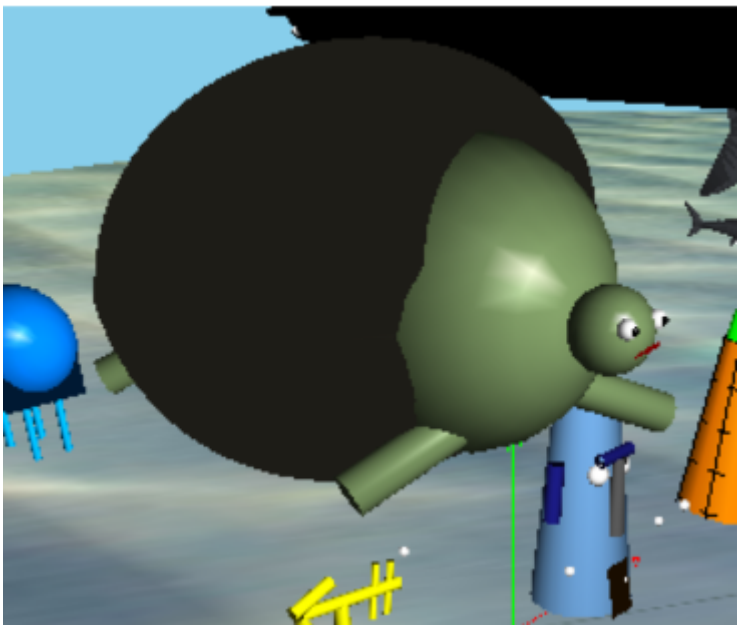
Shark

In order to create a very realistic living shark, we decided to load the model from an obj file which we downloaded from an available source. We modified a bit for this model by creating shark babies that stay under their mom's belly wherever the mother goes. We also modified the model by adding animation to it which allows it to swim forward only. Image below shows the shark models that we created.



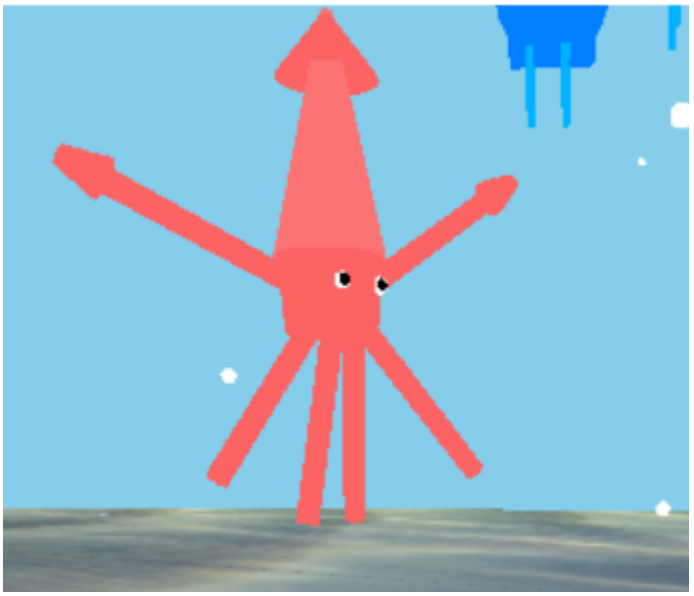
Turtle

We used a bezier curve to create the shells of the turtle. The reason that we do not use this quadric sphere is because the shape of the quadric sphere is too rounded for a turtle shell and it does not have a hole for the turtle's head and butt to pop out. Therefore, a bezier curve from lab 9 is used to create the shells. The head, eyes, and cloaca (butt) of the turtles are created using the quadric sphere. Both the hands and legs are created using quadric cylinders. The mouth of this turtle is created using a quadric disk. In order to make it more lively, we added an animation which allows the turtle to swim forward slowly. Image below is the turtle that we created.



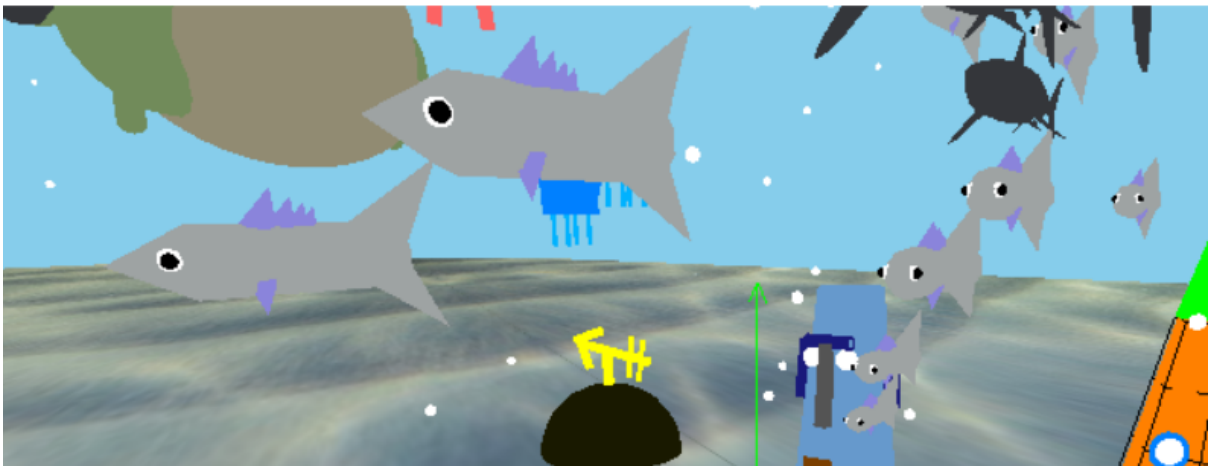
Squid

From what we learned in the lab session, we figured out that we can create a giant squid using “gluCylinders” and “gluSpheres” only. We build the top of the squid head using a cylinder with a tiny top base, so it does look like a cone, while the head of the squid is built with a cylinder and spheres for the eyes of the squid. The tentacles of the squid are built with cylinders as well, the flexible use of “glTranslatef” and “glRotatef” can make how the squid become our desired patterns and shape, so tons of translation and rotation is used in building this model. To make it more realistically, we decided to add an animation for the squid to swim upwards. Picture below shows how is the squid looks like.



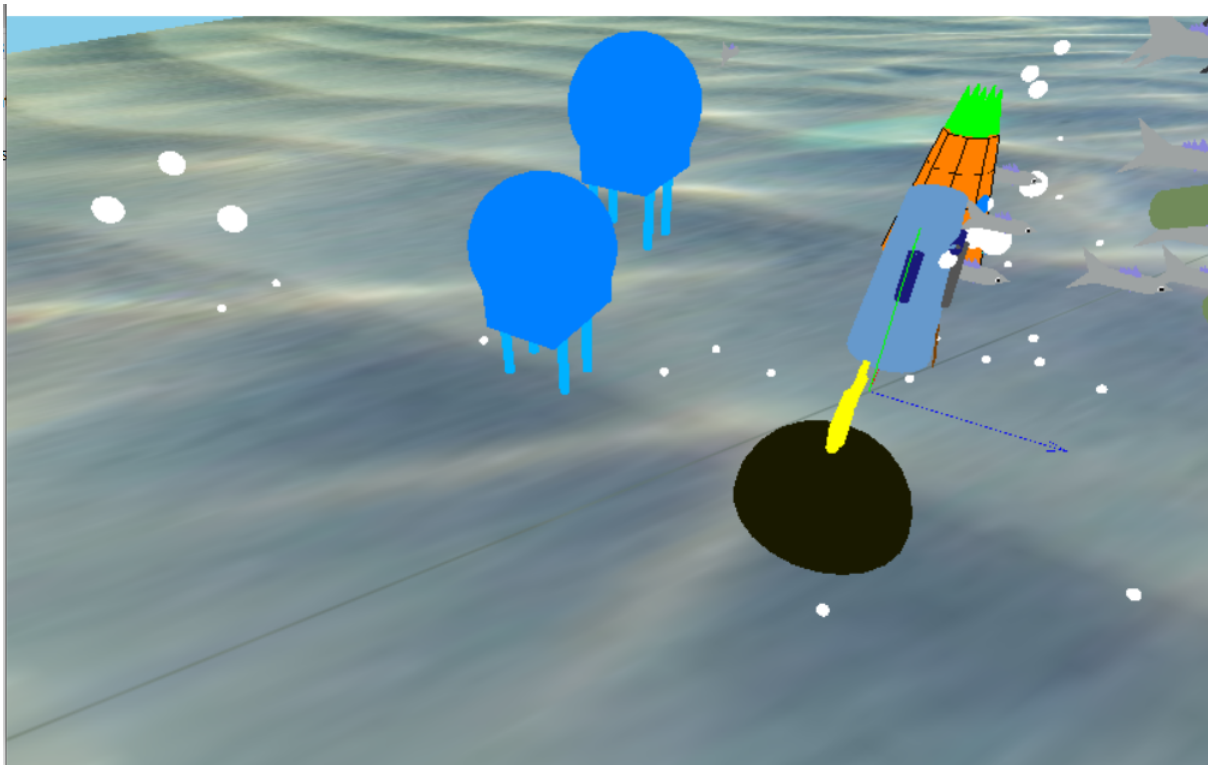
Little fishes

To increase the diversity of the underwater world that we had created, we came up with an idea which designed a little fishe for the shark to chase off. This little fish is made by “gluCylinder” and “gluSphere” as well, like the giant squid. We had referred to what is inside the ocean, we realised there is always a school of little fishes swimming around, so we decided to increase the amount of little fish, which we use to randomly spawn the little fish in a certain area within tick time. The picture below shows these little fishes.



Jellyfish

In order to create a jellyfish model that is more realistic, we decide to sphere the body of the jellyfish. Since we cannot create a hemisphere, we decided to create a 3D rectangle with 5 faces with 4 vertices for each of them. We put the 3D rectangle at the lower part of the sphere to make it look more similar to the jellyfish in real life. After creating the body, we also create 4 cylinders at the bottom of the jellyfish's body as its leg. Since some of the jellyfish are living in groups, we decided to put two jellyfish together. At the end, we add animation to the jellyfish by making them move upwards.



Submarine

To make the underwater scene more interesting, we decided to add something other than different types of fish. So, we use the submarine as the model by loading an obj file downloaded online. After doing research, we decided to make the colour of the submarine black because most of the colour of the submarine is black. We also add animation for the submarine by making it move forwards.