**webgl.js – Project 4**

**Lily Forry**

**CMSC 405 6380 Computer Graphics (2225)**

**Professor: Lauren King**

**Due: July 12, 2022**

**User’s Guide**

The goal of Project 4 was the use of webgl.js to develop a unique 3D animated scene. The scene needed to be a minimum of 640 x 480 pixels, contain 10 unique shapes, use multiple lighting, texture effects, and let the user change different aspects of the scene through toggles. For further information, see Project Requirements. Diskplay was what I chose as the starting point for this project, since it included everything that would be required and only needed elements added and modified.

**Getting Started**

The main file, Project4HTML.html needs to be within the webgl folder, as it pulls elements from webgl.  
During testing, I used Waterfox so I can guarantee that it functions correctly, any browser should work.  
When Project4HTML.html is opened, the following scene should appear.  
A picture containing graphical user interface

Description automatically generated

This scene consists of three possible lighting effects, 10 trees, a pole with night-time lighting, a car with night lighting, the sun with day lighting, and an optional UFO attempting to abduct the car.

The viewpoint can be changed by using the mouse to rotate the scene freely.

The animate checkbox can be checked in order to start the animations and change between day and night.  
Graphical user interface

Description automatically generated

Graphical user interface, application

Description automatically generated

The Car checkbox can be unticked in order to remove the car and the car’s lighting from the scene  
Graphical user interface, application

Description automatically generated

The UFO checkbox can be checked in order to add a stealthy UFO to the scene  
A screenshot of a computer

Description automatically generated with medium confidence

The Red, Yellow, and White Sunlight radio buttons can be chosen between to change the sunlight’s color  
Graphical user interface

Description automatically generated  
Graphical user interface

Description automatically generated with low confidence  
Graphical user interface

Description automatically generated

Finally, the reset button can be clicked to reset the scene as it would be when the project was originally launched  
A picture containing graphical user interface

Description automatically generated

**Project Requirements**

Include 10 different objects

10 Trees – 1 cylinder and 1 cone

A pole – 1 cylinder and 1 sphere

Car – 14 cylinders, 4 torus, 2 rectangles

Sun – 1 sphere

Ufo – 1 torus and 2 spheres

Use multiple lighting effects

Viewpoint lighting

Ambient Lighting via sun

Spotlights via Car healights and UFO abduction light

Include widgets to control different parts of the scene

Checkboxes for animation, the car, and the UFO

Radio Buttons for which color of sunlight

Reset button to reset the scnee

Use webgl

**Test Plan**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Case | Input | Expected Output | Actual Output | Pass/Fail |
| 1 | Create Scene Canvas | A 700 x 700 canvas colored black should appear on the page beneath all texts and widgets | A 700 x 700 canvas colored black appears on the page beneath all texts and widgets | Pass |
| 2 | Create Widgets | Three Checkboxes labeled “animate, car, and UFO” should be displayed, as well as three radiobuttons labeled “White, Yellow, and Red Sunlight”. Lastly, a Reset click button should be displayed. | Three Checkboxes labeled “animate, car, and UFO” are displayed, as well as three radiobuttons labeled “White, Yellow, and Red Sunlight”. Lastly, a Reset click button is displayed. | Pass |
| 3 | Create Text | A header labeled “Forry Final Project: Lighting” should appear at the top of the screen. Corresponding Text for all of the widgets should also be displayed, as well as a prompt for how to move the scene. | A header labeled “Forry Final Project: Lighting” appears at the top of the screen. Corresponding Text for all of the widgets is displayed, as well as a prompt for how to move the scene. | Pass |
| 4 | Create Platform | A disk should appear with a grass texture and road on it. | A disk appears with a grass texture and road on it. | Pass |
| 5 | Create Car | A red car object made out of red rectangles, blue torus wheels, and yellow axle cylinders should appear on top of the road. | A red car object made out of red rectangles, blue torus wheels, and yellow axle cylinders appears on top of the road. | Pass |
| 6 | Create Trees | 10 Basic trees made from brown cylinders and green cones should appear on the platform. | 10 Basic trees made from brown cylinders and green cones appear on the platform. | Pass |
| 7 | Create View Light | A white light should appear from the camera. | A white light appears from the camera. | Pass |
| 8 | Create Day Lights | During the day, the only lights that should be displayed are the viewpoint light and the sunlight. | During the day, the only lights that are displayed are the viewpoint light and the sunlight. | Pass |
| 9 | Create Night Lights | During the night, the only lights that should be displayed are the viewpoint, car headlights, ufo abduction light, and pole light. | During the night, the only lights that are displayed are the viewpoint, car headlights, ufo abduction light, and pole light. | Pass |
| 10 | Check Sun Light Colors | When “white sunlight” is checked, the sun should put out a white light. The same should be true for red and yellow lights. | When “white sunlight” is checked, the sun puts out a white light. The same is true for red and yellow lights. | Pass |
| 11 | Perform Viewpoint Transforms | The scene should be able to be clicked on and dragged around with the mouse. | The scene is able to be clicked on and dragged around with the mouse. | Pass |
| 12 | Select Animate | When the animate checkbox is checked, the sun should rotate counter-clockwise and emulate a day-night cycle. Additionally, the car and UFO should be rotating along the road in a clockwise rotation. | When the animate checkbox is checked, the sun rotates counter-clockwise and emulates a day-night cycle. Additionally, the car and UFO rotate along the road in a clockwise rotation. | Pass |
| 13 | Select UFO | When checked, this box should display a UFO that will chase after the car in a clockwise rotation and  have a spotlight during the night. | When checked, this box displays a UFO that will chase after the car in a clockwise rotation and have a spotlight during the night. | Pass |
| 14 | Select Car | When checked, this box should display a car that will rotate along the road in a clockwise rotation and have 2 headlights during the night. | When checked, this box displays a car that will rotate along the road in a clockwise rotation and have 2 headlights during the night. | Pass |
| 15 | Click Reset | When clicked, this button should reset the scene as if the program has just been opened. | When clicked, this button resets the scene as if the program has just been opened. | Pass |
| 16 | Close the Page | When the browser is closed, it should close. | When the browser is closed, it closes. | Pass |

**Lessons Learned**

My lessons are basically the same as the last project. I don’t like working with computer graphics manually like this or programming GUIs. I still think that a project like this would take a third of the time if something like Unity or Unreal was used.  
Working with javascript and HTML was a lot better this time around. It would have been really cool if we went from Java to JS to Angular or React with this final project.  
After dealing with JOGL and threejs, webgl was a lot more granular but also seemed more intuitive for some reason. I’m glad I was able to do what I wanted to after the last project, with having sun/moon cycles, and adding lighting effects onto different objects and dealing with shadows.