CMSC 405
Computer Graphics:

Project 4

Instructor: *Dr. James Robertson*
Student: *Dakin T. Werneburg*

October 15, 2017

TABLE OF CONTENTS

Question Analysis

Purpose of Assignment

The assignment was to design and implement a 3D graphics scene using WebGL graphics by programing shaders. The scene should include 10 different shapes and multiple lighting effects. It should include radio buttons, slider bars or other widgets to turn on off certain components of the animation.

Questions to be Answered

- What will the 10 different shapes be?
- Should the shapes include textures?
- What lighting will be applied?
- What widgets will used?
- How to structure the code?
- How will I handle different views?

Program Design

Design Decisions

- I decided to used blender and parse the obj file to extract vertices and normal using a youtube video for reference and create a Halloween scene.
- Using textures was part of the assignment part due to complexity with switching shaders and applying lighting to textures this was waived.
- I utilized mostly point and spot lights positional lights.
- I decided to use checkboxes to turn on/off all lights individually, a slider was used to move the spotlight cutoff and button to stop animation.
- I decided to break up the code and I still think my javascript files could be broken up as well, but to keep file count down, I separated style and javascript from html. I also used put the obj files in side javascript files as strings.

Assumptions

- Lighting will be key to seeing the 3d objects.
- Different Lighting will have different effects

Screen Shots

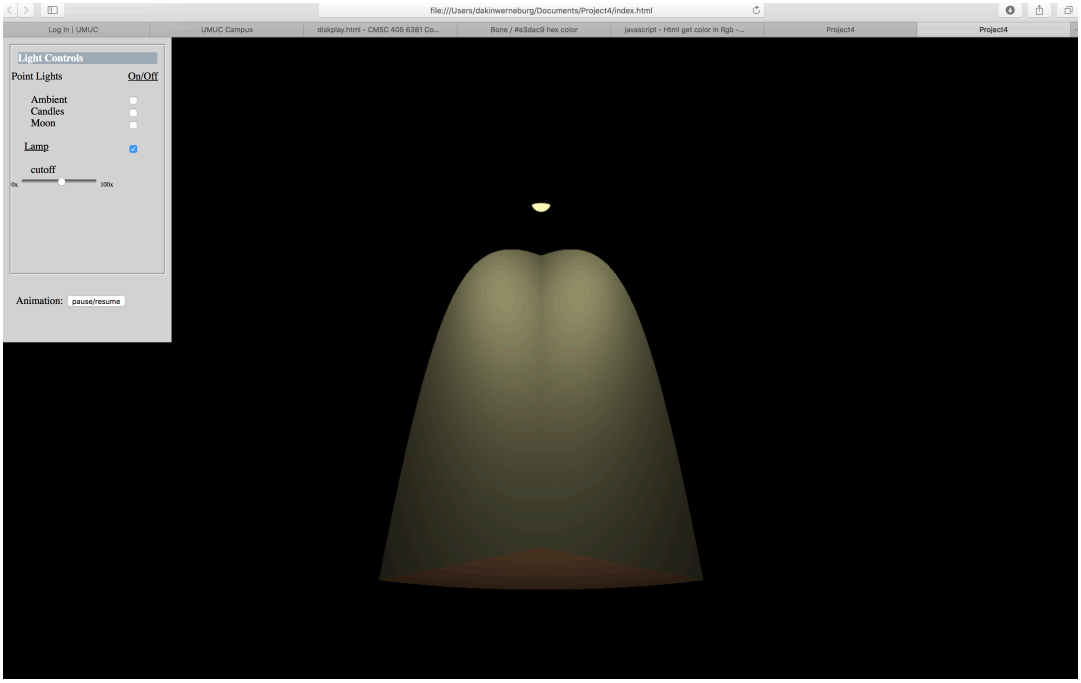

Figure 1: Example 1

Figure 2. Example 2.

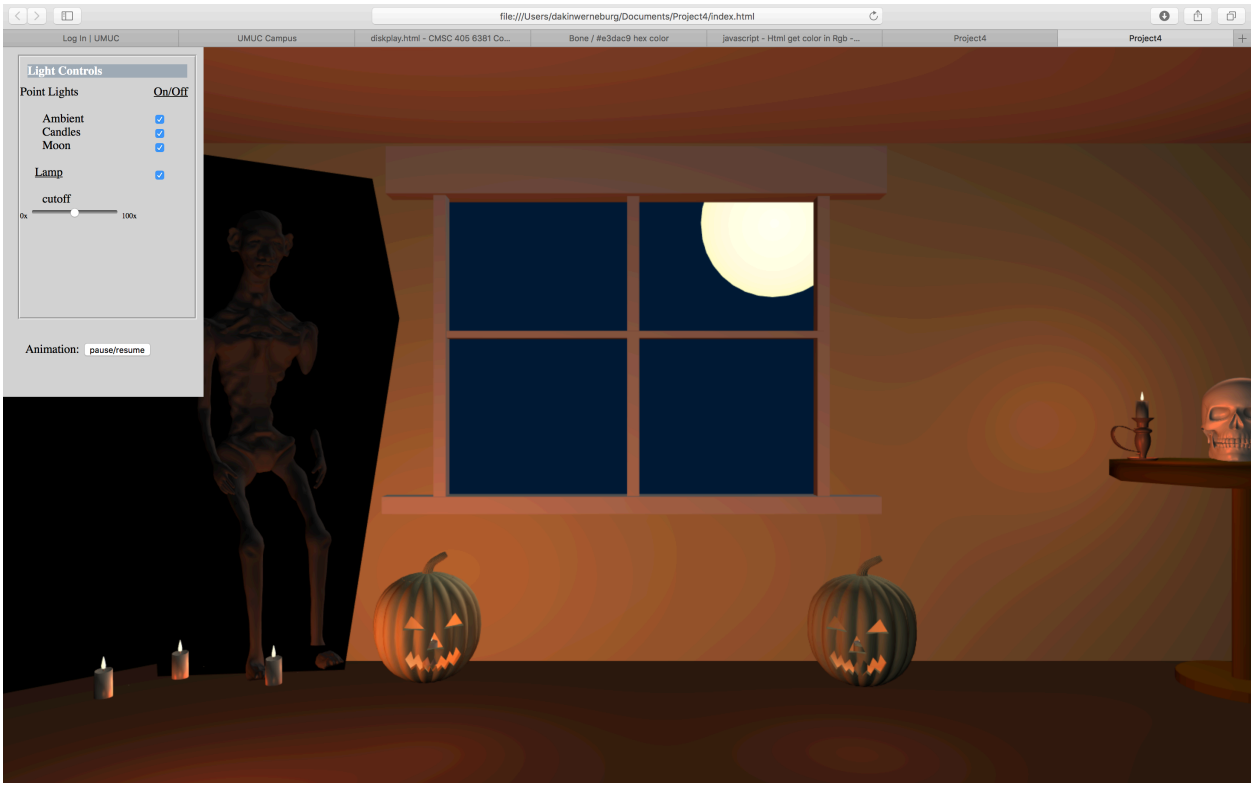

Figure 3. Example 3.

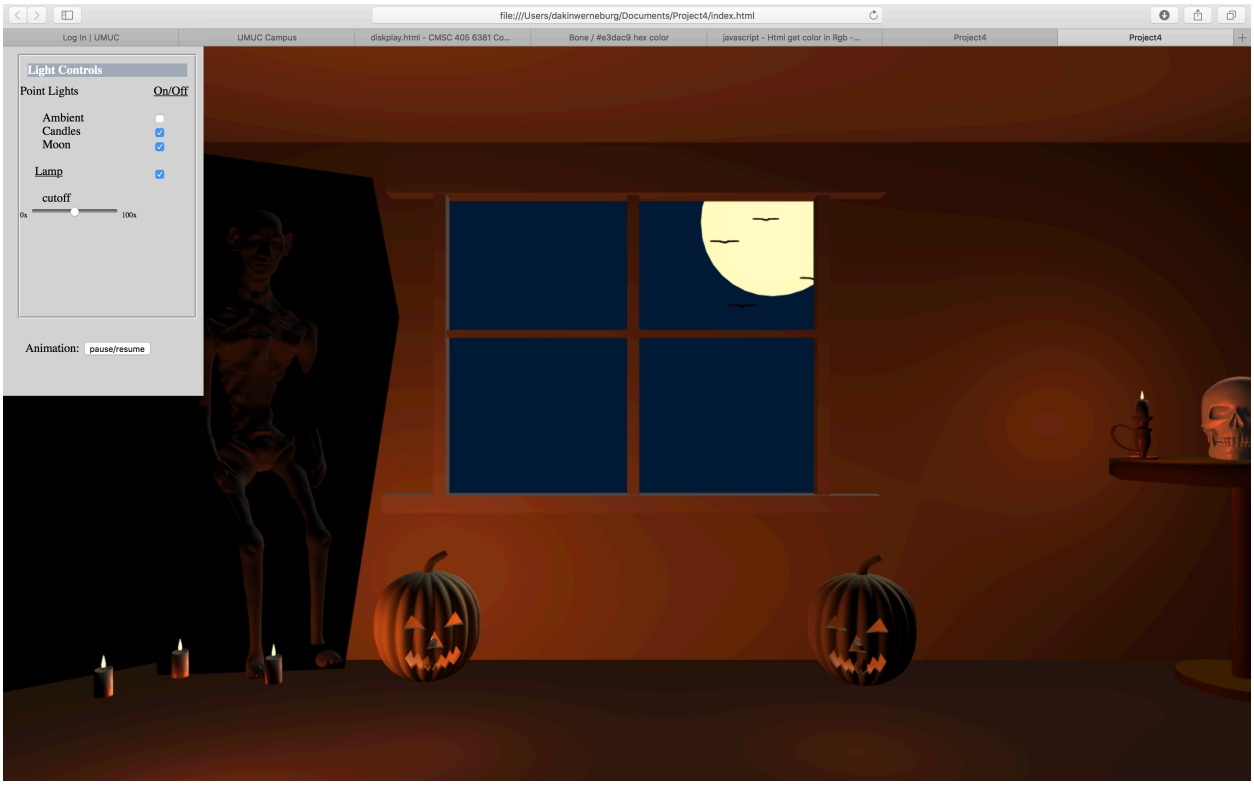

Figure 4. All lights off except yellow flashlight

Possible Improvements

- Could have done something creative in the shaders to give some lighting effects with the candles
- Could have added textures to bring the scene to life.
- Could have split up the javascript code.

.

Test Plan

Input	Expected Output	Actual Output	Passed	Corrections made
addListeners()	flipSwitch() method called when checkboxes were clicked	flipSwithch() methods called, no errors	Y	
addListeners()	animated variable toggled between true and false when button was clicked	Variable updated correctly	N	
flipSwitch()	Lights turned on and off based on the light selected	The correct light was turned on and off	Y	
Init()	Canvas created with camera and control and adjust viewport	All window and view elements created successfully	Y	
Draw()	All object created and displayed in proper location	Objects were displayed	Y	
createModel	Models correctly created and loaded	Models correctly created and loaded	Y	
Animate()	Position light positions updated and sphere moving back and forth	All transformations work	Y	
animate(	Animation working	Animation working	Y	

Error Handling

- **Try-Catch Block** – validated each function call and displays an error with the type error.

Lessons Learned

- WebGL was by far the most complicated project and is very low level. If time permited I am sure I would have loved to program some shaders rather that revert to one provided in the course.
- Models from blender are more complicated to load since textures and materials need to be parse as well.
- Framebuffers allow memory savings and provide capability to do shadows and mirror effects.

References

1. Fun with WebGL 2.0 : 011 : Parsing .OBJ. (2017, March 04). Retrieved October 15, 2017, from https://www.youtube.com/watch?v=0duMYbBPPMU three.js - Javascript 3D library. (n.d.). Retrieved from https://threejs.org