Emily Godwin
CMSC 405 - 2215
6/29/21

Project 3

Developer's Guide

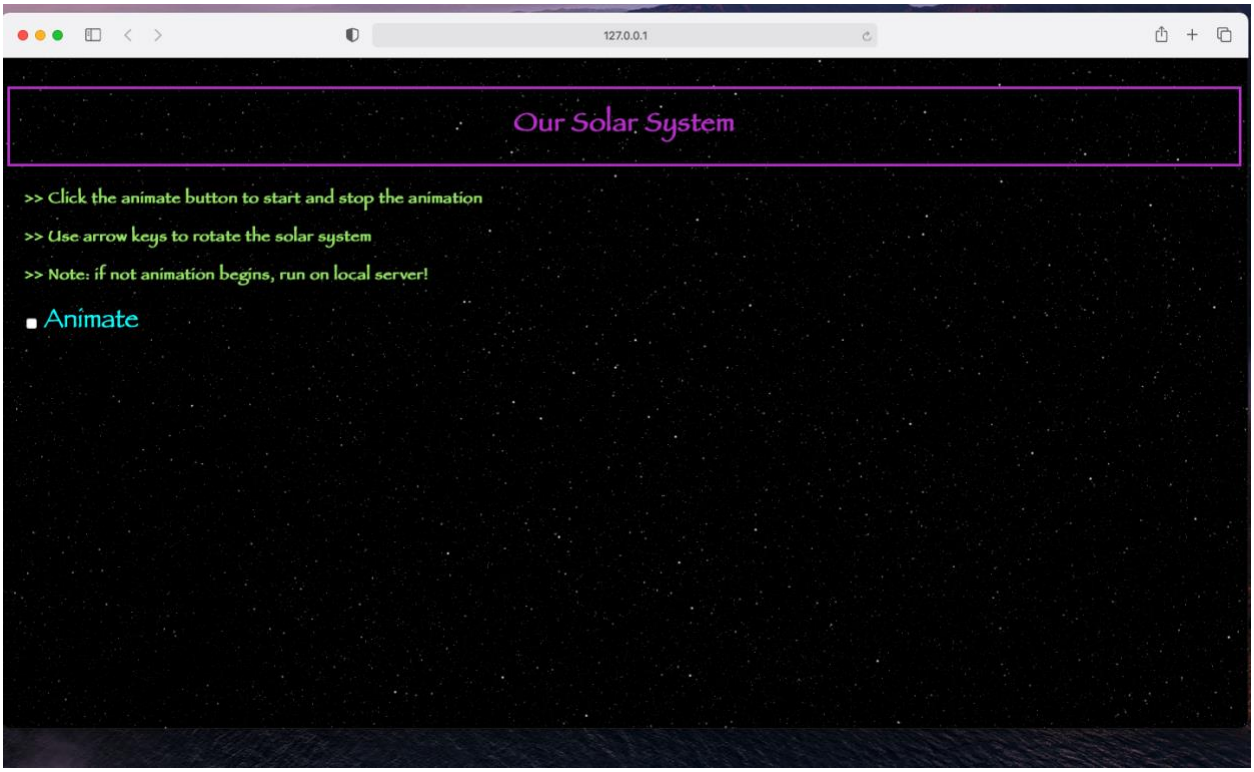
To run the project, double click "Project 3.zip" from the UMGC submission portal. This will download the files. If the zip file does not unzip, unzip it. In a local server, open the SolarSystem.html file and it will display the scene in your browser.

- **Note:** The scene requires running from a local host server due to the textures employed in its design. Running the file in a public web browser will not display the solar system scene.

Test Plan

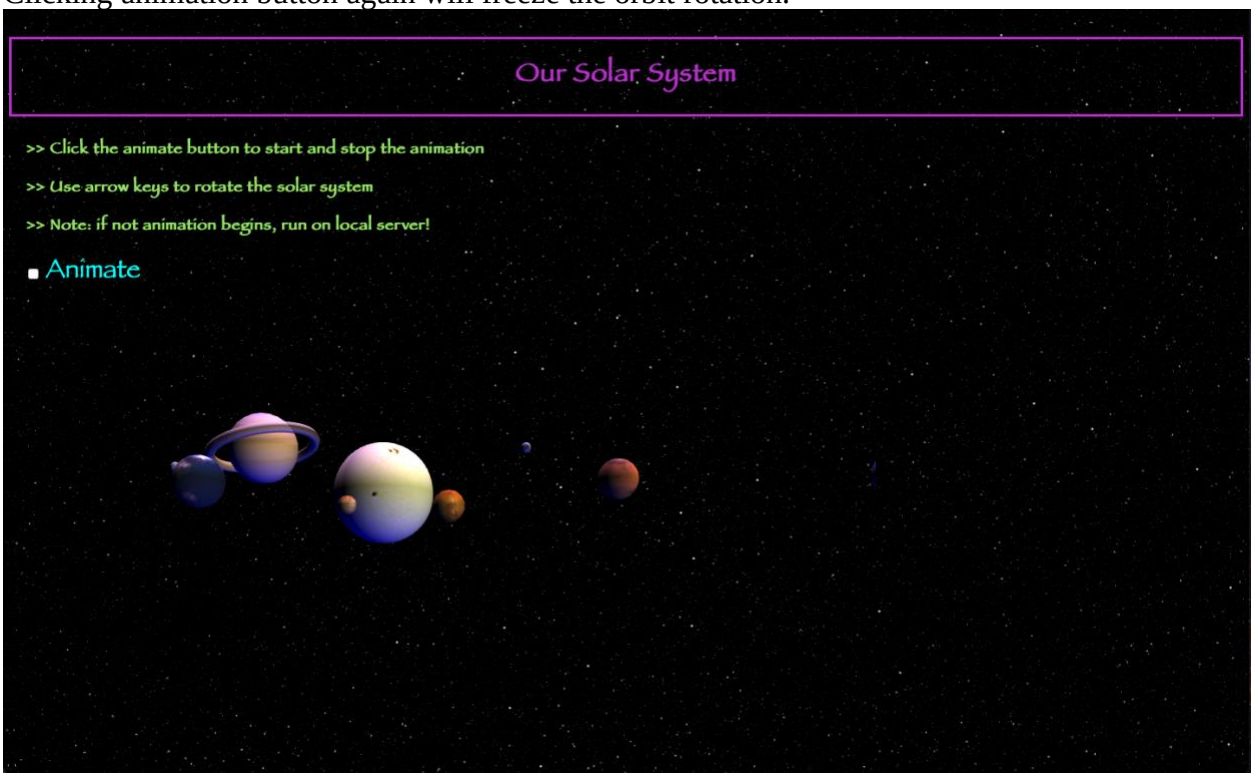
Test - Name	Input	Expected Output	Actual Output	Pass?
Running the SolarSystem Animation	Open the SolarSystem.html file in a local server. Click "Animate". Use arrow keys to rotate the model	A 3D animated model of a solar system, with planets orbiting the sun. Using arrow keys will rotate the model to realign the planets.	A 3D animated model of a solar system, with planets orbiting the sun. Using arrow keys will rotate the model to realign the planets.	Yes

Screen Captures
Before the animation begins, there are no planets. Once the animation button is clicked, the Animation will start.

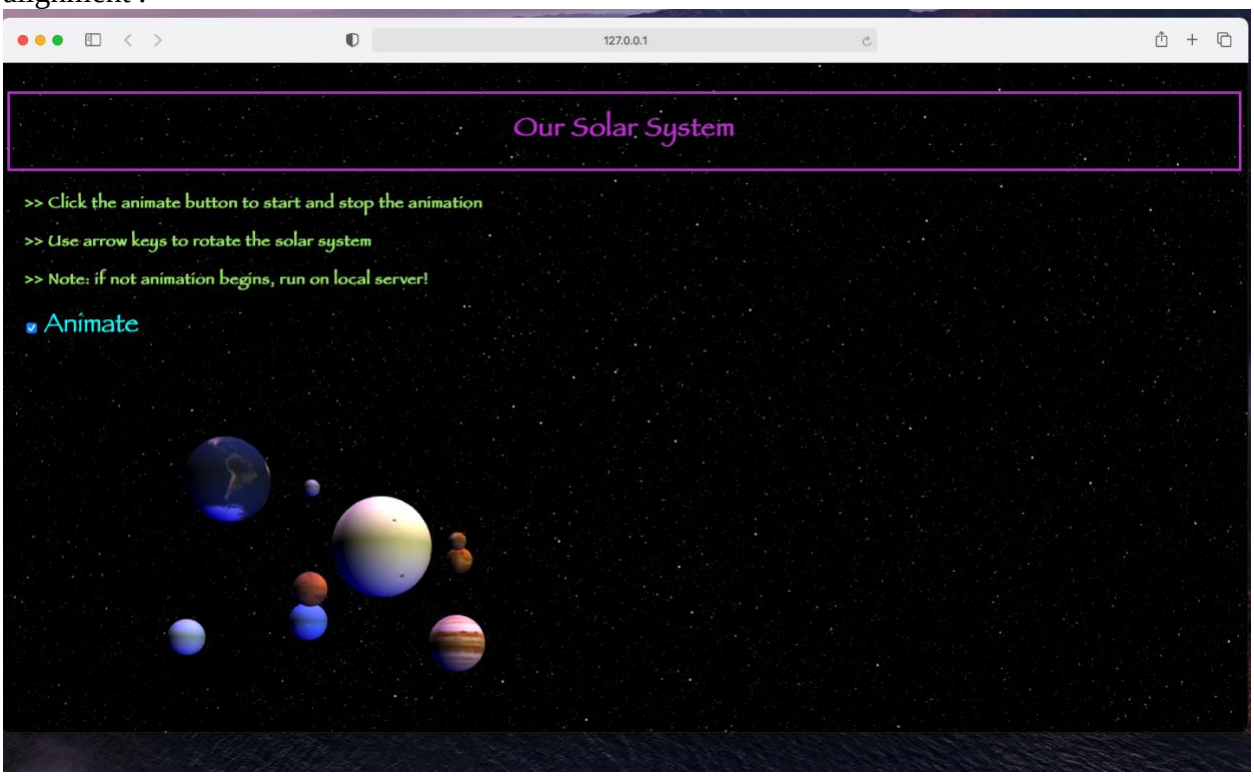


The animation is active, so the planets orbit the sun

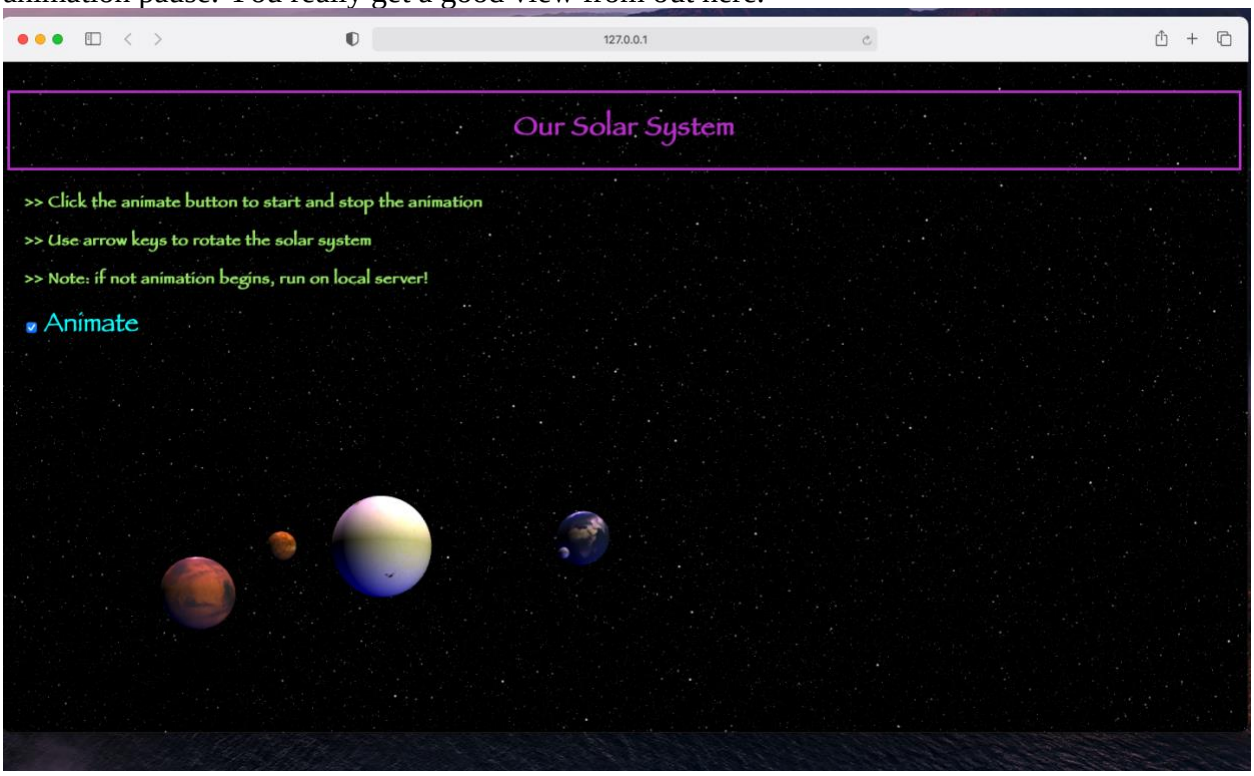
Clicking animation button again will freeze the orbit rotation.



Clicking the animation button again restarts the orbit. Using arrow keys rotates the planetary alignment .



Another example, using down arrow keys to realign planets. Realignment is also allowed during animation pause. You really get a good view from out here.



Lessons Learned

Three.js is a whole new world when it comes to 3D animation graphic design for the web. Having had a small background in html, and css, I felt that this took my ability to include graphics into my webpages to a whole new level. Opening up the Mr.doob website and exploring all of the possibilities that Three.js can have was magical, and I definitely feel that Three.js is the next step in my path to becoming more proficient at animation design. Javascript feels like a cousin of Java, which made reading the text and examples a little bit easier, as the class and object structure made me feel as though I could work through the examples and understand how they were developed. Definitely a more abstract and encapsulated way to design geometric structures than OpenGl, Three.js gave me the power to understand how to design and animate quickly without having to call two different Graphics environments. Having had experience with

html, I felt at home diving into the basics of the development of a scene in a webpage. I was able to get basic animations going and understood why they worked the way they did. I wanted to use textures in this project, however and so the obstacle of locally run imaging confused me until I read through the Three.js documentation a little more After understanding why my textures and the textures in examples weren't loading, and finally getting some progress going, I became inspired to try to animate a solar system, which I had tried to do in the past without success. After finding some textures at Solar System Scope, and reading up on why they weren't showing up when mapped to an object, I implemented my mesh design with maps, bumps, and specular images. I used the sphere Rotator to implement my orbits, and used the BufferedGeometry.js class to create the ring for Saturn, which I also applied a mesh to. This was a really fun animation to create, and I really enjoyed learning these foundational skills.

I hope to continue to learn more about interactive design as well, as I did implement an Animate start button, and used the arrow key rotation functions, but certainly could've taken these concepts farther. In the future I hope to apply these skills and my knowledge of html to allow users to have a larger, more interactive experience with my designs. I also hope to further understand the concept of field of view, as I did my best to get my animation with in the frame but overall wish that the field of view could've encapsulated the entire rotation without sacrificing the scale of the images. I also wish to do more with the zoom of the camera, and would've liked to understand more about allowing the user to change the field of view with the keyboard or mouse.

Sources:

Graphics and Three.js

R. Cabello. https://threejs.org/

D. J. Eck.(2018, January).*Introduction to Computer Graphics V 1.2*.Hobart and William Smith Colleges.(pp. 171-211).

https://learn.umgc.edu/d2l/le/content/580433/viewContent/21625861/View

Provided Three.js example files, including modeling-starter.html

Torus

R.Cabello.https://threejs.org/docs/#api/en/geometries/TorusGeometry

Textures

Solar System Scope. https://www.solarsystemscope.com/textures/