Orbital Mechanics Simulator

Lucius Kwok (VC1B)

<lk@felttip.com>

Supervisor: Brian Papa

<bpapa@icloud.com>

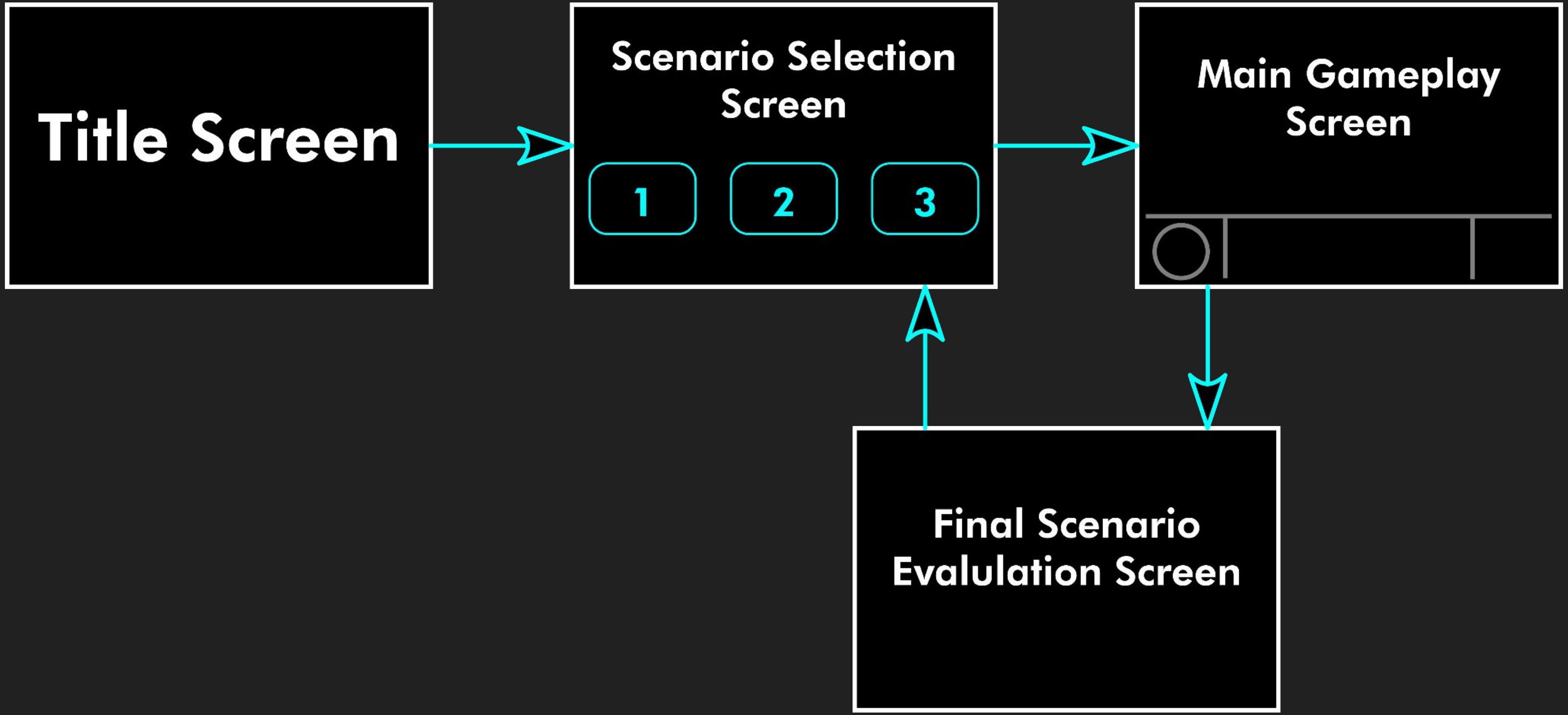
Abstract

Orbital mechanics are an often misunderstood aspect of spaceflight, but one which can be explained through hands-on piloting of a simulated spacecraft which follow a few basic mathematical equations. In this project, I will focus on teaching how a few basic laws of Newtonian physics can result in many types of orbits around celestial bodies, and how to figure out the most efficient way to get from one planet or moon to another. Along the way, I may touch upon the Oberth effect, Hohmann transfers, gravity assists, patched conics, the rocket equation, and other spaceflight-related concepts. By turning it into a game to find the most efficient solution, learning becomes fun and accessible to everyone.

Tools

- Software
  - Unity 6 engine
  - Microsoft Visual Studio Community 2022
  - Autodesk Maya 2025
  - Adobe Photoshop 26
  - Panic Nova (text editor)
  - GitHub suite
  - Google suite
- Hardware
  - Development:
    - Windows PC with Intel Core i7-11700 & NVIDIA GTX 1080 Ti
    - Apple MacBook Air (M3, 2024)
  - Target Platforms: Windows, macOS, WebGL

Diagram 1: Screen UX Flow



Title Screen

Scenario Selection Screen

1

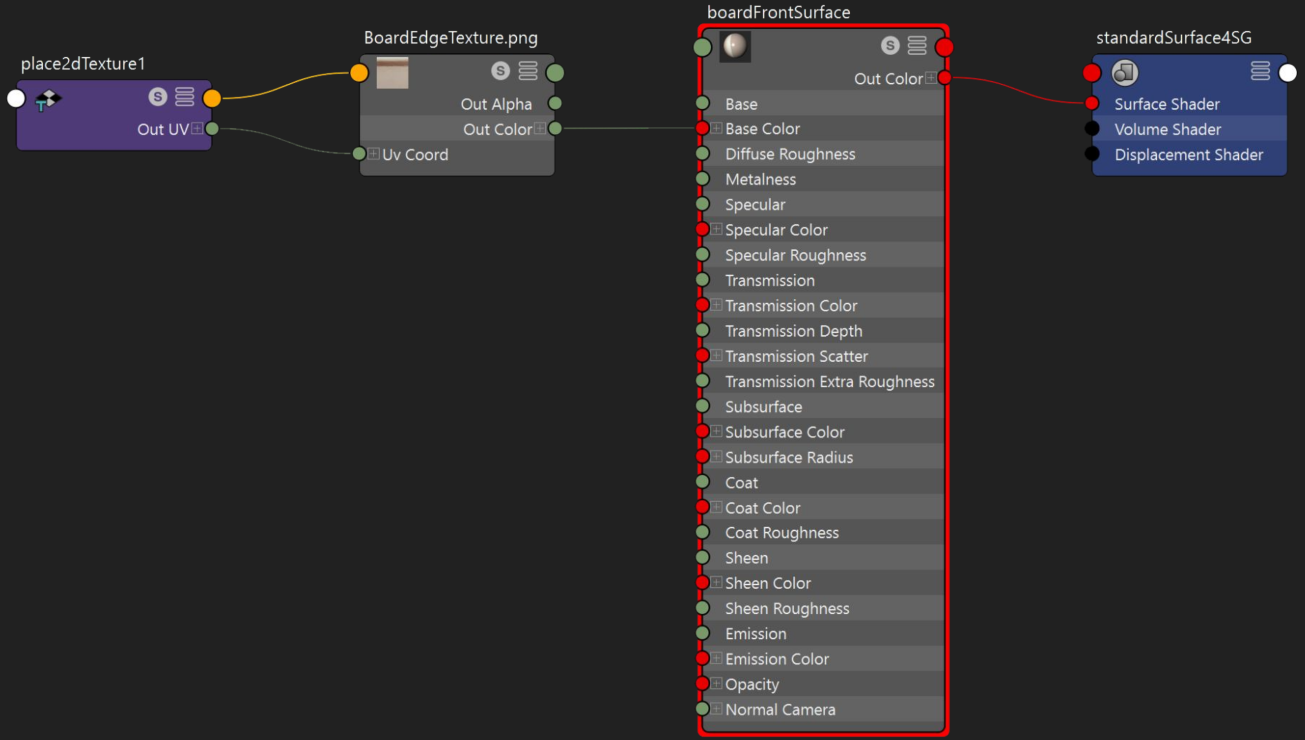
2

3

Main Gameplay Screen

Final Scenario Evalulation Screen

Diagram 2: Example Shader Network



Simple texture mapping shader for 3D rendering

Diagram 3: Example of Patched Conics

Transfer orbit with entry and exit from different spheres of influence.

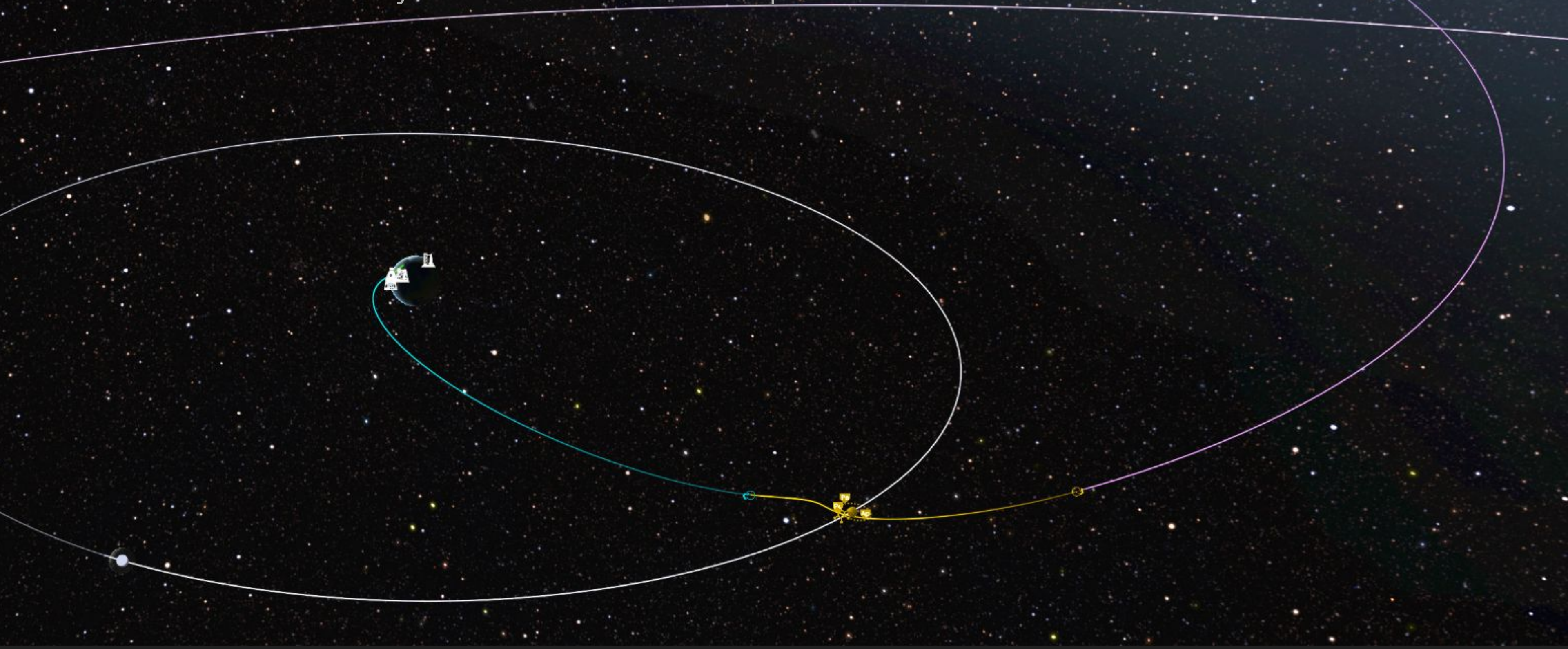
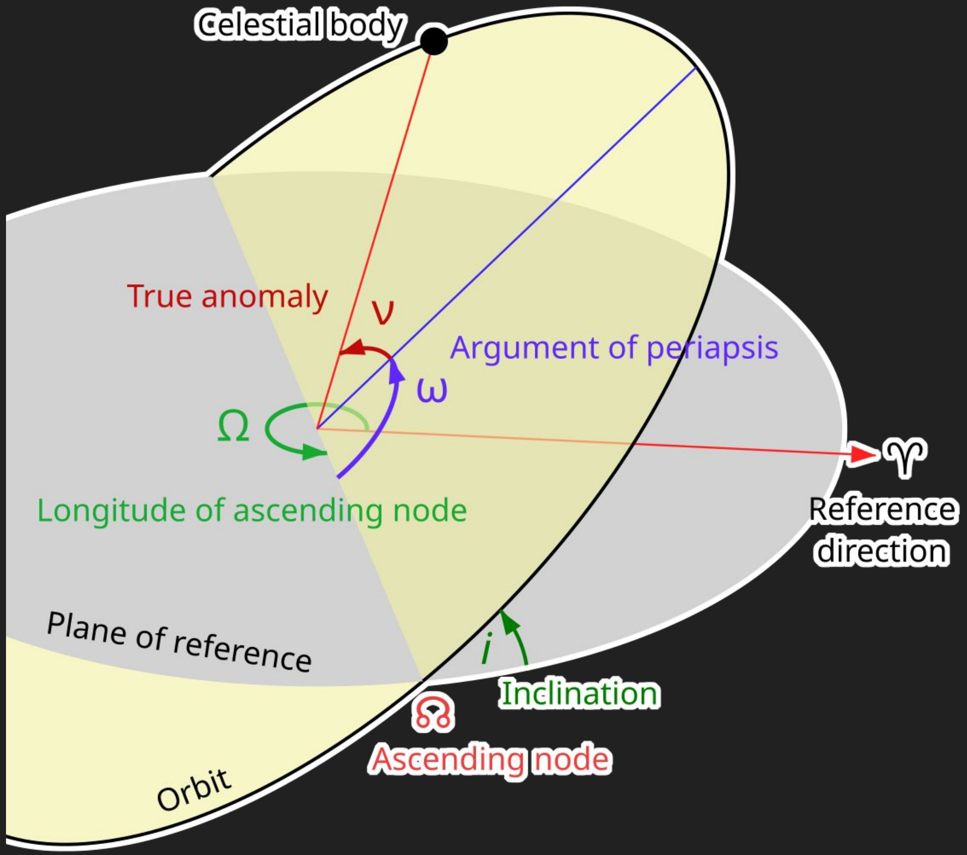


Diagram 4: Keplerian Orbital Elements

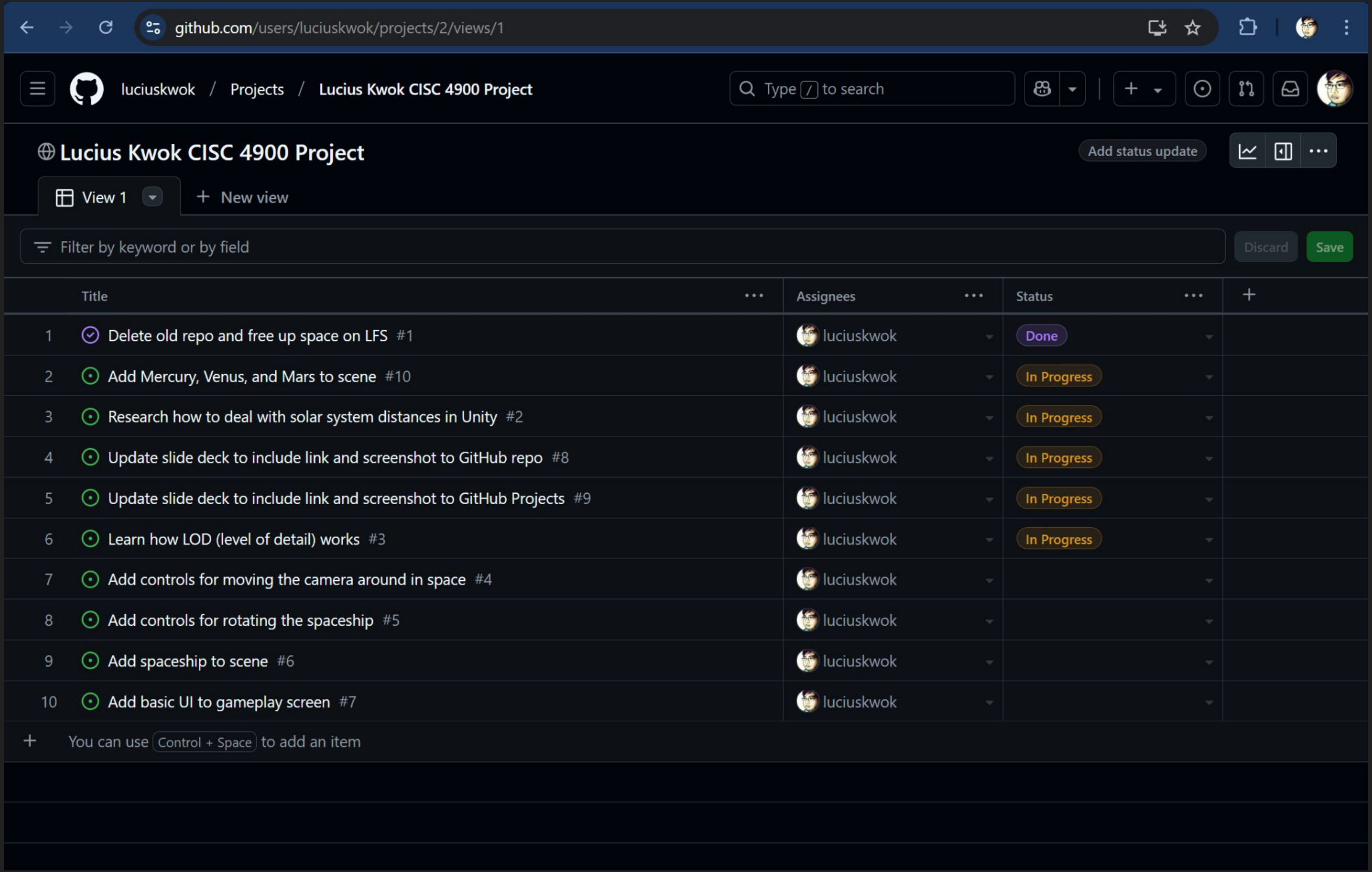
Showing an inclined elliptical orbit



Tentative Schedule

<table>
<tr><th>Week 1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>10</th><th>11</th><th>12</th><th>13</th><th>14</th><th>15</th><th>16</th></tr>
<tr><td colspan="3">Project planning & research. Set up project repo & issue tracker.</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td colspan="3">Identify users, use cases, and proposed solutions. Sketch UI/UX.</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td colspan="3">Develop prototype/demo. User testing, presentation.</td><td></td><td></td><td colspan="2">Further testing</td><td></td><td colspan="2">Final testing</td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td colspan="5">Refine and build out UI and features based on user and presentation feedback.</td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="4">Triage features and issues to decide on MVP</td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="3">Feature lock & focus on fixing issues.</td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="3">Prepare final presentation & submission</td></tr>
</table>

GitHub Projects


github.com/users/luciuskwok/projects/2/views/1
luciuskwok / Projects / Lucius Kwok CISC 4900 Project
Type / to search
Add status update
View 1
New view
Filter by keyword or by field
Discard
Save

Lucius Kwok CISC 4900 Project

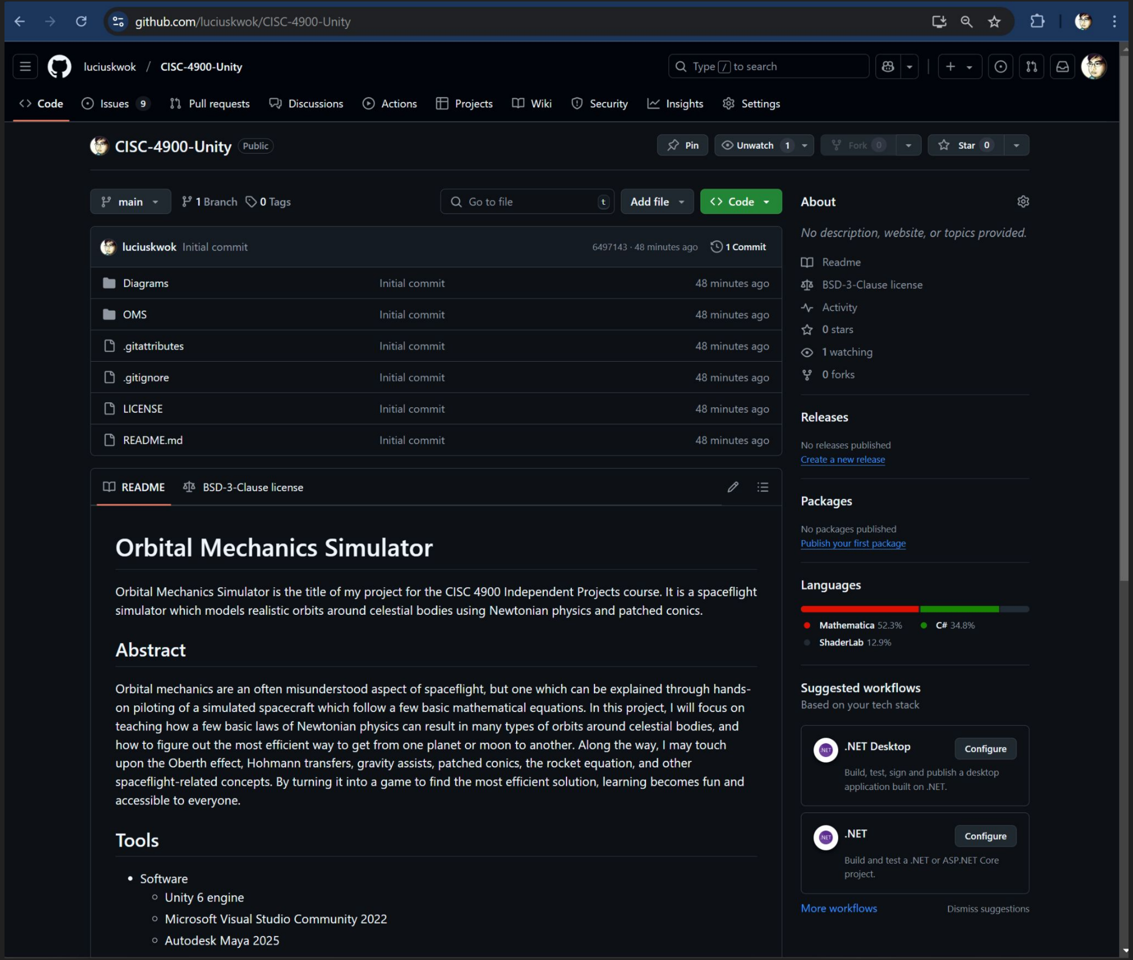
	Title	Assignees	Status
1	Delete old repo and free up space on LFS #1	luciuskwok	Done
2	Add Mercury, Venus, and Mars to scene #10	luciuskwok	In Progress
3	Research how to deal with solar system distances in Unity #2	luciuskwok	In Progress
4	Update slide deck to include link and screenshot to GitHub repo #8	luciuskwok	In Progress
5	Update slide deck to include link and screenshot to GitHub Projects #9	luciuskwok	In Progress
6	Learn how LOD (level of detail) works #3	luciuskwok	In Progress
7	Add controls for moving the camera around in space #4	luciuskwok	
8	Add controls for rotating the spaceship #5	luciuskwok	
9	Add spaceship to scene #6	luciuskwok	
10	Add basic UI to gameplay screen #7	luciuskwok	

+ You can use Control + Space to add an item

https://github.com/users/luciuskwok/projects/2

GitHub Repository

https://github.com/luciuskwok/CISC-4900-Unity



CISC-4900-Unity Public

luciuskwok Initial commit 6497143 · 48 minutes ago 1 Commit

Diagrams	Initial commit	48 minutes ago
OMS	Initial commit	48 minutes ago
.gitattributes	Initial commit	48 minutes ago
.gitignore	Initial commit	48 minutes ago
LICENSE	Initial commit	48 minutes ago
README.md	Initial commit	48 minutes ago

README BSD-3-Clause license

Orbital Mechanics Simulator

Orbital Mechanics Simulator is the title of my project for the CISC 4900 Independent Projects course. It is a spaceflight simulator which models realistic orbits around celestial bodies using Newtonian physics and patched conics.

Abstract

Orbital mechanics are an often misunderstood aspect of spaceflight, but one which can be explained through hands-on piloting of a simulated spacecraft which follow a few basic mathematical equations. In this project, I will focus on teaching how a few basic laws of Newtonian physics can result in many types of orbits around celestial bodies, and how to figure out the most efficient way to get from one planet or moon to another. Along the way, I may touch upon the Oberth effect, Hohmann transfers, gravity assists, patched conics, the rocket equation, and other spaceflight-related concepts. By turning it into a game to find the most efficient solution, learning becomes fun and accessible to everyone.

Tools

- Software
  - Unity 6 engine
  - Microsoft Visual Studio Community 2022
  - Autodesk Maya 2025

About

No description, website, or topics provided.

- Readme
- BSD-3-Clause license
- Activity
- 0 stars
- 1 watching
- 0 forks

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Languages

Mathematica 52.3% C# 34.8%
ShaderLab 12.9%

Suggested workflows

Based on your tech stack

.NET Desktop Configure
Build, test, sign and publish a desktop application built on .NET.

.NET Configure
Build and test a .NET or ASP.NET Core project.

More workflows Dismiss suggestions

Use Cases

1. Astrophysics course to illustrate different kinds of orbits and how velocity varies with the distance.

2. Casual user who wants to play with spaceships in orbit and solve challenging scenarios such as finding the most efficient transfer from Earth orbit to Moon orbit.

3. Video maker (for YouTube, for example) who wants to generate animations to show the orbits of asteroids, satellites, and celestial bodies in their videos.

Data Sources

NASA: Gravitational constants; distances and diameters of the sun, planets, and moons.

University of Texas: Orbital elements.

Wikipedia: Information on celestial bodies (planets, moons, stars).