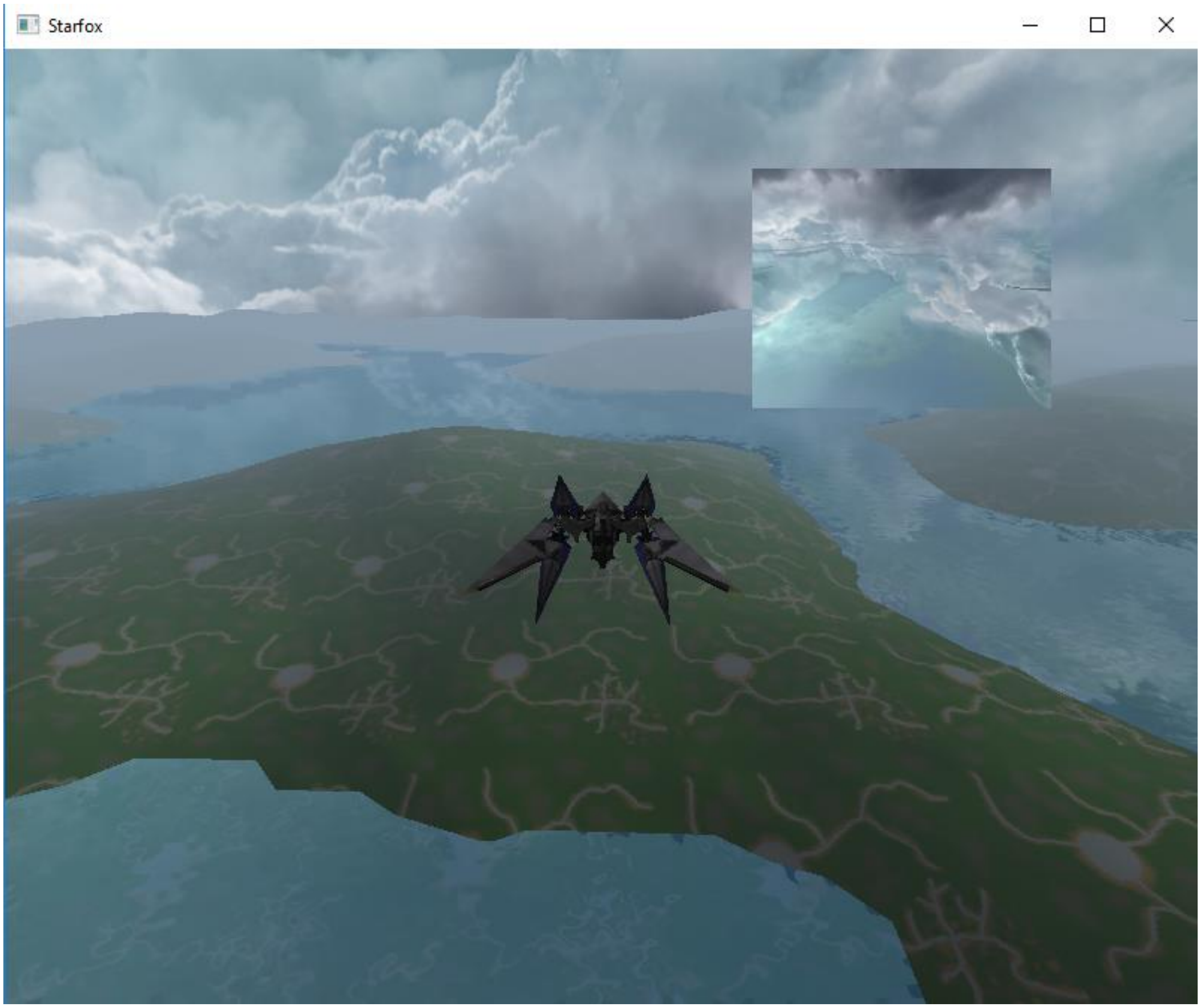
Starfox
Zachary Davids
100796282



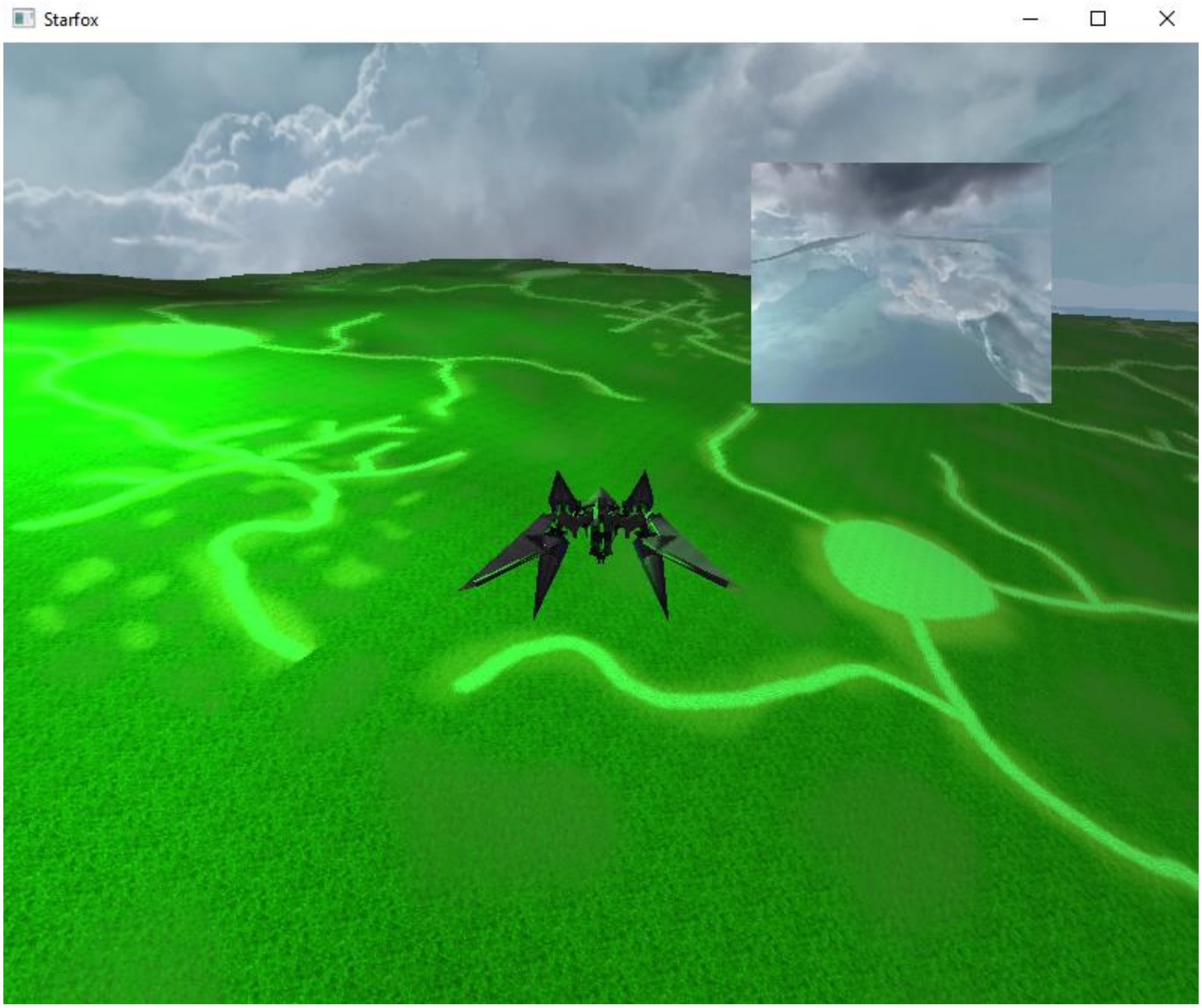
Description

The object of the game is to destroy all of the enemies using your weapons, while avoiding enemy projectiles. The game can be played using WASD controls to move the ship forward, backward and rotate left and right, and using R and F the player's ship can be strafed up and down. Pressing spacebar will fire the player rocket launcher. Using shift the camera perspective can be switched between 1st and 3rd person.

Features

The game features three different enemies, the gatling enemy, laser enemy, and rocket enemy, and each have their own distinctive model and share behavior defined with a deque based finite state machine. These enemies will search for the player and change their behavior states based on the distance from the player, weapon range, and their own health.

The scene is illuminated with a sun, and three point lights of different colors spread throughout the scene which interact with every surface present in the game. A skybox has been included that features day and night cycles that switch between morning, afternoon, early night and night. Water has been implement using environment mapping, using framebuffers to combine the refractive and reflective texture of the scene with time-offset dudv texture to create the semblance of motion. Collision detection between the player and the terrain has been implement but not for any other object. Weapons use hierarchical transformations to take the transformation of their parent entity. There is currently only a single weapon type implemented completely, while the skeleton classes for the others have been fleshed out. The game features very beautiful aesthetics and visuals through object loading and complex height-map based terrain generation, and blend mapping between five textures for ground tiles. GUI has been implemented and currently displays the reflection texture used for water.



Features Missing

2/3 Weapons, particles, collision detection for projectiles.

Technical Notes

All game objects such as weapons, enemies, and the player, are entities which contain a model (used for meshes) and have their own renderer. Terrain, skybox, GUI, and water objects (also using models) all have their own separate shader and renderers that operate differently based on their needs. The master renderer controls all of these specialized renderers from its render function which is accessed from the Game class's main loop. Models are loaded in using the ASSIMP library with code modified from https://learnopengl.com/. Wavefront .obj files were used for all models, found on https://www.models-resource.com/gamecube/starfoxassault/ and modified slightly in Blender. The engine itself was built following Java OpenGL tutorials by https://www.youtube.com/user/ThinMatrix/videos mixed with some code from demos provided for assignments. Unfortunately as a solo group and balancing other course work I did not have time to implement all of the features. I do feel however that I accomplished quite a lot by myself, and while incomplete, I feel satisfied with the amount of features I implemented.