Project Plan: Team Foxtrot

David Hite, Kelby Faessler, Austin Welborn

Introduction:

Our team was formed around the idea of creating a 2D game. We agreed that we wanted to use Unity and C# in the project, so those will be the main technologies behind our game. Several ideas were proposed for the game itself including a Super Mario clone and a side-scrolling shooter such as Metal Slug. Ultimately we decided to stick with the side scrolling theme but based in the Star Trek universe. The game will be similar to games like Gradius and Tyrian.

Program Description:

We'd like to create sprites of space ships that would represent several different popular ships from Star Trek such as the U.S.S Enterprise B, U.S.S Enterprise Vengeance, and several others. Similarly, we would create sprites that represent popular enemies in Star Trek such as Klingons, Borg, and Romulans, etc.

The game would be created in such a way where each level would have enemies come from the right screen while the player's ship moves at a constant speed towards the right. The player can also maneuver their ship in a 2D plane across the screen along the x-axis and the y-axis. Movement in both directions simultaneously would allow the player to move diagonally. Each level would have plenty of different enemies that the player will try to avoid and fire at using their ship's specific weapons. At the end of each level would be a unique boss such as a Romulan mothership, or others (not wanting to ruin the surprise here).

The player would select a ship then have a limited number of hit points that they can sustain from enemy fire or running into obstacles. After a player sustains a certain number of hits, the ship will be destroyed and the player will have to use a coupon in order to continue.

The enemies will be represented by different sized sprites and have different firing sequences and different number of hit points, so the game will be fun and challenging. Some enemies will be a single hit enemy and others would be 2-3 hits, and bosses would be upwards of 20-40+ hits depending on the damage the player's ship can do. Also, we are thinking of adding different kinds of guns that the player's ship can obtain through the game by beating certain enemies or bosses which would drop a gun that will upgrade the ship's firing damage and/or hit multiple enemies at once.

Software Structure:

Unity is a platform that has a lot of game making functionality built in, but there is plenty of software development to do as well.

Many properties are easily configured in Unity, such as physics-related properties and animations, but some things such as user input and animation transitions are best done in C#.

Each level can be represented with a scene object that has a variety of objects and properties inside of it. The scene contains a player object, a camera object, enemy objects, and obstacle objects.

Software Libraries, API's, and Tools:

The Unity engine will be our main development tool. We'll use Unity to include sprites in our game as well as include the C# scripts that control the behavior of the game. We may make use of 3rd party sprite models, although this has not been decided yet. If we create our own sprites, we may use graphical design tools for the sprites such as Gimp, GraphicsGale, or Firealpaca.

Team Assignments:

David

Player control handling, player and enemy health setup, object collisions

Kelby

Level generation, Menu creation, Camera/Side-scrolling effect, Level scoreboard

Austin

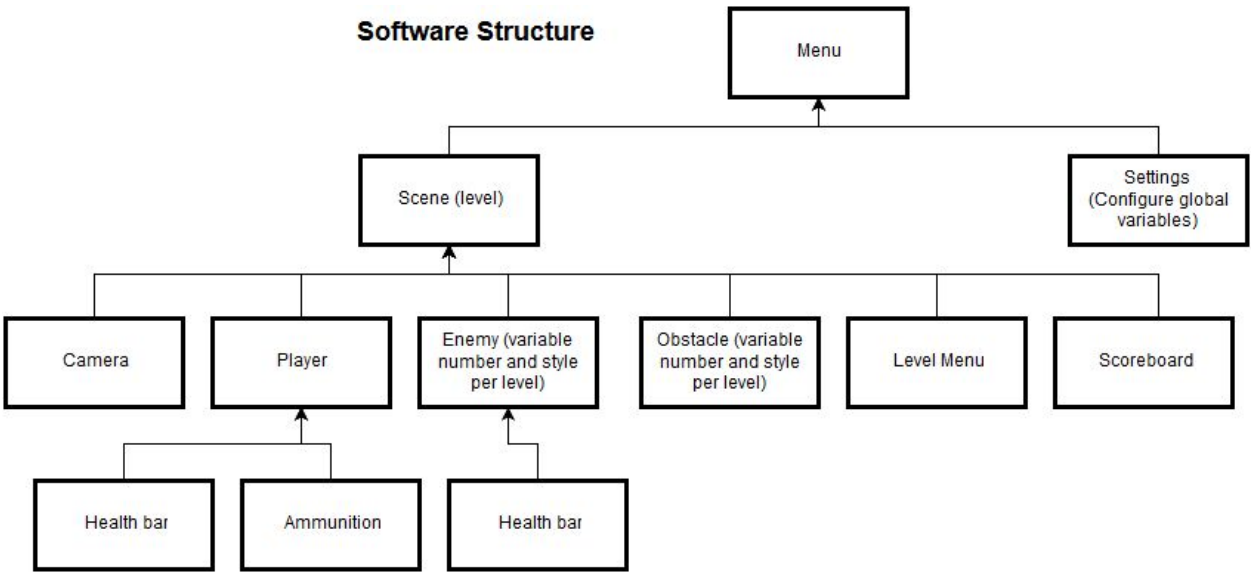
Creating sprites for player ships and enemy ships, "bullets" (firing and damage).

Time Management:

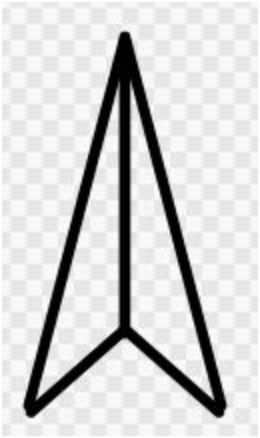
Each of us expects to spend about 10 hours a week on this project, with more time some weeks, for a total of 100 hours each. We will likely expand the project to get it sufficiently complex to keep us busy.

Graphical Examples:

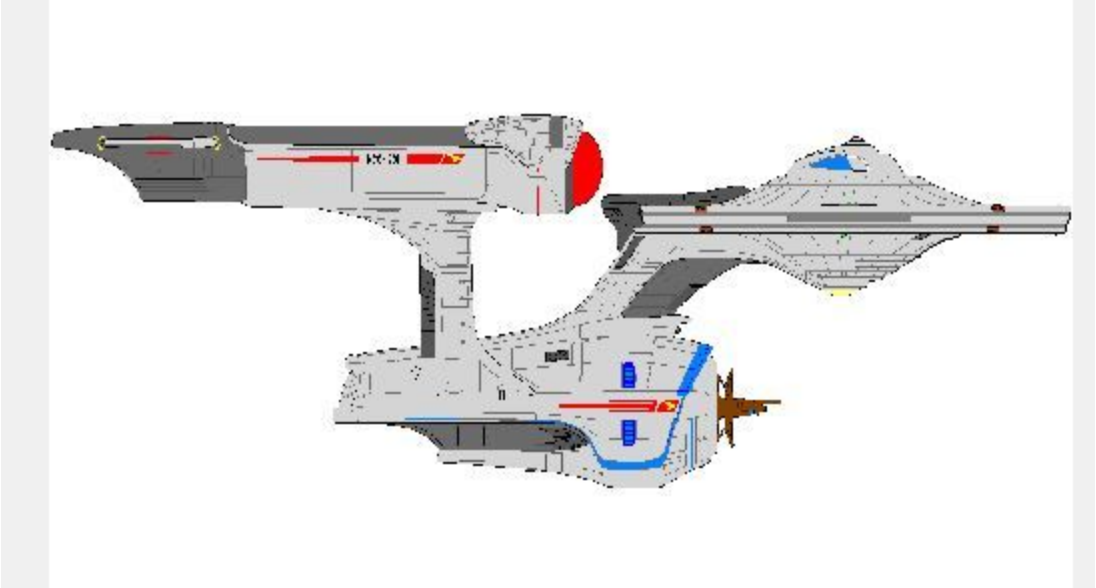
The structure of our software architecture:



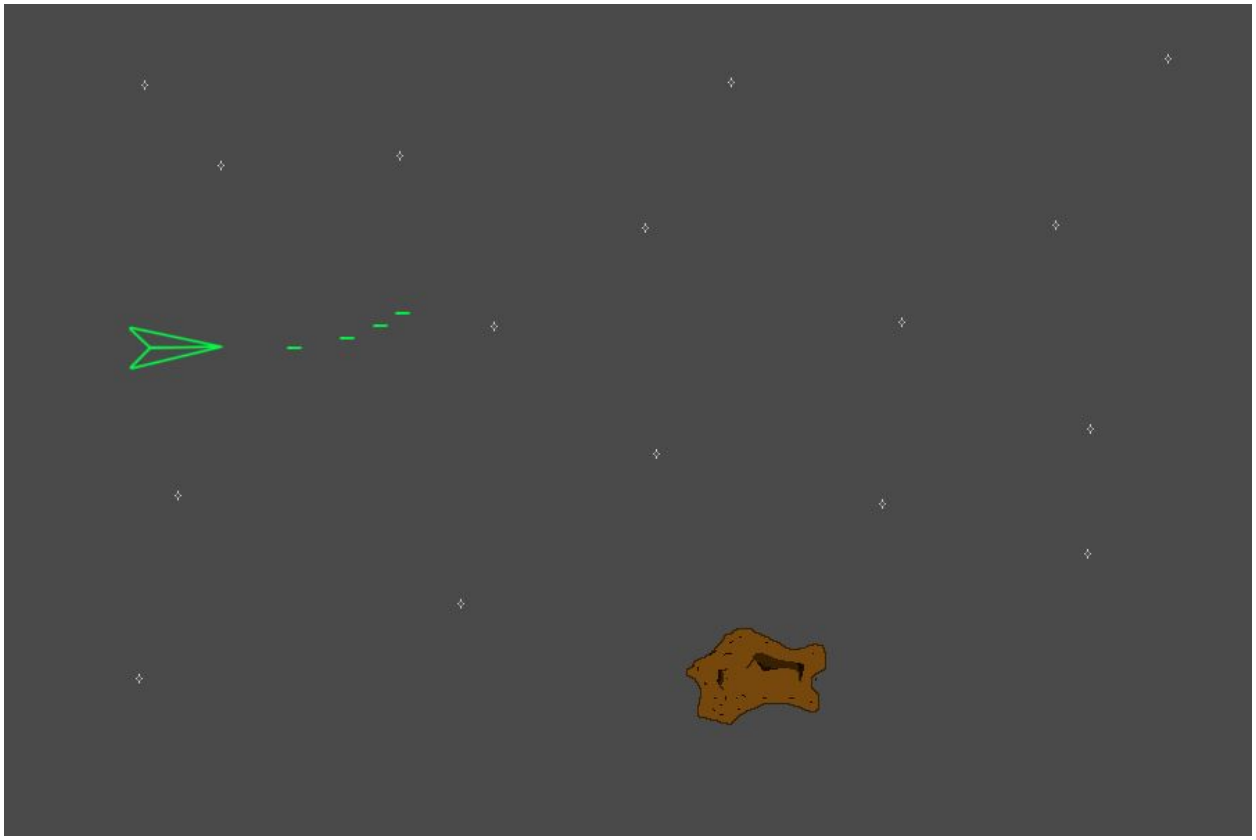
Example basic ship sprite (we'll start with basic geometries at first and add detailed sprites later):



Example detailed ship sprite:



Example scene, with ship shooting and asteroid obstacle (stars in background). No enemies present



Conclusion:

Overall the game would be a simple 2D side-scrolling Star Trek Space Ship shooter. The player gets to choose from a number of ships and maneuver around the screen and avoid or fire at enemies that come from the right edge of the screen. The goal is to get a high score, collect different guns and experience the universe of Star Trek and its unique races, ships, and environments.