

Term Project Proposal

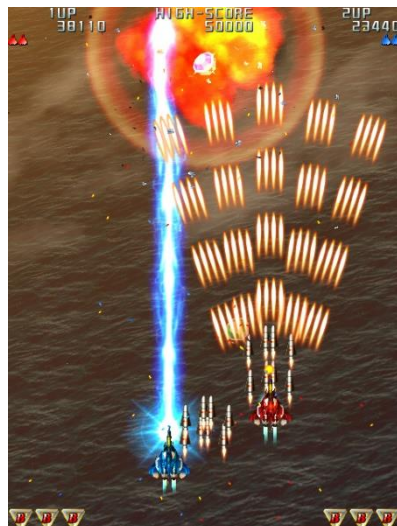
Name: Junke Zhao

AndrewID: junkez

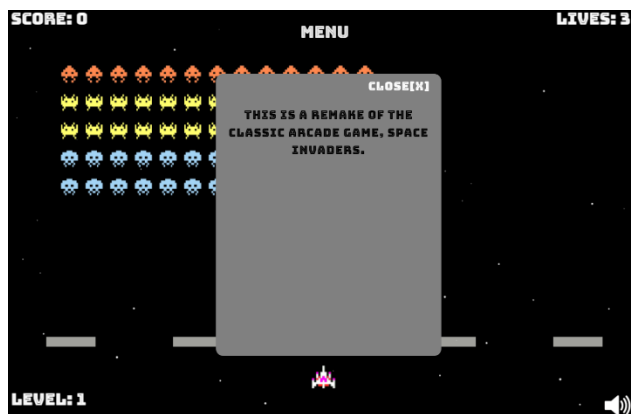
Project Name: Thunderbolt112

Project Description: The world's first video game was the famous Space War, and this is a classic horizontal flying shooting game. I hope to pay homage to this game in this project by developing a similar flying shooting game.

The game was developed with reference to the game Thunderbolt on the Sega Genesis. The player operates the plane in the center of the screen as it navigates through the universe and the space base, attacking enemies by shooting, dodging obstacles and enemies in the scene, and finally reaching the destination.



Similar Projects: One of the most famous similar project is Space Invaders Remake. (url: <https://buckyz.github.io/SpaceInvaders/>) and Space Shooter. These two games are implemented well using python, and a lot of images and sound clips are used to make the experience better.



In my project, I'll probably involve more scenarios and levels, as well as a greater variety of enemies and attacks, which will probably involve the use of a lot of graphic material. Also, both projects use the python library pygame, which is not currently involved in using in my project.

Structural Plan: The main objects in the game include the player-operated airplane, enemies, bullets, drop objects and obstacles. The player character will be a separate class, containing values such as current level, life, attack mode, speed, etc. Enemy planes are in a different category, containing the enemy's life, attack power and means of attack, etc. Bullets are divided into two categories, player and enemy, and have similar effects, including direction of movement, speed, attack power and other values. Additionally, Drops and Obstacles are two similar classes, the difference being that when the player comes into contact with them, one causes the player to gain a buff (attack power, life, etc.), while the other causes the player to lose life.

The classes above will be organized into different python files. Besides, there will be a main game script (main. py) to manipulate the running of the game, and some additional files to control the outside input, like images, sound and other resources.

Algorithmic Plan: One of the most challenging part is to add enemies' AI, and I make some preliminary conceptualization:

1. Creating Enemy Movement Patterns: Create a set of predefined paths or algorithms to simulate realistic flight patterns, making enemies harder to predict, such as straight paths, zigzags, loops, and random trajectories.
2. Customize Targeting Strategies: Implement varying targeting strategies for enemies. Some enemies might directly target the player, while others may prioritize avoiding the player's shots, or advanced targeting logic, , such as predictive targeting, where enemies anticipate the player's future position based on their current trajectory.
3. Dynamic Generating: The generation of enemies may not be fixed. It's a good way to generate dynamic enemy based on the player's progress and performance.

Timeline Plan:

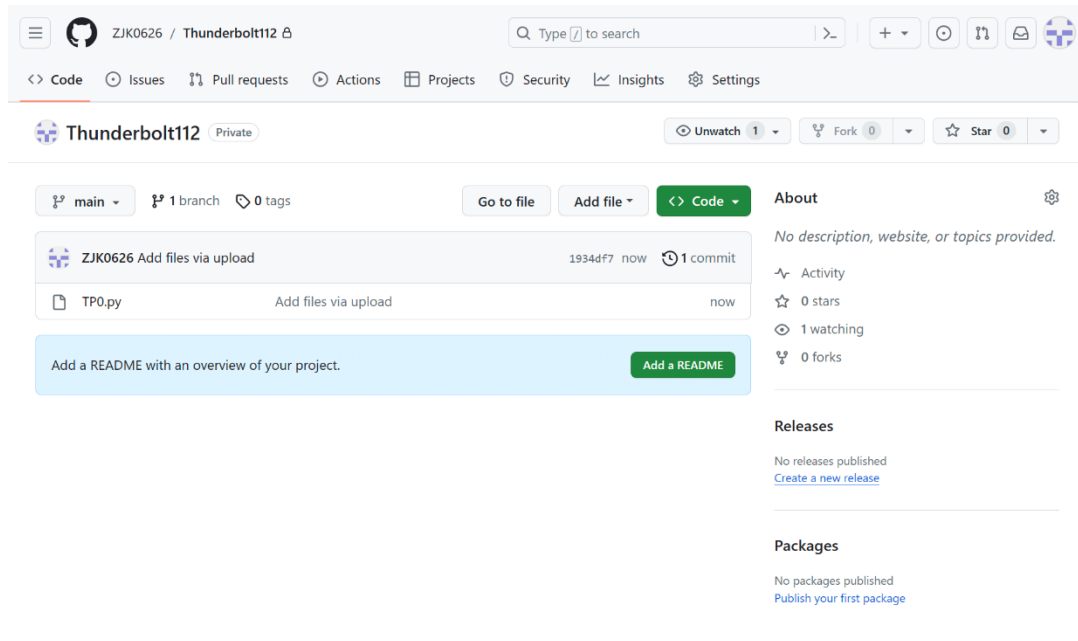
TP0 (Nov. 20) – Submitting preliminary proposal;

TP0-TP1(Nov 20 – Nov 27): Implement basic functions, such as the player can move around, attack enemies, and dodge player attacks. Completing most of the user interface design.

TP1-TP2(Nov 27 – Dec 1): Optimize enemy AI, try to add more levels, enemy types and attack styles.

TP2-TP3(Dec 1 -Dec 6): Import art and music clips into the project to make it look better. Also optimize the game functions.

Version Control Plan: I will use GitHub for the version control, and at the same time use Google Drive to back up my code:



Module List: None

TP2 Update:

1. A full-functioned level 1
2. Player and enemies can shoot bullets now
3. Bullets will cause damage
4. Add two types of obstacles
5. Add new types of enemies and bullets
6. Create a winning page
7. More art resources and UI design