Project Title: Bullet Hell

Group Members: Lewis Cook Jakarta International School

Jakarta, Indonesia

Jan Lin American Heritage School

Plantation, Florida

Daniel Olis Ridgeview High School

Orange Park, Florida

Wesley Wang Cupertino High School

Cupertino, California

In the early 1950's the first video games were invented. They reached mainstream popularity in the 1970’s with the appearance of games such as Pong, Pac-Man, and Space Invaders. Games then began to evolve over time, becoming more complex with a wide variety of genre’s, better graphics, and greater user interaction. However, the games of today require new consoles with greater data storage and processing capability, and are difficult for someone new to gaming to play due to their complex rules, reliance on familiarity with a gamepad and events where the player is forced to react in less than a second to certain game elements and stimuli. This project aims to create a fun game that is less technologically demanding and is friendly towards the less experienced gamer.

Bullet Hell is a video-game subgenre where the player must dodge projectiles thrown at them by various enemies and ultimately beat the final boss to complete the game. Compared to games currently in the market, there are few of its kind, giving the subgenre a relative novelty. Also, due to its simplicity in game objective and mechanics the game can be easily learned, even by those unfamiliar with videogames.

In this project, Python was used, an open source programming language that allows for efficient audio and visual rendering along with Pygame, a set of Python modules designed for game creation. Users can click the start button in the menu to initiate gameplay. Using the keyboard, the user controls the movement of their playable character. Enemies spawn automatically at the top of the screen. By pressing the spacebar, the ship then fires bullets that destroy enemies upon collision. If the character is hit by an enemy bullet they lose one of three lives and temporarily gain invincibility so they can recover. The game finishes when the user loses all lives, then gives the user a score based on performance. The game then compares, and ranks the user’s score among other local scores