CyberShooter

Official Game Documentation

This game was designed, developed, and tested by Group U which consists of the following members:

- Adrian-Gabriel Aduculesei
- Ertugrul Akay
- Anton Nyagolov

GitHub Repository Link: https://github.com/antpngl92/GameDev

Gameplay Video Link: https://www.youtube.com/watch?v=gRp8_5jIVLs&feature=youtu.be

Concept

Our game is an arena-shooter inspired run-and-gun FPS that requires players to defeat waves of enemies and collect power cells in order to progress through the levels. Each level has a certain number of waves that the player must defeat, and a certain amount of hidden power cells they must collect in order to advance. Each level also features a new weapon unlock that the player can use. The player initially starts off with the Plasma Pistol and can acquire the Machine Gun and the Shotgun when reaching Level 2 and Level 3, respectively.

There are various types of enemies in the game, including melee units that follow the player and attempt to attack at close range, as well as stationary gatling turret enemies that will only engage the player when they are within range. Each level features new, visually distinct enemies. Each wave for each level also presents tougher enemies for the player to fight.

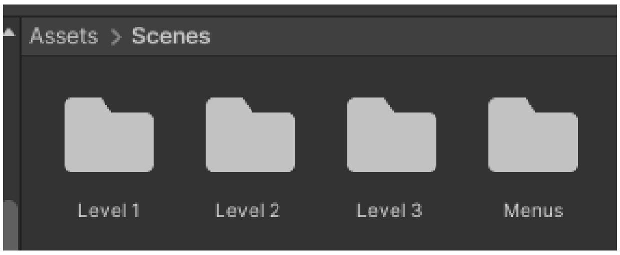
How to play

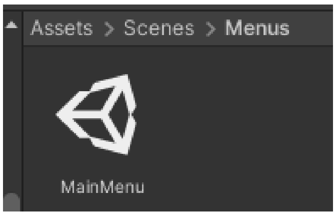
Starting Unity Project

To start the unity project, add the project folder located at “GameDev\ScifiShooter\MultiPlatform Game Dev Project” to Unity, and then launch.

Starting the game

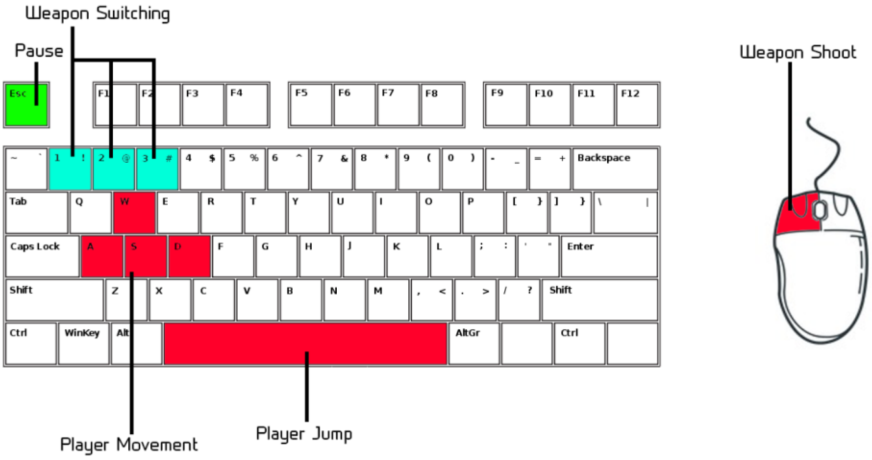
The game features a total of three levels (named Level 1, Level 2, and Level 3) that reside in the “Scenes” folder. To start the game, one can start the StartMenu Scene first (Located in the “Menus” folder in the “Scenes” folder) in order to adjust volume settings if needed and then start the game. After that, the game follows a sequential order of levels (Level 1 leads to Level 2, and Level 2 leads to Level 3) after the completion of a level.





Assets > Scenes > Menus

Game Controls



Implemented Features

Environment

- Level 1
- Level 2
- Level 3
    - All three levels are made using a Sci-fi modular asset pack and other sci-fi props.

Audio-visual

- Audio
    - We have various sounds throughout the game ranging from music to player and weapon sounds.
- Lighting
    - We make use of Realtime lighting within the game and our levels feature a neon-style of lighting.
- Cameras
    - We make use of a main character camera through which all action is viewed, and also a separate gun camera that renders only the weapon to avoid wall clipping issues.
- Particle Effects
    - We make use of a Particle Effects pack that is used when shooting weapons and enemy death particles.
- Materials
    - We use own-made materials to create various environment assets and also make use of some shader material assets for various props like forcefields and water.
- Animations
    - We use a simple own-made animation for the weapon holster which applies to all weapons; this animation is played when the weapon is being reloaded. We also use

    scripts to add a Sway and Recoil effect to weapons when the character is moving and firing.

Gameplay

- Player Controls
    - Our player is fully controllable using Keyboard and Mouse controls.
- Interactions with entities
    - We have a pick-up system where the player can walk over items such as health, ammo, and power cells to pick them up and restore statistics and advance the game.
- NPC Behaviour
    - We have various enemies within the game that act as the main opponents of the player.
- Difficulty Levels
    - Each level has various waves of enemies, and these enemies get progressively more difficult due to statistical improvements.
- Player adjustable parameters
    - We have player adjustable parameters within the options menu (see non-gameplay).

Non-gameplay

- Main menu
    - Features a "play game button", a "options" button, and an "exit game" button.
- Options menu
    - Features a volume slider that controls the volume for the entire game
- Pause menu
    - Can be brought up using the Escape key and completely freezes the entire game until resumed.

Asset List

- An asset list is included within the README.md file on our GitHub repository.