HGD Progress Report

Name: Jay Hawkins
Course number: ENT2960
Team #: 5
Team/game name (if known): Team 5 Shorty Shooters
Current semester (eg., Fall 2030): Fall 2024
Expected game completion (e.g., Fall 2030): Spring 2025

Sprint 4

Completed & In-progress tasks

Update Environment Assets (100% Complete)

Commit link (or other evidence of completed work):
https://github.com/HuskyGameDev/2024f-team5/commit/c3ae741c19686e41cade7e8288d0e8cb844e6076
Story: As a user I want a good looking map or something

Screenshots/videos/GIFs:
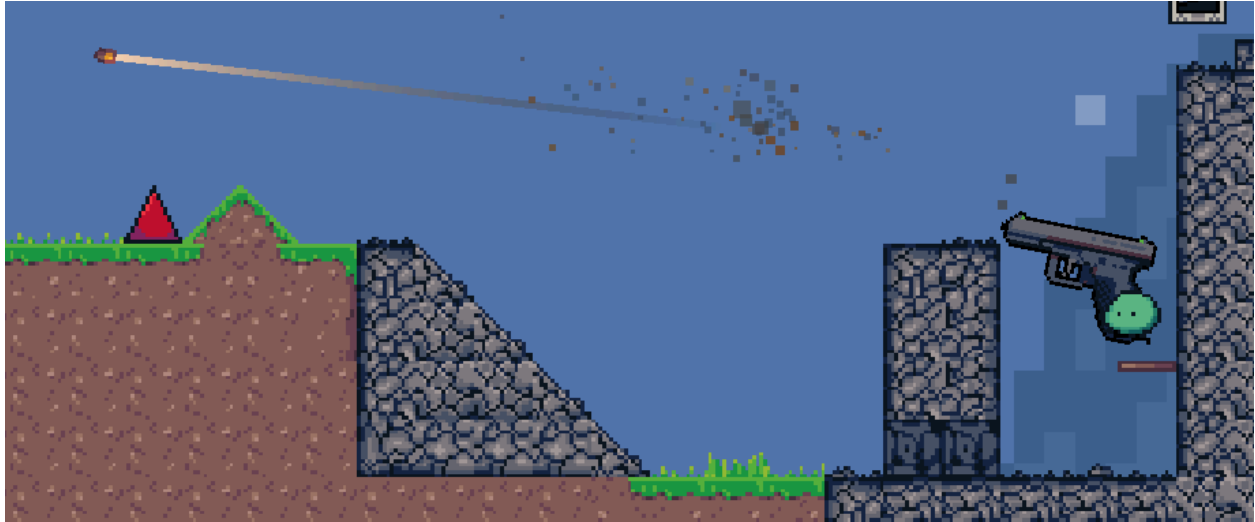

Old: New:

Create Bullet Trails (100% Complete)

Commit link (or other evidence of completed work):
https://github.com/HuskyGameDev/2024f-team5/commit/174575d408234ad1eb9e4a446b66ef5292b48033
Story: As a user, I want the bullets on screen to have good visibility so I can understand what is happening better.

Screenshots/videos/GIFs:

Add Weapons and Weapon Animations (100% Complete)

Commit link (or other evidence of completed work):
https://github.com/HuskyGameDev/2024f-team5/commit/f4b6cd4a22af08080db700cbe6c35fa0f012a4c6

Story: As a user, I want variety between different weapons so that I can explore more content and experiment with the sandbox.
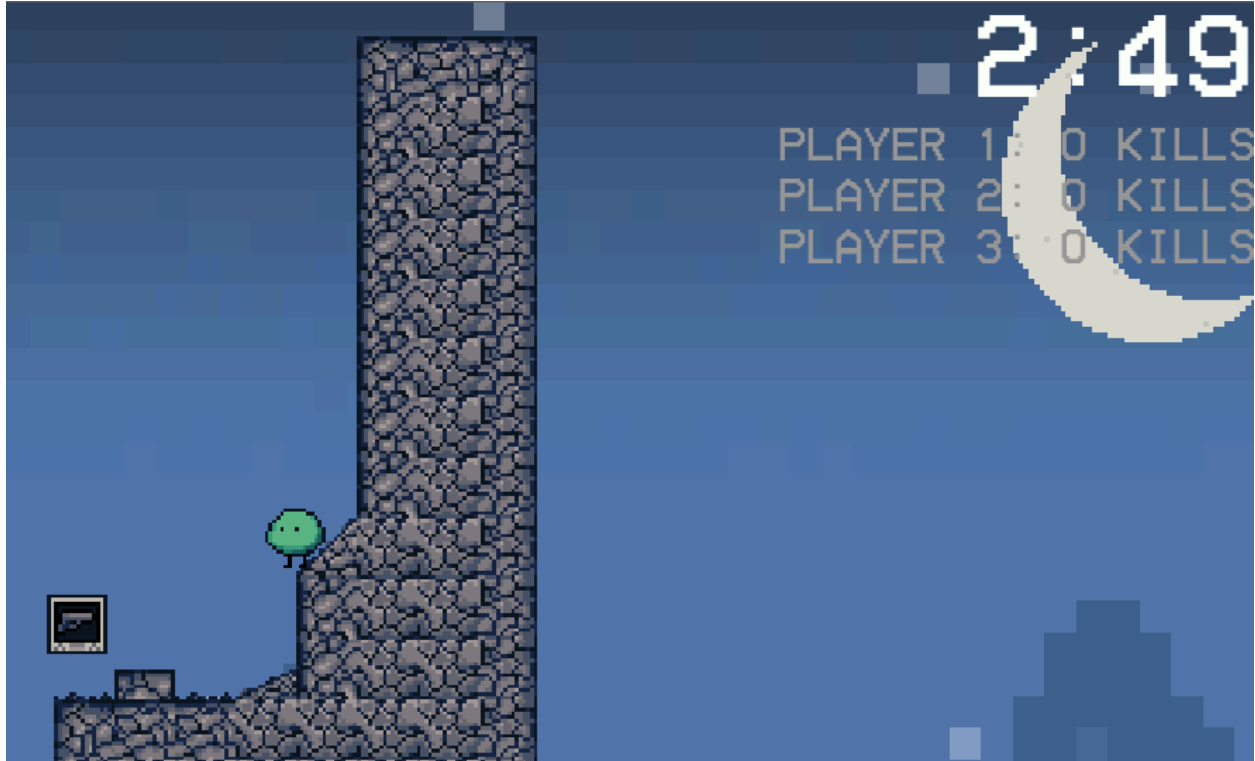
Screenshots/videos/GIFs:

Deathmatch Gamemode Logic (50% Complete

Commit link (or other evidence of completed work):

Story: As a user, I want an objective so that there is meaning behind my actions

Screenshots/videos/GIFs:

Roadblocks

If there are no roadblocks, you must state that. Do not leave this section blank.
If there ARE roadblocks, include them here AND a solution for them.

A current road block is that the gamemode data isn't synced between players. This hasn't been fully fixed, but the solution is to handle gamemode data from the host instead of locally.

Sprint 3

Completed & In-progress tasks

Reformat Code (100% Complete)

Commit link (or other evidence of completed work):
https://github.com/HuskyGameDev/2024f-team5/commit/df90591df41ec94bfde1aa1f6b3b9

Story: As a developer, I want to have easily readable and organized code so I can develop new features easier and so that I can showcase this work in my portfolio.

Screenshots/videos/GIFs:

```gdscript
## % of grip strength to hold current item
var grip: float = 100


# ====================== [ CLASS METHODS ] ====================================

## Handles *most* player animations
func _animate() -> void:
    # Reset animations
    if(_crouching && Input.is_action_just_released("crouch")):
        uncrouch()
    if(anim.current_animation == "look_up" &&
      Input.is_action_just_released("look_up")):
        anim.current_animation = "idle"

    if is_on_floor():
        if(Input.is_action_just_pressed("look_up")):
            anim.current_animation = "look_up"
    else:
```
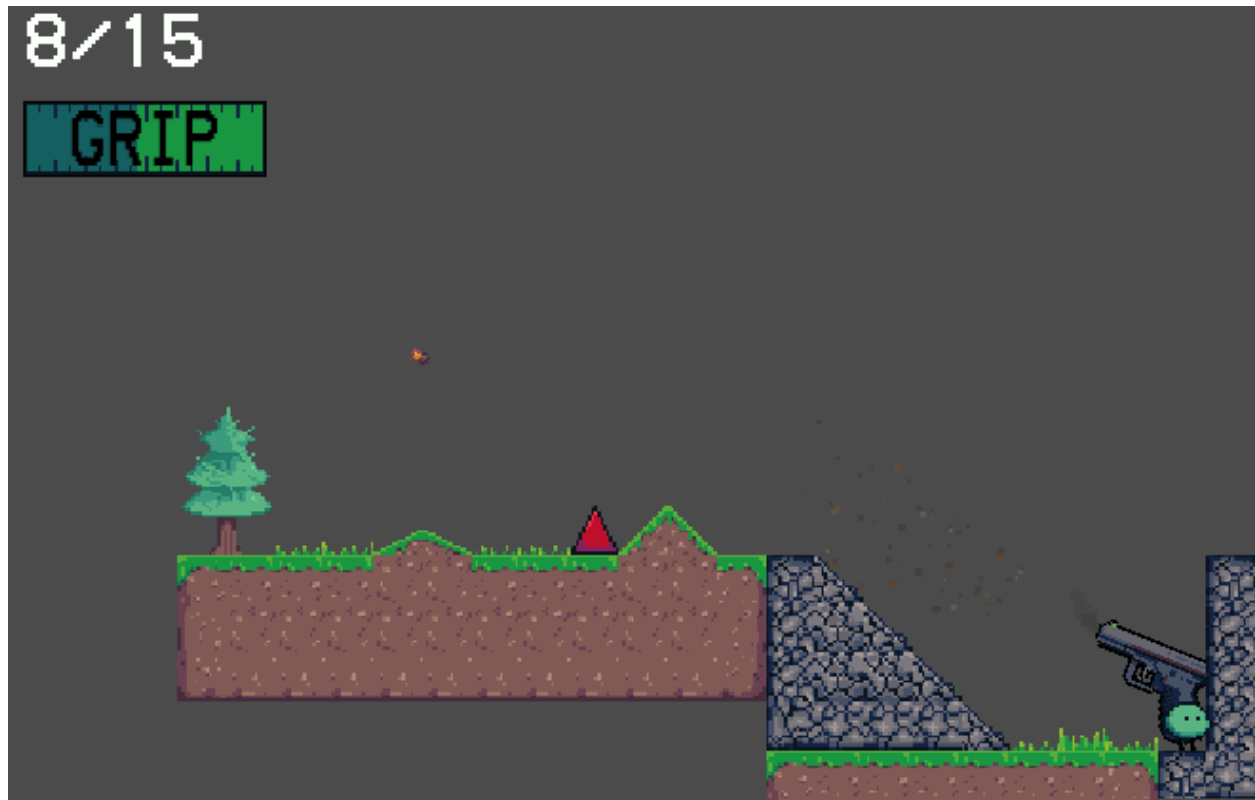
Losing grip mechanic (100% Complete)

Commit link (or other evidence of completed work):
[https://github.com/HuskyGameDev/2024f-team5/commit/4ef40923d99782b33ab361768b5e
d97f6144344b](https://github.com/HuskyGameDev/2024f-team5/commit/4ef40923d99782b33ab361768b5ed97f6144344b)

Story: As a player, I want my gun to block projectiles to a certain extent and have a risk-reward system for shooting so that the game is more challenging.

Screenshots/videos/GIFs:

Roadblocks

Godot doesn't natively have a trail effect like Unity has, so the bullet isn't so visible on the screen. I will create a custom bullet trail script eventually to help improve visibility.

Goal Setting

Goals AND a plan to accomplish them (time put aside (# of hours), where you will look to find new material, etc.) **Not required for Sprint 4.**

I hope to have more weapons done next sprint and have the movement and shooting feeling solid. (Expected: 5hrs)

Sprint 1&2 (Game started this semester)

Completed & In-progress tasks

Basic Player Animation (100% Completed)

Commit link (or other evidence of completed work):

https://github.com/HuskyGameDev/2024f-team5/commit/62e89f2286108effdf27451581aacbf041 33976b

Story: As a player, I would like to see the character I control respond to the movement I input so the game will be more visually appealing and easy to read.

Screenshots/videos/GIFs:

Gun aiming mechanics (100% Completed)

Commit link (or other evidence of completed work):

https://github.com/HuskyGameDev/2024f-team5/commit/aafedba2cb78cbb5cad7c3008303175c 1f1a8b5f

Story: As a player, I would like to have accurate and responsive aiming mechanics so that I can accurately predict where my bullets will land.

Screenshots/videos/GIFs:
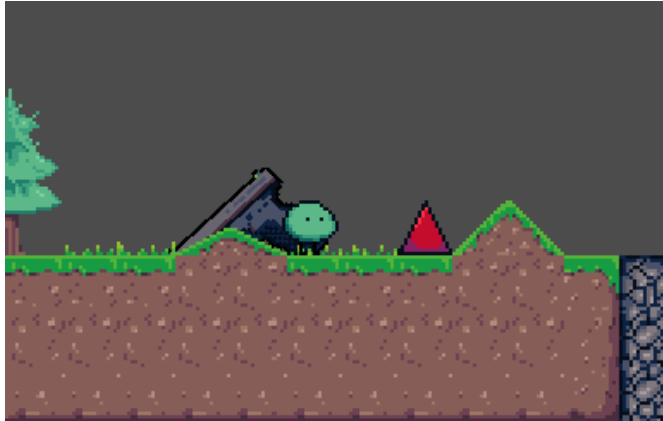
(it's looking at the bottom right because that's where my mouse was when I was selecting the area for the screenshot)
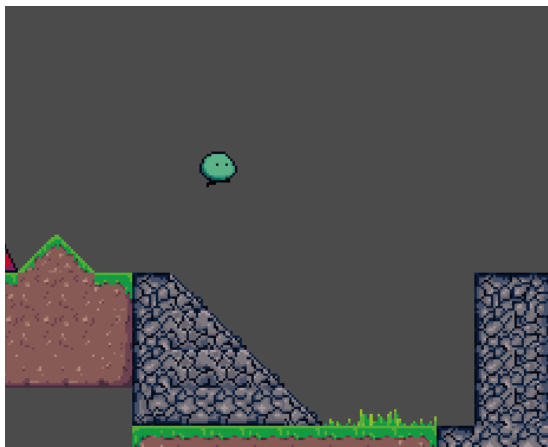
Basic Player Movement (100% Completed)

Commit link (or other evidence of completed work):
https://github.com/HuskyGameDev/2024f-team5/commit/b8f596ed06221cfe74f3a69af0844656e4321bde

Story: As a player, I would like to be able to control my character so that I can actually play the game.
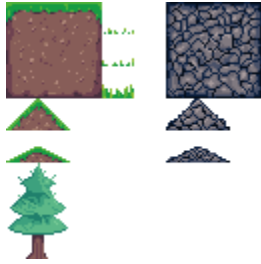
Screenshots/videos/GIFs:

Environment Artwork (25% Completed)

Commit link (or other evidence of completed work):
https://github.com/HuskyGameDev/2024f-team5/commit/4ed177ca1889b7248ab2a3a8d71b6b7578156e2b

Story: As a player and viewer, I would like to see variety in the looks of maps so that my experience can be more varied.

Screenshots/videos/GIFs:

Roadblocks

Godot's AnimatedSprite2D node was very easy to use, but it required each animation to be changed if you wanted to use a different sprite sheet. As a team, we decided we wanted to have skins, so it would have been very difficult to make animations for each skin had we continued using this system. I reworked the animation to use AnimationPlayer2D and Sprite2D so that only the spritesheet needed to be changed to apply to all animations.

When aiming, there is a distance between the anchor and the barrel that makes the aiming slightly inaccurate by about 10-30 degrees (varying by angle). Using some trigonometry, I was able to fix the difference and allow for the gun's barrel to actually point at the mouse.

Goal Setting

I plan to continue developing the gameplay to the point where there's an actual gameplay loop. By the end of the next sprint, I want to have a full map, weapon spawners, and complete the weapons by adding projectiles in order for there to be a semblance of a real game.