Project Name: The Hidden Path

“The Hidden Path” is a level-based game where you play as a character who is inside of a maze-like box, and has to find the exit. However, there’s a catch. The walls can be changed whenever the character presses SPACE. This means that walls will pop up where they were previously empty, and some other walls will disappear. Your goal is to use this mechanic in order to make your way through the maze and out the exit.

Analysis:

This game is influenced by another game called “Close Your Eyes”. In this game, closing your eyes allows you to see things in the world that you couldn’t see before. There is a specific section of this game in which you are going down a hallway, being chased by an entity behind you. In order to make it to the end however, you have to keep closing an opening your eyes, removing obstacles in front of you. This is similar to my project “The Hidden Path”, as it’s based on the same principles. To get through the maze, you have to constantly switch between two different views. The only difference is that in my levels, there is no entity chasing you. Instead, there’ll be a timer ticking down until it reaches 0.

Structural Plan:

There will be multiple files each with classes and methods that contribute to the final project. The first file will be used with a class that allows one to process a maze’s properties. The second file contains the classes that creates all the object to contain all the sprites to be used in the final game. The third file will contain a class with methods for when there is a result to a level. The fourth file contains the class that opens a level. This file will use all the classes and files previously mentioned. The Fifth File will use all these classes and potentially more in order to create the main menu, and the methods to access the different levels of the game.

Algorithmic Plan: How to Make Walls Disappear and Reappear.

In order to do this, I will create two separate images of the same maze. The difference is that one maze will have some walls and openings that the other does not have. I will set one of these images as the image on the screen. Then inside the pygame main loop, I’ll have a loop that checks for every event that the game causes. Whenever the event is the player pressing SPACE, I’ll switch the screen’s image to the other maze. This will allow them to change the walls or their surroundings.

However, this doesn’t address everything. I need to be able to make a wall that’s solid disappear, before reappearing into a solid wall. To do this, I’ll make all the walls black, then prevent the character from moving onto a space where the screen is black. This way, when the screen image changes, the character will be able to move right on through.

Timeline Plan:

I intend to finish the following before Tuesday October 25th:

* Creating all the Levels and their Images
* Creating a Main Menu where the player can play through each level one by one
* Creating an Endless Mode, where the player can be chased by an enemy while they attempt to gain coins.

I intend to finish the following before Tuesday November 1st:

* Creating a Shop Menu to Use Coins
* A Leaderboard on the Endless Mode for people to compare their scores to others
* Any Other Finishing touches like Character Animation and Sound Effects

Module List:

* Pygame: for main display and controlling character
* ImageWriter: for potentially processing images of the mazes
* Sys: Primarily for the Exit Function

Possibly:

* Tkinter: for Main Menu and Shop Menu