Final Project Devlog

11/30:

- Preproduction
    - I decided to create a 3D game where the player must navigate a course as quickly as they can.
    - To-do:
        - Implement player movement.
        - Get the camera to track the player.
        - Implement a timer.
        - Create a track of platforms.
            - Including some platforms that can break/disappear.
        - Implement windmill-style blades that the player has to avoid.
        - Implement a game-over screen when the player reaches the end (success) and when the player falls off the track (failure).
- Aesthetic Goal One: The game should require practice for the player to feel successful.
    - Signs of success:
        - The game is difficult but becomes easier the more times it is played.
        - Players want to win.
        - Players are able to tell how good or bad they are doing.
    - Signs of failure:
        - The game is too easy to feel worth playing.
        - Players don't care about winning.
        - Players are not able to tell how good or bad they are doing.
- Aesthetic Goal Two: The game should feel at least slightly unique on each attempt.
    - Signs of success:
        - Players do not get immediately bored with the game.
        - Players do not feel overpowered in the game world.
        - There are some factors outside the player's control.
    - Signs of failure:
        - Players master the game almost immediately.
        - Gameplay requires almost the same action by the player on every attempt.
    - NOTE: My second aesthetic goal has led me to decide that I will try to implement dynamic creation of the track of platforms, vanishing platforms, and blades.
- Core Mechanics:
    - Player navigates the track
    - Player avoids vanishing platforms and blades
- Core Loop:
    - Player moves forward, player pauses to assess vanishing platform or blade, player navigates vanishing platform or blade, player moves forward.

    - This core loop serves my aesthetic purposes because it becomes easier as the player practices reacting to each obstacle and the obstacles appear in different places/combinations, making each time playing the game unique.

BEGAN PRODUCTION

- Implemented a track of basic platforms
- Created the player
- Implemented player movement along the x axis and z axis using arrow keys.
- Changed the camera orientation and implemented camera tracking of the player.

12/1:

- Realized that my implementation of the track (with spaces in between flat platforms) would make dynamic implementation of the track at runtime more difficult). Thus, I re-implemented the track using cube-shaped blocks touching at their edges and made the track spawn dynamically and randomly at runtime.
- Added vanishing blocks that spawn randomly at positions on the track.
    - Used a coroutine to disable the mesh renderer and box collider of a vanishing block for a second every couple seconds.
- Fixed a bug where blocks could spawn inside other blocks by checking if a given position is already occupied (using Physics.CheckSphere) before instantiating there.
- Implemented player movement along the y-axis (jumping) when the spacebar is pressed.
- Changed jumping so that pressing the spacebar only has an effect when the player is touching a block of the track, thus preventing the player from just jumping repeatedly in the air over the track.
- Changed the camera orientation again, to provide a better visual of how the player can navigate the track.
- Added a skybox background
- Added three more layers of track underneath the original track so that if the player falls, they have three chances to land back on track and thus complete the course. Also, the tracks sometimes combine, making a more interesting map.

12/2:

- Added ending bricks to each of the four tracks that, upon collision with the player, will produce the game-completed screen.
- Fixed a bug where the ending bricks spawned at the end of every brick, not the last one in the track, by moving the call to instantiate outside the track-generating for loop.
- Slightly increased the distance between the tracks to make landing on them more feasible.
- Decreased the proportion of vanishing blocks to regular blocks to make the game slightly easier.

- Once again, changed the camera orientation, to provide a better visual of how the player can navigate the track.
- Altered player velocity to make the game slightly easier.
- Decided to add a feature: gems that decrease the time on the timer by 5 seconds. Also decided to probably not include the blades, because the game doesn't need to be harder.
- Added dynamic and random instantiation of gems above certain blocks at runtime.
- Implemented deletion of a gem object if the player collides with it.
- Implemented an updating text object that tracks time in seconds since the attempt began.
- Implemented subtracting five seconds from the time when the player collides with a gem.
- Changed it so gems decrease the time on the timer by 3 seconds, not 5.
- Implemented GameOver screen appearing and physics pausing when the player collides with an ending block.
- Implemented GameOver screen appearing and physics pausing when the player falls lower than the lowest block.
- Implemented ending text messages, one for failure and one for success, appearing on the GameOver screen.
- Implemented the game restarting when the player presses escape.
- Fixed a bug where the timer did not reset after pressing escape by setting MyTime equal to 0 in the Start() function of Timer.

12/3:

- Postmortem:
    - My goals for this game did not change significantly from my original design. I implemented player movement, camera tracking, a timer, a dynamically generating track of platforms (including vanishing platforms), and a game-over screen as intended. It is also in 3D.
    - However, I made some modifications. I decided against implementing windmill-style blades, because the game was already difficult enough, and I added gems that subtract 3 seconds from the timer when collected by the player.
    - The most difficult part of this project was figuring out how to dynamically generate the track, but I'm really glad that I did it. I'm also glad that I chose to do the project in 3D because it taught me a lot about using and orienting a camera in 3D games.