Research Project Thesis – First Person Platformer

Abstract/Summary

Recreate the movement system of the video game Titanfall as a time trial first-person platformer video game. The player with be able to walk, run, jump, double jump slide and wallrun. This project will also entail a level creator and editor and a leaderboard for best times for both pre-built (developer made) and user created levels. This game will utilize the control schemes of keyboard and mouse, as well as controller (360/XBO controllers). This project will also use a server hosting a database for levels and best times that the game will communicate with to download necessary leaderboards and upload new best times done by the user.

Tools and Software

* Environment
  + Desktop
    - Windows 10
    - AMD FX 6300 Processor
    - 16 GB Ram
    - NVidia GeForce 7400 GPU
  + Laptop
    - Windows 10
    - Intel Core i7 7th Gen Processor
    - 12 GB Ram
    - NVidia GeForce 940MX
  + Server
    - To be decided through further discussion/research
    - Windows Server or Linux Server
    - Physical or Google Cloud Server
    - Utilizes a MySQL Database
    - Online Accessible
* Software
  + MySQL
    - Database used for Leaderboards and program queries
    - Will need to research how to have the program connect and query the database
    - Database list will contain Level tables that will describe Player times in order of fastest and will only contain 5 listings.
      * Time
      * Player Name
      * Date
      * Level Name
    - Level Tables will be an updated table listing received level creations by users and developers.
      * Level Name
      * Created by
      * Date Created
      * Date Modified
      * Linked to best times table(described above)
  + Unity3D
    - One of two video game engines to choose from for this project
    - Most knowledgeable in
    - C# will be the programming language used
  + Unreal Engine 4
    - One of two video game engines to choose from for this project
    - Research will be necessary for use and coding
    - Least knowledgeable in
  + Blender
    - 3D modeling program to be used for creating 3D assets to export to chosen 3D game engine
  + GarageBand
    - iOS mobile app for creating and editing music
  + GIMP
    - Image creation and editing program for creating 2D art assets
  + GitHub
    - Version control program to be used when creating the game and server

Project Description

Create a video game program that replicates the movement system of Titanfall. There will be limitations; for example, may not be able to run on some vertical surfaces. The game will implement a movement system, a level creator/editor, and online service functionality with a server host.

The movement of the game will let the player walk in any direction, run in one direction, turn, jump, double jump, slide, wallrun, and jump off objects even while wallrunning. The movement will be controlled by keyboard and mouse or by a 360 controller. Some physics based movement will be implemented, which Unity does support. The physics for forces, such as multiplied force when jumping quickly off a wall to increase speed, will affect the player.

The level editor will allow a user to create their own levels within a given area. They will be able to place and lock objects, spawn points, start points, end points, and hazards. They will also be able to modify the transform, position, and rotation elements of any object before and after placing. The user will be able to save any additions made and be able to return to it later for further work. The information on the created level will be saved as a configured text document to be used later for loading the level, as well as, uploading the level to the server for other users with the game to download and play.

The server will host a leaderboard for best times accomplished on any level created and registered to the server. The game will communicate with the server and ask for times held for any given level in its database. The database will hold a table of levels and a corresponding table for the top five best times for that level. The server will also host level information that users of the game can download and be able to play user created levels.

The end goal for this project is purely to recreate the movement of Titanfall in a time trail game set in First Person. The Level Editor and Leaderboard/Level hosting Server are addons created for this project. This is a project I have been planning to make shortly after I graduated and will continue working on it no matter the state the project is in after the time allowed to work on it.

Plan

(Subject to Change)

* Sprint 1
  + Choose a engine (Unity 3D)
  + Create a basic movement system
    - Walk
    - Run
    - Slide
    - Jump
  + Create Basic level for testing and troubleshooting systems and gameplay
    - Utilize generic assets
      * Blocks
      * Pyramids
      * Cylinders
* Sprint 2
  + Movement System
    - Add Double Jump
    - Add Wallrun mechanic to movement system
      * Attach to Vertical surface
      * Move forward along surface while player is moving forward somewhat parallel to surface
      * Be able to jump off the wall
      * If player is moving away from the surface, detach from surface.
      * Be able to jump off the wall
  + Start creating server
    - Create test code with engine to test communication and data sharing with server
  + Create Relational Diagrams for the database
* Sprint 3
  + Troubleshoot Movement System further
  + Create 5 basic levels
  + Create Hazards
    - Blades
    - Projectiles
    - Pits
    - Kill Walls
  + Have it so the game can now communicate to the configured server.
  + Create Database
* Sprint 4
  + Create Level Editor
    - Objects
      * Obstacle/Ground/Wall
      * Be able to spawn objects from a list
      * Be able to edit object properties
        + Size
        + Rotation
        + Position
      * Be able to change color
    - Set Player Spawn Point
    - Set Level End Point
    - Be able to Save level
    - Be able to Name Level
    - Be able to return for further editing
  + Upload Created Level Information to the Server
    - Name
    - Created By
    - Date Created
    - Level Structure
      * Objects
        + Size
        + Rotation
        + Position
      * Start Point
      * End Point
  + Download Created Level Information from the server
    - Name
    - Created By
    - Date Created
    - Level Structure
      * Objects
        + Size
        + Rotation
        + Position
      * Start Point
      * End Point
    - Cannot allow user to edit downloaded levels that are not theirs