

Will Loughlin
Homework 4 Individual Project
Wills Game

My Github Project Link: <https://github.com/WillLoughlin/WillsGame>
My Most Recent Commit:
<https://github.com/WillLoughlin/WillsGame/commit/a658c0514f2e3653976a4b63a64485cfe32db583>

This document can be found in the HomeworkX folder in my repo.
A video explaining current features can be found in the HomeworkX folder as well.

Section 1: What I planned on doing for this checkpoint

For this deadline I had planned to finish 3d model rendering, gun firing, and 3d collision. In my initial project proposal I had planned to be working on the player info database and user interface applications for this checkpoint but I have moved those to the final checkpoint to focus on the gameplay which is the most important aspect of the game. At my last checkpoint I was able to load 3d models but I could not manipulate them successfully to aim correctly or give the player a weapon. My plan for this checkpoint was to implement gravity and player collision with blocks, as well as a shooting system that had its own collision detection system to check for players.

Section 2: What I accomplished for this checkpoint

I accomplished exactly what I had planned to finish by this checkpoint. First I finished the 3d model rendering and gave the player a weapon that could aim and fire. Now each player model rotates only on the x axis to indicate direction, but the player's own model rotates in all directions to allow aiming on the y axis. The weapon is a

seperate gltf 3d model loaded separately from the player-models and placed in the hands of the models.

The 3d collision with the map is also finished. The player checks collision against each block separately, allowing for any kind of arrangement of blocks on the map. The map creation is not done yet, but it is a very simple task as it just involves the placement of blocks without any more coding. The player can walk off the map currently, so the player is stopped before they reach a y of -3 in world coordinates. Player collision with blocks is checked twice at each frame to allow left to right movement if forward is blocked or forward and backward movement if left or right is blocked. This allows for smooth player movement around blocks. I also added a sprint function, which can be used by pressing left shift while moving to move faster.

Shooting was the final feature implemented for this checkpoint, and while it is functional it requires some small tweaks before the game is finished. When the mouse is clicked a bullet is fired from the position of the player's camera. At each frame the bullet is checked for collision against all other players using the position of their cameras. At this point you must hit a player in the head to kill them, but it will be very simple to add more collision points on the players as the code is already completed, it is just a matter of adding more player coordinates to check collision against. If a player is hit by a bullet the bullet is removed from the server and each player's scene individually, and the player is killed. As of now you spawn 50 units in the air when you are killed and you drop back onto the map. In the final version there will be spawn points around the map where players will spawn when killed

I also added a small crosshair to the screen to help with aiming, which is the first step to a gui. At this point the crosshair is not present on the heroku hosted version because there is an issue with the gui and online hosting. To see the crosshair check out the videos of current game progress in my repository. The crosshairs and gui proved to be more difficult than expected because Three JS does not allow for image overlay in the renderer. To create the crosshairs an entire new scene is created with an html canvas used as a texture on a planar object in front of an orthographic camera. Once the original game scene is rendered, this gui scene is rendered on top of the game which is causing problems in the version hosted on heroku.

Section 3: What I had planned for next deadline, and what plans changed

In my original proposal I had planned to complete 3d collision between bullets, players, and objects by the next checkpoint along with a gui. Instead of waiting to finish the 3d collision checking until the end I tackled that for this checkpoint, leaving the player database and the gui for the next checkpoint. I will also complete the map with spawn points in different areas. This is a very reasonable amount of work as the bulk of the coding has been completed. The next checkpoint will round out the final touches of the game. At this point I have learned almost all of the new information I needed to complete this project. All I need to do is refresh on database management and complete the gui implementation that is already somewhat functional.

Section 4: Screenshots of current accomplishments

I will include some screenshots of current progress below, and a narrated video of the game in action can be found in the homeworkX folder in my repository.

