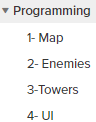
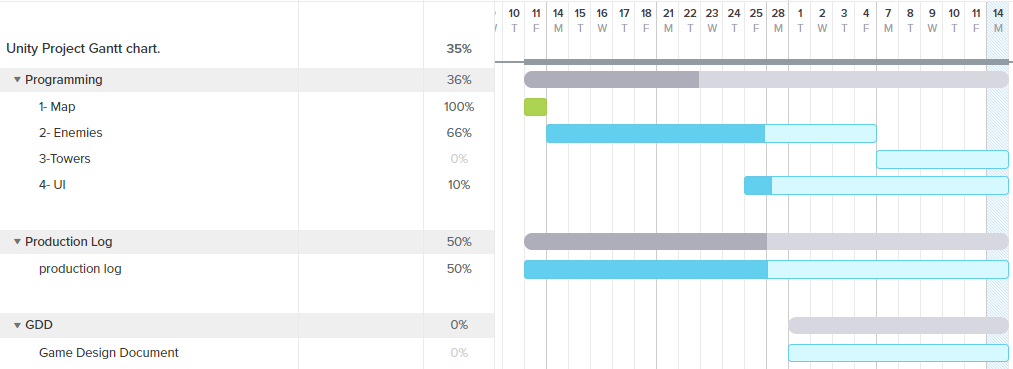
**Planning:**



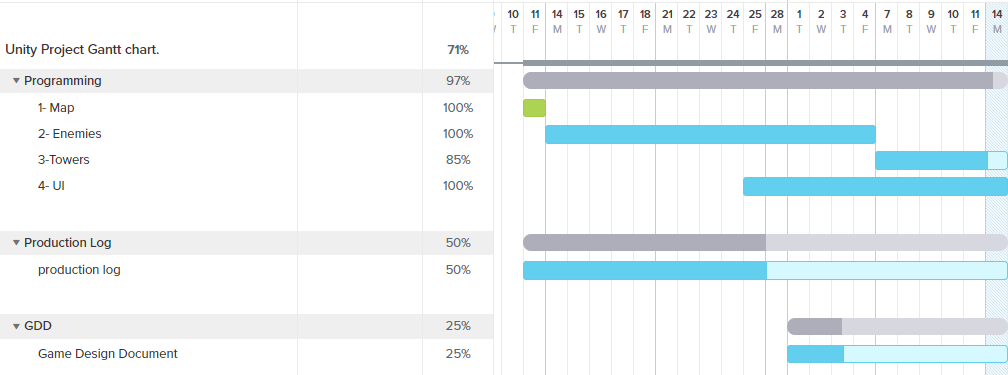
These are the 4 main aspects of the Unity project that I needed to complete. The only part that gave me issues was the “Towers” part. This is because I needed to detect if an enemy is in the range of the tower, check which enemy is the closest to the tower, and then shoot at the enemy. Then it checks if the bullet collides with the target, and damage it. Unfortunately, the hand-in date crept up on me and I couldn’t fully finish the Towers script. Once the bullet had hit the target, it destroys the bullet, but doesn’t damage the enemy. I had to cut this out of the final showcase because there wasn’t enough time to complete this part.

**Project management:**

25th Feb:

On the last lesson before the initial hand-in date, I was further behind than other people, this was because I couldn’t access my unity project from home. This means I could only work on the project when I was in college during the first few weeks. After this, I managed to figure out how to access my work.

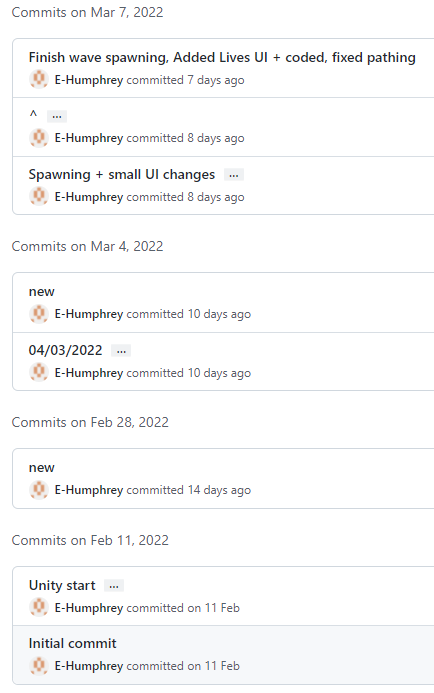
14th March:

  
On the new hand-in date, all the necessary Unity part of the assignment had been completed. This is because I fully focused on the scripts and making them function as intended. I had forgotten to complete the production log as I was going along. I was going to leave some of the GDD until the end, such as black box testing and source control, but I've had to improvise and change some plans because of my circumstance.

**Black-box testing:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test** | **Testing What:** | **Works?** | **Comments?** | **Resolution?** | **Works?** |
| **1** | **Enemy movement.** | **Y** | **Rotation didn’t work.** | **Change enemy model to capsule.** | **Y** |
| **2** | **Wave Spawning.** | **N** | **Spawned wrong enemy and didn’t increment difficulty.** | **Fixed enemy prefabs and edit code to make a sort of tutorial at the start.** | **Y** |
| **3** | **Lives.** | **Y** | **Shows lives on UI and updates every frame.** | **-** | **-** |
| **4** | **Turret targeting.** | **N** | **The turrets were targeting the enemies correctly, but it wasn’t losing them as soon as they left the turrets range.** | **Added lines of code to make sure it only targets enemies INSIDE the range, and if they leave the turret’s range, lose sight of the enemy.** | **Y** |
| **5** | **Turret Shooting.** | **Y** | **Bullet instantiates and travels to the enemy. As soon as it collides with the enemy, the bullet gameObject will get destroyed.**  **(Doesn’t damage/kill enemies yet.)** | **Will add code to make sure the enemies lose health as soon as the bullets hit them.** | **-** |
| **6** | **Turret Building.** | **Y** | **The building of the turrets works as intended. They will build when the player presses a tile and has the correct amount of cash.**  **(Can build more than 1 turret in a tile.)** | **Add a few lines of code saying when the turret is built, a boolean is changed to true. If they try to build on that same tile, it won't work because the boolean is set to true** | **-** |

**Source control:**

Source control is basically just managing changes to your code, so that you can easily track your past versions and even access them if a game breaking bug appears that you cannot edit/remove. Because I had never used source control properly before, I didn’t know how important it was, and got lazy with it.  
  
This can be seen with the very very brief titles of the commits. The names ”New”, and ”04/03/2022” are proofs of the source control problems I made. Next time, I will know that the source control is a very important part of the project, and keep is as professional as I can.