# Imported Assets and Libraries

There are some things I didn’t make in this game and some things I made before starting this project. There is of course Unity which I didn’t make, but I also didn’t code some of the packages I used for this game. These packages are:

* Cinemachine, for better cameras and transitions
* TextMeshPro, for better text display
* LeanTween, for UI animation

Cinemachine and TextMeshPro are official Unity packages while LeanTween is third-party.

I also imported assets which I frequently use for all my games such as Audio and UI, but these are made by me.

The first commits on the GitHub repository were all imported from the aforementioned sources.

# The barebones prototype

I have a lot of ideas (some of which won’t make it before the end of the assignment, or won’t make it into the game at all), but first I need to build the crude version of the game to see if there is potential for fun. However, the concept of a top shooter is decades old so the prototype will need to be a bit richer than normal.

With that in mind, I made a GDD (Game Design Document) and wrote some ideas, I set up a Trello board and made a list of what I should do first before any play testing is done. First the player movement, which was something I already had done dozens of time so it went pretty easy. I used a physics-based movement with Unity’s Rigidbody to allow for smooth controls as well as potential use of physics later (such as wind). The next thing was shooting, this is also relatively easy, just a little bit more verbose for the input since I have to detect when the player presses and releases the button for firing.

The firing was working just fine, but then I set up the main menu (which took 5 minutes since I already made a template) and it was not working anymore. The player was moving according to the input, but it was not shooting. Although I don’t know why it is like this, the likely culprit is the PlayerInput component I have on my Canvas (which was necessary for my menu) which meant I had two PlayerInput components in my scene at once which broke something (the Input System package is still relatively new so it’s still a bit funky to use at times). The movement input was using messages and worked fine, while the shooting was using with C# events and didn’t work. I couldn’t use messages for the shooting since I need to detect whether the input was pressed or released so I switched the PlayerInput to use C# events and made sure to change the movement input accordingly. After that, the shooting and movement were both working fine, at the expense of concise code since it probably tripled the amount of line needed for this.