48 Hours Exam Dev log.

March 15, 2024 19:30 – March 17, 2024 – 19:00

Planning and Setup – (2 hours)

Started planning out the how I'd like to handle the game setup decided that since I am most probably going to be stuck at lot in the asset integration and learning curves of dealing with those integrations I opted to use the Third person template for handling at least the movement of the character though I would edit some stuff as the movement of the character in the template is meant to be on foot movement.

Plan of action decided at the time.

- Setup
- Asset gathering
- Asset integration
- Movement editing initial.
  - Controls using WASD where W is for accelerating/speed up.
  - A and D are turning.
  - S will be slowdown.
- UI and menu.
- Obstacle jumping detection and Scoring.
  - Scorer class attached to the player actor and will detect the obstacles jumped over by line tracing downward and if the objects jumped over were tagged with 'obstacle' then it would score it.
- Level Design.
- Replay function.
- Optimizations/Bug fixes.
- Power up – score multiplier. (post completion)

Studying asset integrations into unreal – (3.5 hours)

When on YouTube to look for asset integration videos and learning some stuff on animation retargeting amongst other stuff.

3D Asset gathering and Integration – 8 hours

This became the sink hole in which I ended up wasting so much time on as I had initial planned on using a different character from the unreal engine mannequin and was handling some animation retargeting and gathering which I ended up abandoning eventually as it took too much time.

I also made the mistake of thinking the city park asset referred in the instructions were going to be an asset pack but ended up as a whole project.

Movement and UI – 5 hours.

Development of the actual game mechanics finally began and from movement editing, up until the UI and Menu was smooth sailing at least until it came time to look for the GUI assets.

Jump Line tracing and scoring - 6 Hours

This was the final dead lock on my development as It seems the method I thought off was a bit off or maybe I just don't have enough understanding of the method itself but essentially I just couldn't get to make the detection work when jumping over the obstacles and handling scoring.

Conclusion of performance during development

What went right:

- I was able to partially achieve what I had hoped to create in terms of the system of a more component-like based approach on mechanics.

What went wrong:

- Spent too much time on the asset integration when I should have just settled on having the first skate animation to begin with and improve it later on. Also trying to find a way to import the skateboard used in a way that it should have skeletons when I have little 3D rigging knowledge should've been not attempted in my opinion.
- Got called on Saturday to come to work and handle an emergency issue in the live prod was really such a waste that could have at least given me time to get most of the core mechanics done in.
- Overthinking and trying to make it right immediately.