**Project Evaluation Document**

Name of project: LeapFrog.

Project participants: Mark Aigbe & Dennis Dusmetov.

Project duration: 1/10/19 – 08/12/19.

**Intro:**

In this project evaluation I(Mark) will be discussing various aspects in relation to our project such as:

1. The important aspects of our game.
2. The development process from the alpha to the final state of the game.
3. The application of HCI in relation to our game mechanics and UI.
4. The challenges we faced during this project and how we dealt with them.

At the very end I will leave my conclusion as to how I felt about the project.

**1) Important aspects of our game:**

The main aim of our project was to make a game that was convenient in the sense that it required no high skill cap while playing the game, making it more presentable to a broad spectrum of people. We did this by only needing one key for control of the player(frog), by using the spacebar, meaning literally anyone can play.

Another aspect we considered convenient was having a non-complex user interface (UI). We decided to have big, spacious words for various things like our buttons, score count and death text. We also added a red tint as to when the player dies it’s more recognizable that the player has died.

The final main aspect we thought of was difficulty. Having the snakes spawn at the same height would lean on the too easy side, so instead we decided to reduce the spawn rate so the snakes are approaching the frog more often and we got the snakes to spawn on a random Y position using the random.RangeY(min, max) method.

**2) Development Process**

For the approx. 10 weeks we worked on the project, we have been constantly been adapting the look of our game. At the start we were considering doing a 3rd-person view but deemed that camera position to be too awkward for the player being able to perceive the gap between the snakes and whether the frog can pass through. We were also stuck on finding assets and ended up paying for one of them (frog assets).

Midway through the project is when things started to flow properly, we managed to get the snakes spawning at a random y position to add variation to the snake obstacles in order to challenge the player We also implemented an “Invisible ceiling” in order to prevent the player from jumping to high and leaving the play area!

As we reached the final state of the project, we both split up on separate parts of the project. Dennis started the UI process, and I focused on adding the finishing touches to the game by implementing the particles, sound effects and fixing any bugs that occurred in our game.

I also worked on the UI, by adding the start menu, while Dennis focused on the game over menu.

**3) Application of HCI in relation to our project:**

Of course, computers had a huge impact on our project. At the start we were sharing our progress we did at home through USB until we found out how to use Unity collab…

Having a functional spacebar on a keyboard attached to a desktop/laptop is pivotal to our project, it’s the only control a user has to the frog through code!

Being able to interact with our game through code is a very satisfying especially when we add a new aspect to our game.

**4) Challenges we faced and how we dealt with them**

The first big challenge was getting the Snakes to spawn at random heights. This took us a week to solve as we weren’t sure how to get this to actually happen. How we overcame the challenge was by getting an array of snake obstacles using Game Object [] and creating 4 copies of our snake obstacle. We then used the random.RangeY() code to give a random height position to each obstacle.

The next big challenge we faced was allowing the count to go up as the frog passed the gap in between the snakes. This took longer to solve because we intended in putting a physical, visible object inside the gap between the snakes and apply a collision to that in order to get the score up. But that wouldn’t work, and the object would fall to the ground even with gravity off and is kinematic on… How we overcame this was by adding a physical object that filled the whole gap, had a z of 0( so it was thin enough for the player to pass through even though it had rigid body applied to it) and gave it a tag “cube”, we also made it invisible by removing the mesh texture applied to it. We then applied the comparetag () method in order to check if it collides with player, making the score go up!

Our hardest challenge was to get the background to repeat, simultaneously, and infinitely. Unfortunately, we couldn’t solve this problem. I personally spent the whole duration trying to solve this but no matter what I did there would always be a stutter when the background would try to repeat. As we were approaching the end we eventually let this issue go unsolved and focused on other aspects of the game by making the game as compact as possible.

**Conclusion:**

All in all, there are many positives I take from this project we worked on. I had to work with another person, so I made sure I was putting in my end of the work in order to make the workload balanced and fair. To be fair, I did want most of the influence as I had a vision from the very start as to how I wanted this game to turn out and fortunately I had a partner allowing me the freedom of expression for this project but also adding his own input and rationalizing my spontaneous ideas!

I’ll definitely remember and appreciate this project as I really enjoyed working on it, who knows maybe a few months down the line I might further develop it into a more fleshed-out game.

**References:**

Sound:

<https://www.youtube.com/watch?v=pgI-sVAbsEo&list=PLTmomcjY1VRx8N6MpwuJkLIpKfC3N7I_7>

<https://www.youtube.com/watch?v=NOsI_3i5QKk>

<https://www.youtube.com/watch?v=_4vQ6ZQGdnE>

Assets: