**Assignment 3 – B.I.TS**

* **Peoples Section**

Name: Michael Mountjoy

Student email: [s3841493@student.rmit.edu.au](mailto:s3841493@student.rmit.edu.au)

My ethnic background is a mix between Italian and British. I am passionate in the gaming industry within I.T. Additionally, Java and functional programming like F# has been fascinating to learn.

I am still considered a beginner in programming and I.T but I am excited to learn more about programming and how to make video games, like through unity.

I hope to develop vital skills and experience relevant to creating video games throughout the course of this assignment. The way I see this happening is using the unity engine to create assets, mechanics, scripts and content with supplementary programming through collaboration with my group members.

Throughout the third part of this project, I will be endeavouring to learn C# and become involved in the creation of the scripts with the rest of the team. I am

* **Tools and Technologies**

The group used a variety of tools and technologies. These included communicative technologies, as well as software technologies such as designing assets for games, coding technologies, and more. These included but were not exclusively limited to Microsoft Teams, Visual Studio Code, Unity, Blender, Zbrush and Github.

Firstly, the team used Microsoft Teams in order to collaborate and organise our efforts in the completion of each milestone as it related to the assignment parts. This included routine, weekly meetings, and it followed a flexible but well-defined schedule. All team members in advance of a day were informed of certain tasks or requests, and they needed to respond to them within that timeframe. This also extends to the familiarity each team member had with the program in knowing how it functions, and how to use it effectively. Additionally, the interactivity each team member had with the program and their communicative contributions also are included in their capacities to interact with the program within a meaningful way.

Secondly, the team used Visual Studio Code in order to code the website in HTML and CSS. This program is also being used for coding the mechanics and functional aspects of the game that is being created in Unity. Members within the team have worked together using this program in order to code up the website and allocated parts of our team will work on the Unity script programming. The team will also be using C# to code the scripts for Unity, as C# is a programming language designed exclusively for video game programming.

Thirdly, the game is going to be finalised and created within Unity. The team will work collaboratively on a shared project and will create assets, maps, characters, mechanics, and code all the necessary structures to function as a fully fleshed out game. There will also need to be a time investment in teaching the members of the team how Unity functions if as a team we are to make full use of its capabilities as a software that has the ability to create games. Assets have also been created with Unity, which means it is also a viable program other than Blender to create 3D models. However, Unity, as described below, might lack some of the tools that Blender has in order to create authentic and visually distinctive 3D models, which Blender might be more equipped to deal with.

Fourthly, some members of the team decided to create some assets within Blender, Zbrush. Blender is an open sourced, free program along with Zbrush for specialize tasks that has a variety of toolsets which can be used to create visual effects, 3D models, animations, and assets. Some members of the team will collaborate together to create unique and original assets for the purpose of being integrated and used within the game. Although a majority of assets were already created in Unity, there are some tools which are not present in Unity that some members of the team may find useful in designing and creating their own characters.

Finally, the team is using Github in order to coordinate the amalgamation of all of our work. We are using Github Desktop to push work onto the cloud and the system allows for a team to do that. The structure of how it is setup also allows for ease of updating existing files from other members, or individual work. This is because you can locally edit the files and then commit it to the designated repository through Github desktop. However, there can be merge issues or file integrity collisions if certain members of the team push their work simultaneously, or if there are different word processors used to edit something created by another word processor.

The following table below illustrates the corresponding proficiencies with each medium:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Names of Members | Microsoft Teams | Visual Studio Code | Unity | Blender/Zbrush | Github |
| Michael M. | Medium | Low | Low | Low | Medium |
| Nathan C. | High | High | High | Low | High |
| Tanya T. | Medium | Medium | Low | Medium | Medium |
| Long | Medium | High | High | High | High |
| Rachael B. | Medium | Low | Low | Medium | Medium |

* **Windfall**

As a group, we decided to shift the “first level” from the extended features and integrated it as part of the Minimum Viable Features. We needed to optimise this because there was structural changes that occurred to the team at certain points in time that necessitated this shift. However, despite this adaptation, this optimised the efficiency of the completion of the assignment.

There were also some discrete workarounds the group discovered when trying to coordinate the assignment. For example, striking an appropriate balance between the designation of tasks and the duration it takes to complete them resulted in less time that it would take to complete and achieve certain milestones within the assignment. Also, using Github desktop was a streamlined and intuitive way to upload and edit files to Git. This avoided having to navigate to the website, login, and manually upload each file. Instead, the members of the team merely had to edit the files locally, and then forward it as a commit, which had a cumulative compounding effect in saving time.

Additionally, the construction of the timetable was a well-optimised effort because each group member focused on filling out the estimated time they would spend working on each applicable part of the project. Members of the group filled this out on a Google Drive document which all of the team had access to, which ensured that each member did not have to waste time creating a table or trying to format one. Therefore, this optimised the amount of time focused on productive work and not on work that does not contribute to the completion of the assignment within a timely manner.

* **Spike trap sound ideas**

Below is a few ideas regarding the sound that the spike trap should make once triggered.

<https://www.youtube.com/watch?v=O5ZWNkXI3KU> – Stock sound

<https://www.youtube.com/watch?v=ZbNwxu2kuFM> – Royalty free spike sound

<https://www.youtube.com/watch?v=sPpGUEwejGo> – Fortnite spike trap