**PROJECT POSTMORTEM SUBMISSION FRIDAY 10TH MAY 2019**

|  |  |
| --- | --- |
| **STUDENT NAME** | Alpeche Pancha |
| **PROJECT NAME** | End Of The Line |
| What do you think went well on the project? | Overall, our group worked well together and as a team to deliver a playable game. Our game meets the requirements of the SUMO Digital brief. In terms of design, the sprites are of very good quality and make the game look more polished. As a nice finishing touch, the designers added animations to the character, water and lava assets, making them look even more appealing.  The main gameplay mechanic went through iterations and we responded to feedback given to us during the various stages of the project. Although we were not able to deal with all the feedback, and our mechanic still needs improvement, we have documented and acknowledged all the feedback given to us.  As seen in our Discord chat logs, our level of communication improved throughout the course of the project. During the last few sprints, we were actively sharing our progress with tasks and clarifying anything we were unsure about with each team member. We also used voice chat for a remote meeting. JIRA tasks were well planned and allocated, mostly by the lead project manager Matthew. |
| What do you think needed improvement on the project? | Firstly, a couple of game features were incomplete due to time constraints. I believe that we could’ve planned a little better to have these features implemented in the final game.  The first feature was the distance tracker with milestone beams. Matthew was very keen on having light beams that informed the player about how much distance they had travelled, at different milestones (e.g. 500 meters, 1000 meters, etc). This was a technical challenge because our only level in the game was randomly generated during gameplay. It was not possible to incorporate the distance beams in the generated content easily, so we decided that this feature should only be implemented if there was time at the end for it.  The next feature was the random content generation. Although our level does generate content infinitely during gameplay, the way I implemented this feature is not optimal. There are only 6 different types of “chunks” that get generated, some with platforms and walls. This leads to the gameplay being too easy and predictable, even when the player speed is gradually increased to raise the difficulty.  Due to the way I implemented level chunks, they were not flexible enough to allow entities such as gold pickups, water, lava, and distance beams to be used. Although all these assets, and more, are visible in our demo levels, the final level does not include them because of this technical limitation, and it makes the game look bland and repetitive.  Finally, the use of version control in the project was not always effective. The fact that we used Blueprint for our programming made it hard to track changes and view differences in files across branches. Additionally, Github Desktop or other Git clients were not used by other team members. Several commits had default commit messages such as “Added files via upload”. As a programmer, this made it harder for me to track project changes and see what new files were uploaded when. Many commits were also made on the master branch by all team members, which is not often good practice. |
| What do you think of your own contribution to the project?  Reflect on the quantity and quality of your work. Whether you were reliable as a team member, your general behaviour, whether you were proactive in spotting problems. These are the key qualities of a professional. | As the only programmer in the team, I feel that my contributions to the project were satisfactory. The game is functional and most features we planned for were implemented. I’ve already mentioned that I am not happy with the two features that were incomplete, and in hindsight I should’ve spent more time planning and researching better ways to implement them in the game. However, I communicated these issues with the other team members and let them know that I was not able to deliver the work ahead of time, so that we could prioritize other tasks in the backlog.  As a team member, I attended every meeting and was always reachable via Discord, as seen in our chat logs. I often suggested ideas to the designers and always clarified their requirements before implementing the features they wanted. I was always open to criticism and feedback and used these to iterate during development.  I also recorded several gameplay videos for our presentation, walkthrough and final submission. I kept sharing different versions of these videos with the rest of the team and addressed their feedback before making the final uploads. I did my best to showcase all the content of our game in these videos.  In terms of hours put into the project, I believe I could’ve spent more time towards the end of the project improving the random level generation feature of our endless runner. The number of hours I spent slightly decreased at the end, as I shifted my focus to other assignments. Better time management could have prevented this. |
| **OVERVIEW** |  |
| **Thinking about the project you have worked on this year, what are the important lessons that you will take away from the experience for your next group project?** | As a part-time student, it is vital to manage my time more appropriately. Several negative aspects of the project that I have mentioned could’ve been avoided if I had better time management. I’ve learned that spending small amounts of time each day working on a project is better than spending sleepless nights towards the end of a project trying to finish everything at once.  I’ve also learned that feedback is important, whether it is from other players or lecturers and team members. The more feedback received, the more iteration and improvements can happen. This leads to a great polished final game. If there isn’t enough feedback, or if feedback is ignored, then the final game will never be the best it can. |

**Asset List**

My contribution to the project was all the Unreal Engine 4 Blueprints and anything else related to the programming of the game. Here’s an overview of the assets:

* “MineCart/MineCart.uproject”: The Unreal Engine 4 project for the game
* Character and game mode programming:
  + MineCart/Content/2DSideScrollerBP/Blueprints/**2DSideScrollerGameMode.uasset**
  + MineCart/Content/Character/**2DSideScrollerCharacter.uasset**
* Game items programming:
  + MineCart/Content/Items/**MilestoneBeam.uasset** (unused)
  + MineCart/Content/Items/Pickups/**ScorePickup.uasset**
* Level programming:
  + MineCart/Content/Level/**DemoLevel.umap**
  + MineCart/Content/Level/**MainLevel.umap**
  + MineCart/Content/Level/**MainMenu.umap**
  + MineCart/Content/Level/**TutorialLevel.umap**
* All files in “MineCart/Content/Level/Chunks”, which contain the parts of the level that get randomly generated during gameplay
* All files in “MineCart/Content/Level/Placeable”, which contain the Blueprints needed for placing rail tracks in the game.
* Level content design and programming:
  + MineCart/Content/Level/Background/**CaveWall1.uasset**
  + All files in “MineCart/Content/Level/Platform”
* All files in “MineCart/Content/Widgets”, which contain the Blueprints for the game HUD, player inventory, main menu and game over screen. They also contain the following unused Blueprints that were removed from the game:
  + HealthBar.uasset
  + Anything with “drag” or “dragging” in the file name
* Other work:
  + Demo and walkthrough videos of the game
  + Contributions to presentation