

Phase 4 Report - Group 21

Overall Description

Raccoon Raiders is a game that takes place in the Academic Quadrangle at Simon Fraser University. The main character of the game is a computer science student who is struggling with the challenges of his coursework, leading to exhaustion. To help the student overcome his fatigue, the game requires him to collect energizing rewards like coffee and bubble tea while avoiding failed academic work that are scattered throughout the map. The items all affect the score and/or speed of the student: coffee grants 50 points, bubble tea grants 100 points and doubles the speed, and punishments remove 50 points and reduce the speed. Once all coffees are collected a portal is opened for the student to complete the game.

However, the game isn't just about collecting rewards, as there are also deadly raccoons that need to be avoided. The raccoons chase the student and pose a significant threat to his survival. The student must gather all the coffees, dodge the raccoons and punishments, and locate the portal to complete the game successfully.

At the end of the game, the player's score and time are reported to create a competitive environment surrounding the game so that users can compete with friends.

Original Plan & Design vs Final Product

Our original plan and design for the game varied quite a bit from the final product. During the planning phase of the project we had decided to include many additional features beyond what was required by the instructions of the project. These features included a scoreboard, multiplayer system, fighting mechanics, and a boss battle at the end of a level. We also wanted to include multiple levels and maps, each showcasing different areas within SFU that had their own unique set of obstacles and rewards.

However, as we began to develop the game, we soon realized that we may not have enough time to implement all these features and were not comfortable enough with coding in Java yet. Unfortunately, we had to remove most of the additional features from the original plan so that we could complete the project on time and with a high level of quality. This resulted in us having very different UML and use cases from what we had created early on in Phase 1 and we had to update these documents accordingly.

By removing the extra features, we were able to focus more on the core requirements, such as the reward and punishment system and pathfinding algorithm of the enemies. But we were still able to stay true to the original design by not altering the background story for the game and using the same initial retro theme we had planned on. Overall these changes allowed us to focus more on the software development process as a whole rather than just the product we were creating.

Important Lessons Learned

We learned several important lessons working as a team to develop this game. With the first important lesson being the importance of accurately assessing the scope of the project. We realized that it is essential to ensure that the team has the skills and time necessary to complete the project on time and end up with a result which we are satisfied with. The initial overestimation of our capabilities led us to

removing many additional features we had planned for the game, and required us to re-engineer our requirements multiple times, which was a challenging but necessary decision. As a result of this challenge we also learned the need to prioritize game features and focus core gameplay.

Furthermore, another important lesson we learned was to have effective time and project management. By breaking down the project into smaller tasks and setting deadlines for each task we were able to stay on track and meet deadlines. To assist us in this process we used an application called ClickUp. By using this app, we were able to create a list of tasks and then each of us was able to select the tasks we wanted to perform and assign deadlines. As a result of this process, we learned that it's essential to remain flexible throughout the development process and be willing to adapt to changes.

Finally, we learned that effective communication among team members is also really important when working on software as a team. We realized that we needed to frequently update each other on what we completed and what needed to be done next, so that we could avoid overlapping each other's work. We found that by holding team meetings at the start of each phase and updating each other through Discord, this allowed us to ensure that everyone was on the same page and that we were sticking to the timeline.

Overall, our team learned that game development is a complex process that requires careful planning, proper time management, and effective communication. By applying all of these lessons, we were able to create a game that we were proud of and enjoyed playing.

Tutorial/Video: <https://www.youtube.com/watch?v=d7QG1P4O0Fk>

Artifacts Locations:

Javadocs: /RaccoonRaiders/target/apidocs/

JAR Executable: /RaccoonRaiders/target/RaccoonRaiders.jar

(our just use the make run command)