Sprint 3 Retrospective

CS 307 Group 16 Fathomless Caverns of Peril | Kyle Day, Jason Seeley, Stefan Gerber, Yuchuan Huang

What went well? (0.5 point) (a)

You may write this section in sentences and/or list successful user stories and tasks with detailed discussions.

1. As a user, I would like to see animated sprites and textures.
    a. This story covered five acceptance criteria relating to the visuals corresponding to the game. These types of criteria have proven to be quite easy to implement tasks as they are generally simply interpreting data that is already made available from each server request. A visually appealing game is necessary for an immersive experience. Although considerable work was completed for this user story during this sprint it was ultimately not substantial enough to qualify as a visually appealing experience.
2. As a user, I would like the ability to change textures, meaning edit player and mob sprites as well as the map components.
    a. Again this story covers acceptance criteria that rests completely within the browser. There were no issues at all with the implementation of these criteria as the required changes had been anticipated and correctly staged throughout the entire duration of the project.
3. As a user, I would like there to be changes in the map textures based on the different level biomes.
    a. This story required expanding the biomes class to contain more level information. Implementing the textures based on the biome was a very straightforward mapping process. However, there were some changes that had to be made in level generation relating to this new biome information. Also it was hard to find adequate textures for each different map object since there are so many.
4. As a user, I would like to have an overall score that increases with defeating monsters, collecting items. As a user, I would like the score to be displayed that will be displayed within the game and on a competitive leaderboard
    a. This story builds upon the very rudimentary scoring function made in the previous sprint, which was entirely user experience based. The new score function takes multiple factors into account, prioritizing user stats like turns and items gained, and is dynamically accessible for the player to see. Displaying the score on the frontend (for player and competitive leaderboard) was a very simple task because all of the calculations are done in the backend.
5. As a user, I would like flying creatures to fly over lava, and others to avoid lava.
    a. This story is not as simple as it seems, we spent a lot of effort on determining which terrain will damage the mob because different creatures could be damaged by different terrain.
6. As a user I would like certain mobs to have the ability to destroy the terrain.

    a. This story includes algorithms that determine which terrain can be destroyed, a new path-finding algorithm that finds the path that includes destructible terrain, and calculates which path costs less turn. It is a time consuming story.
7. As a user, I would like there to be an overarching narrative/story that is presented to the user through text (boxes and screens), so that I understand why I am going through the dungeon and defeating enemies.
    a. This was able to be accomplished two-fold by creating types of text that might appear for the player. Warnings may be issued to the user upon the traversal of a hazardous tile. These are to guide the player so they do not unintentionally damage themselves during their progression. Additionally there are text boxes that will appear when reaching a new level that will make the player conscious of the theming of the levels.
8. As a user, I would like there to be non-playable characters with optional dialogue. The NPCs will talk with the player by displaying text on the screen if the player chooses the option to talk.
    a. Finally the implementation of NPCs that are only interactable and not destructible requires considering them as any ordinary decoration game object. The shop GUI had to be built as well to interpret the shop inventories which allows for the buying and selling of shop NPCs.

2 What did not go well? (0.5 point)

(a) Include general retrospective review for this sprint.

Unfortunately as before, the main weakness for the sprint were generally simply failures in integration. Each teammate has a different idea of what needs to actually be accomplished and how. Even though there are extensive conversations regarding the specifics of requirements, not all details can be covered which leaves room for interpretation. These realized interpretations, regardless of if they are tested or not within the environment of the codebase, there have always been integration issues that limit production efficiency. There has almost never been a case in which committed code functions without necessary modifications for implementation into the production build.
Additionally another large setback for this sprint that was present within Sprint 2 was beginning on a flawed foundation. There has been a large emphasis on implementing as many features as possible which has led the production to be overall sloppy and rushed. This is unfortunate as elegant and efficient solutions are not even considered. Even though the product is functional, the lack of consistent standards or even code reviews has left the codebase to be bloated and very loosely related to the original plan. Fortunately server response times have been proven to not be compromised and the largest paths are called most infrequently.

(b) Ensure to list ALL unsuccessful user stories and tasks with detailed discussions. (They should be in line with the Sprint Planning Document for the next sprint.)

3 How should you improve? (1.0 point)

(a) Mention at least two ways to improve your work in the next sprint and be as detailed as you can.

Between the competition of the Sprint 3 review and the final presentation there will be general maintenance and upkeep to the codebase. This will allow for processes to be streamlined and heavily simplified. Now that new content will not be added it is important to ensure the final optimization of the project occurs and verify that everything is working properly. Potentially in the far future the server calls could even be ported to fully run on PHP and even shift to a leaner server which would greatly improve latencies.
Finally in the future, testing should occur not only within the integration step as this is costly in time. API calls could be simulated to verify that newly created code does not cause any catastrophic errors in dependencies, syntaxes, or proper task completion. Since the system is quite interconnected with its object oriented model, a single improper line will cause an HTTP 500 server error to occur within The Common Gateway Interface Binaries.