**Simmer Team Post-Mortem Statement**

**Production**

Good 1: Full-team meetings and the scheduled opportunity for both teams to meet and discuss the requirements of accomplishing their respective sides of a deliverable. The structure within what is discussed in different parts of the meeting is a helpful method to allow a digestion of information.

Good 2: The production process, communication, and task management was much better off during the second half of the project.

Good 3: Most if not all of the deliverable deadlines were met on time except for a few exceptions regarding weekly recaps and some internal expectations we set for ourselves at the beginning of the project.

Bad 1: We never got as mature of a player goal as we may have wanted. What the player was working towards as an experience could be clearer in conveyance.

Bad 2: Maintenance and pushing of the use of good practice in DevOps to maintain a clear direction for our own reference and an idea of what scope the team can handle.

Bad 3: The initial half of the project felt shaky and not as focused in terms of assigning tasks, asking for deliverables, and communicating between team members.

**Art/Visual**

Good 1: Variety of art styles and individuals to work on assets while still maintaining a consistent style across the game.

Good 2: Art direction was pretty well understood early on and each aspect of the game connected with that original idea. Whether it was from other games that inspired the direction or visual support from mood boards and other reference images.

Good 3: Developed better connections between the software team with regards to what can be implemented and what each team can do respectively.

Bad 1: Lack of designated art leader to help organize how long items would take to be completed, a specific style direction, and keep track of which assets are being completed by who.

Bad 2: Narrative for the player felt lacking by the end of the project and it didn’t feel like there was a complete point from A to B.

Bad 3: More rigid deadlines and consistent deadlines to reduce the fluctuation of work week by week. There were more work items towards the end of the project, which we would like to have avoided.

**Programming/Software**

Good 1: Did a good job mitigating bugs and we didn’t have any major issues that we weren’t able to solve.

Good 2: Felt support through the team and wasn’t afraid to ask for help while learning Unity.

Good 3: Appreciated the transition from miscommunication early on to developing synergy later on in the team.

Good 4: We felt we were able to get things done at most stages of the project.

Good 5: We did a good job limiting the scope each week and we didn’t have to rush to implement features each week.

Bad 1: Some of the code became unreadable and without documentation it made it hard to traverse or add on top of it.

Bad 2: Design and programming was not always connected and sometimes ideas were not always shared across teams before implementation. Stricter rules about design to implementation would help organization and prevent unused work.

Bad 3: Wish that we could have gone further on the project both in scope and in the content.

Bad 4: Design was not finalized before programming began. Modules would have been produced faster had there been design decisions made at the outset.

**Audio**

Good 1: We had a lot of external tools to implement in the game and we were able to insert the sounds easily for the game.

Good 2: Was able to get free sounds assets from the unity play store

Good 3:

Bad 1: We don’t have as many sounds as we would have liked for small interactions in the game.

Bad 2: Establish the sound and music role earlier in planning.

Bad 3: Inexperienced with how to implement sounds in unity.