

The first week (i.e, p1\_milestone) consisted of becoming disillusioned with the reality of a stark and daunting learning curve. It was very hard to get started with the project simply due to the fact that it's hard to use the tools at your disposal if you do not even know they exist. Without much/any previous experience working with the Unity engine. Consequently, the p1\_milestone week of development was arguably the most difficult. Much time was spent learning the basics of the engine: collision detection, physics, attaching components, etc. However, the second week of the development cycle met with more satisfaction with regards to the perspective of a developer. Once we were able to get the hang of the Unity engine, workspace, and workflow, we were able to implement features more efficiently. It was rewarding to see our game progress as we worked on the p1\_alpha; albeit, there were still a lot of man hours involved. Furthermore, an increased familiarity with the game engine allowed us to quickly identify and fix the issues with our game as per the p1\_milestone feedback. Lastly, the p1\_gold stage of development was a refreshing change of pace, as we were able to explore the creative side of game development with a refined and improved skill set with regards to programming in Unity. With a drastically increased level of familiarity with Unity, we were able to resolve the bugs from our alpha and work on our custom level and remaining features more comfortably. The JIRA project-management system was helpful in keeping track of tasks and managing time-allocation in general. We made sure to log our progress on our active sprint upon working on tasks, and we used JIRA often. I found that JIRA was useful in estimating how much time we could expect to invest on remaining tasks left to do with a deadline approaching. Thus, it led to decisions to drop certain features and prioritize others. Throughout the development of this project, we found out (sometimes the hard way) the importance of frequent playtesting. We found it was a good practice to test everything after creating a new component/script even if small in scope. Sometimes things that were working before could be thrown off and require

reworking/refactoring. Thus, it was important to playtest when we had the minimally viable solution to a new task. Moreover, assignment feedback was helpful in identifying and addressing bugs in our game. With so many moving parts to keep tabs on, it was easy to forget about a detail that needed some minor tweaking before the finished product (for example, Link stepping backwards during attacking animations).

The initial idea for the custom mechanic was to create a portal gun that would allow the player to link two locations. The idea for the level was that the player would have to choose carefully where to place the portals in order to succeed. One additional idea was that other objects could somehow be transported through the portals to add a layer of complexity. These objects ended up being the projectiles shot by the portal gun and arrows. The inspiration for me came from watching a minecraft video on youtube showing off a portal gun mod which I thought could be interesting in zelda. Zelda does not have the 3D complexity of minecraft but since it is a top down game, by utilizing the walls correctly, it is possible to still create an immersive experience. When actually trying to configure the level, it was clear that there needed to be spaces that were inaccessible to the player, the solution to which would be the portal gun. Otherwise the player could just walk through the level. If these spaces were regular obstacles, it would confuse players as to why only some walls triggered portals. If they were impassable like the water it would limit the teleportation mechanic. By instead adding lava tiles to the level that force the player back to the start of the room and cause damage upon contact, it makes the player have to be much more conscious about how they use the portal gun (lest they teleport into lava and lose progress). The intended player impact with the combination of the portal gun and the lava tiles was to create an experience where the level starts out simple (teleporting across a chunk of lava) and progress to where the player has to be really aware of where they walk and where they place portals as they become constrained on small platforms surrounded

by lava. Then, we add the twist of forcing them to shoot the portals to introduce a new mechanic to expand their options in using the portal gun. Finally, everything comes together with the addition of enemies to increase the time pressure to move and complete the rooms while avoiding being hit. The experience is much different to the original zelda since precision on movement becomes key. You need to control exactly how much you move with short and long taps to avoid ending up in lava. In addition, rather than thoughtlessly hitting the sword button in fights, the portal gun requires far more thought before you shoot to avoid teleporting into lava. The goal is to create momentary frustration if you accidentally end up in lava which becomes more punishing in later stages. At the same time, though, this fosters a sense of accomplishment by the end.

Perhaps the main and most important point as to what went right was that every task was submitted complete and on time. Especially since there were so many moving parts to the project and additional assignments on the side, making sure nothing slips through the cracks is very important. Time management was arguably the most challenging part of this project; it was easy to get tunnel vision on a single task. When you start working on a task, it is hard to just give up on it and move on to the next when you get stuck; you want to see it through to its completion. Thus, certain tasks took much longer than they should have perhaps due to said tunnel vision, and, in consequence, led to a time crunch with regards to other tasks. That being said, going forward we should make an effort to write down the issues we're having if we get stuck and try to move on to tasks we know we can complete; if we cannot make progress on a task, we should try to resolve it in office hours instead of spending valuable time going nowhere.