Introduction

The project I worked on is called “Tournamental” and it is a top-down tile-based strategy game. My job was to design and implement a mechanic during a one-week sprint, then to implement a mechanic designed by another designer in another one-week sprint. Throughout the project, I managed my progress and communicated with the project’s team through a Trello board. The first mechanic that I implemented was the Worn Out Tile mechanic that would break when players came within a neighboring radius of the tile then if a player or non-ball interactable enter the tile it will cave in and the object or the player will fall. The tile can only be surpassed if players push the interactable ball into the worn-out tile. The second mechanic I implemented that was designed by Thomas Morin was the Reveal the Way mechanic. This mechanic allows the player to pick up an orb interactable, each orb gives 2 reveal uses, the player can then use the reveal uses to reveal the hidden path on the map.

What Went Right

Mechanics Complemented Each Other Well - after implementing the first mechanic I was left with a decision of choosing another mechanic to implement in my game. There were a lot of mechanics to choose from but most were similar to my worn-out tile, a mechanic where the tile falls through, some accounted for the time spent on the tile while others relied on the number of times entered. The reveal the way mechanic was one of the more unique mechanics on the list so I decided to implement it and the two mechanics came together in a great way to complement each other. I used the reveal the way mechanic to hide the worn-out tile and this helped emphasize the cracking tile sequence when players enter a neighboring tile. This also made the ball movement more meaningful because instead of pushing it against a wall, the player can now push the ball off into the reset pit which would cause them to reset the level. These mechanics working well together made the game and levels come together nicely.

Additions to Reveal the Way increased usability- After implementing the base mechanic that Thomas Morin had designed, I decided to make some slight changes to the mechanic and its feedback. I decided to change it from hiding the tiles to fading in and out of the opacity. This allowed for a more smooth transition between revealing tiles. I decided to use translucent materials to achieve this and Lerped in between 0 (completely Translucent) and .9(Mostly solid, slightly translucent). This allowed me to easily set up a blinking effect too and better expressed the mechanic through feedback. Not only did I add the translucent materials, but I also decided to add a for each loop to check for the number of uses from each reveal ability. This made the mechanic much more versatile and usable, allowing the designers to add as many reveal interactables to the level as they would like and the uses will only be accounted for if they have been interacted with. Though minuscule, these few iterations to this mechanic made it much easier to design levels around this ability and solved the issue that I was struggling with, which was players not having enough time or reveal uses to see the pathways.

Concise and Commented Blueprints - through the first sprint I focused a lot on refining my blueprints, making concise and clear edits and comments to them and I ended up with a well-refined mechanic by the end of the first sprint. This helped a ton going into the second sprint because I could go through my blueprint quickly and make iterations by reading comments to see where everything was happening. Not only did commenting help me stay organized but it also made my mechanic more presentable to others if they were to implement the mechanic in their model.

Rapid Iterations and Building Up Mechanics - Through the first sprint I was making rapid iterations but more so to stabilize the mechanic and make the blueprint more bulletproof. The second sprint is where I noticed the impact of rapidly iterating and building up the mechanic. During the second sprint, I began using more breakpoints to see where and when blueprints were triggering and this helped me pinpoint how each blueprint was communicating with one another. This helped me solidify my mechanics and make much-needed iterations to stabilize the mechanic and make it usable. These rapid iterations helped me refine and build up this mechanic designed by Thomas Morin and allowed me to pinpoint areas of need to increase the stability of the mechanic before polishing the project.

Managing Progress with Trello - Before this project, I managed most of my projects through game design documentation on google docs and systems through google sheets, but I hadn't used any project management software like Trello or JIRA. With this being my first time using Trello and project management software, I was a bit hesitant about what I should include in posts and decided to be extra thorough by adding screenshots of blueprint snippets, playthrough videos, mechanic walkthroughs, and updates on changes or issues I had. This helped me later on during the stretches because I was able to go back through the progress I made and see the areas I had previously worked on. For instance, I believe I was working on the interactable ball's movement and noticed that in the Trello I had talked about adding a second push bool and now for the ball's movement I didn't need this bool instead I could use an integer to track the ball's location. This saved me a lot of wasted time spent going through the blueprint and setting up breakpoints to figure out problem areas. Instead, I replaced this variable with the needed variable and went on with my day. This happened numerous times through both sprints where I would check through the Trello card to see if I had made prior changes that would need to be altered. While time management was an issue in this sprint, using Trello improved my time management tenfold and I was able to implement both mechanics within a day allowing me to focus the rest of the week on solidifying blueprints and polishing the mechanic.

What Went Wrong

Interactable Ball needed multiple iterations - The interactable ball blueprint started as a blueprint that would find the nearest non-enterable tile and push the ball to the tile before that non-enterable tile. Eventually, I realized that this led to the ball being pushed into the corner, to counter this I made the ball push two tiles before the non-enterable tile, then I made it so the player could push the ball a second time. After all of the iterations, I still had to add in the second mechanic, the "Reveal the Way" mechanic, which added more tiles that I needed the interactable ball to check for. This only made tracking variables more difficult because I needed to make sure that each of the OnEnter and OnExit events was calling the proper tiles. The need for these many iterations of the Interactable Ball ate into a lot of the time needed to spend on implementing and polishing these mechanics.

Lack of Depth Leading to Poor Readability - One of the major issues that I faced while implementing the "Reveal the Way" mechanic was trying to increase player readability. The game is from a top-down isometric perspective and I was trying to use one static mesh plane with a simple green material but noticed that it was very unclear where each invisible tile was. Since this mechanic requires memorization and the game requires players to count spaces to understand where they need to move and push the interactable, I needed to make sure these spaces could be differentiated and to start all of the spaces would blend. I made multiple iterations from different materials to different meshes until I ended up realizing that one reason I couldn't differentiate the height of the invisible tiles and why falling didn't feel right was because the objects were floating over a flat void of a background, so I added another plane under all tiles to create a base of lava in my level. After adding in the base lava planes I noticed it helped but everything still blended, so I opened google drawings and made a quick lime green square with a black border and added this texture to a material that I then used on the squares. This helped differentiate the tiles, but the depth still wasn't there so I had to eventually add another translucent material to a cube static mesh that made the pathway look more like a hidden rock pathway suspending the player above the lava. While the outcome of the readability is much better, it took a lot of iterations to come to this final version and the time that I took reiterating the visuals of the pathway could've been used to polish other aspects of the game like the user interface and HUD.

Difficult starting Level Designs - After spending many hours reiterating the mechanics I still needed to work on proper level designs that gave players puzzles, introduced mechanics, and eased the player into the game. After playing through the build I had made I noticed I could've benefited from more outside playtesting because I thought my levels were easy just because I had played through them hundreds of times, in actuality the first couple of levels should have introduced the mechanics more gradually and given more UI and HUD feedback to players. I think using more outside testing or reaching out to the team with more questions on level design would have benefited the pacing of the gameplay and improved overall understanding of the mechanic in the three-level prototype.

Lacking Unique Player Facing Interface - During the first sprint I was more focused on ironing out my mechanic and making my code more readable, commented, and unbreakable and I started the second sprint with the same attitude making a rough iteration of the mechanic by the halfway point of the second sprint, unfortunately around the same time halfway through this week I also became sick, so a lot of time was wasted because I was too delirious to make proper changes to my blueprints. I eventually got to a point after 2 or 3 days where I was able to go back through and make necessary fixes to the new mechanic I was implementing, but this also left me no time to make great changes to the HUD or UI. I was able to implement tracking of reveal uses and put that on the HUD, but wasn't able to make major changes to the HUD's aesthetic. While this didn't affect gameplay drastically it feels less unique and more basic with the basic compass and HUD text. I think adding unique text and focusing on visual feedback and UI design would have benefited the overall aesthetic of the build.

Time Management and Time Crunches - At the end of the two sprints I realized that a lot of the issues and troubles I had with this project stemmed from time management. I would often find myself obsessing over fixing one area of the mechanic when I needed to step back and take a minute to evaluate the whole process. Nearing the end of the second sprint I used more breakpoints which I will be utilizing more in the future to pinpoint and narrow down problematic areas in my blueprints. I think that adding daily task lists also helped with time management. During the second sprint when I found myself in a time crunch I decided to write down short, concise task lists that I could go over and figure out what areas I needed to work on and refine before focusing on polishing other areas. For instance, I decided to wait to add audio feedback to the Reveal the Way mechanic because I wanted to make sure the movement and mechanic interaction was working flawlessly. After I had finished the mechanic and it interacted and worked flawlessly, I then added the audio. Time management is a crucial area for development and since I was sick for the majority of the second sprint this put me into a time crunch nearing the last half of the sprint. This could've been avoided if I had better planned out my time and taken an hour a day during my sick days to implement things that were less blueprint intensive like audio, materials, or even designing the UI. While nobody wants to work while they're sick, the reality is that sometimes we need to push through it to make deadlines and this time I feel like I learned that even through sickness it is important to make consistent and steady progress to avoid time crunches.

Conclusion

This project was a very enjoyable one and I was able to challenge and push myself to my limits by designing, implementing, and polishing these two mechanics. I am satisfied with the outcome of both mechanics and I am looking forward to building on this project to display it in my portfolio. I am looking forward to using some of the things I've learned throughout the process like project management software, rapid iterations, and well-commented, refined blueprints. I also look forward to finding ways to better improve my time management and prevent time crunches, so that in future projects I can focus the time needed on more player-facing information, UI design, and HUD design to give my games a better overall aesthetic and increase player understanding in my games/ levels.