Aberration Games - Deeper

P3 – Post Mortem

4/20/2022

Pitch

Deeper is a 2D action-survival-adventure game where players are tasked with drilling into a dangerous and mysterious dig site in search of a valuable diamond below. Many dangers lurk below, extraterrestrial life seems to have invaded the underground. Drill strategically, upgrade your gear, and do not run out of light. Discover the mysteries that awaits for you when you go Deeper.

Creative Process

Our team, at the beginning, was inspired to work on a rogue-like horror digging game. We would have a player combat system and the player would be tasked to fight off the enemies, extraterrestrial life that has invaded earth and had burrowed itself underground to regain strength, in order to save the planet. On top of the combat system, we would have upgrades to our combat system where players would gradually get stronger and be faced with stronger and more difficult enemies.

We came up with this idea by jotting down many different genres and types of games we could come up with, and a bunch of the best ideas that we like together. As for player impact, we wanted players to have a sense of an ominous vibe throughout the game, have the wondering what kinds of creatures will appear next. We also wanted players to feel as sense of adventure, so we made the map as big and open as possible so that players would have room to explore the depths of the map. We did this by having multiple directions in which the players can go to complete the game. Doing this made the game feel less linear, and more “open” in terms of freedom in exploration and choosing where to go. We still had to add some linearity to our game so that the players wouldn’t just feel as if they are lost, and at the same time feel the sense of progression towards the end of the game.

Development and Iterative Process

Given our initial idea of the game, we went forward and tried to make our rogue-like-horror-adventure game. Through the process of building this game and wanting the ominous vibe, we began by creating our scene and making it as dark as possible, but to complement this we first needed to revolve our mechanics around light. With this, we gave our player a light source that gradually becomes darker, and of course we had to add the digging functionality, this was our gold spike. With what we already had in our gold spike and the feedback, our team had begun to spark new, yet different ideas. Upgrades that revolve around light source, enemies that 1 hit you, and consumables that helps you survive in the dark. With many ideas every week, we eventually steered away from our original rogue-like game and went more into a tunneling-survival-like-game, similar to Motherload, eventually we ended up here with Deeper.

We utilized the Analyze, Design, Implement, Playtest (ADIP) pattern throughout the entire process of building the game. For example, at the beginning of the game, we needed a minable block, the dirt, and we began designing it and implementing it. Through playtesting and feedback, we found that the dirt was not organic and felt blocky. We then ended gave it a rough dirt-like texture with a couple of different dirt sprites, and we randomize through them for each dirt to keep them feeling like dirt. We went through multiple different ways of making dirt, before ending up with our current results. Another way we utilized ADIP is for our upgrades.

Throughout each week from playtesting, we had to constantly change our upgrade system and abilities, given our feedback from the games. Some feedback made us realize that players don't use the abilities enough, other feedback made us realize that players don't want to constantly buy consumables, so we made crucial items a permanent upgrade. Some abilities felt lackluster, so we reworked the abilities and changed the light beam to a Shock stun that stuns all enemies around you, and another ability called the mining beam the tunnels all breakable blocks in front of you, and at the same time stunning enemies, but is required to aim to stun. We also added in shields, so players don't die as much, and can progress faster. These are some of the many things that we came up with utilizing ADIP and player feedback.

Playtesting was performed within our team, our peers, and our instructors for this class. We received a ton of feedback, about our light source, our upgrades, power ups, sound, shop system, UI, controls, level design, etc. We tried our best to follow through with most of these feedbacks, ended up creating the game that we have today based off of mostly feedback from gold spike all the way to beta. Our game, Deeper, had basically became what it is today, solely from feedback. Our task time estimates were always off. I noticed that we were always off by around an hour more, sometimes our estimates were off by a few ours less, but at least it was somewhat consistent. I do not believe that our task time estimates improved overtime, as we always tried to overestimate each task to give us a clear idea on how much time we need to spend.

Future Improvements

There were a lot of things that went right during this project. Of the many things that went right, I believe that we had a great idea for a game, and it was implemented well, I would say. The game was sure entertaining for me, and I felt it was entertaining for some others as well.

For me personally, what went right in this project was, really, creative process and learning many new things about unity. We enjoyed seeing others play the game that we worked so hard on for the past six weeks, and at least I believe that's what mattered most for this project.

For what went wrong in this project was most likely our last submit for alpha. As a lot of us are preparing for our exams and final projects/assignments in other classes this last week, we did not have a lot of time to work on the game. Despite this, we were ambitious enough to have decided on creating a boss level on top of fixing bugs, and adding a few more additional things, for our final submit. We thought it would have been a great ending, with final boss music, and an overall great vibe to our game. Aside from the final boss, we had finished adding all the stuff that we wanted to add and fixing all the noticeable bugs by Monday night. Despite being late already, we wanted to add the final boss, so some of us had grinded all night to finish the boss. With only a few hours a sleep, we eventually had the boss and the final level completed by 5:40pm Tuesday. Everything was looking fine, and we took 20 minutes to build everything and submit it before the 6pm deadline. When we saw Austin play our game on the live stream and encountered a bug that made the game unplayable, we were crushed. We did not test the built game and apparently the bug only appeared in the built game, but not on unity itself. But worst of all, we did not get to see Austin or Max play through our debugged version of the game.

A measure that we will take to ensure a better result in the future is to always playtest the built version of the game before submission. On top of this, this taught us to better manage our time, and not be ambitious in the last minute. If we were to just have a 3rd zone without making a completely new things, AKA the boss level, but instead just build a harder level, we would have been able to catch the bug, as we would have time to build everything, and playtest through it and really polish up the game.

EECS 494 Reflection

My time here in EECS494 was overall enjoyable. I met and worked with some really amazing people and really got to express my creative side, in building these games, something I haven’t done in a long time. But I believe my biggest takeaway from this is the fact that I get to learn Unity and get an insight on the general game development environment. Game development could be a path that I take for my future, so I believe that EECS494 really helped me get my foot in the door.