YOOBEE COLLEGES

Crop Duster

Game Development Report

Table Of Contents

Introduction

This report is discussing the development process of the project “Crop Duster”, the main purpose of the development of “Crop Duster” is a group project assignment for CS205. The goal of the project was to develop a Game using Unity Game Engine that would incorporate unfamiliar technology within the development, in this case it was originally decided among the team to create a Sandbox VR Game being Crop Duster but due to having a lack of access to a VR headset for the development and testing process the controls had to be compromised down to a Keyboard and Mouse controls. With the development of the project it was also a learning experience along the way to use unfamiliar technology within the project to further develop and streamline certain processes that could help the workload during development.

Project Overview

Within the team there are three members Jackie, Jordan, and Dan that developed the project focusing on different sections of the game. Dan was tasked with creating world tiles that the player would use to place into the world, Jordan was tasked with incorporating the player controls and doing the world placement of the tiles, Jackie was tasked with creating the UI for the game and creating Ai for the NPCs within the game to gather resources. As many of the functions of the game would have to work with each other it was important to remain in communication with each other and also help work on certain tasks to help the future development for the game as after each section of the game was to be completed, the project was to merged using Gitlabs and would possibly cause errors and conflicts between each branch if everyone did each section not to specification.
The main target audience of the game was “G” for General, as the game achieves to be able to be played by anyone which could be ranged from kids to adults, the focus of the game was to be fun with no blood or violence involved, with this type of target audience it was possible for the game to incorporate fun assets that would appeal to many people.

Main Core Functions

- Build
- Be Creative
- Provide for AI

Secondary Functions

- Generate Disasters
- Defend or Destroy
- Procedural Events

Implementation

The game was developed using Unity Game Engine that uses C# scripts, packages and assets that were used in the game were downloaded from Unity asset store and Github. Since the game was originally planned to be a VR game SteamVR package was used and the preferred package to incorporate VR as it could be used for different types of headsets and had in-built settings streamline the development for the game. The assets chosen were Low Poly asset packages as it was the design style chosen for the game and made it work well with the tile building. The entire project had 6-8 weeks to be completed with the workload being equally distributed, with each member's contributions being uploaded to Gitlab within the respective branches to check the progress of the game's features and check if the project was on schedule for the end of the project to reach the deadline.

For the development of the Player Controls the time and research was put into understanding the development for VR however Due to Covid-19 restrictions and having a lack of access to a VR headset the idea was changed to convert the player controls into Keyboard and Mouse controls. The controls are standard with WASD controls to move the player around with SPACE and Left CTRL to move the player up and down, MMB was used to display the cursor in the game for the player to interact with the HUD and place tiles into the world. Tile placement in the game proved to be a bit difficult as the there were many ways to tackle using placing objects from HUD to World with each way having their own strengths and weaknesses, in the end the the best solution was for the player to press a button from the HUD which would spawn an object that would follow a Ray Cast of the mouse in the world, and when the player would confirm the placement with Left Mouse Button the object would be replaced with a permanent tile in the world. The player ability was under a time constraint so only one ability was added which would cause an earthquake in the game, this was easily done by adding a Camera shake to simulate an earthquake and have a chance to destroy buildings in the game. At the end of the development the Merging all branches proved to be a bit difficult for some branches as the each contribution had conflicts with other branches that could only be sorted manually but proved even more difficult, luckily with Unity it was possible to create export packages that could be added into another project, which was used for some branches that had too many conflicts.

Conclusion

Overall the entire project was a good learning experience to develop on Unity Game Engine, although with the constraint of Covid-19 restrictions hindering the ability of achieving more for this game and having to change features for what could be realistically achieved for the game, all the main functions were completed. The game is currently in a state where more features can be added later in the future, where VR could possibly be added and updates that could add more of the features that were originally stated within the GDD. Through the development process of the entire course of the project all members have a better understanding and knowledge on how to work with Unity Game Engine within a team setting. In the future the game does indeed have potential to have more features updated or added in the Game and could possibly use more advanced packages that could improve AI. At the current state of the game it functions with bare basic controls and gameplay and it does have the fundamentals to be changed in the future.