Project Design Document

11/29/20121
Allyn Alves - Frameplay Demo

Planning

1 Player Control

You control a: A player object on a grid area

in this: Top Down 3D | game

where: WASD / arrow keys

makes: The player move around the screen

2 Basic Gameplay

During the game, Collectible objects | appear

from: Random positions on the playable ground

and the goal of the game is to: Collect as many of the objects as you can and get a high score

3 Gameplay Mechanics

As the game progresses, The player grows in length

making it: Harder to avoid colliding with themself and obstacles around them

Notes: This is mainly a demo for coding, so focus will be on the item trailing, visual path/ground trailing effect, traversed area tracking, and re-visited area tracking.

4 User Interface

The: score

will: increase

whenever: Collectible objects are attained

At the start of the game, the title: “Reactive Ground” | will appear

and the game will end when: The user closes the game, it is just a demo for procedural coding :)

5 Other Features

I want the code in this project to be easily modifiable. I will do this by separating out individual scripts for different behaviors and adding easy-to-modify components/variables that show in the Unity editor Inspector area.

Documentation

Possible Technical Issues	-Sometimes things work great in the editor and not as expected in the build, so I want to make sure to test that as I go by making builds each time I think I have the project playable with the desired behavior -I am making the main part of the coding in a 1x1 grid format (I really like the idea of making a Snake-like game), so some of the features may need extra modding to apply smoothly to a non-grid type game if the code were to be polished up to become a plugin for multiple game types. -I am also using Lists to store the data for visited tiles and re-visited tiles, and I think that can probably be optimized for better performance later, but for the demo purpose I think it will be ok -Might not be technical, but managing my time for the project. I work a lot of hours right now throughout the week/weekend and even though a project is complete it may take a while to refine for playable builds and sharing with others so I wanted to make sure to set aside time to get everything into a shareable format. I did this by setting aside time in my phone schedule after work in the beginning of the week before visiting family for the Thanksgiving holiday, and setting aside more time before or after the jobs I had Sat/Sun when I returned. .
Experimentation	The main thing I experimented with on this project was how the positions the player visited and re-visited were handled. I also experimented with how different combinations of behavior felt while playing the game, and decided to make that a main feature in the demo.
Things I had to lookup	I look up a lot of things when I work on projects :) The first thing I did was watch a couple of tutorials on how to build a Snake clone game and picked the features I like the best from each video and incorporated a similar method into my project. This included the way objects trail behind the player object and the simple top-down view to make it as straightforward as possible. How to properly adjust the FixedUpdate timestep in project Time settings so the player moves at a speed that was challenging but doable and fun. How to procedurally generate a simple grid, used that and made it modifiable to be a 3D grid if the user wants to go that route

Bugs found and bugs fixed	At first I made the visible traversed/visited tracking tiles spawn from the player and trail behind in that way, and it looked cool and worked well in the editor but when I went to build the game, the .exe game did not work properly. Tiles were spaced out with 1x1 empty spaces between them and the re-visited tiles spawned in the empty spaces between tiles when they were only supposed to show up on top of the Visited tiles, and only when those tiles have been triggered on a second visit by the player. After spending a couple hours trying to figure out the issue I decided to scrap that code, take some time to think of a new solution, and try a completely different method. I ended up making a grid of tiles spawn separately from the player, and having each individual tile have scripted behavior. This worked great and I think it is also easier for users to modify the ground behavior in this way.

Project Timeline

Milestone	Description	Date Finished
#1	- *Full functionality for growing object trail that follows player. First (buggy) draft of coding for visited path, re-visited path, and ground trail effect.* - *Set up version control with GitHub*	*11/24*
#2	- *Redesigned the scripts for or visited path, re-visited path, and ground trail effect. All fixed :)*	*11/26*
#3	- *Demos now available - Web GL link to Unity Play added*	*11/27*
#4	- *Functional feature(s) by milestone #4*	*mm/dd*
#5	- *Functional feature(s) by milestone #5*	*mm/dd*
Backlog	- *Feature on backlog - not a part of the minimum viable product* - *Feature on backlog - not a part of the minimum viable product* - *Feature on backlog - not a part of the minimum viable product*	*mm/dd*