

Game Design Document
[PLAN B: Renegade]

By

Student #2980656
Michael Mitter
COMP486
Dr.Ali Dewan

Table of Contents

Game Mechanics.....	3
Game Story.....	6
Target Audience.....	6
Levels.....	6
User Interface.....	6
Technical.....	7
Assets.....	7

Game Mechanics

The finished product is unfortunately a little incomplete. I have the previous game design document at home and I am in Mexico and likely won't be returning for years until the madness at home dies down a bit, I wrote this previous to the war act being declared so it looks like I won't be getting my former document back at all.

I can give you my recollection of the original design document. I wanted to continue with the mechanics behind my first platformer but after arriving in Mexico, I noticed that would be impossible as I accidentally left key Unity files at home.

This game was initially going to be a way for people to learn English, I had all sorts of battle systems drawn out so that players could navigate through mini-games designed to help them improve their English while at the same time being drawn into a story. The battle specifics were going to be based on the ULTIMA game series for the original NES with a battle scape that the player could navigate their party around and use long distance melee weapons and spells that would surge across the field.

The ESL aspect would have taken place during the weapon/spell usage. This would have resulted in a sort of micro game to keep the mechanics fresh and interesting. I also wanted to include an element of one of my favorite games XCOM, specifically the base building aspect. The RPG was going to be based on some simple 8 bit graphics and followed an alien invasion story starring a cat and a young hero. The control scheme for the over world was going to be simple but also include elements of SIM games where you could build the area around your base with a functioning economy. I was intent on using some of the conspiracy theories I heard on a channel for the story, unfortunately, they've now become fact so I changed up the story a little again as this is supposed to be a sci-fi not a historical representation.

Eventually I would like to re-integrate some of my game ideas from the original document to make the battles a little more interesting. I loved playing through Shadowrun on the Genesis as it was sort of a sandbox game with a few story flags that triggered when you met / accomplished a goal. I used that as the basis of my story flags mechanic. This was ultimately removed after the co-routine disaster. Hopefully, I'll be able to re-integrate it in the future. It also included two different battle systems, one an action RPG and the other a turn based affair reminiscent of the Phantasy Star series. It showed that you didn't always need to have a party of 4+ characters to keep things interesting during a battle, you could just have one character in an interesting play system.

I also removed most of the complexity involved in the game, simply because I don't have the time to put it in after so many design workarounds. It would be difficult to implement the intricacies after tweaking the engine to work without co-routines.

The game was supposed to be kind of complex in its workings but after trying to use several different (expensive) Unity assets [and failing] I determined that I might as well just code it from the ground up and so I abandoned several weeks worth of work to pursue a more traditional course.

I used tutorials all over YouTube and Zenva to help me construct and merge different systems. After writing the battle system, which took a horrendous amount of time through an old Brackey's tutorial designed for a different version of Unity, I figured I would have to abandon 90% of the 'ESL' system if I was to complete the assignment on time. It turned out that I had to abandon 99% of it and start almost from scratch with just a basic ATK/DEFEND system. I wanted to add an action RPG battle system, somewhat like Chrono Trigger, but the NPC movements kept getting stuck and the amount of work involved tripled so I went with a dumbed down the system that resembled Pokemon where you just walk around and get into encounters. I realize this is a bit annoying but it was a much more logical fix without a lot of added headache. I was able to re-purpose trigger colliders for this and use them as random encounter zones. The Random Encounter zones constantly roll numbers in real-time and have to match a single digit in order to trigger an encounter. Surprisingly, the encounters are frequent despite the criteria.

The game's base building aspect was replaced with a more traditional JRPG overworld but when I had problems merging the scenes together, I abandoned that idea for a SSI computer RPG feel like the one in the DarkSun series where you are placed in a map and there is a pretty obvious pathway. I initially had a HUGE script written up for the game which unfortunately needed to be cut down to just a few sentences for each interaction. Upon compiling the game, it turned out that the asset that I used to invoke co-routines to pause the in-game time, gave the engine all manner of problems and literally BROKE Unity. It told me that the file didn't exist in the right directory, which seemed like an easy fix, I just moved the file to that directory. Boom, the entire Engine vomited my scripts from their objects and triggers and refused to acknowledge their presence. I thought my entire project was going to have to be scrapped and that I would fail the course. I eventually found a solution, which doesn't make any sense at all but I returned the file which fixed all those errors to its 'non-existent' status in the previous folder and lo and behold, the Unity editor righted itself without me doing anything. Even the tinkering I had done to get it back together was completely over-written and all the scripts returned to their rightful places. I have no explanation for this, it's almost as if the file was haunted. So with this in mind, I had to rebuild my co-routines, which unfortunately did not work as intended and ended up causing all manner of additional errors. As of tonight, I've gone about deleting most of the dialog and leaving in just the base. I can't afford to spend additional time trying to build a system for this without abandoning my other courses.

As for the platform, of course WebGL was assigned but I also purchased an asset that allowed me to build the assignment for the Switch. It cost around \$50 but I felt it was worth it as I could release my 'game' on the Switch with only a bit more work. That was until Nintendo and Unity decided that they were no longer co-operating and took away that functionality. That was about a week in total right into the bin.

I bought several assets to assist in the creation of an inventory and save system. None of them worked out of the box and the CORGI templates require a separate engine which I don't have extra time to learn. Hopefully in the future I'll be able to make games with them, as of now, they are currently not useful.

The graphics were originally supposed to be fairly simplistic and I tried using TILED for Phaser but for some reason I couldn't get Phaser to work properly. I even purchased the IDE from the Phaser webpage, still ran into problems. Then I tried using TILED for unity and kept running into problems with the triggers not importing correctly. I bought an asset meant to convert TILED maps to Unity, of course it didn't work in the version of Unity I was building the game in. I eventually used Youtube tutorials to learn to build the maps through Unity using a 3 layer system. It wasn't easy and there were all sorts of glitches in the map, you might still find a few here and there, please let me know if you do. I tried to test the heck out of my game, but I still find them every now and then in development play throughs. I purchased a new set of graphics from Itch.io that I thought made the game look cooler and then went about over-writing my maps with them. They proved to be pretty difficult to do because of the layers required to make the images seem 3D-like. So far I've somewhat succeeded but the shadow tiles seem to be missing in certain places and I'm not sure how to deal with this at the moment.

The audio is mainly soundtracks I purchased on the Unity asset store. The battle system utilizes an old Nintendo sound pack which I bought on itch.io, it actually turned out better than expected. The control systems were initially built with Playmaker but the Playmaker code is almost entirely incompatible with regular Unity C# code and the syntax to do something simple like add two numbers together is insanely complex. I ended up totally abandoning Playmaker for this reason and had to convert my player's control scheme to the game I had already made.. Another problem was that in Playmaker, you use teleportation more than the physics system. This probably works well in VR and FPS games but for 2D RPG's this makes it difficult to interact with the map and Unity's native trigger system, so I had to rewrite that aswell. Basically every 'time saving device' that I purchased to help me build the game was abandoned in favor of just coding through the traditional method of scripts. You would have to dedicate yourself to never learning to code in order to get the fullest use out of the Playmaker asset. Sad that the developers didn't consider rewriting more compatible code for the 2021 version of Unity but Playmaker is 8 years old at this point.

Influences

Dragon Warrior 1 through 5 (NES)

Final Fantasy 1 (NES)

ULTIMA (NES)

Chrono Trigger (SNES)

Robotrek (SNES)

Secret Of Mana (SNES)

Shadowrun (GENESIS)

UFO:Enemy Unknown / XCOM 1 (PC)

Game Story

The game story was sacrificed due to co-routine problems. It takes place in the present day where a scientist has a head on collision with the player on a road. The contents of a vial the scientist was holding somehow make their way into the player's body. The player comes to find out through a voice in their head that an 'otherworldly' creature now inhabits their body in some form. The creature is a spoiled princess and tyrant, forcing the player to walk towards a series of passage ways in hopes of re-uniting with her people and taking over the player's body. Along the way it is discovered that the player can harm the parasite by harming themselves. It also comes out that the otherworldly creature is a type of alien cat and the two share a series of interesting philosophical dialog eventually culminating in a sort of romance. Later, the player would be able to deliver the parasite into a catlike body which then the two become inseparable as they try to both save and takeover the world, followed by an army of cats who the parasite can communicate with.

Target Audience

The audience for this game would have been JRPG lovers and casual gamers. There seems to be an entire following dedicated to this genre, so I thought this story would be a fresh take on the series. Most of the games of this type are usually in fantasy settings, rarely do you get a horror RPG that doesn't include Zombies. I thought the anti-hero aspect would add some fresh gameplay to a generation of boring 'save the princess' troupes. Princesses are rarely worth the trouble and I wanted the game to reflect this.

Initially, the ESL aspect was put in place to appeal to ESL students. This may be re-implemented in the future.

Levels

At first the game was to be an openworld / sandbox/ tower defense game with elements of strategy interspersed. Later as difficulties with TILED arose, the level progression was turned into a simple map based system. The Player navigates their way through the map to the end where the next in the series starts.

User Interface

I wanted this to be as simplistic as possible. Currently, the control keys are W/A/S/D and the mouse is used a little as well to click the onscreen buttons during the battle scene. I would like to simplify this in the future to ALL keyboard interactions [and onscreen presses for a mobile version]. Other than that, any additional elements will take the form of on-screen buttons so that the user won't require an 'instruction' manual to use the game. I don't want the game hindered by a complex set of controls, everything should flow naturally so that the player can get down to the important stuff like the story or play mechanics.

Technical

The game is built in Unity 2021 with about \$515 worth of game assets. It was coded in C#.

Update [8/6/2021]:

Assets

Graphics

Runtime Pixel 3D Mesh \$10
Ultimate SFX \$30
SmearFX \$15
JuiceFX \$15
Pixatool \$15
Horror City - Sewer and Factory \$4
Horror City - Retail Tiles Add on \$6
Horror City - Tile Pack \$20
Horror City Sprite Pack #2 \$5
DithArts Sci Fi Characters \$5
All In 1 Sprite Shader \$34.95

Audio

Universal Soundscapes \$20
Ultimate Sound Effects Bundle [90 Game Assets] \$10

User Interface

Text Animator \$30

Pipeline

Universal Device Preview \$44.99
Playmaker \$65
Script 3 Inspector 3 \$40
Feel \$15
Stream Deck Integration \$15
Universal Device Preview \$45
Top Down Engine CORGI \$30
Full Screen Editor \$10

Engine 2D Corgi \$30