Software Development Project

SD4 – “RPG”

Report By: Kevin Kleber, Scott Ohlrich, Ricky Sanders, & Steven Spangler

Course: Intro to Software Engineering, 3626, 1

Project Dates: 3/26/2015 – 4/30/2015

Date Submitted: 4/9/2015

# Introduction:

* Purpose:

Role Playing Games are very popular right now; however, there are very few that are free to play and allow you to play with your friends.

* Mission:

Develop an RPG with distinct classes and enemies that allow for an immersive multiplayer experience.

* Vision:

The player gets to enjoy playing a new RPG by themselves or with friends. The player will enjoy a new form of entertainment that is easily accessible and free to play without worrying about monthly payments.

# Description and Scope:

The goal of this project is to develop a Role Playing Game that allows for network multiplayer. This program will be completed by four group members over the course of about a month. This game will be created using Unity 5.0 with C# as the main coding language. The final product will be a relatively small prototype to act as proof of concept for a larger game. The objective of the game is to strengthen your characters through increasingly difficult levels of enemies and to avoid character death or your character will be lost forever.

# Estimates/Plans:

The overall estimate for this project is that it will take 32 hours to complete. We have also found it more helpful to measure our game in terms of number of scripts rather than lines of code due to the nature of the project. We have estimated a total of 16 scripts will be necessary. This time estimate takes into account that some tasks will require team meetings with all four team members. As a result, the time for completion of these tasks has been multiplied by four. We estimated that the preliminary planning, requirements, and high-level design phases should take around 3.5 hours. We have noted 5 major components for our game: characters, enemies, environment, user interface, and networking. Overall, designing, coding, and testing for the characters should take around 6.75 hours and 5 scripts. One script will be a basic shell containing base character stats and movement. Each of the four classes will also have their own script. Enemies will take 6.75 hours and 4 enemies. Again, there is one master script, and one for each type of enemy. The environment should take 2 hours and 3 scripts with the exact same breakdown. The user Interface should take 4.75 hours and three scripts and will be responsible for handling all of the menus. Finally, networking should require 1 script and 2.75 hours for development. We have also included 5 hours for debugging and .5 hours for analysis in these estimates. These time estimates have been based off of our previous project experiences.

# Requirements:

* The player shall be able to choose between four classes of characters
* The player shall be able to control his character’s movement
* The player shall be able to activate his character’s abilities
* Each character shall have distinct attributes (ex: speed, damage, health)
* A player shall not be able to play as a character that has been defeated (no more health)
* A player shall be able to make a new character of a class that has been defeated
* The game shall have at least two distinct environments (ex: desert, tundra…)
* The player shall be able to save his progress and resume it at a later date
* The game shall have at least three enemy types for the player to defeat
* The player must see growth in his character through experience and/or stat boosts
* The game shall be made using unity
* The game shall be orthographic
* Enemy strength shall scale (way to scale is to be determined)
* The game may allow for multiple players to be in the same environment at the same time\*
* The game may allow for players to play together over the network\*
* Levels shall be generated pseudo-randomly\*
* The player shall have an inventory\*
* The game shall have items for the player to store in his inventory\*

\*Requirements with asterisks are not required but would be nice

# Networking Requirements:

* Players can connect to a server (client)
* Players can host a server (host/server)
* Clients can send commands to their character through the server
* Hosts can safely disconnect clients, including voluntary disconnects and when clients are not responding
* Hosts should not leak memory from network calls
* Hosts update clients through information trackers
* Hosts have authority, clients have none

Scripts:

Network Menu – Hosting/Connecting server

Network Connector – Connects players

Network Controls – Sending/Receiving player commands

Network Trackers – Updates clients with information

UI -