Requirments Document Rev 0
A World Apart

Team SantaHatesPoorKids
Jim Wu, 001409055
Ian Yang, 001217664
Gabriel Castagner, 001412885
Junhao Wang, 001215428

October 13, 2017

Contents

1 Purpose of the Project

1.1 Background

This game is to being developed as the main project for the SFWR ENG 4GP6 capstone course. This game had been an original concept that never made it out of it's idea developing stage and the development team decided that the game's idea and asthetic would be a great project to make.

1.2 Goals

The development team wants to create a enjoyable strategic rogue based game as a personal achivement. This game has a few aspects that have open ended solutions such as map generation and random chance variables that will give us the knowledge of how to make them and improve upon them in the future. We as a development team want to deveope this project so that we can improve our abilities to become excellent game developers and understand the process of creating a video game using the Unity Engine. The Development team wants our audience to have an enjoyable, relaxed experience while playing this game. Roguelikes can usually be very stressful and frustraiting due to its random and trial and error nature, but we wish to change this experience and base it more on long term thinking and reward thoughtful resource management.

2 Stakeholders

There are a few stakeholders in for this project.

Player - The Player stake in the game is that they are the ones who are playing and enjoying the game. They are important to take into account as we will need to ask questions like, "Is this portion of the game fun" or "Does this feature benefit the player's understanding of a mechanic". The main concern for the player will be to ensure that the game is fun and understandable.

Development Team - The Development Team's stake in the game is that we are the ones who are creating and organizing the game. This project will reflect our knowledge that we have acquired over the course of our university career so we must show that we can develop this game properly and so that it can be improved upon in the future.

Instructor - The instructor's stake is that they will have constructive control over the duration of the project. They will provide feedback and give guidance so that the development team can create the best version of their game.

Judges - The judges stake is that they will give a review based on merit and specific guidlines. They decide how good the game is and how well it fits the development criteria.

3 Mandated Constraints

There are a few constraints that the instructor wants us to follow.

MC1. The project must be developed in the Unity Game Engine

MC2. The project's documents and source files must be managed in the respective GitLab files

MC3. The project must be a fully fledged, standalone video game that must be created from scratch with the exception of game assests (models, sprites, audio, etc).

3.1 Solution Constraints

There are a few constraints that the development team can use as solutions wants us to follow.

SC1. The project must be developed in the Unity Game Engine

SC2. Any game assests in the Unity assest store must be referenced in both the project documents and in the game folders.

SC3. Any game assests outside of the Untity assest store and not original to the development team must have documented concent to use such game assest if not under free use terms and conditions.

3.2 Implementation Environment of the Current System

There are a few enviroment implementation constraints.

IE1. The game will be published to be played on Windows, IOS and Linux Machines.

IE2. The game will be able to run on a player's laptop or desktop machine.

IE3. The game will not need the internet to functions.

3.3 Partner or Collaborative Applications

N/A

3.4 Off the Shelf Software

There are a off the shelf software that will be needed to develop the game's assests.

OSS1. The game's textures and character sprites will need to be created in either Photoshop or Gimp 2 depending on the developer.

OSS2. The game's models will need to be created in either Blender or Maya.

3.5 Schedule Constraints

There are a few Schedule constraints.

SC1. The project's Sales pitch and first demo will need to be presented October 17th 2017

SC2. The Game requirement's document will need to be ready by October 19th 2017

SC3. The first Implementation document will need to be ready by December 7th 2017

SC4. The first V and V document will need to be ready by January 4th 2018

SC5. The last Game requirement's document will need to be ready by February 27th 2018

SC6. The last Implementation document will need to be ready by March 29th 2018

SC7. The last V and V document will need to be ready by April 6th 2018

SC8. The final demo will need to be presented sometime in April 2018

3.6 Budget Constraints

There are a few Budget constraints.

BC1. The development team will allowcate $10.00 for any neccisary game assests.

BC2. The development team members will be responsible for purchasing any 3rd party software needed to develop game assests.

4 Naming Conventions and Terminology

The following are name conventions that the development team will use:

Project - Interchangable with Game and Application.

Game Assests - Used to reference the asthetics of the game. These include character models, sprites, textures, character specific scripts, and audio.

Game Developers - References the four groupmates responsible for creating the game.

5 Relevant Facts and Assumptions

5.1 Relevant Facts

There are a few facts about game enviroment the developers made.

RF1. Maps are generated using a psuedo randomly generated seed. It is theortically possible to generate the same map given the same seed.

5.2 Assumptions

There are a few assumptions the developers made about the players.

A1. We assume that all users of this product are capable of using a computer and a keyboard.

A2. We assume that all users of this product will be able to differentiate between their character and the map.

A3. We assume that all users will have an Apple or Windows machine.

A4. We assume that all users have used a WASD + spacebar control scheme before.

A5. We assume that some users have played a some sort of game that uses turn based rules.

A6. We assume that when the user achieves a game over that that specific game is no longer replayable.

6 Scope of the Work

6.1 Existing Inspirations

There are a few existing game that was are using as inspiration to model our game's mechanics and theme off of.

EI1. FTL- We are using FTL's map generation as a general concept on how the level will be connected together.

6.2 Context of the Work

There are a motivational factors that the development team will keep in mind when we are developing the project.

CW1. The development team will challenge each feature with the idea of fairness, is the feature implemented fair to the player. The team will as to see if the feature can be accounted for and deflected. If the feature can't be accounted for, will it terminally upset the player causing game breaking moments.

CW2. The development team will ensure that the art style, gui and map design is considered clean and clear. The developers will question if features like tooltips, indications and menus are easy to read and that selections are simple for the user to understand. There should be limit of the clutter on screen, both entity and texture wise.

7 The Scope of the Product

7.1 Product Boundary

This section will cover what the project will include and what it will not include.

PB1. The development team will challenge each feature with the idea of fairness, is the feature implemented fair to the player. The team will as to see if the feature can be accounted for and deflected. If the feature can't be accounted for, will it terminally upset the player causing game breaking moments.

PB2. The development team will ensure that the art style, gui and map design is considered clean and clear. The developers will question if features like tooltips, indications and menus are easy to read and that selections are simple for the user to understand. There should be limit of the clutter on screen, both entity and texture wise.

7.2 Product Use Case Table

PUC	PUC Name	Actors	Input/Output
1	Save Game	System	Save Game button initialization, Save game file
2	Load Game	System	Load Game button initialization, Save game file, Reinitalization of Previous Game
3	New Game	System	New Game button initialization, Initalization of New Game
4	Display ToolTip Info	System	Mouse x and y coords, GUI displayment of ToolTip Info
5	Perform Character Action	System	Player Input, Appropriate System output, GUI output
6	Perform Character Modification	System	Player Input, Appropriate System modification, GUI output
7	Audio Modification	System	Player Input, Appropriate Audio modification
8	GUI Modification	System	Player Input, Appropriate GUI modification

7.3 Individual Product Use Cases

This Section will cover the Product Use Cases in detail. They will include their trigger conditions, preconditions, procedures, outcomes and any use cases that are related.

PUC1. **Save Game -** The user wants to save the current game session.

Trigger Conditions: Player Selects Save Button

Preconditions Conditions: Game Session is paused, Player is in Game Menus

Procedures: Current Game session is formatted and outputted to coressponding save file.

Outcomes: System displays game file is saved, Game file is saved in proper location.

Related Use Cases: N/A

PUC2. **Load Game -** The user wants to load a desired game session.

Trigger Conditions: Player Selects Load Button

Preconditions Conditions: Game Session is paused, Player is in Game Menus, Player is in Main Menu

Procedures: System asks if they wish to leave their current game session. If selected yes then system reads desired game file and reinitializes desired game data.

Outcomes: If player selected no to leaving current game session, then the system returns the player to the game menus. Otherwise, the desired game is displayed on screen in the same state that it had been saved in.

Related Use Cases: N/A

PUC3. **New Game -** The user wants start a new game session.

Trigger Conditions: Player Selects New Game Button

Preconditions Conditions: Player is in Main Menu

Procedures: System asks to select difficulty, name the game type and waits for player input to start game. The player selects start game, the base save file is created with the corresponding difficulty and seed. Game file is generated using the corresponding seed.

Outcomes: New game is initialized, new game save file is created.

Related Use Cases: N/A

PUC4. **Display ToolTip Info -** The user wanst to view specific character values and how they are derived.

Trigger Conditions: Player has mouse icon hover over the GUI of a value for 0.5 seconds.

Preconditions Conditions: The player is viewing a menu where that value is displayed.

Procedures: The System retrives the desired value's equation and formats the dependencies in a listed format similar to:
value - dependency_name.

Outcomes: Returns a GUI format of the values dependancies and names.

Related Use Cases: N/A

PUC5. **Perform Character Action -** The user wants to perform a specific action related to a character.

Trigger Conditions: Player right clicks on any object in the scope of possible options.

Preconditions Conditions: The player has selected a character that is able to perform an action.

Procedures: The System determines if the selected option is possible, if not, then the system will provide and audio feedback that indicates that is not a possible action to the player. If it is a possible action, then the system will perform that function by updating the values of the character(s) and entities involved.

Outcomes: Returns the Graphical representation of outcome of the actions and updates the current game session.

Related Use Cases: N/A

PUC6. **Perform Character Modification -** The user wants to apply a specific modification to a character.

Trigger Conditions: Player matches a modification with a desired character.

Preconditions Conditions: The player is in a character editing menu.

Procedures: The System determines if the selected option is possible, if not, then the system will provide and audio feedback that indicates that is not a possible action to the player. If it is a possible action, then the system will perform that function by updating the values of the character.

Outcomes: Returns the Graphical representation of outcome of the modification and updates the current game session.

Related Use Cases: N/A

PUC7. **Audio Modification -** The user wants to apply a specific modification to the audio of the game.

Trigger Conditions: Player makes adjustment to an audio feature.

Preconditions Conditions: The player is in the audio menus.

Procedures: The System applies to the change to the game session base on the value coressponding to the menu.

Outcomes: Returns the Graphical representation of outcome of the modification and returns a audible que to the player representing the selection that they have changed.

Related Use Cases: N/A

PUC8. **GUI Modification -** The user wants to apply a specific modification to the graphics of the game.

Trigger Conditions: Player makes adjustment to a graphical feature.

Preconditions Conditions: The player is in the graphics menus.

Procedures: The System applies to the change to the game session base on the value coressponding to the menu. The system may need to reajust it's output values depending on the input selection.

Outcomes: Returns the Graphical representation of outcome of the modification and returns a graphical indication to the player representing that the change has been made.

Related Use Cases: N/A

8 Functional Requirements

8.1 Core Mechanics

8.2 Primary Gameplay Mode

8.3 Alternate Game Modes

8.4 Menus and other Systems

9 Look and Feel Requirements

9.1 Apperance Requirements

test

9.2 Style Requirements

test