Software Requirements Specification

for

Project Pluto

Version 0.1.1

Prepared by Brian Bach, Damien Rodriguez,  
Randy Heckard, and Matthew Montney

CSCD 488  
Group 8

1 October 2020

Table of Contents

Table of Contents 1

Revision History 2

1. Introduction 3

1.1 Purpose 3

1.2 Document Conventions 3

1.3 Intended Audience and Reading Suggestions 3

1.4 Project Scope 3

2. Overall Description 3

2.1 Product Perspective 3

2.2 Product Features 4

2.3 User Classes and Characteristics 4

2.4 Operating Environment 4

2.5 Design and Implementation Constraints 4

2.6 User Documentation 4

3. System Features 4

3.1 Player Movement 4

3.1.1 Description 4

3.1.2 Stimulus 4

3.1.3 Requirements 5

3.2 Player Interactions 5

3.2.1 Description 5

3.2.2 Stimulus 5

3.2.3 Requirements 5

3.3 Player Combat 5

3.3.1 Description 5

3.3.2 Stimulus 5

3.3.3 Requirements 5

4. External Interface Requirements 6

4.1 User Interfaces 6

4.2 Hardware Interfaces 6

4.3 Software Interfaces 6

4.4 Communications Interfaces 6

5. Other Nonfunctional Requirements 6

5.1 Performance Requirements 6

5.2 Software Quality Attributes 6

6. Other Requirements 7

Appendix A: Glossary 7

Appendix B: Analysis Models 7

Appendix C: Issues List 7

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Matthew | 6/10/2020 | Explain more features | 0.1.1 |
| Everyone | 1/10/2020 | Initial version | 0.1 |

# Introduction

## Purpose

*Project Pluto* is a singleplayer 2D fantasy roguelite inspired by other games of its genre.

## Document Conventions

**Formatting**

Items in italics refer to specific entities, such as the name of the project or a particular user class

**Feature Priorities**

A priority of 10 is the highest priority; the game is unplayable without this feature

A priority of 1 is the lowest priority; potentially crosses the line into feature creep

**Feature Requirement Prefixes**

Feature requirements are prefixed by a capital letter, which refers to the category of the feature, and a number, which refers to the feature index within the category. The feature categories are explained below, and the feature index starts at one and increments as the document continues.

**Feature Categories**

C Combat

D Dungeon

E Enemy

I Inventory/Items

P Player

T Town

## Intended Audience and Reading Suggestions

**Project Manger**

A *Project Manager* is a who is person overseeing the project, ensuring that tasks and requirements are completed on time.

**Developer**

A *Developer* is a person who is creating *Project Pluto*.

## Project Scope

**Goal**

Our primary goal is to complete *Project Pluto* v1.0 by the end of March 2021.

**Version Scope**

v0.1

*Players* starts in a single room with a basic weapon

The room has doors and items that can be interacted with

A dummy enemy is in the room that can take damage from player attacks

Future Versions

The room is populated with enemies, which must be defeated to “win”

The enemies drop their weapons, which can be picked up and used by the player

The initial two weapons will be either melee (sword) or ranged (bow)

# Overall Description

## Product Perspective

*Project Pluto* is a self-contained product. It is not a replacement or continuation of another product.

## Product Features

**General Features**

A basic introduction cutscene

Potentially a basic quest system

**Player Features**

Realtime combat with varying combat styles

Enemy weapons can be picked up and used

Currency can be spent in the base to gain permanent bonuses

**Dungeon Features**

Randomly generated rooms

Clearing a room provides a temporary bonus or currency

## User Classes and Characteristics

**Player**

A *Player* is a person who plays the game

They can potentionally have many different skill levels

## Operating Environment

*Project Pluto* will be able to be ran on computers running modern operating systems, and it will have the capacity to be ported to nearly any system, such as video games consoles.

## Design and Implementation Constraints

**Software Dependencies**

*Godot* Game engine responsible for creating and running *Project Pluto*

## User Documentation

*Players* will be provided with a short tutorial that runs when playing *Project Pluto* for the first time. If a saving system is implemented, the tutorial will instead run when creating a new game.

# System Features

## Player Movement

**Priority** 10

### Description

*Players* can move around the world.

### Stimulus

*Players* move by using the arrow keys.

### Requirements

P1-1 A player object exists and can be interacted with

P1-2 *Players* can control the player object

P1-3 The player object collides with obstructions, such as walls

## Player Interactions

**Priority** 7

### Description

*Players* can interact with certain objects, such as picking up items.

### Stimulus

*Players* interact with objects by pressing the Interact button.

### Requirements

P1-1 See Requirement 3.1 *Player Movement*

P1-2 See Requirement 3.1 *Player Movement*

P2-1 The player object must be within a small range to interact with an object

P2-2 Only certain objects can be interacted with, such as items

## Combat

**Priority** 9

### Description

*Players* and enemies can fight with a variety of weapons.

### Stimulus

*Players* perform a weapon’s normal attack by pressing the Primary Attack button and perform the weapon’s special attack by pressing the Secondary Attack button.

Enemies use their weapons when their AI triggers an attack.

### Requirements

P1-1 See Requirement 3.1 *Player Movement*

P1-2 See Requirement 3.1 *Player Movement*

C1-1 A weapon object exists and can be used

C1-2 An enemy object exists and can be interacted with

C1-3 The player and enemy objects can take damage

C1-4 The attack deals damage to entities that are hit by the attack

## Attack Combos

**Priority** 5

### Description

A *Player* or enemy’s third normal attack plays a different attack animation and has a different effect.

### Stimulus

The combo attack will only occur if the *Player* or enemy uses their weapon’s normal attack three times in quick succession.

### Requirements

P1-1 See Requirement 3.1 *Player Movement*

P1-2 See Requirement 3.1 *Player Movement*

C1-1 See Requirement 3.3 *Combat*

C2-1 Normal attacks in quick succession are kept track of

C2-2 The third normal attack has a different effect than the first two

## Stamina

**Priority** 6

### Description

Stamina represents how much a *Player* or enemy can do before needing to “rest” for a short time.

When stamina reaches zero, the *Player* or enemy is severely slowed for some time.

### Stimulus

Stamina always increases slowly and is decreased in varying amounts when performing certain actions, such as attacking or dodging.

### Requirements

P1-2 See Requirement 3.1 *Player Movement*

C3-1 A system exists to track the stamina of a player or enemy object

C3-2 Stamina slowly increases all the time

C3-3 Stamina decreases when performing actions

C3-4 When stamina is depleted, the player or enemy object is slowed and cannot perform stamina depleting actions

## Weapon Inventory

**Priority** 8

### Description

Both *Players* and enemies can have one weapon. *Players* have the potential to increase the number of weapons they carry.

### Stimulus

*Players* can cycle between their weapons by pressing the Swap button.

### Requirements

P1-2 See Requirement 3.1 *Player Movement*

C1-1 See Requirement 3.3 *Combat*

I1-1 Player and enemy objects can hold a weapons

I1-2 *Players* can increase the number of weapons they can hold

I1-3 *Players* can cycle between held weapons

## Elite Enemies

**Priority** 4

### Description

An elite enemy has a modifer applied that makes it stronger in some way, such as being larger and having increased stats, or splitting into two regular enemies upon death.

### Stimulus

Elite enemys spawn to increase difficulty, such as in later rooms as a replacement for existing enemies or as a miniboss in early rooms.

### Requirements

C1-2 See Requirement 3.3 *Combat*

E1-2 Enemies can have modifiers applied to them

## Dungeon Rooms

**Priority** 9

### Description

A room where *Players* fight enemies, gain rewards, and travel through to get to other rooms.

### Stimulus

Rooms are loaded when the *Player* enters a new floor.

### Requirements

D1-1 A room object exists

D1-2 A room has one to four doors that connect to other rooms

D1-3 Enemies spawn in a room when the *Player* enters for the first time

D1-4 The doors are locked while enemies are still alive

D1-5 Enemies do not respawn after clearing a room

D1-6 Clearing a room grants the *Player* a reward

## Dungeon Floors

**Priority** 9

### Description

A dungeon floor is a series of interconnected rooms and defeating a boss allows the *Player* to move to the next room.

### Stimulus

A new floor is generated when the *Player* starts a new run or completes the previous floor.

### Requirements

D1-1 See Requirement 3.8 *Dungeon Room*

D1-2 See Requirement 3.8 *Dungeon Room*

D2-1 The layout is randomly generated, picking rooms from a pool

D2-2 Defeating the boss moves the *Player* to the next floor

## Shops

**Priority** 5

### Description

A themed location to spend coins, such as for upgraded weapons or new skills. Most shops will be in town, but some dungeon floors can have a room that has a shopkeeper inside.

### Stimulus

*Players* start trading by talking to the shopkeeper with the Interact button.

### Requirements

D1-1 See Requirement 3.8 *Dungeon Room*

T1-1 A shopkeeper object exists and can be interacted with

T1-2 Interacting with the shopkeeper will open a trading menu based on the shop

## Wandering NPCs

**Priority** 1

### Description

NPCs that wander around town, and can be talked to by *Players*.

### Stimulus

*Players* talk to NPCs by pressing the Interact button.

### Requirements

T2-1 An NPC object exists and can be interacted with

T2-2 NPCs respond with dialogue when interacted with

T2-3 NPCs can stand in place, wander, or follow a path

## You Can Pet the Dog

**Priority** 1

### Description

There is a dog in the town that can be pet. After petting it, the dog will follow the *Player* as they wander around town.

### Stimulus

*Players* pet the dog by pressing the Interact button.

### Requirements

T2-1 See Requirement 3.11 *Wandering NPC*

T2-3 See Requirement 3.11 *Wandering NPC*

T3-1 The dog follows the *Player* after being Interacted with

T3-2 The dog stops if the *Player* gets too close to the dungeon entrance.

## Quests

**Priority** 5

### Description

Optional challenges that *Players* can complete to earn bonuses, such as an increased weapon inventory.

Some quests are special and will always appear in a playthrough, and some quests are randomly generated. The special quests will be mostly the same between playthroughs, and the random quests grant smaller rewards.

### Stimulus

*Players* gain quests by talking to NPCs and shopkeepers.

### Requirements

T1-1 See Requirement 3.11 *Shops*

T2-1 See Requirement 3.11 *Wandering NPC*

T4-1 Some NPCs can grant quests

T5-1 Completing a quest grants a reward

# External Interface Requirements

## User Interfaces

**Menus**

*Title Screen* *Players* can start the game or enter the options menu

*Options* *Players* can modify controls and game settings, such as volume

*Pause Screen* Only accessible after starting the game

Freezes the game so the *Player* can leave it running while doing other things

Allows access to the options menu

*Trading* *Players* can trade coins for services from shopkeepers

*Dialogue* Shows text

**GUIs**

*General HUD* Shows information about the player object within the dungeon

## Hardware Interfaces

*Players* interact with the game by using a mouse and a keyboard.

## Software Interfaces

The *Godot* engine and its libraries will be responsible for creating the game and porting it to all operating systems.

## Communications Interfaces

*Project Pluto* will have no communication with external machines or services.

# Other Nonfunctional Requirements

## Performance Requirements

*Project Pluto* will maintain a framerate of at least 60 frames per second on all tested machines, and the input system will be highly responsive.

## Software Quality Attributes

For *Players*, *Project Pluto* will be able to be played on nearly any operating system, and will be as accessible as possible via options, such as Colorblind mode, or the ability to disable certain graphical flourishes, such as motion blur.

For *Developers*, the codebase will be clean and modular, allowing for easy modifications and additions.

# Other Requirements

Appendix A: Glossary

**Coin:** A room reward that is kept after completing a run and can be spent to purchase permanent character upgrades.

**Trinket:** A room reward that is lost after completing a run.