Test Plan Template:

(Name of the Product)

Prepared by:

(Names of Preparers)

(Date)

TABLE OF CONTENTS

1.0 INTRODUCTION

2.0 OBJECTIVES AND TASKS

2.1 Objectives

2.2 Tasks

3.0 SCOPE

4.0 Testing Strategy

4.1 Unit Testing

4.2 System and Integration Testing

4.3 Performance and Stress Testing

4.4 User Acceptance Testing

4.5 Batch Testing

4.6 Automated Regression Testing

4.7 Beta Testing

5.0 Test Schedule

6.0 Control Procedures

7.0 Features to Be Tested

8.0 Features Not to Be Tested

9.0 Resources/Roles & Responsibilities

10.0 Schedules

11.0 Risks/Assumptions

12.0 Tools

Introduction

The aim of this test plan is to test ‘The Pixel Wizard’. This is a 2d side-scrolling rpg cross platforming game. There are several main areas that need to be tested for this game before it can go be released.

Each of the main areas that will be tested, will have sub sections that will also need their functionality to be tested. The main areas are:

* The main menu
* The pause menu
* Character controls
* Items
* Enemy functionality
* How the level functions

Objectives and tasks

Objectives

The objective of these tests is to verify the functionality of The Pixel Wizard. To ensure that it meets the specifications outlined in the appendix. The test will test the scripts and fix any bugs encountered and then test for non-script related bugs.

The results of the test should be that the game is ready to launch and be given to the client and that the scripts are re-usable for future projects.

Tasks

In this test plan the tasks are to:

* Test the functionality of the parts of the game
* Test the game in full
* Report problems back to the game designer

Scope

General

The general scope

The main sections that will be tested will be the main menu of the game, the in-game pause menu, the character controls, items and enemy and level functionality.

Within those areas there is sub areas that will also be tested. For the main menu if a save file feature is implemented in the game the load a file feature needs to be checked to ensure that it loads the correct file to the point is was saved. The options were the game settings are set will be tested to ensure the settings can be changed and once the game loads these settings carry into the game.

The range of controls for the character will be tested. Movement in each direction, jump, attack and controls to open pause menu. There are several different types of pickups in the game, when the player interacts with these either with attacks or walking over them there needs to be two tests. One to ensure on collision they interact as expected and that once they are triggered, they function correctly. This includes pickups such as potions which will replenish the players health.

Enemies will be tested to ensure they function correctly, these functions include dealing the correct amount of damage, having the right health settings and dropping or giving the correct pickups and health. The first build of the game has a set layout for enemies, if these were to be randomised the script would need to be checked so that there is a limit to how many enemies are spawned for two reasons. To keep the game balanced and not overpower the player and secondly the game would crash.

The last area to test is the levels. Testing the level includes testing that the level assets are solid, and the player will not fall through the level. The platforming aspects also need to be checked, if the gaps are to big and the player is unable to jump and make the gap the game will be unplayable.

Tactics

The first thing to be checked will be the start menu. This alongside the pause menu will take the lowest amount of time to test as they do not interact with other elements of the game.

The time frame for testing the game menu will be three days. It will be tested in order it is displayed on the screen. During this time the sub menu of options/settings will be tested for functionality.

The next item on the list to be tested is the pause menu. There will be two days assigned to the pause menu. It has a smaller list of options and if the main menu is working a lot of the script is transferrable and applicable to the pause meu.

Four days will be assigned to testing the player and the control system. The main function of these tests is to ensure the player functions correctly. An example of this would be that when the move button is pressed, they move. This will also test that the health bar is working.

Two days will be assigned to testing the enemies. What will be tested is that their health decreases on impact of the players attack, that they are attacking correctly, dropping the right pickups and that they move around the scene correctly.

Two days will be spent testing the level. This includes that the colours work correctly, that the terrain is correctly spaced, and the player is able to traverse the landscape correctly. Scene progression also needs to be tested that when a player completes the level they are brought to the next level, and also the save game function will be tested. Pickups will also be tested during this time.

The way these tests will be tracked and check of will be both physically and digitally tracked. There will be a chart set up in the main office with a break down of each section that is to be tested and a check list accompanying it to mark of if it has been completed, if the test is a fail or a pass, if the test has not been done yet and how many times it has been tested. If it has failed there will be a chart recording the changes that are necessary to fix the game.

Digitally the way this checklist will be done is by a shared excel sheet. This will be shared on google docs. Every member on the team will have access to this and they will be able to add the entries for their section.

Testing Strategy

Unit Testing

Definition

Participants

Methodology