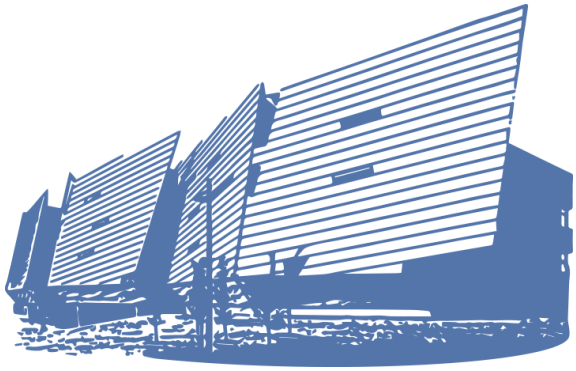

Galway Mayo Institute of Technology, Dublin Rd Campus

BSc Project

Software Testing Project

Tomas O'Malley (G00361128)

47197 – Software Testing
Bachelor of Science in Computing in Software Development
Department of Computer Science

May 14, 2020

Contents

1 Introduction **4**
- 1.0.1 Test Plan Template . . . 4
- 1.0.2 Prepared by . . . 4

2 Objectives and Tasks **5**
- 2.0.1 Objectives . . . 5
- 2.0.2 Tasks . . . 5

3 Scope **7**
- 3.0.1 General . . . 7
- 3.0.2 Tactics . . . 7

4 Testing Strategy **9**
- 4.0.1 Unit Testing . . . 9
- 4.0.2 System and Integration Testing . . . 9
- 4.0.3 Performance and Stress Testing . . . 9
- 4.0.4 User Acceptance Testing . . . 9
- 4.0.5 Batch Testing . . . 10
- 4.0.6 Automated Regression Testing . . . 10
- 4.0.7 Beta Testing . . . 10

5 Test Schedule **11**
- 5.0.1 Schedule . . . 11

6 Control Procedures **12**
- 6.0.1 Procedures . . . 12
- 6.0.2 Change Request . . . 12

7 Features to Be Tested **13**
- 7.0.1 The Features . . . 13

8 Features Not to Be Tested **14**
- 8.0.1 Features . . . 14

9 Resources-Roles and Responsibilities **15**
- 9.0.1 Roles . . . 15

10 Schedules **16**
- 10.0.1 Schedule . . . 16

11 Risks and Assumptions **17**
- 11.0.1 Risks . . . 17

12 Tools **18**
12.0.1 The Tools 18

Chapter 1

Introduction

1.0.1 Test Plan Template

Game Development International Ltd 2D game

1.0.2 Prepared by

1. Tomas O'Malley(G00361128)
2. John Smith
3. Laine Louhglin
4. Tyler Durden
5. Grace Kelly
6. Mia Wallace
7. Jackie Brown
8. June Carter
9. Jack Daniels
10. John Harris

Welcome to our Test case documentation for the latest game developed by Game Development International Ltd. The company provided me with a short brief informing me of the features/fundamentals of their new game title. The product developed by Game Development International Ltd is a 2D side-scrolling platformer that borrows behaviors from other titles such as 'Shovel knight' to 'Skyrim'. Here are a list of the characteristics/functions set in their title.

Game fundamentals

1. Allow the player to control a specific character, that has an important fictional/narrative role.
2. Have game statistics and/or relational attributes with other game objects, enemies, and/or the player character.
3. Allow the player to take on and navigate the levels using an easy-to-use user interface.
4. Have obstacles that the player must overcome, such as enemies and bosses.

The game holds a rich number of features and should allow me to create lots of different tests. Game testing is a huge part of our team's experience and we will apply the different methodologies to test and document their Product. Overall we were happy with the game Design Document provided by'Game Development International Ltd' which my whole team studied. Exclusive access to the game allowed to Alpha test the game.

Chapter 2

Objectives and Tasks

2.0.1 Objectives

Here some of the primary objectives of the test Plan

1. Defining Tasks and Responsibilites
    - One of the main areas of the test plan is to precisely define the tasks and the responsibilities of each member involved in the company and mark the Software testing life.
2. Effective Communication
    - Communication is vital when working in a team and can be a barrier to successful testing. The test plan allows the team to know the issues e.g Unit Testing bug and can allow other members to collaborate when appropriate.
3. Precise Documentation
    - Overall this test plan is being a documentation mapping the continuous integration and or progress of the Software product we are testing. This document will allow us to record the overall quality of the Companie products and will allow us to excel in the manner we overcome/analyze problems in future applications.

2.0.2 Tasks

Here are some of the tasks that I will go into detail later

1. Report Document
    - Each member of the team is required to report the obstacles they encountered during their interaction with the software lifecycle process.
2. Create Validation
    - Validation refers to the action of checking or proving the validity or accuracy of something. We aim to validate each component in the Product for example character attributes such as health using testing strategies such as unit testing etc.
3. Test Products Components
    - Each component of the product will be marked to a tester and they will be responsible for documenting their discoveries and including files such as .csv mapping the time and the component where the program passed or failed.

4. Bug Report

    - A bug report must be created and maintained throughout the testing process. Bug report systems such as Bugzilla will be used to optimize our productivity and diagnose the severity of the faults in the program.

5. Testing methodologies

    - The testing methodologies will be defined by the complexity and the severity of the Product attribute. For example if the game has an issue with users losing their save games this is more vital than in-game health issues. We must access what are the most important components e.g user security.

Chapter 3

Scope

Game Development International Ltd has a rich catalog of features in their game which I cover briefly in the Introduction.

3.0.1 General

Here are some of the technical features

1. **Menu Systems :** Test the products menu such as the start pause menu screens
2. **In game controls :** Test the basic used to interact with the game e.g W- To move forward
3. **Character Attributes :** Test the basic character attributes such as health and points.

   All of the features listed are also interfaces allow the user to interact with the 2D game and set the foundation for the game.Test cases will be designed and allocated to developers to throughly test the intergation of each componet/module.

3.0.2 Tactics

All tests will be carried out in Alpha Testing Environment.

Alpha testing is a type of acceptance testing; performed to identify all possible issues/bugs before releasing the product to everyday users or the public. Alpha testing is carried out in a lab environment and usually, the testers are internal employees of the organization.

The Scope of this approach is to find as many bugs in the game and understand how these bugs congregate in other areas of the application. To test all of these existing functions I will be incrementally testing using a procedural approach for all, An example. When testing the Save Game I would note the steps taken e.g

1. Launch game on browser/Unity Etc
2. Click PLay Button
3. Press the pause button in game e.g ESC
4. Click the Save Gamwe button listed using mosue or Arrwo keys and press enter button on keyboard
5. Exit Game by pressing exit button using arrow keys and enter keys

6. Click Resume button

7. If game resumes from save file . Test case is a Pass if not , a fail.

This is a very specific example but the incremental step , dumb user approach is the most successfully method as we can accurately test the links from small modules to larger modules.

Chapter 4

Testing Strategy

4.0.1 Unit Testing

Unit Testing is a level of software testing where individual units/components of a software are tested. A unit is the smallest testable part of any software. It usually has one or a few inputs and usually a single output. In procedural programming, a unit may be an individual program, function, procedure,etc.

The unit Testing will be carried out by myself (Tomas O'Malley).I plan to create test scripts using automated tools such as selenium to for example test the in game menus.

4.0.2 System and Integration Testing

System Integration Testing is defined as a type of software testing carried out in an integrated hardware and software environment to verify the behaviour of the complete system.It is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirement.

System and Integration Testing will be carried out by John Smith.John tested each component in the game by manually testing inputs into the game e.g Testing the save game function from in game and pre game menus.

4.0.3 Performance and Stress Testing

Performance testing is carried out to check the system's performance under varying loads.Stress testing is carried out to check the behaviour of the system under the sudden increased load of the system or software application. It only checks the stability of the system or software application.

Performance and Stress Testing will be carried out by Laine Louhglin.Laine carried out the tests by researching the scripts in the game and attacking the game by using scripts to see how many levels of the game can be played before crashing reflecting the game memory allocation.

4.0.4 User Acceptance Testing

User acceptance testing (UAT) is the last phase of the software testing process. During UAT, actual software users test the software to make sure it can handle required tasks in

real-world scenarios, according to specifications.

User Acceptance Testing will be carried out by Tyler Durden.Tyler tested the product by deploying the game on both mobile and pc platforms to simulate how the game will be interacted with in normal situations like with and without wifi enabled

4.0.5 Batch Testing

Batch testing is a comprehensive test on your current trained model to measure its performance in LUIS. The data sets used for batch testing should not include example utterances in the intents or utterances received from the prediction runtime endpoint.

Batch Testing will be carried out by Grace Kelly.Grace used automated tools to test the quality of the product by simulating random events for the product to handle.

4.0.6 Automated Regression Testing

Regression Testing is defined as a type of software testing to confirm that a recent program or code change has not adversely affected existing features. Regression Testing is nothing but a full or partial selection of already executed test cases which are re-executed to ensure existing functionalities work fine.

Automated Regression Testing will be carried out by Mia Wallace.Mia wrote a series of tests to make sure previous compoennts did not directly old modules tests on enemies shooting before and after power ups.

4.0.7 Beta Testing

A beta test is a type of testing period for a computer product prior to any sort of commercial or official release. Beta testing is considered the last stage of testing, and normally involves distributing the product to beta test sites and individual users ("beta testers") outside the company for real-world exposure.

Beta Testing will be carried out by Jackie Brown.Jackie was responsible for securing a session with a local college to play the product.Jackie asked many questions and reported back to the scrum master.

Chapter 5

Test Schedule

5.0.1 Schedule

A scrum meeting was organized and all members involved in testing or management attendance were mandatory.We mapped out an Agile Development Plan, working in work periods named 'sprints' essentially a sprint is a focus point for some time.

A milestone is a significant event in the course of a project that is used to give visibility of progress in terms of achievement of predefined milestone goals. Failure to meet a milestone indicates that a project is not proceeding to plan and usually triggers corrective action by management.

Underneath is the testing schedule outlying the timing and milestones approved by our Scrum master.

The Testing Schedule

1. Month 1 : Enviroment Set Up
    - Testers will set up their environment for concurrent testing of the game. All developers will successfully loaded the 2d game on their company machine alongside big tracking software Bugzilla and Selenium for automated testing.
2. Month 2 : Bug Testing
    - Each developer has received their testing ground e.g integration testing, Back-end, front-end testing. Developer approaches software with different strategies and successfully finds issues/bugs to report.
3. Month 3 : Bug Repository
    - After a month of testing all bugs are documented inside of a repository alongside information such as its severity, component, developer, date and time, etc.
4. Month 4 : Alpha Testing
    - Management will organize outside users to play the 2D game to simulate a public release without releasing it to the public. The Player's impressions recorded and the product is documented and reported to developers.
5. Month 5 : Management Meeting
    - Management will organize a meeting with the company 'Game Development International Ltd ' and discuss the statistics/discoveries and answer any questions the company has about the current state of the game.

Chapter 6

Control Procedures

6.0.1 Procedures

When testing the product features when a bug is found I report it in a .csv file containing under a template I created. The template will be as follows

Example Bug Reported

Bug reporting System			
Bug Name	Component	Tester	Risk
Character Movement	Player Movement in main Scene	Tomás	Severe

6.0.2 Change Request

When testing the product and modification needs to be made the developer must follow this template

Example Change Request

Change Request System				
Bug Name	Component	Tester	Manager	Modification
Character Movement	Player Movement in main Scene	Tomás	Ms Carter	Need to add new Script to fix errors

Approve Y/N :n Manager NAme :xxxx Date : xxx

Chapter 7

Features to Be Tested

7.0.1 The Features

Some Features to Be Tested

1. Menu Systems
    - The game has a number of in game and pre game menus which must be tested indivually and tested on how they integrate at different moments in-game.
2. Character Movement
    - The Character Movement is a priority in the game and different combinations specified such as up down left and right must be tested as they are a core function to the gameplay.
3. Enemies
    - The Enemies are a massive component in the game and their attributes such a health and must be tested throughly .E.g If an emeny is killed they must leave the scene or if hit they must lose -1 health etc.
4. Save Game
    - The Save Game feature allows the user to resume game at another apprioate time.We must invesigate the components that hold the counters and search for bugs to prevent users from explotiing the feature to skip levels etc.

Chapter 8

Features Not to Be Tested

8.0.1 Features

Some Features Not to Be Tested

1. Performance Testing

    - In Modern computer systems we have an excess of power e.g in cores, RAM (Random Access Memory), volatile memory etc.2D games are not demanding to run on modern systems, and 2d games have been popular on computers since the 1960s.

2. Stress Testing

    - Stress testing is a type of software testing that verifies the stability and reliability of the system. It is evident the game is very light in terms of computationally power and trying to play the game for hours would be extremely unproductive.

3. Multiple Mobile Platforms

    - We will not test this game for multiple smartphone platforms e.g Apple (Ios) and Android. Smartphones have near-identical architecture and we can presume with high levels of confidence that this game will show the same behavior independent of the Operating System.

4. Usability testing

    - Testing for user-friendliness is subjective and will depend on the targeted end-user or customer. Nearly all computer/mobile games use the most common movement systems e.g WASD and Gestures. We are highly confident the user experience is traditional and usability testing would be redundant.

Chapter 9

Resources-Roles and Responsibilities

9.0.1 Roles

Here are the The Resources and Roles - Responsibilities

1. Testing Group

    - The testing group consisted of the staff members 1.John Smith 2.Laine Louhglin 3.Tyler Durden 4.Grace Kelly 5.Mia Wallace 6.Jackie Brown and 7.Tomas Omalley .The main roles of the testing groups was the create test cases e.g User acceptance test and study the behaviour of the processes.Each member of the testing team was delivered a role and its their repsonsible to test and report back to their management whether there is a success or a failure.

2. Report Group

    - The Report group consisted of the staff members 1.Jack Daniels and 2. John Harris Each week we would take part in an online meeting via Microsoft teams where the report Group would deliever a report with an overview of productivty , bugs tested , failed test cases and new bug discoveries.Each week the developers will hand over an individual report and the report group would prepare , manage and design report merging all the developers feedback into a single report.

3. Team Management

    - The manager of the team for the product was June Carter.We all tested the product in an agile Enivroment where all developer would test in short iterations.June acted as the Scrum master and her role was to manage developers and manage the process of how information is exchanged.

Chapter 10

Schedules

10.0.1 Schedule

Scheduling in project management is the listing of activities, deliverables, and milestones within a project. A schedule also usually includes the planned start and finish date, duration, and resources assigned to each activity. Effective project scheduling is a critical component of successful time management.

Scheduling was a very difficult area to manage when testing their product. We had to create an incremental approach starting with the bare bones of the Product and working from the ground up e.g testing the transition from Start menu to the in-game. Underneath are Schedules created during testing.

Test Plan

The overall testPlan was successful and we did not have to change many of the milestones.THe test plan set a pace for the team to work at and worked very well in our agile development approach.The product and test plan was a success

Test Cases

The overall test cases created by developers were solid and successful.Near all components were tested and only a few minor bugs were not resorted.

Test Incident Reports

The test incident Reports were created by nearly all developers when an issue was incountered.There was a template .csv sent to all users to doocument their bugs to.This report summarised bugs and was informative and beneficial to the team.

Test Summary Reports

The Test Summary Reports were created by the Managers and dispersed to the whole team at the end of sprints or milestones.We found the informative summary report straight to the point and informed the entire team on their strengths and their weaknesses.

Chapter 11

Risks and Assumptions

11.0.1 Risks

The Risks/Assumptions

Contingency plan for unexpected Bugs

- A risk is the chance, high or low, that any hazard will actually cause somebody harm.
- An assumption is a thing that is accepted as true or as certain to happen, without proof.
- Through testing it was found during intergation that the user may be able to get unlimited health if the user hits the enenie while in an invincible mode.This bug has delayed our team and we must re schedule the unit testing for another week until this severe bug is fixed.
- I believe ya the team will be able to finish up by the scheduled beta testing period which is in month 4 we have set a contingency plan where developers can help eachother when they have completed their test cases.
- From the positive test cases create and monitored we can assume the game is near ready to be deployed to the public.
- A large risk for us would be to not collobarate together to effectively test all areas e.g to test on android instead of both IOS and Android.

Chapter 12

Tools

12.0.1 The Tools

A program used for software development or system maintenance. Virtually any program or utility that helps programmers or users develop applications or maintain their computers can be called a tool. The Tools used during testing are as follows :

1. Bugzilla - Bug Report Tool
    - Bugzilla is a web-based general-purpose bug tracking system and testing tool originally developed and used by the Mozilla project , and licensed under the Mozilla Public License.
    - This bug reporting has been a go to for our company and we will not be changing as its an industry standard.
2. Selenium (software) - Automated Testing
    - Selenium is a portable framework for testing web applications. Selenium provides a playback tool for authoring functional tests without the need to learn a test scripting language.
    - We found our development team was very pleased with the product. The documentation online was very clear and can be also run in browsers which is a bonus for our web-developers.
3. LaTeX - Documentation System
    - LaTeX is a document preparation system. When writing, the writer uses plain text as opposed to the formatted text found in "What You See Is What You Get" word processors like Microsoft Word, LibreOffice Writer, and Apple Pages.
    - Each developer was required to upload a .pdf outlying their discoveries throughout the week and this open-source Tex editor was suitable. It is cross-platform (Mac, Windows, Linux) so our developers on other platforms had no issues installing. We may consider Microsoft Word in the future due to its steep learning curve.
4. Microsoft Teams - Online Management Platform
    - Microsoft Teams is a persistent chat-based collaboration platform complete with document sharing, online meetings, and many more extremely useful features for business communications. Having an excellent team space is key to being able to make creative decisions and communicate with one another.

- The whole team must be connected 24/7 and we found this platform suited our criteria 1. Responsive 2.Private Chatting 3.Instant Messaging. We will use this platform again in the future to monitor our workplace/employee's productivity.

5. GitHub - Version Control

   - GitHub is a Git repository hosting service, but it adds many of its own features. While Git is a command line tool, GitHub provides a Web-based graphical interface. It also provides access control and several collaboration features, such as a wikis and basic task management tools for every project
   - The whole team must use GitHub in case the newest installment in testing breaks the entire product.GitHub allowed us to collaborate on our code and work in parallel.