# Software Testing Plan

## Introduction

This software test plan will utilize an Agile testing methodology which forms part of the overall Agile SDLC model used for developing the project. Given the simplistic nature of the project, static testing may not be necessary. The program may also be tested manually; as there is no need for numerous, repetitive actions to test the relatively few required functions. The main test metrics used will be error count and error frequency.

The requirements of this project suggest the employment of a Behavioral Testing Strategy which involves the use of Black Box tests on the application. Testing will focus primarily on key functionality from the user’s perspective. Project scope does not specify requirements pertaining to user experience.

## Scope

### In Scope

* User authentication
* Display database
* Search database
  + With any combination of parameters
* Top ten movies
* Operation on three different sized devices
  + PC
  + Tablet/Laptop
  + Phone

### Out of Scope

* User experience
* Login security
* User account recovery
* Thumbnails with theatrical posters/cover art
* User reviews

## Quality Objective

The quality objectives will connect measures of software quality to tangible and achievable goals for the development team. It is through these objectives that software quality assurance will be performed.

### Objectives

1. Ensure functional requirements are met
2. Test all relevant code pathways
3. Ensure multi-device compatibility
4. Ensure program is free of errors that:
   1. *Break vital functionality*
   2. *Cause crashes/ freezing*
   3. *Significantly affect reasonable usability*
   4. *Threaten data integrity*

### CITE MS QA Standards

The following are the tasks and objectives of CITE Managed Services Quality Assurance System which will guide the quality assurance standards of the project, as detailed on the CITEMS Website:

* Elaboration and implementation of procedures and regulations for software development process based on industry standards and best practices;
* Product lifecycle monitoring to ensure compliance with established processes and guidelines
* Product quality verification and validation to ensure that it complies with clients’ business needs and expectations;
* Establishment of an effective collaboration between all project team members.

## Roles and Responsibilities

### Scrum Master

The Scrum Master will set forth the testing plan. They will delegate and share testing duties among the team. They will determine which code pathways/scenarios are to be tested, and their priority. They will also outline the test methodology, including test levels.

### Configuration Manager

The Configuration Manager will be responsible for approving changes committed, and monitoring the effects of these changes. They will identify any issues caused by fixes/additions to the project and determine when they occurred, and who was responsible. They will also use this information to assist in the reproduction of successful configurations.

### Developer

Developer will produce a Bug Report detailing any new bugs with each iteration, and assist the team in determining the priority of each bug based on Severity, Frequency, and Risk.

## Test Methodology

### Overview

The Agile test methodology will be used for this project. This methodology is part of the Agile SDLC model used for the project’s development. Each software iteration is built and then tested thoroughly before moving on to the next iteration, ensuring no recurring issues will carry over between development cycles.

### Test Levels

Given the relatively simple nature of the project, some of these testing levels may be executed concurrently to eliminate redundant efforts e.g. Unit + Integration, System + Acceptance.

#### Unit Testing

Each unit of the program is tested independently to ensure proper function in accordance with project requirements.

#### Integration Testing

Units are tested when integrated with each other. Interactions between units are scrutinized and verified, and errors in these interactions are identified.

#### System Testing

The complete program is tested with all units fully integrated. Determines program’s ability to function as a whole.

#### Acceptance Testing

Determines whether a program is acceptable for the end user. Finished program is tested against business requirements.

### Suspension Criteria and Resumption Requirements

#### Suspension Criteria

Testing will be suspended if:

1. Any vital functionality is broken
2. An error directly prevents further testing
3. Error(s) considered too numerous or too frequent
4. An error is discovered that renders further testing redundant

#### Resumption Requirements

Testing will resume when:

1. Vital functions repaired
2. Errors preventing testing are resolved
3. Errors are reduced to an acceptable level
4. Testing is no longer obstructed by dependencies

### Test Completeness

The criteria to determine that testing is complete, are as follows:

1. All test cases carried out successfully
2. All bugs fixed
   1. OR mitigated, with plans for full resolution in next iteration

### Test Deliverables

|  |  |  |  |
| --- | --- | --- | --- |
| **SPRINT ONE** | **SPRINT TWO** | **SPRINT THREE** | **HANDOVER** |
| Test Strategy | Test Cases | Test Cases | Client Signoff |
| Test Cases | Bug Report | Bug Report |  |
| Bug Report | Bug Triage | Bug Triage |  |
| Bug Triage |  |  |  |

Table 3- Timeline of Test Deliverables

## Resource and Environment Needs

### Testing Tools

1. Linux Server
2. Movie Database (Populated)

### Test Environment

PC REQUIREMENTS

* **Hardware**
  + Intel Pentium 4 CPU or later
  + 4GB RAM
  + Networking capabilities (LAN, Wi-Fi)
* **Software**
  + OS: Windows 7 or later
  + Any leading web browser

MOBILE DEVICE REQUIREMENTS

* Android 5.0 Lollipop or later

## Glossary

|  |  |
| --- | --- |
| **TERM/ACRONYM** | **DEFINITION** |
| Bug | An error within a computer program that causes undesired or unexpected actions |
| GitHub | Online service providing version control for software development |
| LAN | Local Area Network |
| OS | Operating System; software which gives computers a user friendly interface, and acts as a foundation for running programs |
| RAM | Random Access Memory. Programs are loaded into this short term storage, allowing fast access to any part of the program |
| Wi-Fi | Wireless Local Area Network.  The name Wi-Fi has no further meaning, and was never officially a shortened form of "Wireless Fidelity” |