**Test Plan-Test Specification**

**Version 1.0**

**Project Management App**

**Team A**

**CSC-355**

**Spring 2016**



02/24/2016

Author: Jennifer Li – System Tester

Submitted To: Dr. Joo Tan

**TABLE OF CONTENTS**

Revision History…………………………………………………………………………………..ii

1. Introduction……..….....…………………………….…...…………………………….……....1
2. Features to Be Tested…………………………………………………………………………..
   1. Intermediate Release……………………………………………………………………
   2. Final Release……………………………………………………………………………
   3. Scenarios……………………………………………………………………………….
3. Test Approach ………………………………………………………………………………….
   1. Unit Testing…………………………………………………………………………….
   2. System Testing…………………………………………………………………………
4. Software Defects States………………………………………………………………………...
5. Software Defects Severity………………………………………………………………………
6. System Test Configuration……………………………………………………………………..
7. Test Responsibilities……………………………………………………………………………
8. Test Entrance Criteria…………………………………………………………………………..
   1. Developers……………………………………………………………………………..
   2. Testers…………………………………………………………………………………..
9. Test Schedule…………………………………………………………………………………...
10. Risks and Contingencies……………………………………………………………………

**REVISION HISTORY**

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Author | Description | Date |
| 1.0 | Jennifer Li | I created the first draft. | 1/31/2016 |
| 1.1 |  |  |  |

**1.0 INTRODUCTION**

This document presents the test and specification plan on how and what, the Project Management App will be tested. This document will outline, test procedures formulated to validate the Project Management App. This document will describe the test strategy, the test system, and provide an estimation of testing effort.

**2.0 FEATURES TO BE TESTED**

The following features will be tested to ensure that The Knowledge Box satisfies all functionality as specified in the Business Requirements I and per the request of the client and approval of the development team will be included and tested after the intermediate release.

* RS 3.2.1.1 - Admin login
* RS 3.2.1.2 - Tech login
* RS 3.2.2.1 - Tech logout
* RS 3.2.2.2 - Admin logout
* RS 3.2.3 - Admin add an account
* RS 3.2.4.1 - Tech view all active calls
* RS 3.2.4.2 - Admin view all active calls
* RS 3.2.5.1 - Tech view all inactive calls
* RS 3.2.5.2 - Admin view all inactive calls
* RS 3.2.6.1 - Tech view an individual call
* RS 3.2.6.2 - Admin view an individual call
* RS 3.2.7 - Admin back-up database
* RS 3.2.8 - Active and inactive calls sorted
* RS 3.2.9 - Admin add a new call
* RS 3.2.10 - Admin edit an existing call
* RS 3.2.11.1 – Tech open a call
* RS 3.2.11.2 - Admin open a call
* RS 3.2.12.1 - Tech close a call
* RS 3.2.12.2 - Admin close a call
* RS 3.2.13 - Admin approve a call
* RS 3.2.14 - Help documents accessible

**2.1 Intermediate Release**

Stuff

**2.2 Final Release**

Stuff

**2.3 Scenarios**

The system will also be tested using various real-to-life scenarios. This will verify workflow and coordination of functions:

* RS 3.2.16 – Complete and approve call scenario
* RS 3.2.17 – Delete active user scenario
* RS 3.2.18 – Add new admin scenario
* RS 3.2.19 – Admin sort scenario
* RS 3.2.20 – Add numerous new calls scenario
* RS 3.2.21 – Add new call with an error scenario

**3.0 TEST APPROACH**

The overall testing approach includes feature testing, regression testing, product installation testing, backup/restore testing, usability testing, and scenario testing. Each type of testing is described in more detail in the following sub-sections.

**3.1 Unit Testing**

Stuff

**3.2 System Testing**

Stuff

1. **SOFTWARE DEFECTS STATES**

For the purpose of tracking the status of all bugs, bug states are defined. The six bug states are defined as follows:

* **New** – a defect that has not been acknowledge or addressed by developers
* **Defer** – a defect that will be postponed and addressed at a later date
* **Open** – a defect that is currently being worked on by the Development Team
* **Fixed** – a defect that has been fixed by the Development Team and is ready for re-testing
* **Closed** – the Test Team has tested the fix using the same test configuration and test procedure that was used to find the defect originally
* **Trash** – a defect that is not valid or a defect that will not be addressed

1. **SOFTWARE DEFECTS SEVERITY**

Stuff

1. **SYSTEM TEST CONFIGURATION**

For the test process, we will use the KU Computer Lab in Lytle Hall. The Knowledge Box is designed to run on Windows XP. It was created using Visual Basic 2005. The test product will be in the form of an executable (.exe) program which the testers will download from the Yahoo group. The use of one shared drive for all users to connect is being coordinated.

1. **TEST RESPONSIBILITIES**

Both the Development and Test Teams have responsibilities for ensuring the quality of the intended product.

*Development Team responsibilities include:*

* Unit Test Features as they are developed
* Fix 100% of major bugs and at least 90% of moderate bugs. Minor bugs will be fixed if time allows.
* Perform Integration Testing on features before they are packaged in a build for the test team
* Prepare the application for delivery to the test team according to the WBS
* Fix bugs submitted via MRs

*Test Team responsibilities include:*

* Run planned tests and create MR reports for developers
* Prepare a MR Summary Report on bugs found
* Prepare a Test Status Report & Test Metrics at the end of the testing phase
* Prepare Test Logs and a Bug Status Report

1. **TEST ENTRANCE CRITERIA**

Before the Test Team will accept the system from the Development Team the following Entry Requirements must be met.

*Entrance Criteria include:*

* + Development unit and integration testing completed
  + System must install properly without undesirable side effects (see Section 8.0)
  + 100% of Major bugs found in development testing must be fixed

**8.1 Developers**

Stuff

**8.2 Testers**

Stuff

1. **TEST SCHEDULE**

It is expected that all unit and integration testing done by the development team will occur prior to the start of Test Cycle #1 and that all Major bugs found in that testing will be fixed prior to the start of system testing. Table 15.1 provides brief starting and ending dates for tasks during each cycle.

**Table 15.1: Task Start & End Dates**

|  |  |  |
| --- | --- | --- |
| **Task** | **Start Date** | **End Date** |
| Test Cycle #1 | 4/3 | 4/13 |
| Smoke Test | 4/3 | 4/3 |
| Summarize MR Reports | 4/13 | 4/13 |
| Test Cycle #2 | 4/17 | 4/20 |
| Smoke Test | 4/17 | 4/17 |
| Summarize MR Reports | 4/20 | 4/26 |
| Summarize Test Results | 4/23 | 4/26 |

1. **RISK AND CONTINGENCIES**

The following Table 16.1 outlines possible risks and contingency plans affecting the testing of the Knowledge Box system:

**Table 16.1: Risks and Contingencies**

|  |  |  |  |
| --- | --- | --- | --- |
| **Risk** | **Likelihood** | **Impact** | **Mitigation** |
| Hardware/Software required are not available at testing time. | 25% | Major | The testing will be done using the Lytle Hall computer room. If this room is not available, other backup labs should be found and reserved. |
| Requirements change during test development. | 25% | Moderate | Requirements should be constantly reviewed and test cases need to be revised if requirements change |
| Test cases are not ready. | 25% | Moderate | Testing of available test cases should be performed while other test cases are finalized |
| User interface changes during test development or at start of test. | 25% | Moderate | If interface changes, development team needs to inform test team of any changes. |
| Inclement weather forces a delay in schedule | 25% | Moderate | If people are not available due to weather, others from the team may be asked to help in the testing phase. If the university closes, testers may need to download the system at home to perform testing. |
| People scheduled to test are not available at test time. | 25% | Minor | If people are not available, then we should consider asking another team member to assist in testing. |
| Personnel do not give 100% effort to the project | 25% | Moderate | If members are not giving 100% to the project, this should be brought to the attention of Upper Management and the Lead Tester. Upper Management and test Lead will address the issue and take necessary action. |
| Test cases in Test Specification do not correspond to System Interface delivered | 50% | Moderate | Test Cases may need to be modified to the System Interface that is delivered. |
| Test documents are inconsistent | 25% | Minor | Documents will be review and must be updated according to the schedule outlined in the WBS [6]. |
| Test documents are not completed on time | 25% | Minor | The documents must be completed within 24 hours and all team members notified. |