

**CAPSTONE PROJECT 1**

**ROOMY SYSTEM MANAGEMENT**

**TEST PLAN FOR SPRINT 1 DOCUMENT**

Version: 1.1

Date: Sep 8, 2018

**MENTOR**: MSc. Phan Van Son

**PROJECT TEAM**: BLUE TEAM

**TEAM MEMBER** : Huynh Quoc Nhat

Ho Trung Anh

Le Hoang Quoc

Huynh Vu Ha Lan

Huynh Thi Thanh Van

**INTERNATIONAL SCHOOL**

**Document Approvals:** The following signatures are required for approval of this document.

|  |  |  |
| --- | --- | --- |
| MSc. Vo Van Son  *Mentor* |  | Date: |
| Huynh Vu Ha Lan  *Scrum Master* |  | Date: |

**Revision history**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Date** | **Version** | **A, M, D (\*)** | **Description** | **Author** |
| Sep 8, 2018 | 1.0 | A | Test plan for sprint 1 | Blue  Team |
| Sep 11, 2018 | 1.1 | M | Test plan for sprint 1- Update | Blue  Team |

(\*) **A**-added, **M**-modified, **D**-deleted

**TABLE OF CONTENT**

[Definitions 5](#_Toc466235365)

[1 Introduction 6](#_Toc466235366)

[1.1 Purpose 6](#_Toc466235367)

[1.2 Scope 7](#_Toc466235368)

[1.3 System overview 7](#_Toc466235369)

[2 Test Requirements 8](#_Toc466235370)

[3 Testing Strategy 9](#_Toc466235371)

[3.1 Types of testing 9](#_Toc466235372)

[3.2 List of use cases 9](#_Toc466235373)

[3.3 Test detail 10](#_Toc466235374)

[3.3.1 Functional testing 10](#_Toc466235375)

[3.3.2 Interface testing 10](#_Toc466235376)

[3.4 Test method 10](#_Toc466235377)

[3.4.1 Unit test 10](#_Toc466235378)

[3.4.2 Integration test 10](#_Toc466235379)

[3.4.3 System test 11](#_Toc466235380)

[4 For each testing 11](#_Toc466235381)

[5 Resource 12](#_Toc466235382)

[5.1 Human 12](#_Toc466235383)

[6 Risk/ Contingency/ Mitigation plan 13](#_Toc466235384)

[7 Plan for testing 13](#_Toc466235385)

[8 Test result 14](#_Toc466235386)

**THE TABLE**

[Table 3.1: List of use cases 9](#_Toc355816232)

[Table 4.1: Each testing 12](#_Toc355816233)

[Table 5.1: Human resource 13](#_Toc355816234)

[Table 6.1: Risk/ Contingency/ Mitigation plan 13](#_Toc355816235)

[Table 7.1: Plan table 14](#_Toc355816236)

# Definitions

|  |  |
| --- | --- |
| **Definitions/Acronyms/Abbreviations** | **Description** |
| GUI | Graphical User Interface |
| UI | User Interface |

# Introduction

## Purpose

* Determine what information the project should be tested:
  + System Objectives.
  + System Context.
  + Functional Requirements.
  + Configuration.
* List the Test Requirements
  + System Objectives.
  + System Context.
    - The system provides an easy-to-use UI.
  + Functional Requirements.
* Provide methods, testing strategies should use
  + User Test.
  + Acceptance Test.
* Resources for Testing
  + 3 testers.
  + 3 laptops.
  + Excel.
* List the results, after the document was made:
* Test plan, test case document, report for each User test, acceptance test.
* Test plan document.
* Unit test.
* Acceptance test and performance test document.

## Scope

Describe the stages of testing and which kind of test would be used in this plan.

* Stages of testing:
* Test plan.
* Test case.
* Test unit.
* Test report.

## System overview



Figure 1.1: Context diagram of system

# Test Requirements

The list below shows the category (functional requirements, system requirements, and functional requirements outside) should be tested.

* List the Test Requirements
* System Objectives
* Build a system to help user manage their events.
* System context
* The system provides an easy-to-use UI.
* Functional requirements:
* Login.
* Manage accounts.
* Manage contacts.
* Manage Transaction History.
* Manage Cart.
* Manage Post.
* Logout.
* Configuration
* Stand-alone.
* Non Functional Requirement test:
* Security requirement.
* Usability.
* Compatibility.
* Extensible.
* Performance.

# Testing Strategy

Test Strategy presented the methods for testing software applications. At the testing requirements shall describe what should be tester while the testing strategy outline the ways used to test.

In this section, techniques and evaluation criteria are the main contents of interest.

## Types of testing

All test cases will be done manually by the tester.

## List of use cases

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Tool** | **Company** | **Version** |
| Management of testing activities | Excel | Microsoft | 2016 |
| Bugs tracking | Excel | Microsoft | 2016 |
| Function tests | By hand |  |  |
| Project management | Word,  Excel | Microsoft | 2016 |
| Database System  Management | MySQL | Sun |  |

Table 3.1: List of use cases

## Test detail

### Functional testing

Goal: Ensure proper target of test functionality, including navigation, data entry, processing, and retrieval.

### Interface testing

Goal: Ensure system have interface same as interface have on interface system design.

## Test method

### Unit test

* Purpose: A unit test is used to test the unit based on its logic and structure. Developer can use unit test when finish unit.
* Specification for the testing: use white box testing: Basic path.

### Integration test

* Purpose: use test when integration unit tests to test performance, function of system.
* Specification for the testing:
* Check the structure (structure): Black Box Test Similarly, Boundary test.
* Function tests (functional): Black Box Test Similarly, use test coverage.
* Check performance (performance): Check the operating system.

### System test

* Purpose: System testing is performed on the entire system in the context of a System Requirement Specification (SRS).
* Specification for the testing:
* Interface document.
* Database document.
* Architecture document.

# For each testing

|  |  |  |  |
| --- | --- | --- | --- |
| **Test type** | **Definition** | **Start criteria** | **Stop criteria** |
| Functional testing | Function testing of the target of test should focus on any requirement for test that can be traced directly to use cases or business functions and business rules. The goals of these tests are to verify proper data acceptance, processing, and retrieval, and the appropriate implementation of the business rules. | After any function is finished, can begin to test it. | * When all essential requirements implemented and all high-priority defects have been fixed. * When defects are found fewer than 5% total number of the first defects and it doesn’t make the system is failure. |
| Interface testing | User Interface (UI) testing verifies a user’s interaction with the software. The goal of UI testing is to ensure that the User Interface provides the user with the appropriate access and navigation through the functions of the target of test. In addition, UI testing ensures that the object within the UI function as expected and conform to corporate or industry standards. | After any function is finished, can begin to test it. | * When all essential requirements implemented and all high-priority defects have been fixed. * When defects are found fewer than 5% total number of the first defects and it doesn’t make the system is failure. |

Table 4.2: Each testing

# Resource

## Human

|  |  |  |
| --- | --- | --- |
| **Human** | **Minimum**  **Quantity**  **(full time)** | **Note** |
| Test Project Manager:   * Thanh Van | 1 | Create test plan.  Manage testing activities:   * Technical guide. * Manage reports. |
| Tester:   * Huynh Vu Ha Lan * Le Hoang Quoc * Huynh Thi Thanh Van | 2 | Perform the test:   * Conduct test cases. * Do acceptance test. * Write test results. |

Table 5.3: Human resource

# Risk/ Contingency/ Mitigation plan

|  |  |  |
| --- | --- | --- |
| **Risk** | **Describe** | **Mitigation** |
| Lack of requirements | Means not having sufficient  Requirement documents, testcases, execution of testcases. | Planning to prepare full document: test cases |
| Behind schedule | If the product to release is not yet completed tothe date mentioned then there exists a risk. | Creating test schedule, deliver clear tasks in testing phase. |
| Insufficient team | Means a company with small team whereall the work assigned to those team members and having lackof time to deliver the product. | Work breakdown structure is clearly, have to assign special for each member base on each role. Can increase work time if necessary. |

Table 6.4: Risk/ Contingency/ Mitigation plan

# Plan for testing

|  |  |  |  |
| --- | --- | --- | --- |
| **Task** | **Start date** | **End date** | **Responsibility** |
| Test plan | 08/09/2018 | 08/09/2018 | Le Hoang Quoc |
| Create test case | 22/09/2018 | 23/09/2018 | Le Hoang Quoc,  Huynh Vu Ha Lan,  Huynh Thi Thanh Van |
| Implement test | 24/09/2018 | 27/909/2018 | Huynh Quoc Nhat,  Le Hoang Quoc,  Huynh Vu Ha Lan,  Ho Trung Anh,  Huynh Thi Thanh Van |
| Report test and send to developer fix error/problem | 27/09/2018 | 30/09/2018 | Huynh Quoc Nhat,  Le Hoang Quoc,  Huynh Vu Ha Lan,  Ho Trung Anh,  Huynh Thi Thanh Van |
| Retest | 01/10/2018 | 02/10/2018 | Huynh Quoc Nhat,  Le Hoang Quoc,  Huynh Vu Ha Lan,  Ho Trung Anh,  Huynh Thi Thanh Van |
| Closing | 03/10/2018 | 03/10/2018 | Huynh Quoc Nhat,  Le Hoang Quoc,  Huynh Vu Ha Lan,  Ho Trung Anh,  Huynh Thi Thanh Van |

Table 7.5: Plan table

# Test result

Test results will be written in excel file.