

COMPANY RATING WEBSITE

Test Plan

*Report #3*

|  |  |  |
| --- | --- | --- |
| **COMPANY RATING WEBSITE – *KẾ SINH NHAI TEAM*** | | |
| **Group Members** | Trần Quốc Hưng | SE02970 |
| Nguyễn Sơn Lâm | SE03090 |
| Nguyễn Việt Hùng | SE03293 |
| Nguyễn Nhật Quang | SE03122 |
| Trần Đăng | SE03039 |
| **Supervisor** | Bùi Đình Chiến | |
| **Project code** | CRW | |

**- Hanoi, 20/06/2016 -**

SIGNATURE PAGE

AUTHORS: Nguyễn Sơn Lâm 01/07/2016

Team Member

Nguyễn Nhật Quang 01/07/2016

Team Member

REVIEWER: Trần Quốc Hưng 01/07/2016

Team Member

APPROVAL: Bùi Đình Chiến --/--/2016

Supervisor

Record of change

\*A – Added; M – Modified; D – Deleted

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Effective Date | Changed Item | A,M,D\* | Change Description | Reason for Change | Rev. Number | |
| 01/07/2016 | Test Plan | A | First version | Create Test Plan | | 1.0 |
|  |  |  |  |  | |  |
|  |  |  |  |  | |  |

Contents

[SIGNATURE PAGE 2](#_Toc455293775)

[Record of change 3](#_Toc455293776)

[1. INTRODUCTION 5](#_Toc455293777)

[1.1 Purpose 5](#_Toc455293778)

[1.2 Definitions and Acronyms 5](#_Toc455293779)

[1.3 References 5](#_Toc455293780)

[1.4 Background information 5](#_Toc455293781)

[1.5 Scope of testing 6](#_Toc455293782)

[1.6 Constraints 7](#_Toc455293783)

[1.7 Risk list 7](#_Toc455293784)

[1.8 Guarantee the quality model 8](#_Toc455293785)

[2. REQUIREMENTS FOR TEST 11](#_Toc455293786)

[2.1 Test item 11](#_Toc455293787)

[2.2 Acceptance test criteria 12](#_Toc455293788)

[2.3 Feature not to be tested 12](#_Toc455293789)

[3. TEST STRATEGY 13](#_Toc455293790)

[3.1 Test type 13](#_Toc455293791)

[Function testing 13](#_Toc455293792)

[User Interface testing 14](#_Toc455293793)

[Data and Database Integrity testing 15](#_Toc455293794)

[3.2 Test stages 15](#_Toc455293795)

[4. RESOURCES 16](#_Toc455293796)

[5. TEST ENVIRONMENT 17](#_Toc455293797)

[5.1 Hardware 17](#_Toc455293798)

[5.2 Software 17](#_Toc455293799)

[5.3 Infrastructure 17](#_Toc455293800)

[6. TEST MILESTONES 18](#_Toc455293801)

[7. DELIVERABLES 19](#_Toc455293802)

# INTRODUCTION

## Purpose

This is the comprehensive test plan of the CRW project. The purpose of the document describes scopes of test and activities which need to be taken during test process of project. It addresses the following items: Scopes of Testing, Requirements for Testing, Test Strategy, Test Resources, Test Environment, Test Milestones and Deliverables.

## Definitions and Acronyms

|  |  |  |
| --- | --- | --- |
| Acronym | Definition | Note |
| CRW | Company Rating Website | Project name |
| GUI | Graphical User Interface |  |
| IT | Integration Test |  |
| KLOC | 1000 line of code |  |
| PM | Project Manager |  |
| QA | Quality Assurance |  |
| SRS | Software Requirement Specification |  |
| ST | System Test |  |
| TC | Test Case |  |
| TL | Test Leader |  |
| TP | Test Plan |  |
| TR | Test Report |  |
| UT | Unit Test |  |

**Table 1:** Definition and Acronyms

## References

|  |  |  |
| --- | --- | --- |
| Document | Filename | Notes |
| Software Requirement Specification | CRW\_Software Requirement Specification\_v1.0\_EN |  |
| Project Plan | CRW\_Project Plan\_v1.0\_EN |  |
|  |  |  |

**Table 2:** References

## Background information

The target of testing is ensured all functions will be run correctly as SRS description. In addition, restrict maximum of defect during the user access in the application. To do this target, website will have to:

* Passed the stages of testing: Unit Testing, Component Testing, Integration Testing, System Testing, Acceptance Testing.
* Passed the types of testing: Function Testing, User Interface Testing , Data and Data Integrity Testing.
* Run normally in required devices/browsers.

## Scope of testing

CRW will be tested in 5 phases

**Phase 1: Unit testing**

* Unit testing will be done by developers.
* Developers use White Box Testing technique to do.
* When executing unit testing, if any bug is found, developers will have to log bug on “Defect Log Management” file and fix it until it is correct.

|  |  |
| --- | --- |
| Rule for filling test result | |
| Test result pass | Passed |
| Test result fail | Failed |
| Do not test | Untested |
| Cannot Test | N/A (Not available) |

**Phase 2: Integration testing**

* After finishing with component testing, integration testing will be performed by testers.
* Material will be integration test cases, high-level design and test tools.
* Integration test focuses on specific areas of use cases whereas all requirements are completed.
* Integration test should be performed to ensure all components incorporate well.
* When executing integration testing, if any bug is found, testers will have to log on “Defect Log Management” file and assign to developer to fix it and loop this process until it is correct.

|  |  |
| --- | --- |
| Rule for filling test result | |
| Test result pass | Passed |
| Test result fail | Failed |
| Do not test | Untested |
| Cannot Test | N/A (Not available) |

**Phase 3: System testing**

* After finishing integration testing and developers collected all functions and items, testers will perform system testing, which means testing the system as a whole.
* Material will be system test cases and SRS.
* When executing system testing, if any bug is found, developers will have to fix it and testers will verify as needed.
* System test will end only when all test cases are passed and no bug is found.

|  |  |
| --- | --- |
| Rule for filling test result | |
| Test result pass | Passed |
| Test result fail | Failed |
| Do not test | Untested |
| Cannot Test | N/A (Not available) |

**Phase 4: Acceptance testing**

* Based on customer/user requirement specification, system will be tested again to ensure that there is no longer any missing feature or mistake regarding requirement.
* When executing acceptance testing, if there is any problem, developers will have to fix/update it accordingly and tester will verify as needed.
* Acceptance testing will end only when the system met requirement specification.

## Constraints

* Deadline for testing can only be met when the development progress is on time.
* Test execution can only be performed when system passes Unit Test Inspection.
* At least one round of testing must be performed for requirements.
* Have multiple environments to be tested: Windows XP, Windows 8… and multiple browsers: Firefox 40 and Google Chrome 50…

## Risk list

* Dependant on the UI design and develop progress.
* Not enough time to write the required amount of test cases for test phases execution or re-test of fixed bug.
* Tester(s) could be ill during the testing phase.

## Guarantee the quality model

CRW Follows the V-Model process



**Figure 1**: V-Model

Testing progress is divided into 5 phases: Unit test, Component test, Integration test, System test and Acceptance test

* Unit test:
  + *Unit testing is used to verify a single minimal unit of source code. The purpose of unit testing is to isolate the smallest testable parts of DDL and verify that they function properly in isolation.*
  + Unit testing is the first level of testing and is performed prior to component testing
  + Unit testing will be done by developer.
* Component test:
  + *Component testing is used to validate a combined many minimal units of source code.*
  + *Component testing is performed after unit testing and before integration testing*
  + *Component testing will be done by tester*
* Integration test:
  + *Integration testing is a level of the software testing process where individual units or component are combined and tested as a group.*
  + *The purpose is to expose faults in the interaction between integrated units.*
  + *Integration testing is performed after component testing*
  + *Integration testing will be done by tester*
  + *There are two methods of doing integration testing: Bottom-up Integration testing and Top Down Integration testing:*

|  |  |
| --- | --- |
| No | Integration Testing Method |
| 1 | Bottom-up integration  This testing begins with unit testing, followed by tests of progressively higher-level combinations of units called modules. |
| 2 | Top- Down integration  This testing, the highest-level modules are tested first and progressively lower-level modules are tested after that |

**Table 3:** Integration test

* System test:
  + *System Testing is a level of the software testing process where a complete, integrated system is tested*
  + *The purpose is to evaluate the system’s compliance with the specified requirements*
  + *System testing is performed after integration testing*
  + *System testing will be done by tester*
* Acceptance test:
  + *Acceptance testing is performed after system testing*
  + *Acceptance testing will be performed by the test leader and team leader.*
  + *The acceptance test will be done for a period of 1 weeks after completion of the system test process*

# REQUIREMENTS FOR TEST

## Test item

|  |  |  |  |
| --- | --- | --- | --- |
| TC No. | Group of Fuctions | Function | Glossary |
| User | | | |
| TC001 | Review | View Review |  |
| TC002 | Search Review |  |
| TC003 | Company | View Company Information |  |
| TC004 | Search Company |  |
| TC005 | Comment | View Comment |  |
| Guest | | | |
| TC006 | Common | Register Account |  |
| TC007 | Review | Post Review (Anonymous) |  |
| Member & Administrator | | | |
| TC008 | Common | Login |  |
| TC009 | Logout |  |
| TC010 | View Notification |  |
| TC011 | Get Notification |  |
| TC012 | Account | View Account Information |  |
| TC013 | Update Account Information |  |
| TC014 | Change Password |  |
| TC015 | Forgot Password |  |
| TC016 | Comment | Add Comment |  |
| TC017 | Update Comment |  |
| TC018 | Delete Comment |  |
| TC019 | Profession | Suggest Profession |  |
| TC020 | Add Profession |  |
| TC021 | Company | Suggest Company |  |
| TC022 | Add Company |  |
| Member | | | |
| TC023 | Company | Manage Followed Company |  |
| TC024 | Follow Company |  |
| TC025 | Unfollow Company |  |
| TC026 | Review | Manage Review |  |
| TC027 | Post Review |  |
| TC028 | Update Review |  |
| TC029 | Delete Review |  |
| TC030 | Report Review |  |
| TC031 | Upvote Review |  |
| TC032 | Remove Review Upvote |  |
| TC033 | Bookmark | Manage Bookmark |  |
| TC034 | Add Bookmark |  |
| TC035 | Remove Bookmark |  |
| Administrator | | | |
| TC036 | Company | Manage Company |  |
| TC037 | Add Company |  |
| TC038 | Update Company |  |
| TC039 | Delete Company |  |
| TC040 | Review | Manage Report |  |
| TC041 | Revise Report |  |
| TC042 | Change Review Status |  |
| TC043 | Account | ManageAccount |  |
| TC044 | Search Account |  |
| TC045 | Change Account Status |  |
| TC046 | Warn User |  |
| TC047 | Profession | Manage Profession |  |
| TC048 | Search Profession |  |
| TC049 | Update Profession Name |  |
| TC050 | Delete Profession |  |

**Table 4:** Test item

## Acceptance test criteria

* Criteria for Development team’s Unit test, for Test team accepts to start testing:
  + *Number of TC/KLOC: 40TC/KLOC*
  + *Number defects/KLOC: 3-4 defects/KLOC*
  + *Statement coverage: 100%*
  + *Branch coverage: 100%*
  + *Path coverage: 100%*
* Criteria for Integration test:
  + *Number of TC/KLOC: 30 TC/KLOC*
  + *Number defects/KLOC: 2-3 defects/KLOC*
* Criteria for System test:
  + *Number of TC/KLOC: 20 TC/KLOC*
  + *Number defects/KLOC: 4-6 defects/KLOC*
* Criteria for Acceptance test:
  + *Number defects/KLOC: 1-2 defects/KLOC*

## Feature not to be tested

* The stablility of website when internet connection is lost.
* Over than 5000 concurrent user at any given point.

# TEST STRATEGY

## Test type

### Function testing

* Functional testing is a type of software testing whereby the system is tested against the functional requirements/specifications.
* Functions are tested by feeding them input and examining the output. Functional testing ensure that the requirements are properly satisfied by the website. This type of testing is not concerned with how processing occurs, but rather, with the results of processing.
* During functional testing, [Black Box Testing](http://softwaretestingfundamentals.com/black-box-testing/) technique is used in which the internal logic of the system being tested is not known to the tester.

|  |  |
| --- | --- |
| Test Objective | This type of test is to ensure proper target-of-test functionality, including user interaction and all function defined in specification document is to be implemented correctly. |
| Technique | Execute each use case, use-case flow or function, using valid and invalid data, to verify the following:   * The expected results occur when valid data is used. * The appropriate error or warning messages are displayed when invalid data is used. * Each business rule is properly applied. |
| Completion Criteria | * All planned tests have been executed. * All identified defects have been addressed and closed. |
| Special Considerations | Testing may be stopped when:   * Time runs out * A certain number of defects found * Test coverage > 97% * Stop when testing becomes unproductive |

**Table 5:** Function testing

### User Interface testing

* GUI testing is the process of ensuring proper functionality of the GUI for a given web and making sure it conforms to its written specifications.
* GUI testing evaluates design elements such as layout, colors, [fonts](http://whatis.techtarget.com/definition/font), font sizes, labels, text boxes, text formatting, captions, buttons, lists, icons, links, content and more.

|  |  |
| --- | --- |
| Test Objective | Verify the following:   * Navigation through the target-of-test properly reflects business functions and requirements, including window-to-window, field-to-field and ease of access methods (tab keys, mouse movements, accelerator keys) * Window objects and characteristics, such as menus, size, position, state and focus conform to standards. |
| Technique | Create or modify tests for each window to verify proper navigation and object states of each application window and objects. |
| Completion Criteria | Each window successfully verified to remain consistent with benchmark version or within acceptable standard. |
| Special Considerations | Not all properties for custom and third party objects can be accessed. |

**Table 6:** GUI testing

### Data and Database Integrity testing

The databases and the database processes should be tested as a subsystem within the Project. These subsystems should be tested without the target-of-test’s User Interface as the interface to the data.  Additional research into the Database Management System (DBMS) needs to be performed to identify the tools and techniques that may exist to support the testing identified below.

|  |  |
| --- | --- |
| Test Objective | Ensure database access methods and processes function properly and without data corruption. |
| Technique | * Invoke each database access method and process, seeding each with valid and invalid data or requests for data. * Inspect the database to ensure the data has been populated as intended, all database events occurred properly, or review the returned data to ensure that the correct data was retrieved for the correct reasons. |
| Completion Criteria | All database access methods and processes function as designed and without any data corruption. |
| Special Considerations | * Testing may require a DBMS development environment or drivers to enter or modify data directly in the databases. * Processes should be invoked manually. * Small or minimally sized databases (limited number of records) should be used to increase the visibility of any non-acceptable events. |

**Table 7:** Data and Data Integrity testing

## Test stages

Clarified state of the stage in which the test will be executed. Identified below are the stages in which common tests are executed

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Types of Test | Stages of Test | | | |
| **Unit** | **Integration** | **System** | **Acceptance** |
| Function Testing | X | X | X | X |
| User Interface Testing | X |  | X | X |

**Table 8:** Types and stages of test

# RESOURCES

## Human resource

|  |  |  |
| --- | --- | --- |
| Worker/Doer | Role | Specific Responsibilities/Comments |
| LamNS | Test Leader | Manage Test Resource and assign test tasks.  Create and review Test Plan.  Create and review Test Case.  Execute test.  Create and review Test Report |
| QuangNN | Tester | Create and review Test Case.  Execute test.  Create Test view points  Create and review Test Report |

**Table 9:** Human resource

# TEST ENVIRONMENT

## Hardware

|  |  |  |
| --- | --- | --- |
| Name | Purpose | Detail |
| Laptop HP | Device for create and execute test | Windows 10 Core i5 |
| Laptop Asus | Device for create and execute test | Windows 8 Core i7 |

**Table 10:** Hardware

## Software

|  |  |  |
| --- | --- | --- |
| Name | Purpose | Detail |
| Test Plan | Managing test | Microsoft Word 2013 |
| Test Case | Executing test | Microsoft Word 2013 |
| Test Report, Test Checklist | Tracking test | Microsoft Word 2013, Redmine |
| Chrome | Executing test | Version 50 |
| Firefox | Executing test | Version 40 |

**Table 11:** Software

## Infrastructure

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Purpose | Detail | Vendor/In-house | Version |
| CRW\_Defect Log Management\_v1.0\_EN | Tracking Bug during testing time | Microsoft Excel 2013, Redmine | FPT University | V1.0 |
| Test Effort | Effort for test execution | Microsoft Excel 2013 | FPT University | V1.0 |

**Table 12:** Infrastructure

# TEST MILESTONES

|  |  |  |  |
| --- | --- | --- | --- |
| Milestone Task | Effort (pd) | Start Date | End Date |
| Create Test Plan | 6 |  |  |
| Review & Update Test Plan | 2 |  |  |
| Create Component Test case | 12 |  |  |
| Review & Update Component Test Case | 2 |  |  |
| Create Integration Test case | 2 |  |  |
| Review & Update Integration Test Case | 2 |  |  |
| Create System Test case | 6 |  |  |
| Review & Update System Test Case | 2 |  |  |
| Execute Component test phase 1 | 1 |  |  |
| Execute Component test phase 2 | 1 |  |  |
| Execute Integration test phase 1 | 2 |  |  |
| Execute Integration test phase 2 | 2 |  |  |
| Execute System test phase 1 | 4 |  |  |
| Execute System test phase 2 | 4 |  |  |

**Table 13:** Test milestones

# DELIVERABLES

|  |  |  |  |
| --- | --- | --- | --- |
| No | Deliverables | Responsibilities | Delivered Date |
| **1** | Test Plan | Tester |  |
| **2** | Component Test case | Tester |  |
| **3** | Integration Test case | Tester |  |
| **4** | System Test case | Tester |  |
| **5** | Defect Log Management | All members |  |
| **6** | Test report | PM |  |

**Table 14:** Deliverables