|  |
| --- |
|  |

USEFUL JAPANSES DICTIONARY FOR VIETNAMESE

TEST PLAN

Project Code: UJD\_VN

Document Code: UJD\_VN\_Test Plan\_v1.0\_EN

**Ha Noi, 16/06/2014**

Record of change

\*A - Added M - Modified D - Deleted

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Effective Date | Changed Items | A\* M, D | Change Description | New Version |
| 16/06/2014 | Add new | A |  | v0.1 |
| 23/06/2014 | Update | M | Update comment of document review | v0.9 |
| 30/06/2014 | Update | M | Update comment of document review | v1.0 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

SIGNATURE PAGE

**ORIGINATOR:** Pham Thi Minh 16/06/2014

Test Leader (TL)

**REVIEWERS:** Le Dinh Nam 23/06/2014

Project Manager (PM)

**APPROVAL:** Nguyen Van Sang

Supervisor

TABLE OF CONTENTS

[1 INTRODUCTION 5](#_Toc391813674)

[1.1 Purpose 5](#_Toc391813675)

[1.2 Definitions, Acronyms, and Abbreviations 5](#_Toc391813676)

[1.3 References 6](#_Toc391813677)

[1.4 Background information 6](#_Toc391813678)

[1.5 Scope of testing 6](#_Toc391813679)

[1.6 Constraints 7](#_Toc391813680)

[1.7 Risk list 8](#_Toc391813681)

[1.8 Training needs 8](#_Toc391813682)

[2 Requirements for Test 10](#_Toc391813683)

[2.1 Test items 10](#_Toc391813684)

[2.2 Acceptance Test Criteria 10](#_Toc391813685)

[2.3 Feature not to be tested 10](#_Toc391813686)

[3 TEST STRATEGY 11](#_Toc391813687)

[3.1 Test types 11](#_Toc391813688)

[3.1.1 Function Testing 11](#_Toc391813689)

[3.1.2 User Interface Testing 11](#_Toc391813690)

[3.1.3 Regression Testing 11](#_Toc391813691)

[3.2 Test stages 12](#_Toc391813692)

[4 RESOURCE 13](#_Toc391813693)

[4.1 Human Resource 13](#_Toc391813694)

[4.2 Test management 13](#_Toc391813695)

[5 Test environment 14](#_Toc391813696)

[5.1 Hardware 14](#_Toc391813697)

[5.2 Software 14](#_Toc391813698)

[5.3 Infrastructure 14](#_Toc391813699)

[6 TEST MILESTONES 15](#_Toc391813700)

[7 DELIVERABLES 16](#_Toc391813701)

# 

# INTRODUCTION

## Purpose

The purpose of this document is show scope of testing, test items, test strategy, testing approach, resources, test environment, schedule of intended testing activities …

## Definitions, Acronyms, and Abbreviations

| Abbreviations | Description | Note |
| --- | --- | --- |
| UJD\_VN | Useful Japanese Dictionary for Vietnamese |  |
| TL | Test Leader |  |
| PM | Project Manager |  |
| QA | Quality Assurance |  |
| SRS | Software Requirement Specification |  |
| TC | Test Case |  |
| TP | Test Plan |  |
| ST | System Test |  |
| IT | Integration Test |  |
| UT | Unit Test |  |
| GUI | Graphic User Interface |  |
| TR | Test Report |  |
| KLOC | 1000 line of code |  |

## References

| Title/File name | Author | Version | Effective Date |
| --- | --- | --- | --- |
| SRS | UJD\_VN Team | v1.0 | 20/06/2014 |

## 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 Ting and Regression Testing
* Run normally in required devices/browsers.

## Scope of testing

UJD\_VN will be tested by 5 phases:

Phase 1: Unit testing

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

Rule for filling test result:

|  |  |
| --- | --- |
| Test result pass | Pass |
| Test result fail | Fail |
| Do not test | Untested |
| Cannot test | N/A (Not available) |

Phase 2: Component testing

* After finishing unit testing, component testing will be performed by testers.
* Material are unit test cases, low- level design
* Testers user Black Box Testing technique to do
* When executing component testing, if any bugs are found, testers have to log on “Defect log management” file and assign to developer fix it and redo this process until it is correct.

Rule for filling test result:

|  |  |
| --- | --- |
| Test result pass | Pass |
| Test result fail | Fail |
| Do not test | Untested |
| Cannot test | N/A (Not available) |

Phase 3: Integration testing

* After finishing component testing, integration testing will be performed by testers.
* Material are high- level design and test tools.
* Do test by flow of functions and items which have relation.
* When executing integration testing, if any bugs are found, testers have to log on “Defect log management” file and assign to developer fix it and redo this process until it is correct.

Rule for filling test result:

|  |  |
| --- | --- |
| Test result pass | Pass |
| Test result fail | Fail |
| Do not test | Untested |
| Cannot test | N/A (Not available) |

Phase 4: System testing

* After finishing integration testing and developers collect all functions and items, testers will be performed system testing, it means doing test whole system.
* If any bugs are found, developers have to fix and testers will verify them. System test is ended only when test cases are passed and no bug is found.

Rule for filling test result:

|  |  |
| --- | --- |
| Test result pass | Pass |
| Test result fail | Fail |
| Do not test | Untested |
| Cannot test | N/A (Not available) |

Phase 5: Acceptance testing

* Base on requirement specification, system is tested again, for ensure there is not lacking or mistake any requirement.
* If there is any problem, developers have to fix/update and tester will verify them.
* Acceptance testing is ended only when whole system met requirement specification.

## Constraints

* There are much environments that UJD\_VN should be tested. But the number of tester can’t cover all environments.
* Short time to do test and testers have not experience, so schedule can be missed or miss bugs.
* Not enough time to re-fix bug, write all test case

## Risk list

* Performance test: Cannot test the case which many users connect to website at the same time.

## Training needs

UJD\_VN project follows V-Model process:



**Figure 1: V-Model**

Testing progress is divided to 5 phases include: 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 UJD\_VN and verify that they function properly in isolation.
* Unit testing is the first level of testing and is perform prior to component testing
* Unit testing will be done by developer.
* Component test:
* Component testing is used to validate a single minimal unit 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 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 |

* 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 items

1. *Member functions*

* Search( sentences, conversations, grammar, video, specialized)
* Log in/ Log out
* Register
* Edit profile
* Forget password
* Contribute (content, opinion, Q&A)
* Doing test
* Training listening
* Listening conversation
* Reading document

1. *Admin functions*

* Manage account(member, admin)
* Manage database( vocabulary, grammar, reading, conversation, video, listening, testing)
* Manage contact content( content, opinion, Q&A)

## Acceptance Test Criteria

* Criteria for Unit test of Development team, for Test team accepts to start testing:
* Number of UTC/KLOC: 50 UTC/KLOC
* Number defects/KLOC: 4-6 defects/KLOC
* Statement coverage: 100%
* Branch coverage: 100%
* Path coverage: 100%

## Feature not to be tested

* The stable of website when do not connect internet.
* Many users connect to system at the same time.

# TEST STRATEGY

## Test types

### 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.
* Functional testing is normally performed during the levels of [System Testing](http://softwaretestingfundamentals.com/system-testing/)
* Typically, functional testing involves the following steps:
* Identify functions that the software is expected to perform.
* Create input data based on the function’s specifications.
* Determine the output based on the function’s specifications.
* Execute the [test case](http://softwaretestingfundamentals.com/test-case/).
* Compare the actual and expected outputs.

### 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.
* GUI are checked manually by testers in conformance with the requirements stated in requirements document.

### Regression Testing

* Regression testing is a type of software testing that intends to ensure that changes (enhancements or defect fixes) to the software have not adversely affected it.
* During regression testing, new test cases are not created but previously created test cases are re-executed.
* Regression testing is so important because of the following reasons:
* Minimize the gaps in testing when website with changes made has to be tested.
* Testing the new changes to verify that the change made did not affect any other area of the application.
* Mitigates Risks when regression testing is performed.
* Test coverage is increased without compromising timelines.

## Test stages

Clearly state the stage in which the test will be executed. Identified below are the stages in which common test are executed

| Type of Tests | Stage of Test | | | |
| --- | --- | --- | --- | --- |
| Unit | Component | Integration | System |
| Function Testing | X | X | X | X |
| User Interface Testing |  | X | X |  |
| Regression Testing | X | X |  |  |

# 

# RESOURCE

## Human Resource

|  |  |  |  |
| --- | --- | --- | --- |
| Worker/Doer | Role | Specific Responsibilities/Comments | Location |
| MinhPT | Test Leader | * Manage Test Resource and assign test tasks. * Create Test Plan. * Create and Self Review Test Case. * Execute test. * Review Test Report | FPT, Vietnam |
| TuanNN | Tester | * Create and Self Review Test Case. * Execute test. * Collect data test. * Create Test Report. | FPT, Vietnam |

## Test management

* N/A

# Test environment

## Hardware

|  |  |  |
| --- | --- | --- |
| Name | Purpose | Detail |
| Laptop Asus | Device for executing test | Window 7 Ultimate Core i3 |
| Laptop Vaio | Device for executing test | Window 7 Ultimate Core i3 |

## Software

|  |  |  |
| --- | --- | --- |
| Name | Purpose | Detail |
| Test Plan | Managing test | Microsoft Word 2013, 2010 |
| Test case | Executing test | Microsoft Excel 2013, 2010 |
| Test report, Test checklist | Tracking test | Microsoft Excel 2013, 2010 |
| Chrome, CocCoc | Executing test | Chrome 35.0, CocCoc 35.0 |

## Infrastructure

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Purpose | Detail | Vendor/In-house | Version |
| Defect Log Management | Tracking bug during testing time | Microsoft Excel 2013, 2010 | FPT-University | v1.0 |
| Test Effort | Effort execute test | Microsoft Excel 2013, 2010 | FPT-University | v1.0 |

# TEST MILESTONES

|  |  |  |  |
| --- | --- | --- | --- |
| Milestone Task | Effort (pd) | Start Date | End Date |
| Create Test Plan | 6 | 16/06/2014 | 22/06/2014 |
| Review & update TP | 1 | 23/06/2014 | 23/06/2014 |
| Create Unit Test case (Component test) | 12 | 23/06/2014 | 28/06/2014 |
| Review & update UTC | 1 | 29/0762014 | 29/06/2014 |
| Create Integration Test case | 4 | 03/07/2014 | 04/07/2014 |
| Review & Update Integration TC | 1 | 04/07/2014 | 04/07/2014 |
| Create System Test case | 2 | 07/07/2014 | 07/07/2014 |
| Review & Update System TC | 1 | 07/07/2014 | 07/07/2014 |
| Create Test Checklist | 2 | 08/07/2014 | 08/07/2014 |
| Execute Integration test phase 1 | 4 | 15/07/2014 | 16/07/2014 |
| Execute Integration test phase 2 | 4 | 22/07/2014 | 23/07/2014 |
| Execute System test phase 1 | 2 | 17/07/2014 | 17/07/2014 |
| Execute System test phase 2 | 12 | 24/07/2014 | 29/07/2014 |

# 

# DELIVERABLES

| No | Deliverables | Language | Delivered Date |
| --- | --- | --- | --- |
|  | Test Plan | English | 23/06/2014 |
|  | Unit Test cases | English | 30/06/2014 |
|  | Integration Test Cases | English | 05/07/2014 |
|  | System Test cases | English | 08/07/2014 |
|  | Defect log management | English | 07/07/2014 |
|  | Test reports | English | 30/07/2014 |