****

**FPT UNIVERSITY**

**Taxi Caller Application on Windows Phone**

**F\_Taxi**

**Report #5 – Software Test Documentation**

|  |  |
| --- | --- |
| F\_Taxi | |
| Group Member | SE02705 - Phạm Gia Hữu |
| SE02900 - Tạ Thiên Hưởng |
| SE02268 - Phạm Ngọc Hoàn |
| SE02314 - Nguyễn Văn Lập |
| Supervisor | Nguyễn Văn Sang |
| Project Code | F\_Taxi |

HaNoi, 11/2015

Table of Contents

[1 INTRODUCTION 3](#_Toc364952626)

[1.1 Purposes 3](#_Toc364952627)

[1.2 System Overview 3](#_Toc364952628)

[2 TEST PLAN 3](#_Toc364952629)

[2.1 Scope of Testing 3](#_Toc364952630)

[2.2 Requirements for Testing 3](#_Toc364952631)

[2.2.1 Test Items 3](#_Toc364952632)

[2.2.2 Acceptance Test Criteria 3](#_Toc364952633)

[2.2.3 Testing Risks 3](#_Toc364952634)

[2.3 Test Strategies 3](#_Toc364952635)

[2.3.1 Test Policies 3](#_Toc364952636)

[2.3.2 Test Model 3](#_Toc364952637)

[2.3.3 Types of Testing 3](#_Toc364952638)

[2.3.4 Test Stages 3](#_Toc364952639)

[2.3.5 Tools/Environments 3](#_Toc364952640)

[2.3.6 Resources 4](#_Toc364952641)

[3 TEST CASES 4](#_Toc364952642)

[3.1 <UC 001 - Name> 4](#_Toc364952643)

[3.2 <UC 002 - Name> 4](#_Toc364952644)

[4 CHECKLISTS 4](#_Toc364952645)

[5 TEST LOGS 4](#_Toc364952646)

[5.1 Defect Logs 4](#_Toc364952647)

[5.2 Test Reports 4](#_Toc364952648)

# INTRODUCTION

## Purposes

The target of the test suit is to provide adequate coverage metrics, requirements validation and system quality data such that sufficient data is provided for those making the decision to release. The testing program is used to find errors in the application on Windows Phone to ensure that after fixation of these errors, the end product is defect free. It identifies when the work has been completed so that our project can be closed successfully.

## System Overview

The main objective of the project is creating application for window phone user, help user using app for take a taxi quickly and taxi driver pick up a customer easy.

# TEST PLAN

## Scope of Testing

## Four stages of testing:

* Unit testing: The purpose is to verify the internal logic code by testing every possible branch within the function, also known as test coverage. Unit test will be done by the developers and will be approved by the implementing team leader to ensure that the building blocks of the system uses work independently of each other and the specific function might have multiple tests to catch corner cases in the code.
* Integration testing: The separate module will be tested together to expose faults in the interfaces and in the interaction between integrated components. Integration testing will be done by tester.
* System testing: Compare the system specifications against the actual system. System testing checks if the integrated product meets the specified requirements.
* Acceptance testing: Acceptance testing will be performed by some students at FPT University. The acceptance testing will be done for a period of 1 week after completion of System/ Integration test process. Program will enter into Acceptance testing after all critical and major defects have been corrected. Prior to final completion of acceptance testing all open critical and major defects must be corrected.

## Requirements for Testing

### Test Items

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Use Case ID** | **Actor** | **Group of function** | **Name** | **Number of Testcase** | **Note** |
| UC-01 | Guest |  | Register |  |  |
| UC-02 | Rider | Authenticate | Login |  |  |
| UC-03 | Logout |  |  |
| UC-04 | Forget Password |  |  |
| UC-05 | Manage Profile | View Profile |  |  |
| UC-06 | Update Profile |  |  |
| UC-07 | Change Password |  |  |
| UC-08 | Manage Favorite | List Favorite Driver |  |  |
| UC-09 | Add Favorite Driver |  |  |
| UC-10 | Delete Favorite Driver |  |  |
| UC-11 | Call Favorite Driver |  |  |
| UC-12 |  | Find Lost Asset |  |  |
| UC-13 | Send Application Feedback |  |  |
| UC-14 | Call Taxi Center |  |  |
| UC-15 | Interact with Trip | Create Trip |  |  |
| UC-16 | Cancel Trip |  |  |
| UC-17 | List Complete Trip |  |  |
| UC-18 |  | View Bill Detail |  |  |
| UC-19 |  | Rate Driver |  |  |

### Acceptance Test Criteria

|  |  |  |
| --- | --- | --- |
| **No** | **Test Stages** | **Qualified ratios** |
| **1** | **Unit test** | To pass this stage, all unit test cases must be tested and passed 100% .All defects shouldbe fixed and re-tested |
| **2** | **Integration test** | To pass this stage, all test cases must be tested and passed 100%. All defects should be fixed and re-tested |
| **3** | **System test** | To pass this stage, all test cases must be tested and passed 100%. All defects should be fixed and re-tested |
| **4** | **Acceptance test** | Acceptance Test will be conducted and approved by FPT students |

### Testing Risks

|  |  |  |
| --- | --- | --- |
| **Risk** | **Mitigation** | **Contingencies** |
| PC or Windows Phone Mobile have problem during test (ex: error operation system), not enough device | Find a new PC or Windows Phone Mobile | Borrow and buy more device |
| The server has problem during test | Call center |  |
| Member of test team may be sick in test phase | + Discussing  + Committing  +Assigning tasks  appropriately  + Team building | + Persuading  + Re-organizing  + Reviewing and re-planning the whole project |

## Test Strategies

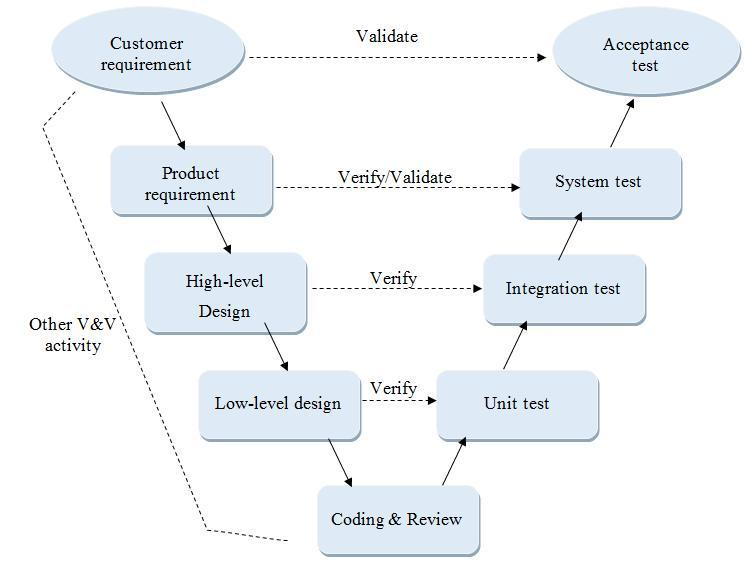
### Test Policies

### Test Model

We choose V-model for this project. Just like waterfall model, the V-model is a sequential path of execution of process. Each phase must be completed before the next phase begins. Testing of the website is planned in parallel with a corresponding phase of development.

Some advantages of V-model:

* Simple and easy to use
* Testing activities like planning happens well before coding
* Time saving, quick
* Works well for small projects where requirements are easily understood.



*Figure 5.1: V-model*

More specifically, the left side of the V represents the analysis activities that decompose the user’s needs into small, manageable pieces while the right side of the V shows the corresponding synthesis activities that aggregate (and test) these pieces into a system that need system that meets the user’s need. The V-model demonstrates the relationships between each phase of the development life cycle and its associated phase of testing. The horizontal and vertical axes represent time or project completeness (left to right) and level of abstraction. Advances in the production of executable requirements, architectures, and designs enable testing begin much earlier on the left side of the V so that requirements, architecture, and design defects can be found and fixed early before they can propagate into downstream work products.

### Types of Testing

### Test Stages

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Type of test** | **Stage of test** | | | |
| **Unit** | **Integration** | **System** | **Acceptance** |
| **Function Test** | X | X | X | X |
| **User case Test** | X | X | X | X |

### Tools/Environments

|  |  |
| --- | --- |
| **Environments** | **Testing Tools** |
| **Hardware** | **Notebooks for developing/testing with the minimum configuration:**  - RAM: 2GB  - HDD: 320GB  - Chipset: Intel dual core 1.8GHz  **A server computer with the minimum configuration**  - RAM: 1GB  - HDD: 120GB  - Chipset: Intel dual core 1.8GHz  **Windows Phone with the minimum configuration**  - OS: Windows Phone 8  - Phone Memory: 8GB Storage, 512MB RAM |
| **Software** | - Visual Studio 2013  - Microsoft Office 2010 |
| **Operation System** | - Windows 10 Enterprise 64bit  - Windows phone 8 |
| **Browser** | - Google Chrome |

### Resources

**Resources and Responsibilities**

|  |  |
| --- | --- |
| **Resources** | **Responsibilities** |
| Developers | * Perform unit text, fix bugs |
| Tester | * Create test plan, test cases, test report, bug list report * Review test plan * Execute test * Log bugs, keep tracking and re-test |
| Project Manager | * Responsible for project schedules * Review test plan |

# TEST CASES

## <UC - 01 - Register>

## <UC - 02 - Login>

## <UC - 03 - Logout>

## <UC - 04 - Forgot Password>

## <UC - 05 – View Profile>

## <UC - 06 – Update Profile>

## <UC - 07 – Change Password>

## <UC - 08 – List Favourite Driver>

## <UC - 09 – Add Favourite Driver>

## <UC – 10 – Delete Favourite Driver>

## <UC – 11 - Call Favourite Driver>

## <UC – 12 – Find Lost Asset>

## <UC – 13 – Send Application Feedback>

## <UC – 14 – Call Taxi Center >

## <UC – 15 – Create Trip>

## <UC – 16 – Cancel Trip>

## <UC – 17 – List Complete Trip>

## <UC – 18 – View Bill Detail>

## <UC – 19 – Rate Driver>

# CHECKLISTS

# TEST LOGS

## Defect Logs

<Report defects found in the project>

## Test Reports

<Test report of the project>

|  |  |  |  |
| --- | --- | --- | --- |
| **Project Name** | Smart Medicine Dictionary | **Creator** | HuongTT |
| **Project Code** | SMD | **Reviewer/Approver** | SangNV |
| **Document Code** |  | **Issue Date** | December 01st 2013 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No** | **Module code** | **Pass** | **Fail** | **Untested** | **N/A** | **Number of test cases** |
| **xxx module** | | | | | | |
| 1 |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |