|  |
| --- |
|  |

First Aid

TEST PLAN

Project Code : FAVN

Document Code : FAVN\_Test Plan\_v1.1\_EN

**Ha Noi, 7/11/2016**

Record of change

\*A - Added M - Modified D - Deleted

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Effective Date | Changed Items | A\* M, D | Change Description | New Version |
|  |  | A | Create document | v1.0 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

SIGNATURE PAGE

**ORIGINATOR:** Nguyen Phuc An //2016

Test Leader (TL)

**REVIEWERS:** Dam Huy Hung

Project Manager (PM)

**APPROVAL:** Nguyen Van Sang

Supervisor

TABLE OF CONTENTS

[1 INTRODUCTION 5](#_Toc469382992)

[1.1 Purpose 5](#_Toc469382993)

[1.2 Definitions, Acronyms, and Abbreviations 5](#_Toc469382994)

[1.3 References 6](#_Toc469382995)

[1.4 Background information 6](#_Toc469382996)

[1.5 Scope of testing 6](#_Toc469382997)

[1.6 Constraints 8](#_Toc469382998)

[1.7 Guarantee the quality models 8](#_Toc469382999)

[2 Risks 10](#_Toc469383000)

[2.1 Schedule 10](#_Toc469383001)

[2.2 Management 10](#_Toc469383002)

[2.3 Personnel 10](#_Toc469383003)

[2.4 Requirements 10](#_Toc469383004)

[3 Environment Requiments 11](#_Toc469383005)

[3.1 Hardware 11](#_Toc469383006)

[3.2 Software 11](#_Toc469383007)

[3.3 Infrastructure 11](#_Toc469383008)

[4 TEST STRATEGY 12](#_Toc469383009)

[4.1 Test types 12](#_Toc469383010)

[3.1.1 Function Testing 12](#_Toc469383011)

[3.1.2 User Interface Testing 13](#_Toc469383012)

[4.2 Test stages 14](#_Toc469383013)

[5 RESOURCE 15](#_Toc469383014)

[5.1 Human Resource 15](#_Toc469383015)

[6 Function To be Tested 16](#_Toc469383016)

[6.1 First Aid Mobile Application 16](#_Toc469383017)

[6.2 Ambulance Mobile Application 16](#_Toc469383018)

[6.3 Admin Web Application 17](#_Toc469383019)

[6.4 Dispatcher Web Application 17](#_Toc469383020)

[7 Acceptance test Criteria 18](#_Toc469383021)

[8 TEST MILESTONES 19](#_Toc469383022)

[9 DELIVERABLES 20](#_Toc469383023)

# 

# INTRODUCTION

First Aid Team developed software system for First Aid (First Aid-software system). The system includes 4 main software system which are: First Aid and Ambulance (Mobile Application), Admin and Dispatcher (Web Application). Test team has been tasked with testing the whole system.

## Purpose

The purpose of this document will help test team to understand more clearly about activities of testing, also help test team members know exactly scope of testing, test strategy, environment requirements, resource etc. and necessary activities for testing software take happens correctly. Set schedules for testing activities

## Definitions, Acronyms, and Abbreviations

| **Abbreviations** | **Description** | **Note** |
| --- | --- | --- |
| FAVN | First Aid Viet Nam |  |
| 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 |  |

**Table 1-1**: Definitions and acronyms

## References

| **Title/File name** | **Author** | **Version** | **Effective Date** |
| --- | --- | --- | --- |
| FAVN\_Software Requirements Specifications\_v1.0\_EN | FAVN Team | v1.0 |  |
| FAVN\_ProjectPlan\_v1.0\_EN | AnhND | v1.0 |  |

**Table 1-2:** Reference files

## 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, system will have to:
* Passed the stages of testing: Unit Testing, Integration Testing, System Testing, Acceptance Testing
* Passed the types of testing: Function Testing, User Interface Testing
* Run normally in required devices/browsers.

## Scope of testing

* FAVN will be tested by 4 phases:

**Phase 1: Unit testing**

* Unit testing will be done by developers
* Developers use 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: Integration testing**

* After finishing component testing, integration testing will be performed by testers.
* Material are integration test cases, high- level design
* 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 3: 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.
* Material area system test case, SRS
* 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 4: Acceptance testing**

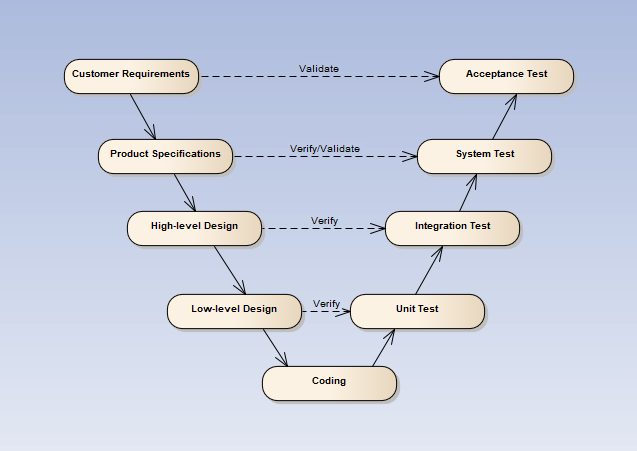
* Base on customer/user requirement specification, system is tested again, for ensure there is not lacking or mistake any requirement.
* If there is any problem, developers have to fix/update and tester will verify them.
* Acceptance testing is ended only when whole system met requirement specification.

## Constraints

* There are only 2 testers and tester don’t have experience in testing.
* Time of Testing is short.
* Have a lot of environments should be tested: Android 4.4.2 and above… and more browsers: Firefox, IE, Chrome … But Test team cannot cover all

## Guarantee the quality models

FAVN project follows V-Model process:



**Figure 1: V-Model**

Testing progress is divided to 4 phases include: Unit 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 FirstAid\_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.
* 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
* 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.

# Risks

## Schedule

The schedule for each phase is very aggressive and could affect testing. A slip in the schedule in one of the other phases could result in a subsequent slip in the test phase. Close project management is crucial to meeting the forecasted completion date.

## Management

Management support is required so when the project falls behind, the test schedule does not get squeezed to make up for the delay. Management can reduce the risk of delays by supporting the test team throughout the testing phase and assigning people to this project with the required skills set.

## Personnel

Due to the aggressive schedule, it is very important to have experienced testers on this project. Unexpected turnovers can impact the schedule. If attrition does happen, all efforts must be made to replace the experienced individual

## Requirements

The test plan and test schedule are based on the current Requirements Document. Any changes to the requirements could affect the test schedule.

# Environment Requiments

## Hardware

|  |  |  |  |
| --- | --- | --- | --- |
| Name | | Purpose | Detail |
| Laptop Dell | Device for create and execute test | Window 10 Education core i5 |
| Laptop Dell | Device for create and execute test | Window 10 professional core i7 |
| Sony Xperia Z3 compact | Device for create and execute test | Android 5.0.1 |
| MI Note 3 | Device for create and execute test | Android 5.0.2 |

## Software

|  |  |  |
| --- | --- | --- |
| Name | Purpose | Detail |
| Test Plan | Managing test | Microsoft Word 2016 |
| Test case | Executing test | Microsoft Excel 2016 |
| Test report, Test checklist | Tracking test | Microsoft Excel 2016 |
| Chrome, CocCoc | Executing test | Chrome 35.0, CocCoc 35.0 |

## Infrastructure

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Purpose | Detail | Vendor/In-house | Version |
| FAVN\_Defect Log Management\_v1.0\_EN | Tracking bug during testing time | Microsoft Excel  2016 | FPT-University | v1.0 |
| Test Effort | Effort execute test | Microsoft Excel  2016 | FPT-University | v1.0 |

# TEST STRATEGY

## Test types

### Function Testing

* Functional testing is a type of software testing whereby the system is tested against the functional require specifications.
* Functions are tested by feeding them input and examining the output. Functional testing ensure that the requirements are properly satisfied by the website and mobile application. 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: | The type of this test is to ensure proper target-of-test functionality, including user interaction, all function defined in specification document implemented correctly. |
| Technique: | Executing 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 3-1:** Function Testing

### User Interface Testing

* GUI testing is the process of ensuring proper functionality of the GUI for a given web, mobile application 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 use 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 for 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 3-2:** GUI Testing

## 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 | Integration | System | Acceptance |
| Function Testing | X | X | X | X |
| User Interface Testing |  | X | X | X |

# 

**Table 3-4:** Test stage

# RESOURCE

## Human Resource

|  |  |  |
| --- | --- | --- |
| Worker/Doer | Role | Specific Responsibilities/Comments |
| AnNP | 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 * Ensures the overall success of the test cycles * Coordinate weekly meetings and will communicate the testing status to the project team |
| AnhND | Tester | * Create and review Test Case. * Execute test. * Create Test view points * Create and review Test Report * Responsible for performing the actual system testing |

**Table 4-1:** Human resource

# Function To be Tested

## First Aid Mobile Application

|  |  |  |  |
| --- | --- | --- | --- |
| FC No. | Group Of Functions | Function | Glossary |
| First Aid Application | | | |
|  | Emergency | View list of injuries |  |
|  | View instruction steps of injury |  |
|  | Search injuries |  |
|  | Call 115 |  |
|  | Read instruction steps |  |
|  | View current location |  |
|  | View list of nearest health facilities |  |
|  | Show direction to health facility |  |
|  | Call to health facility |  |
|  | Learning | View list of learning injuries |  |
|  | View learning instruction steps |  |
|  | View FAQs |  |
|  | More | Send questions |  |
|  | View courses |  |
|  | Share application |  |
|  | Setting | Allow to send information |  |
|  | Information | View application information |  |

## Ambulance Mobile Application

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| UC No. | Group Of Functions | | Function | Glossary | |
| Ambulance Application | | | | | |
|  | Common | Login | | |  |
|  | Logout | | |  |
|  | Task | View ambulance team information | | |  |
|  | Change status | | |  |
|  | Notify task | | |  |
|  | View current location | | |  |
|  | Show location of caller | | |  |
|  | Show direction to caller | | |  |
|  | Report picked up caller | | |  |
|  | Report completed task | | |  |
|  | Report problem | | |  |
|  | Information | View application information | | |  |

## Admin Web Application

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| UC No. | Group Of Functions | | Function | | Glossary |
| Admin/Expert Application | | | | | |
|  | Common | Login | |  | |
|  | Logout | |  | |
|  | Manage user's account | Create account | |  | |
|  | View list of users | |  | |
|  | Search user's account | |  | |
|  | Update user’s account | |  | |
|  | Delete user’s account | |  | |
|  | Manage Q&A | View user’s question | |  | |
|  | View answer | |  | |
|  | Answer question | |  | |
|  | Delete question | |  | |
|  | Manage injury | Add new injury | |  | |
|  | View list of injuries | |  | |
|  | Search injury | |  | |
|  | Update injury | |  | |
|  | Delete injury | |  | |

## Dispatcher Web Application

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| UC No. | Group Of Functions | Function | Glossary | |
| Dispatcher Application | | | | |
|  | Common | Login | |  |
|  | Logout | |  |
|  | Dispatch | Create emergency case | |  |
|  | Dispatch ambulance | |  |
|  | View ambulance information | |  |
|  | View caller information | |  |
|  | Show direction from ambulance to caller | |  |
|  | View list of waiting callers | |  |
|  | Notify ambulances status | |  |
|  | Cancel emergency case | |  |

# Acceptance test Criteria

* Criteria for Unit test of Development team, for Test team accepts to start testing:
  + Number of TC/KLOC: 10TC/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

# TEST MILESTONES

Unit 5/11

System:

|  |  |  |  |
| --- | --- | --- | --- |
| Milestone Task | Effort (pd) | Start Date | End Date |
| Create Test Plan | 2 days | 29/09/2016 | 03/10/2016 |
| Review & update TP | 2 days | 3/10/2016 | 05/10/2016 |
| Unit Test Cases | 19 days | 19/10/2016 | 15/11/2016 |
| Review & update Unit TC | 19 days | 19/10/2016 | 15/11/2016 |
| Create Integration Test case | 4 days | 17/10/2016 | 25/10/2016 |
| Review & Update Integration TC | 2 days | 21/10/2016 | 25/10/2016 |
| Create System Test case | 4 days | 17/10/2016 | 25/10/2016 |
| Review & Update System TC | 2 days | 21/10/2016 | 25/10/2016 |
| Execute UT test phase 1 | 19 days | 19/10/2016 | 15/11/2016 |
| Execute UT test phase 2 | 19 days | 19/10/2016 | 15/11/2016 |
| Execute Integration test phase 1 | 2 days | 8/11/2016 | 10/11/2016 |
| Execute Integration test phase 2 | 1 days | 15/11/2016 | 16/11/2016 |
| Execute System test phase 1 | 2 days | 10/11/2016 | 14/11/2016 |
| Execute System test phase 2 | 1 days | 16/11/2016 | 17/11/2016 |

**Table 6-1:** Test milestones

# DELIVERABLES

| No | Deliverables | Language | Delivered Date |
| --- | --- | --- | --- |
|  | Test Plan | English | 6/10/2016 |
|  | Integration Test case | English | 24/11/2016 |
|  | System Test case | English | 24/11/2016 |
|  | Defect Log Management | English | 26/11/2016 |
|  | Test report | English | 24/11/2016 |

**Table 7-1**: Deliverables