**Chapter 5**

**Test Plan**

**Chapter One | Introduction**

* 1. **Objectives**

The objectives of the test plan of Emergency Information on Mobile project is to establish test plan of the unit testing and system testing and make sure that the bugs or the defects are discovered and fixed. The unit testing covers all of implemented methods in the Emergency Information on Mobile system. The system testing covers the user requirements.

* 1. **Scope**

This test plan describes the unit testing activities to detect the defects on the system and describes the system testing activities for testing a completely integrated system to verify that it meets the user requirements.

* 1. **Acronyms and Definitions**
     1. **Acronyms**

EIOM Emergency Information on Mobile

SRS Software Requirement Specification

URS User Requirement Specification

SDD Software Design Document

UI User Interface

* + 1. **Definitions**

|  |  |
| --- | --- |
| Feature | Transformation of input parameters to output parameters based on a specified algorithm. It describes the functionality of a product in the language of the product. Used for requirements analysis, design, coding, testing or maintenance.[IEEE90] |
| IEEE | Institute for Electrical and Electronics Engineers. Biggest global interest group for engineers of different branches and for computer scientists.[IEEE90] |
| Requirement | (1) A condition or capability needed by a user to solve a problem or achieve an objective.(2) A condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed document.(3) A documented representation of a condition or capability as in definition (1) or (2).[IEEE90] |

|  |  |
| --- | --- |
| Specification | Precise description of an activity or work product which serves as basis or input for further activities or work product. A specification can comprise requirements to a  Product and how they will be solved. Different parts of a specification (e.g. what is to be done, how it will be done) must not be mixed.[IEEE90] |
| UML | Unified Modeling Languages. Standardized notation for  Modeling design descriptions, architectures or scenarios. Not depending on a specific method. Issued and maintained by the Object Management Group (OMG).[IEEE90] |
| Use case | (1)Concept to describe a system based on usage of system resources by its environment. Characterized by an objective-set of interactions within and at the borders of that system. (2) Notation from UML for describing a scenario(Usage approach, operational scenario)from the perspective of this user.[IEEE90] |

**Chapter Two | Test Plan and Test Procedure**

* 1. **Test Objective**

The objectives of testing Emergency Information on Mobile project are:

1. All bugs or defects are detected.
2. Those bugs or defects are fixed.
3. Functions and user interface covered the requirements.
4. All functions and features must be there.
   1. **Scope of testing**

Emergency Information on Mobile will test by white-box testing techniques that are unit testing and system testing and record the test results in the test record.

* 1. **Purpose of Test Plan and Test Procedure**

In this document consist of test plan and test procedure of progress report I. So the stakeholder of Emergency Information on Mobile can review the test in project.

* 1. **Test Duration**

|  |  |
| --- | --- |
| **Progress** | **Date and Duration** |
| Progress Report I | **Perform date:** xxx  **Duration:** xxx |

* 1. **Test Responsibility**

|  |  |
| --- | --- |
| **Item** | **Responsibility** |
| Unit test of web application | Putchakarn, Sawatdiporn |
| Unit test of mobile application | Putchakarn, Sawatdiporn |
| Record unit test of web application | Putchakarn, Sawatdiporn |
| Record unit test of mobile application | Putchakarn, Sawatdiporn |

* 1. **Test Strategy**

Emergency Information on Mobile test strategy will be follow by:

1. Design test case for each feature.
2. Prepare test data for each feature.
3. Determine expected results.
4. Perform testing on individual features.
5. Result of testing will be record.
6. All test files will be store in the project repository.
   1. **Result of Testing**

In the test record the test result will separate into two parts, which are:

1. Actual output: The actual outputs that are performed by each test case.
2. Pass/Fail criteria:
   1. Pass: The result of actual is same like expected result.
   2. Fail: the result of actual result is not same like expected result.
   3. **Test Environment**
      1. **Hardware**

* **Computers**
  + Apple Macbook Pro mid 2013
    - Processor: Intel® Core™ i7-3520M CPU @ 2.90GHz 2.90GHz
    - RAM: 8.00 GB
    - Operating System: Windows 7 Ultimate
  + Dell Inspiron n5110
    - Processor: Intel® Core™ i5-2410M CPU @ 2.30GHz 2.30GHz
    - RAM: 4.00 GB
    - Operating System: Windows 7 Ultimate
* **Mobile phones:** Android Operating System
  + Samsung Galaxy Grand 2 SM-G7102
    - CPU: Quad-core 1.2 GHz Cortex-A7
    - RAM: 1.5 GB
    - Operating System: Android OS, v4.3 (Jelly Bean)
  + Samsung Galaxy S DUOS
    - CPU: 1GHz Cortex-A5
    - RAM: 768 MB
    - Operating System: Android OS, v4.04 (Ice Cream Sandwich)
    1. **Software**
* IntelliJ IDEA
* Eclipse
* Google chrome

**Chapter Three | Unit testing of EIOM**

**3.1 Web Application**

**Unit Test Case 1 (UTC-01):** getHelpPlaces():List<HelpPlace>

**Class name:** HelpPlaceServiceTest **Test Data:** Test Record 1 in appendix A

|  |  |  |
| --- | --- | --- |
| Case | Description | Expected Result |
| Size of help place list |
| 1 | Test get all help place in database | 5 |

**Unit Test Case 2 (UTC-02):** findById(Integer id):HelpPlace

**Class name:** HelpPlaceServiceTest **Test Data:** Test Record 1 in appendix A

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Case | Description | Input | Expected Result | | | | | | | | |
| Help place object | | | | | | | | |
| Name | Address | District | Province | Zip code | Phone Number | Latitude | Longitude | Category |
| 1 | Test find help place by input ID which is in database | 1 | Maharaj Nakorn Chiang Mai Hospital | 10 Suthep Rd | Mueang Chiang Mai | Chiang Mai | 50200 | 053-947700 | 18.789602 | 98.974209 | Hospital |
| 2 | Test find help place by input ID which is not in database | 6 | null | | | | | | | | |

**Unit Test Case 3 (UTC-03):** updateHelpPlace(HelpPlace helpPlace):void

**Class name:** HelpPlaceServiceTest **Test Data:** Test Record 1 in appendix A

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Case | Description | Input | | | | | | | | |
| Help place object | | | | | | | | |
| Name | Address | District | Province | Zip code | Phone Number | Latitude | Longitude | Category |
| 1 | Test edit help place | Maharaj Nakorn Chiang Mai Hospital | 10 Suthep Rd | Mueang Chiang Mai | Lampang | 50200 | 053-111111 | 18.789602 | 98.974209 | Hospital |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Case | Expected Result | | | | | | | | |
| Help place object | | | | | | | | |
| Name | Address | District | Province | Zip code | Phone Number | Latitude | Longitude | Category |
| 1 | Maharaj Nakorn Chiang Mai Hospital | 10 Suthep Rd | Mueang Chiang Mai | Lampang | 50200 | 053-111111 | 18.789602 | 98.974209 | Hospital |

**Unit Test Case 4 (UTC-04):** deleteHelpPlace(HelpPlace helpPlace):void

**Class name:** HelpPlaceServiceTest **Test Data:** Test Record 1 in appendix A

|  |  |  |  |
| --- | --- | --- | --- |
| Case | Description | Input | Expected Result |
| Help place object | Help place object in database |
| ID | ID |
| 1 | Test delete help place in database | 1 | 2,3,4,5 |
| 2 | Test delete help place in database | 4 | 1,2,3,5 |
| 3 | Test delete help place which not in database | 7 | 1,2,3,4,5 |

**Unit Test Case 5 (UTC-05):** getHelpPlacesByCategory(Integer categoryId)**:**List<HelpPlace>

**Class name:** HelpPlaceServiceTest **Test Data:** Test Record 1 in appendix A

|  |  |  |  |
| --- | --- | --- | --- |
| Case | Description | Input | Expected Result |
| Size of help place list |
| 1 | Test get help place by input category id | 1 | 3 |
| 2 | Test get help place by input category id | 3 | 1 |
| 3 | Test get help place by input category id which is not in database | 5 | null |

**Unit Test Case 6 (UTC-06):** getHelpPlacesByProvince(Integer provinceId)**:**List<HelpPlace>

**Class name:** HelpPlaceServiceTest **Test Data:** Test Record 1 in appendix A

|  |  |  |  |
| --- | --- | --- | --- |
| Case | Description | Input | Expected Result |
| Size of help place list |
| 1 | Test get help place by input province id | 1 | 3 |
| 2 | Test get help place by input province id | 2 | 1 |
| 3 | Test get help place by input province id which is not in database | 5 | null |

**Unit Test Case 7 (UTC-07):** getHelpPlacesByCategoryAndProvince(Integer categoryId,Integer provinceId)**:**List<HelpPlace>

**Class name:** HelpPlaceServiceTest **Test Data:** Test Record 1 in appendix A

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Case | Description | Input | | Expected Result |
| Category ID | Province ID | Size of help place list |
| 1 | Test get help place by input category id and province id | 1 | 1 | 2 |
| 2 | Test get help place by input category id and province id | 2 | 3 | 1 |
| 3 | Test get help place by input category id and province id which is not in database | 3 | 3 | null |
| 4 | Test get help place by input category id and province id which is not in database | 2 | 1 | null |

**Unit Test Case 8 (UTC-08):** getCategories():List<Category>

**Class name:** CategoryServiceTest  **Test Data:** Test Record 2 in appendix A

|  |  |  |
| --- | --- | --- |
| Case | Description | Expected Result |
| Size of category list |
| 1 | Test get all categories in database | 3 |

**Unit Test Case 9 (UTC-09):** getProvinces():List<Province>

**Class name:** ProvinceServiceTest  **Test Data:** Test Record 3 in appendix A

|  |  |  |
| --- | --- | --- |
| Case | Description | Expected Result |
| Size of category list |
| 1 | Test get all provinces in database | 3 |

**Unit Test Case 10 (UTC-10):** getCategoryById (Integer id):Category

**Class name:** CategoryServiceTest  **Test Data:** Test Record 2 in appendix A

|  |  |  |  |
| --- | --- | --- | --- |
| Case | Description | Input | Expected Result |
| Category object |
| Name |
| 1 | Test find category by input ID which is in database | 1 | Hospital |
| 2 | Test find category by input ID which is not in database | 5 | null |

**Unit Test Case 11 (UTC-11):** getProvinceById (Integer id):Province

**Class name:** ProvinceServiceTest  **Test Data:** Test Record 3 in appendix A

|  |  |  |  |
| --- | --- | --- | --- |
| Case | Description | Input | Expected Result |
| Category object |
| Name |
| 1 | Test find province by input ID which is in database | 1 | Chiang Mai |
| 2 | Test find province by input ID which is not in database | 5 | null |

**3.2 Mobile Application**

**\*\* ใส่ ตาราง unit test mobile**

**Appendix A**

* Test Record 1

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Help Place Object** | | | | | | | | | |
| ID | Name | Address | District | Province | Zip code | Phone Number | Latitude | Longitude | Category |
| 1 | Maharaj Nakorn Chiang Mai Hospital | 10 Suthep Rd | Mueang Chiang Mai | Chiang Mai | 50200 | 053-947700 | 18.789602 | 98.974209 | Hospital |
| 2 | Lanna Hospital | Chang Phuak | Mueang Chiang Mai | Chiang Mai | 50300 | 053-999758 | 18.812723 | 98.991151 | Hospital |
| 3 | Lampang Hospital | Tambon Phrabat | Amphoe Mueang Lampang | Lampang | 52000 | 054-237400 | 18.285378 | 99.506305 | Hospital |
| 4 | Chiang Rai Police Station | Rattanakeat Road | Mueang Chiang Rai | Chiang Rai | 57000 | 053-603100 | 19.912221 | 99.832526 | Police Station |
| 5 | Ruangchai Yon Garage | Outer Ring Road | Saraphi District | Chiang Mai | 50000 | 053-242999 | 18.750651 | 99.055108 | Garage |

* Test Record 2

|  |  |
| --- | --- |
| **Category Object** | |
| ID | Name |
| 1 | Hospital |
| 2 | Police Station |
| 3 | Garage |

* Test Record 3

|  |  |
| --- | --- |
| **Province Object** | |
| ID | Name |
| 1 | Chiang Mai |
| 2 | Chiang Rai |
| 3 | Lampang |