**The Chiang Mai Red Taxi**

**Service Assistant**

**Test Plan**

**By**

**Kanittee Hongron 542115003**

**Pimchittra Sukkasem 542115042**

**Department of Software Engineering**

**College of Arts, Media and Technology**

**Chiang Mai University**

**Project Advisor**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Ms. Pattama Longanee**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Document Name** | **Version** | **Status** | **Date** | **Viewable** | **Reviewer** | **Responsible** |
| **RoseBeam\_Software Testplan\_13JUNE2014** | CM Red Taxi Service Assistant\_SRS\_0.1 | Reviewed | 13-06-2014 | PM, SA, DEV, QA, Tester, Advisor | Kanittee,  Pimchittra | Kanittee,  Pimchittra |
| **RoseBeam\_Software Requirement Specification\_20\_JULY\_2014** | CM Red Taxi Service Assistant\_ SRS \_0.2 | Reviewed | 20-07-2014 | PM, SA, DEV, QA, Tester, Advisor | Kanittee,  Pimchittra | Kanittee,  Pimchittra |
| **RoseBeam\_Software Requirement Specification\_27 JULY 2014\_** | CM Red Taxi Service Assistant\_ SRS \_0.3 | Reviewed | 27-07-2014 | PM, SA, DEV, QA, Tester, Advisor | Kanittee,  Pimchittra | Kanittee,  Pimchittra |
| **RoseBeam\_Software Requirement Specification\_31JULY 2014** | CM Red Taxi Service Assistant\_ SRS \_1.0 | Reviewed | 31-07-2014 | PM, SA, DEV, QA, Tester, Advisor | Kanittee,  Pimchittra | Kanittee,  Pimchittra |
| **RoseBeam\_Software Requirement Specification\_27AUGUST 2014** | CM Red Taxi Service Assistant\_ SRS \_1.0 | Reviewed | 27-08-2014 | PM, SA, DEV, QA, Tester, Advisor | Kanittee,  Pimchittra | Kanittee,  Pimchittra |

**Table of Content**

**Chapter One | Introduction**………………………………………..………….5

1.1Purpose...…………………………………………………………………….……………..5

1.2Scope…...………………………………………………………………….……………….5

1.3Acronyms and Definitions ...……….…..………………………………………………….5

1.4Abbreviation...……………………………………………………………………………...7

1.5Dependency Document…………………….......…………………………………………..7

**Chapter Two | Test Plan and Test Procedure………………………………7**

2.1Test Objectives…………...………………………………………………………………...7

2.2Scope of Testing…………...……………………………………………………………….8

2.3Test Duration……………………...………………………………………………………..8

2.4Test Responsibility……………………...………………………………………………….8

2.5Test Strategy………………………………………………………………………………..8

2.6Result of Testing………………...…………………………………………………………8

2.7Test Environment………………………………...………………………………………..9

**Unit Test Case-Driver side**……………………………………………………10

Unit Test Case 1 (UTC-01): ...…………………………………………………………….....10

Unit Test Case 2 (UTC-02): …………………………………………………………………11 Unit Test Case 3 (UTC-03): …………………………………………………………………12 Unit Test Case 4 (UTC-04): ………………………………………………………………....13 Unit Test Case 5 (UTC-05): …………………………………………………………………14 Unit Test Case 6 (UTC-06): …………………………………………………………………15 Unit Test Case 7 (UTC-07): …………………………………………………………………16

Unit Test Case 8 (UTC-08): …………………………………………………………………17 Unit Test Case 9 (UTC-09): …………………………………………………………………18 Unit Test Case 10 (UTC-10): ………………………………………………………………..18 Unit Test Case 11 (UTC-11): ………………………………………………………………..19 Unit Test Case 12 (UTC-12): ………………………………………………………………..20

**Unit Test Case-Passenger side……………………………………………….21**

Unit Test Case 13 (UTC-13): ………………………………………………………………..22 Unit Test Case 14 (UTC-14): ………………………………………………………………..23 Unit Test Case 15 (UTC-15): ………………………………………………………………..24

Unit Test Case 16 (UTC-16): ………………………………………………………………..25 Unit Test Case 17 (UTC-17): ………………………………………………………………..26 Unit Test Case 18 (UTC-18): ………………………………………………………………..27 Unit Test Case 19 (UTC-19): ………………………………………………………………..28 Unit Test Case 20 (UTC-20): ………………………………………………………………..29 Unit Test Case 21 (UTC-21): ………………………………………………………………..30 Unit Test Case 22 (UTC-22): ………………………………………………………………..31

Unit Test Case 23 (UTC-23): ………………………………..………………………………32

**System Test Case……………………………………………………………..33**

System Test Case 1 (STC-01): …………………………….………………………………...33

System Test Case 2 (STC-02): …………………………….………………………………...34

System Test Case 3 (STC-03): …………………………….………………………………...35

System Test Case 4 (STC-04): …………………………….…………………………….…..36

System Test Case 5 (STC-05): …………………………….………………………………...37

System Test Case 6 (STC-06): …………………………….………………………………...38

System Test Case 7 (STC-07): …………………………….…………………………….…..39

System Test Case 8 (STC-08): …………………………….…………………………….…..40

System Test Case 9 (STC-09): …………………………….…………………………….…..41

System Test Case 10 (STC-10): ………………………...……………………………….…..42

System Test Case 11 (STC-11): ………………………...……………………………….…..43

System Test Case 12 (STC-12): ………………………...……………………………….…..44

System Test Case 13 (STC-13): ………………………...…………………………….……..45

System Test Case 14 (STC-14): ………………………...…………………………….……..46

System Test Case 15 (STC-15): ………………………...……………………………….…..47

System Test Case 16 (STC-16): ………………………...……………………………….…..48

System Test Case 17 (STC-17): ………………………...……………………………….…..49

System Test Case 18 (STC-18): ………………………...……………………………….…..50

System Test Case 19 (STC-19): ………………………...……………………………….…..51

System Test Case 20 (STC-20): ………………………...……………………………….…..52

**Appendix A …………………………………………………………………………………53**

**Appendix B …………………………………………………………………………………54**

**Chapter One | Introduction**

**1.Introduction**

This Test plan document is the document to describe plan of testing and methodologies that are used in testing The Chiang Mai Red Taxi Service Assistant.

**1.1Purpose**

The purpose of Test Plan describes the plan for unit testing and system test of The Chiang Mai Red taxi Service Assistant system. This document includes test cases for unit testing and system testing, and test data. The Chiang Mai Red taxi Service Assistant test plan consists of the description of each function in the system. From the result of each function can help developer to fix the defects and reducing the cost of the long term maintenance.

**1.2Scope**

The Scope the plan document covers all the events and activities involving the verification of the Chiang Mai Red taxi Service Assistant and System requirements. The unit testing is used to detect the defects in the system. The system testing is used to verify that the system is fully integrated and meets all of the user requirements specified in the Software Requirement Specification.

**1.3Acronyms and Definitions**

**Acronyms**

PMP Project Management Plan

SDD Software Design Document

SRS Software Requirement Specification

UC Use Case

UI User Interface

UTC Unit Test Case

STC System Test Case

**Definitions**

Feature Transformation of input parameters to output parameters based on a specified algorithm. It describes the functionality of a produce in the language of the product. Used for requirements analysis, design, coding, testing or maintenance. [IEEE90]

Design The period of time in the software life cycle during which the designs for architecture, software component, interfaces and data are created, documented, and verified to satisfy requirements. [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 processed by 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]

White box testing Testing process that focus on internal structure. The tester should know the code inside the program and test it through the code and determines the appropriate outputs.

Black box testing Process, device or system that focus on input, output and transfer characteristics without knowledge about it internal structure.

Unit Testing A level of the software testing process where individual units/components of a software/system are tested. The purpose is to validate that each unit of the performs as designed.

System testing A level of the software testing process where a complete, integrated system/software is tested. The purpose of this test is to evaluate the system's compliance with the specified requirements.

**1.4Abbreviation**

Unit Test Case-xx Unit test case number

System Test Case-xx System test case number

**1.5Dependency Document**

This Test Plan is depended on the following documents

* Software Project Management plan
* Software Requirement Specification
* Software Design Document
* Traceability Record

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

**2.1Test Objectives**

The objectives of The Chiang Mai Red taxi Service Assistant are to ensure that the system fulfills the followings:

* Bugs and defects are detected and fixed.
* All user requirements are met.
* All functionalities and features are provided.
* All functionalities and features conform to the descriptions in the Project Requirement Specification and the Project Design Specification.

**2.2Scope of Testing**

The Chiang Mai Red taxi Service Assistant will be tested using both white-box testing technique (i.e. unit testing) and black-box testing technique (i.e. system testing). The test results will be documented in the Test Record.

**2.3Test Duration**

|  |  |
| --- | --- |
| **Progress** | **Date and Duration** |
| Progress Report I | Perform date: 1 July 2014-30 July 2014  Duration: 30 days |
| Progress Report II | Perform date:-  Duration:- |
| Final Progress | Perform date:-  Duration:- |

**2.4Test Responsibility**

|  |  |
| --- | --- |
| **Item** | **Responsibility** |
| Unit Test | Kanittee, Pimchittra |
| Unit Test Record | Kanittee, Pimchittra |
| Integration Test | Kanittee, Pimchittra |
| Integration Test Record | Kanittee, Pimchittra |
| System Test | Kanittee, Pimchittra |
| System Test Record | Kanittee, Pimchittra |

**2.5Test Strategy**

The Chiang Mai red taxi service assistant test strategy will ne follow by:

1.Design test case for each feature.

2.Prepare test data 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

**2.6Result 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:

2.1 Pass: the result of actual result is same like expected result.

2.2 Fail: the result of actual result is not same like expected result.

**2.7Test Environment**

**Hardware**

-Computer : Lenovo Windows 7 500GB HDD, 2GB DDR3 memory

-Smart Phone : Lenovo A316i android operating system 4.2.2

**Software**

-Java Android SDK 4.0

-Google map V.3

-PHP

-JSON

**Unit Test Case-Driver side**

**UTC 01-** **onCreate(Bundle savedInstanceState)**

**Package:**

com.taxiservice.chiangmairedtaxi.test

**Class:**

ChatActivity

**Unit Test Case 01 (UTC-01):** onCreate(Bundle savedInstanceState)

**Description:** Chatroom should be active all the time when Driver clicks to accept Passenger’s request. Test that the chatroom is active when onCreate() method is invoked.

**Prerequisites:**

ReportFunctionCallResult(String functionName, boolean functionCallResult)

**Test data:**

**-**

**Test cases:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description** | **Input** | **Assertion** | **Expected Result** |
| 1 | Test that System that Chat room is active when Oncreate() is invoked | - | assertEquals(tclass.callResult,"onCreate(Bundle saveInstanceState):Success"); | True |

**UTC 02-** **addListenerOnButton()**

**Package:**

com.taxiservice.chiangmairedtaxi.test

**Class:**

ChatActivity

**Unit Test Case 02 (UTC-02):** addListenerOnButton()

**Description:** Chatting process should be active all the time when Driver clicks to send the chat message to passenger.

**Prerequisites:**

ReportFunctionCallResult(String functionName, boolean functionCallResult)

**Test data:**

**-**

**Test cases:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description** | **Input** | **Assertion** | **Expected Result** |
| 1 | Test that System that Chat room is active when Oncreate() is invoked | - | assertEquals(tclass.callResult,"onCreate(Bundle saveInstanceState):Success"); | True |

**UTC 03-** **onCreate(Bundle savedInstanceState)**

**Package:**

com.taxiservice.chiangmairedtaxi.test

**Class:**

MainActivity

**Unit Test Case 03 (UTC-03):** onCreate(Bundle savedInstanceState)

**Description:** Login page should be active all the time when Driver start the application .

**Prerequisites:**

ReportFunctionCallResult(String functionName, boolean functionCallResult)

**Test data:**

**-**

**Test cases:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description** | **Input** | **Assertion** | **Expected Result** |
| 1 | Test that System that log in page is active when Oncreate() is invoked | - | assertEquals(testclass.callResult,"onCreate(Bundle saveInstanceState):Success"); | True |

**UTC 04-** **onCreate(Bundle savedInstanceState)**

**Package:**

com.taxiservice.chiangmairedtaxi.test

**Class:**

MylocatemapActivity

**Unit Test Case 04 (UTC-04):** onCreate(Bundle savedInstanceState)

**Description:** Search page should be active when Driver login to the system successfully.Test the update driving information page is active when invoke method onCreate(Bundle savedInstanceState).

**Prerequisites:**

ReportFunctionCallResult(String functionName, boolean functionCallResult)

**Test data:**

**-**

**Test cases:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description** | **Input** | **Assertion** | **Expected Result** |
| 1 | Test that Search page is active when invoke onCreate(Bundle savedInstanceState) | - | assertEquals(tclass.callResult,"onCreate(Bundle saveInstanceState):Success"); | True |

**UTC 05- showOnMap(double lat,double lng)**

**Package:**

com.taxiservice.chiangmairedtaxi.test

**Class:**

MylocatemapActivity

**Unit Test Case 05 (UTC-05):** showOnMap(double lat,double lng)

**Description:** The map should be active when Driver access to the update driving information.The map should be active when invoke the method showOnMap(double lat,double lng)**.**

**Prerequisites:**

ReportFunctionCallResult(String functionName, boolean functionCallResult)

**Test data:**

**-**

**Test cases:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description** | **Input** | **Assertion** | **Expected Result** |
| 1 | Test that the map is active when invoke method showOnMap(double lat,double lng) | 18.767925361556415, 98.96840095521543 | assertEquals(tclass.callResult,"showOnMap(double lat,double lng):Success"); | True |

**UTC 06- updatePosition()**

**Package:**

com.taxiservice.chiangmairedtaxi.test

**Class:**

MylocatemapActivity

**Unit Test Case 06 (UTC-06):** updatePosition()

**Description:** Updating position should be active when Driver already login to the system. Testing to update position is active when invoke the method updatePosition()

**Prerequisites:**

ReportFunctionCallResult(String functionName, boolean functionCallResult)

**Test data:**

**-**

**Test cases:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description** | **Input** | **Assertion** | **Expected Result** |
| 1 | Testing to update position is active when invoke the method updatePosition() | - | assertEquals(tclass.callResult,"updatePosition():Success"); | True |

**UTC 07- createDestSpinner()**

**Package:**

com.taxiservice.chiangmairedtaxi.test

**Class:**

MylocatemapActivity

**Unit Test Case 07 (UTC-07):** createDestSpinner()

**Description:** Destination spinner (drop down list) should be active when Driver is in update driving information page. Testing Destination spinner is active when invoke the method createDestSpinner().

**Prerequisites:**

ReportFunctionCallResult(String functionName, boolean functionCallResult)

**Test data:**

**-**

**Test cases:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description** | **Input** | **Assertion** | **Expected Result** |
| 1 | Testing Destination spinner is active when invoke the method createDestSpinner(). | - | assertEquals(tclass.callResult,"createDestSpinner():Success"); | True |

**UTC 08- addListenerOnButton()**

**Package:**

com.taxiservice.chiangmairedtaxi.test

**Class:**

MylocatemapActivity

**Unit Test Case 08 (UTC-08):** addListenerOnButton()

**Description:** Updating driving information should be active when Driver click on the OK button ,after inputting driving information . Testing Updating driving information is active when invoke the method addListenerOnButton()

**Prerequisites:**

ReportFunctionCallResult(String functionName, boolean functionCallResult)

**Test data:**

**-**

**Test cases:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description** | **Input** | **Assertion** | **Expected Result** |
| 1 | Testing Updating driving information is active when invoke the method addListenerOnButton() | - | assertEquals(tclass.callResult,"addListenerOnButton():Success"); | True |

**UTC 09- onCreate(Bundle savedInstanceState)**

**Package:**

com.taxiservice.chiangmairedtaxi.test

**Class:**

mysearchActivity

**Unit Test Case 09 (UTC-09):** addListenerOnButton()

**Description:** Request page should be active when Driver click update driving information . Testing Request page is active when invoke onCreate(Bundle savedInstanceState)**.**

**Prerequisites:**

ReportFunctionCallResult(String functionName, boolean functionCallResult)

**Test data:**

**-**

**Test cases:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description** | **Input** | **Assertion** | **Expected Result** |
| 1 | . Testing Request page is active when invoke onCreate(Bundle savedInstanceState). | - | assertEquals(testclass.callResult,"onCreate(Bundle saveInstanceState):Success"); | True |

**UTC 10- createDestListView(ListView view,String url,String type)**

**Package:**

com.taxiservice.chiangmairedtaxi.test

**Class:**

mysearchActivity

**Unit Test Case 10 (UTC-10):** createDestListView(ListView view,String url,String type)

**Description:** Request list should be active when Driver click update driving information . Testing Request list is active when invoke onCreate(Bundle savedInstanceState)**.**

**Prerequisites:**

ReportFunctionCallResult(String functionName, boolean functionCallResult)

**Test data:**

**-**

**Test cases:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description** | **Input** | **Assertion** | **Expected Result** |
| 1 | Testing Request list is active when invoke onCreate(Bundle savedInstanceState)**.** | - | assertEquals(testclass.callResult,"onCreate(Bundle saveInstanceState):Success"); | True |

**UTC 11- testupdateChatStatus()**

**Package:**

com.taxiservice.chiangmairedtaxi.test

**Class:**

mysearchActivity

**Unit Test Case 11 (UTC-11):** testupdateChatStatus()

**Description:** updating Chat status should be active when Driver click accept the passenger request Testing updating Chat status is active when invoke testupdateChatStatus()

**Prerequisites:**

ReportFunctionCallResult(String functionName, boolean functionCallResult)

**Test data:**

**-**

**Test cases:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description** | **Input** | **Assertion** | **Expected Result** |
| 1 | Testing updating Chat status is active when invoke testupdateChatStatus() | - | assertEquals(tclass.callResult,"updateChatStatus(String request\_status):Success"); | True |

**UTC 12-** **onCreate(Bundle savedInstanceState)**

**Package:**

com.taxiservice.chiangmairedtaxi.test

**Class:**

RegisterActivity

**Unit Test Case 12 (UTC-12):** onCreate(Bundle savedInstanceState)

**Description:**

Register page should be active all the time when Driver click on “register” button in login page . Testing Register page is active when invoke the method onCreate(Bundle savedInstanceState).

**Prerequisites:**

ReportFunctionCallResult(String functionName, boolean functionCallResult)

**Test data:**

**-**

**Test cases:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description** | **Input** | **Assertion** | **Expected Result** |
| 1 | Testing Register page is active when invoke the method onCreate(Bundle savedInstanceState). | - | assertEquals(testclass.callResult,"onCreate(Bundle saveInstanceState):Success"); | True |

**Unit Test Case-Passenger side**

**UTC 13-** **onCreate(Bundle savedInstanceState)**

**Package:**

com.taxiservice.chiangmairedtaxi.test

**Class:**

ChatActivity

**Unit Test Case 13 (UTC-13):** onCreate(Bundle savedInstanceState)

**Description:** Chatroom should be active all the time when Passenger clicks to send request to Driver. Test that the chatroom is active when onCreate() method is invoked.

**Prerequisites:**

ReportFunctionCallResult(String functionName, boolean functionCallResult)

**Test data:**

**-**

**Test cases:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description** | **Input** | **Assertion** | **Expected Result** |
| 1 | Test that System that Chat room is active when Oncreate() is invoked | - | assertEquals(tclass.callResult,"onCreate(Bundle saveInstanceState):Success"); | True |

**UTC 14-** **addListenerOnButton()**

**Package:**

com.taxiservice.chiangmairedtaxi.test

**Class:**

ChatActivity

**Unit Test Case 14 (UTC-14):** addListenerOnButton()

**Description:** Chatting process should be active all the time when Passenger clicks to send the chat message to Driver.

**Prerequisites:**

ReportFunctionCallResult(String functionName, boolean functionCallResult)

**Test data:**

**-**

**Test cases:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description** | **Input** | **Assertion** | **Expected Result** |
| 1 | Test that System that Chat room is active when Oncreate() is invoked | - | assertEquals(tclass.callResult,"onCreate(Bundle saveInstanceState):Success"); | True |

**UTC 15-** **onCreate(Bundle savedInstanceState)**

**Package:**

com.taxiservice.chiangmairedtaxi.test

**Class:**

MainActivity

**Unit Test Case 15 (UTC-15):** onCreate(Bundle savedInstanceState)

**Description:** Login page should be active all the time when Passenger start the application . Test that login page is active when the method invoke onCreate(Bundle savedInstanceState).

**Prerequisites:**

ReportFunctionCallResult(String functionName, boolean functionCallResult)

**Test data:**

**-**

**Test cases:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description** | **Input** | **Assertion** | **Expected Result** |
| 1 | Test that System that log in page is active when onCreate(Bundle savedInstanceState).is invoked | - | assertEquals(testclass.callResult,"onCreate(Bundle saveInstanceState):Success"); | True |

**UTC 16-** **onCreate(Bundle savedInstanceState)**

**Package:**

com.taxiservice.chiangmairedtaxi.test

**Class:**

MylocatemapActivity

**Unit Test Case 16 (UTC-16):** onCreate(Bundle savedInstanceState)

**Description:** Search page should be active when Passenger login to the system successfully.Test the Search page is active when invoke method onCreate(Bundle savedInstanceState).

**Prerequisites:**

ReportFunctionCallResult(String functionName, boolean functionCallResult)

**Test data:**

**-**

**Test cases:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description** | **Input** | **Assertion** | **Expected Result** |
| 1 | Test that Search page is active when invoke onCreate(Bundle savedInstanceState) | - | assertEquals(tclass.callResult,"onCreate(Bundle saveInstanceState):Success"); | True |

**UTC 17- showOnMap(double lat,double lng)**

**Package:**

com.taxiservice.chiangmairedtaxi.test

**Class:**

MylocatemapActivity

**Unit Test Case 17 (UTC-17):** showOnMap(double lat,double lng)

**Description:** The map should be active when Passenger access to the search page.The map should be active when invoke the method showOnMap(double lat,double lng)**.**

**Prerequisites:**

ReportFunctionCallResult(String functionName, boolean functionCallResult)

**Test data:**

**-**

**Test cases:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description** | **Input** | **Assertion** | **Expected Result** |
| 1 | Test that the map is active when invoke method showOnMap(double lat,double lng) | 18.767925361556415, 98.96840095521543 | assertEquals(tclass.callResult,"showOnMap(double lat,double lng):Success"); | True |

**UTC 18- createDestSpinner()**

**Package:**

com.taxiservice.chiangmairedtaxi.test

**Class:**

MylocatemapActivity

**Unit Test Case 18 (UTC-18):** createDestSpinner()

**Description:** Destination spinner (drop down list) should be active when Passenger is in search page. Testing Destination spinner is active when invoke the method createDestSpinner().

**Prerequisites:**

ReportFunctionCallResult(String functionName, boolean functionCallResult)

**Test data:**

**-**

**Test cases:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description** | **Input** | **Assertion** | **Expected Result** |
| 1 | Testing Destination spinner is active when invoke the method createDestSpinner(). | - | assertEquals(tclass.callResult,"createDestSpinner():Success"); | True |

**UTC 19- addListenerOnButton()**

**Package:**

com.taxiservice.chiangmairedtaxi.test

**Class:**

MylocatemapActivity

**Unit Test Case 19 (UTC-19):** addListenerOnButton()

**Description:** Updating driving information should be active when Passenger click on the OK button ,after inputting condition. Testing Updating driving information is active when invoke the method addListenerOnButton()

**Prerequisites:**

ReportFunctionCallResult(String functionName, boolean functionCallResult)

**Test data:**

**-**

**Test cases:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description** | **Input** | **Assertion** | **Expected Result** |
| 1 | Testing Updating driving information is active when invoke the method addListenerOnButton() | - | assertEquals(tclass.callResult,"addListenerOnButton():Success"); | True |

**UTC 20- onCreate(Bundle savedInstanceState)**

**Package:**

com.taxiservice.chiangmairedtaxi.test

**Class:**

mysearchActivity

**Unit Test Case 20 (UTC-20):** addListenerOnButton()

**Description:** Search result page should be active when Passenger search for the red taxi . Testing Search result page is active when invoke onCreate(Bundle savedInstanceState)**.**

**Prerequisites:**

ReportFunctionCallResult(String functionName, boolean functionCallResult)

**Test data:**

**-**

**Test cases:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description** | **Input** | **Assertion** | **Expected Result** |
| 1 | . Testing Search result page is active when invoke onCreate(Bundle savedInstanceState). | - | assertEquals(testclass.callResult,"onCreate(Bundle saveInstanceState):Success"); | True |

**UTC 21- createDestListView(ListView view,String url,String type)**

**Package:**

com.taxiservice.chiangmairedtaxi.test

**Class:**

mysearchActivity

**Unit Test Case 21 (UTC-21):** createDestListView(ListView view,String url,String type)

**Description:** Taxi result list should be active when Passenger click search for the taxu .Testing Taxi result list is active when invoke createDestListView(ListView view,String url,String type)

**Prerequisites:**

ReportFunctionCallResult(String functionName, boolean functionCallResult)

**Test data:**

**-**

**Test cases:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description** | **Input** | **Assertion** | **Expected Result** |
| 1 | Testing Taxi result list is active when invoke createDestListView(ListView view,String url,String type) | - | assertEquals(tclass.callResult,"createDestListView(ListView view,String url):Success"); | true |

**UTC 22- testupdateChatStatus()**

**Package:**

com.taxiservice.chiangmairedtaxi.test

**Class:**

mysearchActivity

**Unit Test Case 22 (UTC-22):** testupdateChatStatus()

**Description:** updating Chat status should be active when Passenger click accept the passenger request Testing updating Chat status is active when invoke testupdateChatStatus()

**Prerequisites:**

ReportFunctionCallResult(String functionName, boolean functionCallResult)

**Test data:**

**-**

**Test cases:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description** | **Input** | **Assertion** | **Expected Result** |
| 1 | Testing updating Chat status is active when invoke testupdateChatStatus() | - | assertEquals(tclass.callResult,"updateChatStatus(String request\_status):Success"); | True |

**UTC 23-** **onCreate(Bundle savedInstanceState)**

**Package:**

com.taxiservice.chiangmairedtaxi.test

**Class:**

RegisterActivity

**Unit Test Case 23 (UTC-23):** onCreate(Bundle savedInstanceState)

**Description:**

Register page should be active all the time when Passenger click on “register” button in login page . Testing Register page is active when invoke the method onCreate(Bundle savedInstanceState).

**Prerequisites:**

ReportFunctionCallResult(String functionName, boolean functionCallResult)

**Test data:**

**-**

**Test cases:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description** | **Input** | **Assertion** | **Expected Result** |
| 1 | Testing Register page is active when invoke the method onCreate(Bundle savedInstanceState). | - | assertEquals(testclass.callResult,"onCreate(Bundle saveInstanceState):Success"); | True |

**System Test Case**

**System Test Case 1(STC-01): Passenger can register to the system.**

**Description** This Test case can check that Passenger can register to system.

**Prerequisites or Test input**

The test data is available in Appendix A.

**Test Data**

The test data is available in Appendix B.

**Test script:**

1.Passenger can open an application.

2.Passenger fills information for register information.

3.Passenger clicks confirm button.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case No.** | **Description** | **Input** | **Expected Result** |
| 1 | Test that data valid in database after registration | Use Test input no.3 | The page display message “Registration Complete” and button to redirected to login page |
| 2 | Test that blank data | pass\_mobile=null  pass\_password=null | Display error message “Enter your…” |

**System Test Case 2(STC-02): Passenger can log in to the system.**

**Description** This Test case is for check that Passenger can log in to the system.

**Prerequisites or Test input**

The test data is available in Appendix A.

**Test Data**

The test data is available in Appendix B.

**Test script:**

1.Passenger fills mobile and password.

2.Passenger clicks confirm button.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case No.** | **Description** | **Input** | **Expected Result** |
| 1 | Test that data is valid in database | pass\_mobile=0859796651  pass\_password=1234 | Displays “welcome+name” and re-status of passenger into “online” |
| 2 | Test that data invalid in database | pass\_mobile=999  pass\_password=Rose | Display error message “This username and password is wrong” |
| 3 | Test that blank data | pass\_mobile=null  pass\_password=null | Display error message “This username and password is wrong” |

**System Test Case 3(STC-03): Passenger can log out from the system.**

**Description** This Test case for check that Passenger can log out from the system after log in to the system.

**Prerequisites or Test input**

The test data is available in Appendix A.

**Test Data**

The test data is available in Appendix B.

**Test script:**

1. Passenger select log out from option menu

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case No.** | **Description** | **Input** | **Expected Result** |
| 1 | Test that passenger can log out by select log out from option menu | log out | Redirect to log in page. |

**System Test Case 4(STC-04): Passenger can search for the red taxi.**

**Description** This test case is for check matching condition between User and system by number of seats and destination.

**Prerequisites or Test input**

The test data is available in Appendix A.

**Test Data**

The test data is available in Appendix B.

**Test script:**

1.Passenger input seat and select list of destination.

2.Passenger confirm to search for red taxi.

3.Passenger can get list of red taxi.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case No.** | **Description** | **Input** | **Expected Result** |
| 1 | Test that valid input data | seat=3  destination=CMU | Display the list of 10 red taxi |
| 2 | Test that none of red taxi that match the search conditions. | seat= 9  destination=CMU | Not display list of red taxi |
| 3 | Test that blank input data | seat= null  destination=null | Not display list of red taxi |

**System Test Case 5(STC-05): Passengers can send a request for taxi.**

**Description**  This test case is for check the Passenger send request to Driver.

**Prerequisites or Test input**

The test data is available in Appendix A.

**Test Data**

The test data is available in Appendix B.

**Test script:**

1.Passenger selects red taxi from the list.

2.Passenger send request for red taxi

3.Passenger confirm list.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case No.** | **Description** | **Input** | **Expected Result** |
| 1 | Test that passenger confirm sending request to Driver. | Confirm=Yes | Display the Chat page and request status. |
| 2 | Test that passenger reject sending request to Driver. | Confirm=No | Display search result page. And no request data in database. |
| 3 | Test that Passenger can select the driver from the Driver list | - | Display a confirmation popup with Driver information. |

**System Test Case 6(STC-06): Passenger can chat with driver.**

**Description** The test case is check for Passenger can chat with driver

**Prerequisites or Test input**

The test data is available in Appendix A.

**Test Data**

The test data is available in Appendix B.

**Test script:**

1.Passenger selects list of red taxi.

2.Passenger click chat with driver.

3.Passenger click send message to driver.

4.Passenger confirms send message.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case No.** | **Description** | **Input** | **Expected Result** |
| 1 | Test that passenger can send message | send\_msg= “LLL” | Driver get message “LLL” |

**System Test Case 7(STC-07): Driver can register to the system.**

**Description** This Test case can check that driver can register to system.

**Prerequisites or Test input**

The test data is available in Appendix A.

**Test Data**

The test data is available in Appendix B.

**Test script:**

1.Driver opens an application.

2.Driver fills information for register information.

3.Driver clicks confirm button.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case No.** | **Description** | **Input** | **Expected Result** |
| 1 | Test that data valid in database | Use Test input no.2 | the page display message “Registration Complete” and button to redirected to login page |
| 2 | Test that blank data in database | driver\_id=null  driver\_name=null  driver\_idcard=null  driver\_mobile=null  driver\_password=null  driver\_photo=null  driver\_regdate=null | Display error message “Enter your…” |

**System Test Case 8(STC-08) : Driver can log in to the system.**

**Description** This Test case is for check that driver can log in to the system.

**Prerequisites or Test input**

The test data is available in Appendix A.

**Test Data**

The test data is available in Appendix B.

**Test script:**

1.Driver fills mobile and password.

2.Driver clicks confirm button.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test No.** | **Description** | **Input** | **Expected Result** |
| 1 | Test that data is valid in database | driver\_mobile=222  driver\_password=222 | Displays “welcome+name” and re-status of passenger into “online” |
| 2 | Test that data invalid in database | driver\_mobile=0  driver\_password=0 | Display error message “This username and password is wrong” |
| 3 | Test that blank data in database | driver\_mobile=null  driver\_password=null | Display error message “This username and password is wrong” |

**System Test Case 09(STC-09): Drivers can log out from the system.**

**Description** This Test case for check that drivers can log out from the system after log in to the system.

**Prerequisites or Test input**

The test data is available in Appendix A.

**Test Data**

The test data is available in Appendix B.

**Test script:**

1.Driver select log out from option menu

|  |  |  |  |
| --- | --- | --- | --- |
| **Test No.** | **Description** | **Input** | **Expected Result** |
| 1 | Test that driver can log out by select log out from option menu | log out | Redirect to log in page. |

**System Test Case 10(STC-10): Drivers can update driving information.**

**Description**  This Test case for check that driver can update the driving information.

**Prerequisites or Test input**

The test data is available in Appendix A.

**Test Data**

The test data is available in Appendix B.

**Test script:**

1.Drivers log in.

2.Driver input information.

3.Driver clicks update information.

4.Driver clicks confirm.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test No.** | **Description** | **Input** | **Expected Result** |
| 1 | Test that input valid data | Use Test input no.4 | Display the request page |

**System Test Case 11(STC-11): Drivers respond to Passenger’s request.**

**Description** This Test case is check that the drivers can respond the passenger request.

**Prerequisites or Test input**

The test data is available in Appendix A.

**Test Data**

The test data is available in Appendix B.

**Test script:**

1.Driver selects Passenger’s request message.

2.Driver respond Passenger’s request message by click accept or decline the request

|  |  |  |  |
| --- | --- | --- | --- |
| **Test No.** | **Description** | **Input** | **Expected Result** |
| 1 | Test that Passenger can get response from driver . | - | Show accepted request on passenger “Accepted by driver” |
| 2 | Test that driver can accept the request respond | Confirm = yes | Show accepted request on driver “Accepted by driver” |
| 3 | Test that driver can decline the request respond | Confirm = no | Display the updated request page without the rejected request . |

**System Test Case 12(STC-12): Driver can chat with passenger.**

**Description** This Test case is check that the drivers can chat with passenger.

**Prerequisites or Test input**

The test data is available in Appendix A.

**Test Data**

The test data is available in Appendix B.

**Test script:**

1.Driver selects Passenger’s chat message.

2.Driver chat with Passenger.

3.Driver clicks send message to passenger.

4.Driver confirms send message.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case No.** | **Description** | **Input** | **Expected Result** |
| 1 | Test that driver can send message | send\_msg= “PPP” | Passenger get message “PPP” |

**System Test Case 13(STC-13): Administrator can login to the Administration system.**

**Description** This Test case is check that the administrator can log in.

**Prerequisites or Test input**

The test data is available in Appendix A.

**Test Data**

The test data is available in Appendix B.

**Test script:**

1.Administrator open administration system

2.Administrator fills username and password

3.Administrator confirms log in.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case No.** | **Description** | **Input** | **Expected Result** |
| 1 | Test that data is valid in database | username =admin  password=admin | User is redirected to the main page |
| 2 | Test that data invalid in database | username =ss  password=ss | Redirect back to log in page |
| 3 | Test that blank data in database | username =null  password=null | Redirect back to log in page |

**System Test Case 14(STC-14): Administrator can log out.**

**Description** This Test case is check that the administrator can log out.

**Prerequisites or Test input**

The test data is available in Appendix A.

**Test Data**

The test data is available in Appendix B.

**Test script:**

1.Administrator clicks log out button.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case No.** | **Description** | **Input** | **Expected Result** |
| 1 | Test that administrator can log out by click log out button | log out or exit | Redirect to log in page. |

**System Test Case 15(STC-15): Administrator can add destinations.**

**Description** This Test case is check that the Administrator can add destinations.

**Prerequisites or Test input**

The test data is available in Appendix A.

**Test Data**

The test data is available in Appendix B.

**Test script:**

1.Administrator log in to the system.

2.Admimistrator add destination.

3.Admistrator confirms add destination by clicks submit button.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case No.** | **Description** | **Input** | **Expected Result** |
| 1 | Test that data is valid in database | Destination=Doi Su Thep  Picture= “doisuthep.jpg”  Nearby1=Chiang Mai zoo  Nearby2=CMU  Information= “AAA”  lat=68.09434  lng=55.93366 | User is redirected to the main page |
| 2 | Test that data invalid in database | Destination=&\*&%  Picture=^%&)  Nearby1=&^\*  Nearby2=%$^&  Information=\*(&  lat=68.09434  lng=55.93366 | Redirect back to log in page |
| 3 | Test that blank data in database | Destination=null  Picture=null  Nearby1=null  Nearby2=null  Information=null  lat=null  lng=null | Redirect back to log in page |

**System Test Case 16(STC-16): Administrator can browse the destination.**

**Description** This Test case is check that the Administrator can browse the destination

**Prerequisites or Test input**

The test data is available in Appendix A.

**Test Data**

The test data is available in Appendix B.

**Test script:**

1.Administrator log in to the system.

2.Admimistrator browse menu of destination.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case No.** | **Description** | **Input** | **Expected Result** |
| 1 | Test that can select destination at browse menu | select “Destinations” | Redirect to destination page |
| 2 | Test that can select clear at browse menu | select “ Clear data” | Redirect to database page |

**System Test Case 17(STC-17): Administrator can edit destinations.**

**Description** This Test case is check that the Administrator can edit destinations.

**Prerequisites or Test input**

The test data is available in Appendix A.

**Test Data**

The test data is available in Appendix B.

**Test script:**

1.Administrator log in to the system.

2.Admimistrator edit destination.

3.Admistrator confirms edit destination.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case No.** | **Description** | **Input** | **Expected Result** |
| 1 | Test that data is valid in database | Destination=Doi Su Thep  Picture= “doisuthep2.jpg”  Nearby1=Nimman  Nearby2=CMU  Information= “AAAbbbnnn”  lat=68.09434666677  lng=55.9336688888 | User is redirected to the main page |
| 2 | Test that data invalid in database | Destination=&\*&%  Picture=^%&)  Nearby1=&^\*  Nearby2=%$^&  Information=\*(&  lat=68.09434  lng=55.93366 | Redirect back to log in page |
| 3 | Test that blank data in database | Destination=null  Picture=null  Nearby1=null  Nearby2=null  Information=null  lat=null  lng=null | Redirect back to log in page |

**System Test Case 18(STC-18): Administrator can delete destinations.**

**Description** This Test case is check that the Administrator can delete destinations.

**Prerequisites or Test input**

The test data is available in Appendix A.

**Test Data**

The test data is available in Appendix B.

**Test script:**

1.Administrator log in to the system.

2.Admimistrator delete destination.

3.Admistrator confirms edit destination.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case No.** | **Description** | **Input** | **Expected Result** |
| 1 | Test that data can deleted from database | select “Ping River” | Redirect to main destination page |

**System Test Case 19(STC-19): Administrator can search destinations.**

**Description** This Test case is check that the Administrator can search destinations.

**Prerequisites or Test input**

The test data is available in Appendix A.

**Test Data**

The test data is available in Appendix B.

**Test script:**

1.Administrator log in to the system.

2.Admimistrator search destination.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case No.** | **Description** | **Input** | **Expected Result** |
| 1 | Test that data is valid in database | key=Nimman  from=Destinaion | Display “Nimman” |
| 2 | Test that another data is valid in database | key=Nimman  from=nearby1 | Display “not found” |
| 3 | Test that data invalid in database | key=\*(&)(  from=nearby2 | Display error message “not found” and redirect back to main page |
| 4 | Test that blank data in database | key=null  from=null | Display error message “not found” and redirect back to main page |

**System Test Case 20(STC-20): Administrator can clear data.**

**Description** This Test case is check that the Administrator can clear data.

**Prerequisites or Test input**

The test data is available in Appendix A.

**Test Data**

The test data is available in Appendix B.

**Test script:**

1.Administrator log in to the system.

2.Admimistrator clear data of database.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case No.** | **Description** | **Input** | **Expected Result** |
| 1 | Test that database table can be clear by deleted | select “trequest\_msg” | Redirect to main database page |

**Appendix A. Test Input**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Input** | **Examples** | **Notes** |
| 1 | localhost  theredta\_usr  theredta\_pwd | theredtaxi  mel  321 | -length: 1-20 characters  -type : alphabet or number |
| 2 | driver\_id  driver\_name  driver\_idcard  driver\_mobile  driver\_password  driver\_photo  driver\_regdate | 140522034348  jay  1213  222  222  null  2014-05-22 15:43:48 | -length: 1-20 characters  -type : alphabet or number or date |
| 3 | pass\_id  pass\_name  pass\_idcard  pass\_mobile  pass\_password  pass\_photo  pass\_regdate | 140602012236  brooke  1555  888  999  null  2014-06-02 01:22:36 | -length: 1-20 characters  -type : alphabet or number or date |
| 4 | driver\_lat  driver\_lng  driver\_date  driver\_status  available\_seat  last\_destination | 23.94827394302  12.9479034345  2014-05-23 03:31:06  1  4  wat pra sing | -length: 1-20 characters  -type : alphabet or number or date  -status : 1= available, 0= not available |
| 5 | pass\_lat  pass\_lng  pass\_date  pass\_status  pass\_seat  driver\_id | 18.789553333333334  98.96634666666665  2014-07-28 15:24:49  1  5  null | -length: 1-20 characters  -type : alphabet or number or date  -status : 1= available, 0= not available |
| 6 | send\_date  send\_from  send\_msg | 2014-07-28 21:12:00  140728011812  >< | -length: 1-40 characters  -type : alphabet or number or date |

**Appendix B. Database**

**Passenger Information - tpassenger**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **pass\_id** | **pass\_name** | **pass\_idcard** | **pass\_mobile** | **pass\_password** | **pass\_photo** | **pass\_regdate** |
| 1527434 | balii saa | 12009493 | null | 874734 | null | 0000-00-00 00:00:00 |
| 123456789456 | Dan Klin | 3609900496665 | 0859796651 | 1234 | null | 2014-05-30 00:00:00 |
| 140602012236 | brooke | 1555 | 888 | 999 | null | 2014-06-02 01:22:36 |
| 140616013549 | nornae | 55 | 55 | 66 | null | 2014-06-16 01:35:49 |

**Passenger Information – tpassenger\_position**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **pass\_ id** | **pass\_ name** | **pass\_ lat** | **pass\_ lng** | **pass\_ date** | **pass\_ seat** | **pass\_ dest** | **pass\_ status** | **driver\_ id** |
| 984723 | Adam lav | 39.9845823223434 | 99.1243434664345 | 2014-05-23 03:31:06 | 4 | wat pra sing | null | null |
| 140602012236 | brooke | 18.789553333333334 | 98.96634666666665 | 2014-07-28 15:24:49 | 5 | 140609032237 | 1 | null |
| 140616013549 | nornae | 18.67886 | 98.921585 | 2014-07-03 01:22:26 | 5 | 140609032237 | 1 | null |

**Driver Information - tdriver**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **driver\_id** | **driver\_name** | **driver\_idcard** | **driver\_mobile** | **driver\_password** | **driver\_photo** | **drivrt\_regdate** |
| 140529051946 | kiki | 0123 | 111 | 111 | null | 2014-05-29 05:19:46 |
| 140602035531 | park | 123 | 77 | 11 | null | 2014-06-02 15:55:31 |
| 140616012707 | mel | 123 | 123 | 321 | null | 2014-06-16 01:27:07 |
| 140522030100 | Rose | 5555555 | 9999000 | 1234 | null | 2014-05-22 15:01:00 |
| 140522034348 | jay | 1213 | 222 | 222 | null | 2014-05-22 15:43:48 |

**Driver Information – tdriver\_position**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **driver\_id** | **driver\_name** | **driver\_lat** | **driver\_lng** | **driver\_date** | **driver\_status** | **aviable\_seat** | **last\_destination** |
| 2302323 | lalisa | 23.94827394302 | 12.9479034345 | 2014-05-03 07:06:45 | 1 | 4 | wat pra sing |
| 140602035531 | park | 18.67886 | 98.921585 | 2014-06-09 15:56:27 | 1 | 10 | 140609032237 |
| 140529051946 | kiki | 18.797573 | 98.966533 | 2014-07-28 22:15:02 | 1 | 11 | 140609031901 |
| 140616012707 | mel | 18.8071766666665 | 98.956395 | 2014-07-28 17:34:11 | 1 | 10 | 140609032237 |

**Destination - tdestination**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **destinationid** | **destinationtitle** | **nearby1** | **nearby2** | **latitude** | **longitude** | **Information** | **picture** |
| 140510043030 | Thaphae Gate | 140519040517 | 140519050320 | 8.78801743704621 | 98.9937686920166 | ประตูท่าแพสถานบันเทิง สำหรับชาวต่างชาติ | 1309799402.jpg |
| 140609032237 | Chiang Mai International Airport | 140510052640 | 140609031954 | 18.767925361556415 | 98.96840095521543 | Chiang Mai is a major gateway to Northern Thailand | chiangmaiairport.jpg |
| 140519040517 | Ping River | 140510052640 | 140519050320 | 18.791145815560682 | 99.00191783906553 | Go to Ping River | Ping\_river-1.JPG |

**Menu -tmenu**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **smenu id** | **smenu name** | **sorder by** | **smenu link** | **csec\_ dsc** | **nsec\_ dsc** | **csec\_ edt** | **nsec\_ edt** | **csec\_ del** | **nsec\_ del** | **cuse** |
| 01 | Destinations | 01 | placelst.php | 1 | 1 | 1 | 2 | 1 | 3 | 1 |
| 02 | Clear Data | 02 | cleardatalst.php | 1 | 4 | 0 | 5 | 1 | 6 | 1 |

**Request – trequest**

|  |  |  |  |
| --- | --- | --- | --- |
| **driver\_id** | **passenger\_id** | **request\_date** | **request\_status** |
| 140725123701 | 140728011914 | 2014-07-28 20:27:07 | 3 |
| 140725123701 | 140602012236 | 2014-07-28 20:34:45 | 3 |
| 140725123701 | 140728011812 | 2014-07-28 20:58:27 | 1 |

**Request – trequest\_msg**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **driver\_id** | **pass\_id** | **send\_date** | **send\_from** | **send\_msg** |
| 140725123701 | 140728022201 | 2014-07-28 21:12:00 | 140728011812 | >< |

**tuser**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **suserid** | **suserfname** | **suserlname** | **smobile** | **semail** | **susername** | **spassword** | **ssec** | **scomment** | **dusedlast** | **cmenulast** |
| 1 | administrator | site admin | 0891636750 | maumiv @hotmail.com | admin | admin | 111101 | comment | 2014-07-28 14:20:16 | 02 |