**Test Plan**

**Team Collaboration System for Mobility Water Monitoring**

**By**

**Mr. Peerapong Chompootepa 542115044**

**Mr. Worrasete Tansurat 542115056**

**Department of Software Engineering**

**College of Arts, Media and Technology**

**Chiang Mai University**

**Project Advisor**

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

**Ms.Siraprapa Wattanakul**

**Document History**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Document Name** | **Detail** | **Status** | **Date** | **View able** | **Reviewer& Responsible** |
| **TCS-Test Plan-V.0.1.docx** | -Introduction | Draft | 03/02/15 | SW | PC, WT |
| **TCS –Test Plan-V0.2.docx** | -Unit Testing -System Testing | Draft | 10/02/15 | SW | PC, WT |
| **TCS -Test Plan-V0.3.docx** | -Update Unit Testing -Update System Testing | Draft | 19/02/15 | SW | PC, WT |
| **TCS–Test Plan-V0.4.docx** | -Update Unit Testing -Update System Testing  -Update Test Plan and Procedure | Draft | 14/04/15 | SW | PC, WT |
| **TCS–Test Plan-V0.5.docx** | -Update Test Plan and Procedure | Draft | 30/04/15 | SW | PC, WT |
| **TCS–Test Plan-V0.6.docx** | -Update Test Plan | Draft | 05/05/15 | SW | PC, WT |
| **TCS–Test Plan-V0.7.docx** | -Update System testing | Draft | 20/05/15 | SW | PC, WT |
| **TCS–Test Plan-V0.8.docx** | Update Unit Testing | Draft | 15/06/15 | SW | PC, WT |
| **TCS–Test Plan-V1.0.docx** | -Update Test Plan | Release | 29/06/15 | SW | PC, WT |

**\* SW = Ms.Siraprapa Wattanakul**

**\* PC = Mr.Peerapong Chompootepa**

**\* WT = Mr.Worrasete Tansurat**

Table of Contents

[**Chapter 5-1 | Introduction** 4](#_Toc422843350)

[1.1 Objectives 4](#_Toc422843351)

[1.2 Scope 4](#_Toc422843352)

[1.3Acronyms and Definitions 4](#_Toc422843353)

[**Chapter 5-2 | Test** P**lan and Test Procedure** 7](#_Toc422843356)

[2.1 Test Objective 7](#_Toc422843357)

[2.2 Scope of testing 7](#_Toc422843358)

[2.3 Test Duration 7](#_Toc422843359)

[2.4 Test Responsibility 8](#_Toc422843360)

[2.5 Test Strategy 8](#_Toc422843361)

[2.6 Result of Testing 8](#_Toc422843362)

[2.7 Test Environment 8](#_Toc422843363)

[2.7.1 Hardware 8](#_Toc422843364)

[2.7.2 Software 9](#_Toc422843365)

[**Chapter 5-3 | Unit testing** 10](#_Toc422843366)

[3.1 Web Service Part 10](#_Toc422843367)

[3.2 Mobile Part 19](#_Toc422843368)

[**Chapter 5-4 | System testing** 28](#_Toc422843369)

# Chapter 5-1 | Introduction

## 1.1 Objectives

The objectives of the test plan of Team collaboration system for mobility water monitoring 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 Team collaboration system for mobility water monitoring. The system testing covers the user requirements.

## 1.2 Scope

This test plan describes the unit testing activities to find out the defects in the system and describes the system testing activities for testing a completely integrated system to verify that it cover all of user’s requirements.

## 1.3Acronyms and Definitions

### Acronyms

TCS Team collaboration system for mobility water monitoring

SRS Software Requirement Specification

URS User Requirement Specification

SDD Software Design Document

UI User Interface

UTC = Unit Test Case

STC = System Test Case

SW = Siraprapa Wattanakul

PC = Peerapong Chompootepa

WT = Worrasete Tansurat

### 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.

# Chapter 5-2 | Test Plan and Test Procedure

## 2.1 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 that are define in progress 1, must be follow in project plan.

## 2.2 Scope of testing

Team collaboration system for mobility water monitoring will test by white-box and black-box testing techniques that are unit testing and system testing and record the test results in the test record.

## 2.3 Test Duration

|  |  |
| --- | --- |
| **Progress** | **Date and Duration** |
| Progress Report I | **Perform Date:** 5th February 2015 – 15th February 2015 (Unit and System Test) **Duration:** 11 days |
| Progress Report II | **Perform Date:** 15th April 2015 –  1st May 2015 (Unit and System Test) **Duration:** 17 days |
| Progress Report III | **Perform Date:** 29th April 2015 –  15th May 2015 (Unit and System Test) **Duration:** 17 days |

## 2.4 Test Responsibility

|  |  |
| --- | --- |
| **Item** | **Responsibility** |
| Unit test of web application | WT, PC |
| Unit test of mobile application | WT, PC |
| Record unit test of web application | WT, PC |
| Record unit test of mobile application | WT, PC |
| System test of web application | WT, PC |
| System test of mobile application | WT, PC |
| Record system test of web application | WT, PC |
| Record system test of android application | WT, PC |

## 2.5 Test Strategy

Team collaboration system for mobility water monitoring 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. Record the actual testing result.

## 2.6 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.

## 2.7 Test Environment

### 2.7.1 Hardware

* **Computers**
  + HP Probook 4530s
    - Processor: Intel® Core™ i5-2430M CPU @ 2.40GHz 2.40GHz
    - RAM: 8.00 GB
    - Operating System: Windows 7 Ultimate
  + Asus K550J
    - Processor: Intel® Core™ i7-2410M CPU @ 3.5GHz 2.30GHz
    - RAM: 4.00 GB
    - Operating System: Windows 7 Ultimate
* **Mobile phones:** Android Operating System
  + Samsung Galaxy Ace Plus S7500
    - CPU: Quad-core 1.0 GHz Cortex-A7
    - RAM: 1.0 GB
    - Operating System: Android 2.3 Gingerbread

### 2.7.2 Software

* Eclipse
* Google chrome
* Android AD

# Chapter 5-3 | Unit testing

## 3.1 Web Service Part

**3.1.1 Class name for testing**: Authen

**Unit Test Case 1 (UTC-01):** test\_login(): void

**Class name:** Authen\_test **Test Data:** Test Data 1 in appendix A

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Case** | **Description** | **Input** | | **Expected** |
| **Username** | **password** | **Boolean** |
| **1** | Test login in true case with both correct username and password | win@gmail.com | hahaha | True |
| **2** | Test login in false case with correct username and incorrect password | win@gmail.com | 12345 | False |
| **3** | Test login in false case with incorrect username and correct password | earth@gmail.com | hahaha | False |
| **4** | Test login in false case with both incorrect username and password | earth@gmail.com | 12345 | False |

**3.1.2 Class name for testing**: Member

**Unit Test Case 2 (UTC-02):** test\_newRequest(): void

**Class name:** Member\_test **Test Data:** Test Data 1 in appendix A

|  |  |  |  |
| --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Result** |
| **user\_id** | **Size of Array List** |
| 1 | Test gets size of new user requests in array list | Null | 2 |

**Unit Test Case 3 (UTC-03):** test\_newRequestById(): void

**Class name:** Member\_test **Test Data:** Test Data 1 in appendix A

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Result** | | | | | | |
| **Array List** | | | | | | |
| **user\_id** | **user\_id** | **user\_name** | **user\_email** | **user\_pass** | **user\_tell** | **user\_status** | **is\_admin** |
| 1 | Test gets the detail of new user request in array list by team id | 4 | 4 | Pcpenchamp | prp@gmail.com | 1234 | 0811668436 | 0 | 0 |

**3.1.3 Class name for testing**: Team

**Unit Test Case 4 (UTC-04):** test\_getTeam(): void

**Class name:** Team\_test **Test Data:** Test Data 2 in appendix A

|  |  |  |  |
| --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Result** |
| **team\_id** | **Size of Array List** |
| 1 | Test gets size of teams in array list | Null | 3 |

**Unit Test Case 5 (UTC-05):** test\_getTeamById(): void

**Class name:** Team\_test **Test Data:** Test Data 2 in appendix A

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Result** | | |
| **team\_id** | **Array List** | | |
| **team\_id** | **team\_name** | **team\_leader(user\_id)** |
| 1 | Test gets the detail of team in array list by team id | 4 | 4 | Team 021 | 3 |

**Unit Test Case 6 (UTC-06):** test\_newTeam(): **void**

**Class name:** Team\_test **Test Data:** Test Data 2 in appendix A

|  |  |  |  |
| --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Result** |
| **Size of Array List** |
| 1 | Test adds new team to the array list | $data , $add\_data | 2 |

**Unit Test Case 7 (UTC-07):** test\_deleteTeam(): void

**Class name:** Team\_test **Test Data:** Test Data 2 in appendix A

|  |  |  |  |
| --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Result** |
| **team\_id** | **Size of Array List** |
| 1 | Test deletes a team in array list by team id. Array list contains three teams | 4 | 2 |

**3.1.4 Class name for testing**: Project

**Unit Test Case 8 (UTC-08):** test\_getProject(): void

**Class name:** Project\_test **Test Data:** Test Data 3 in appendix A

|  |  |  |  |
| --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Result** |
| **project\_id** | **Size of Array List** |
| 1 | Test gets size of projects in array list | Null | 4 |

**Unit Test Case 9 (UTC-09):** test\_getProjectById(): void

**Class name:** Project\_test **Test Data:** Test Data 3 in appendix A

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Result** | | | | |
| **Array List** | | | | |
| **project\_id** | **project\_id** | **team\_id** | **project\_name** | **project\_desc** | **project\_created\_date** |
| 1 | Test gets the detail of project in array list by project id | 1 | 1 | 1 | SIA1 | Water quality site 1 | 2014-07-22 21:18:10 |

**Unit Test Case 10 (UTC-10):** test\_getProjectAssign(): void

**Class name:** Project\_test **Test Data:** Test Data 4 in appendix A

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Result** | | | | |
| **Array List** | | | | |
| **user\_id** | **location\_id** | **team\_id** | **location\_name** | **user\_id** | **status\_id** |
| 1 | Test gets the detail of assigned project in array list by user id | 2 | 1 | 1 | place1 | 2 | 7 |

**Unit Test Case 11 (UTC-11):** test\_newProject(): void

**Class name:** Project\_test **Test Data:** Test Data 3 in appendix A

|  |  |  |  |
| --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Result** |
| **Size of Array List** |
| 1 | Test adds new project to the array list | $data , $add\_data | 2 |

**Unit Test Case 12 (UTC-12):** test\_deleteProject(): void

**Class name:** Project\_test **Test Data:** Test Data 3 in appendix A

|  |  |  |  |
| --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Result** |
| **project\_id** | **Size of Array List** |
| 1 | Test deletes a project in array list by project id. Array list contains two project | 2 | 1 |

**Unit Test Case 13 (UTC-13):** test\_getWater(): void

**Class name:** Project\_test **Test Data:** Test Data 5 in appendix A

|  |  |  |  |
| --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Result** |
| **project\_id** | **Size of Array List** |
| 1 | Test gets size of water parameters in array list | Null | 2 |

**Unit Test Case 14 (UTC-14):** test\_getWaterById(): void

**Class name:** Project\_test **Test Data:** Test Data 5 in appendix A

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Result** | | | | | | |
| **Array List** | | | | | | |
| **water\_id** | **water\_id** | **project\_id** | **location\_id** | **user\_id** | **water\_rgb** | **water\_parameter** | **water\_status** |
| 1 | Test gets the detail of water parameter in array list by water id | 1 | 1 | 1 | 1 | 2 | 6b3421 | PH | 8 |

**Unit Test Case 15 (UTC-15):** test\_changeWaterStatus(): void

**Class name:** Project\_test **Test Data:** Test Data 5 in appendix A

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Result** | | | | | | |
| **Array List** | | | | | | |
| **water\_status** | **water\_id** | **project\_id** | **location\_id** | **user\_id** | **water\_rgb** | **water\_parameter** | **water\_status** |
| 1 | Test changes the status of water (8 equal to “Finished” and 7 equal to “Recollect”) | 7 | 1 | 1 | 1 | 2 | 6b3421 | PH | 7 |

**Unit Test Case 16 (UTC-16):** test\_getWaterResult(): void

**Class name:** Project\_test **Test Data:** Test Data 6 in appendix A

|  |  |  |  |
| --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Result** |
| **result\_id** | **Size of Array List** |
| 1 | Test gets size of water results in array list | Null | 3 |

**Unit Test Case 17 (UTC-17):** test\_getWaterResultById(): void

**Class name:** Project\_test **Test Data:** Test Data 6 in appendix A

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Result** | | | |
| **Array List** | | | |
| **result\_id** | **result\_id** | **water\_id** | **result\_parameter** | **water\_rgb** |
| 1 | Test gets the detail of test result in array list by result id | 1 | 1 | 1 | 1 | FFFFFF |

## 3.2 Mobile Part

**3.2.1 Class name for testing:** MainActivity

**Unit Test Case 18 (UTC-18):** testLogin(): void

**Class name:** LoginTest **Test Data:** Test Data 1 in appendix A

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Case** | **Description** | **Input** | | **Expected Result** |
| **Username** | **Password** | **Size of Array List** |
| 1 | Test gets size of users in array list | Null | Null | 3 |
| 2 | Test gets size of users in array list with correct username and password | steve@gmail.com | 1234 | 1 |
| 3 | Test gets size of users in array list with correct username and incorrect password | steve@gmail.com | 123456 | 0 |
| 4 | Test gets size of users in array list with incorrect username and correct password | stevesss@gmail.com | 1234 | 0 |
| 5 | Test gets size of users in array list with incorrect username and incorrect password | stevesss@gmail.com | 123456 | 0 |

**Unit Test Case 19 (UTC-19):** getUser(): void

**Class name:** LoginTest **Test Data:** Test Data 1 in appendix A

|  |  |  |  |
| --- | --- | --- | --- |
| **Case** | **Description** | **Input** | |
| **Username** | **Password** |
| 1 | Test gets the detail of user in array list by a username and password | steve@gmail.com | 1234 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Case** | **Expected Result** | | | | | | |
| **Array List** | | | | | | |
| **user\_id** | **user\_name** | **user\_email** | **user\_pass** | **user\_tell** | **user\_status** | **is\_admin** |
| 1 | 2 | Steve Job | steve@gmail.com | 1234 | Null | 1 | 0 |

**3.2.2 Class name for testing:** FragmentMap

**Unit Test Case 20 (UTC-20):** getProject(): void

**Class name:** ProjectTest **Test Data:** Test Data 3 in appendix A

|  |  |  |  |
| --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Result** |
| **project\_id** | **Size of Array List** |
| 1 | Test gets size of projects in array list | Null | 4 |

**Unit Test Case 21 (UTC-21):** getProjectById(): void

**Class name:** ProjectTest **Test Data:** Test Data 3 in appendix A

|  |  |  |
| --- | --- | --- |
| **Case** | **Description** | **Input** |
| **project\_id** |
| 1 | Test gets the detail of project in array list by project id | 1 |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Case** | **Expected Result** | | | | | | | | | |
| **Array List** | | | | | | | | | |
| **project\_id** | **project\_name** | **project\_desc** | **project\_created\_date** | **team\_id** | **team\_name** | **team\_leader (user\_id)** | **leader\_name** | **user\_id** | **user\_name** |
| 1 | 1 | SIA1 | Water quality site 1 | 2014-07-22 21:18:10 | 1 | Apple team | 3 | Tim Cook | 2 | Steve Jobb |

**Unit Test Case 22 (UTC-22):** getProjectAssign(): void

**Class name:** ProjectTest **Test Data:** Test Data 3 in appendix A

|  |  |  |  |
| --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Result** |
| **user\_id** | **Size of Array List** |
| 1 | Test gets size of assigned projects in array list by user id | 2 | 2 |

**3.2.3 Class name for testing:** FragmentMap

**Unit Test Case 23 (UTC-23):** getProjectLocation(): void

**Class name:** ProjectLocationTest **Test Data:** Test Data 4 in appendix A

|  |  |  |  |
| --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Result** |
| **location\_id** | **Size of Array List** |
| 1 | Test gets size of locations in array list | Null | 3 |

**Unit Test Case 24 (UTC-24):** getProjectLocationById(): void

**Class name:** ProjectLocationTest **Test Data:** Test Data 4 in appendix A

|  |  |  |
| --- | --- | --- |
| **Case** | **Description** | **Input** |
| **location\_id** |
| 1 | Test gets the detail of location in array list by location id | 1 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Case** | **Expected Result** | | | | | | | |
| **Array List** | | | | | | | |
| **project\_id** | **project\_name** | **team\_id** | **team\_name** | **team\_leader** | **leader\_name** | **location\_id** | **location\_name** |
| 1 | 1 | SIA 1 | 1 | Apple team | 3 | Tim Cook | 1 | place1 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Case** | **Expected Result** | | | | | | |
| **Array List** | | | | | | |
| **user\_id** | **user\_name** | **status\_id** | **loc\_status\_desc** | **location\_lat** | **location\_long** | **date\_time\_statuschg** |
| 1 | 2 | Steve Job | 7 | Recollect | 18.7894 | 99.0037 | 2015-05-22 15:55:32 |

**Unit Test Case 25 (UTC-25):** getProjectLocationAssign(): void

**Class name:** ProjectLocationTest **Test Data:** Test Data 4 in appendix A

|  |  |  |  |
| --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Result** |
| **user\_id** | **Size of Array List** |
| 1 | Test gets size of assigned locations in array list by user id | 2 | 2 |

**3.2.4 Class name for testing:** FragmentPicture

**Unit Test Case 26 (UTC-26):** test\_getWater(): void

**Class name:** WaterTest **Test Data:** Test Data 5 in appendix A

|  |  |  |  |
| --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Result** |
| **water\_id** | **Size of Array List** |
| 1 | Test gets size of water parameters in array list | Null | 2 |

**Unit Test Case 27 (UTC-27):** getWaterById(): void

**Class name:** WaterTest **Test Data:** Test Data 5 in appendix A

|  |  |  |
| --- | --- | --- |
| **Case** | **Description** | **Input** |
| **water\_id** |
| 1 | Test gets the detail of water parameter in array list by water id | 1 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Case** | **Expected Result** | | | | | | |
| **Array List** | | | | | | |
| **location\_id** | **location\_name** | **project\_id** | **user\_id** | **user\_name** | **water\_image** | **water\_rgb** |
| 1 | 1 | place1 | 1 | 2 | Steve Jobb | IMG\_20150527\_201418.jpg | 6b3421 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Case** | **Expected Result** | | | | | |
| **Array List** | | | | | |
| **water\_parameter** | **water\_id** | **water\_value** | **water\_date** | **water\_status** | **loc\_status\_desc** |
| 1 | PH | 1 | 0.0 | 2015-06-03 21:23:24 | 8 | Finish |

**Unit Test Case 28 (UTC-28):** addWater(): void

**Class name:** WaterTest **Test Data:** Test Data 5 in appendix A

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Case** | **Description** | **Input** | | | | | | | | | | | **Expected Result** |
| **location**  **\_id** | **water**  **\_id** | **project**  **\_id** | **user**  **\_id** | **water\_**  **image** | **water**  **\_rgb** | **water\_**  **parameter** | **water\_**  **value** | **water**  **\_date** | **water\_**  **status** | **loc\_status**  **\_desc** | **Size of Array List** |
| 1 | Test adds new water parameter into array list | 1 | 3 | 1 | 2 | Null | FFFFFF | KM | 0 | Null | 1 | Initial | 3 |

**Unit Test Case 29 (UTC-29):** deleteWater(): void

**Class name:** WaterTest **Test Data:** Test Data 5 in appendix A

|  |  |  |  |
| --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Result** |
| **water\_id** | **Size of Array List** |
| 1 | Test removes water parameter from array list. Array list contains two water parameters | 1 | 1 |

**3.2.5 Class name for testing:** FragmentPicture

**Unit Test Case 30 (UTC-30):** getWaterResult(): void

**Class name:** WaterResultTest **Test Data:** Test Data 6 in appendix A

|  |  |  |  |
| --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Result** |
| **result\_id** | **Size of Array List** |
| 1 | Test gets size of water results in array list | Null | 3 |

**Unit Test Case 31 (UTC-31):** getWaterResultById(): void

**Class name:** WaterResultTest **Test Data:** Test Data 6 in appendix A

|  |  |  |  |
| --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Result** |
| **result\_id** | **Size of Array List** |
| 1 | Test gets size of water result in array list by result id | 1 | 1 |

# Chapter 5-4 | System testing

**Feature 1: Authentication and member management system**

**System Test Case 01 (STC-01):** Administrator, Team leader, and Collector can log in to the system on a web application.

**Description**

Administrator, Team leader, and Collector can login to the system using his e-mail and password.

**Test Script**

1. On login page.

2. User enters the E-mail and password in the text field.

3. User clicks “Login” button.

**Test Data:**

|  |  |
| --- | --- |
| **Username** | **Password** |
| Admin1@gmail.com | ad001 |

**Test Case:**

| **Test no.** | **Description** | **Input** | **Expect output** |
| --- | --- | --- | --- |
| 1 | User inputs valid e-mail and password. | E-mail: “Admin1@gmail.com”  Password: “ad001” | The system allows a user to access the application. |
| 2 | User inputs invalid E-mail and valid password, then clicks “Login” button. | E-mail: “Ad1”  Password: “ad001” | The system shows error message” Invalid e-mail or password” and returns to the login page. |
| 3 | User inputs valid e-mail and invalid password, then clicks “Login” button. | E-mail: “Admin1@gmail.com”  Password: “abcd” | The system shows error message “Invalid e-mail or password” and returns to the login page. |
| 4 | User inputs invalid e-mail and password, then clicks “Login” button. | E-mail: “Ad1”  Password: “abcd” | The system shows error message “Invalid e-mail or password” and returns to the login page. |
| 5 | User inputs empty e-mail and valid password, then clicks “Login” button. | E-mail: “”  Password: “abcd” | The system shows error message “Invalid e-mail or password” and returns to the login page. |
| 6 | User inputs valid e-mail and empty password, then clicks “Login” button. | E-mail: “Ad1@gmail.com”  Password: “” | The system shows error message “Invalid e-mail or password” and returns to the login page. |

**System Test Case 02 (STC-02):** Administrator, Team leader, and Collector can logout from the system on a web application.

**Description**

Administrator, Team leader, and Collector can log out of the system by clicking “log out” menu to destroy the session of user. This case is for logging out on a web application.

**Test Script**

1. User clicks “Logout” menu.

**Test case:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test no.** | **Description** | **Input** | **Expect output** |
| 1 | User wants to logout from the system. | User clicks “Logout” menu. | The system shows the login page and allow user to input E-mail and password. |

**System Test Case 03 (STC-03):** Team leader and Collector can log in to the system on a mobile application.

**Description**

Team leader and collector can login to the system using his E-mail and password on a mobile application.

**Test Script**

1. On the login page.

2. User enters the E-mail and password in the text field.

3. User clicks “Login” button.

**Test Data:**

|  |  |
| --- | --- |
| **E-mail** | **Password** |
| steve@gmail.com | 1234 |

**Test Case:**

| **Test no.** | **Description** | **Input** | **Expect output** |
| --- | --- | --- | --- |
| 1 | User inputs e-mail and password, then clicks “Login” button. | E-mail: “steve@gmail.com”  Password: “1234” | The system shows  Email: “steve@gmail.com”  Password: “⚫⚫⚫⚫” on the screen. |
| 2 | User inputs valid e-mail and password, then clicks “Login” button. | E-mail: “steve@gmail.com”  Password: “1234” | The system allows user to access the application. |
| 3 | User inputs invalid e-mail and valid password, then clicks “Login” button. | E-mail: “sten@gmail.com”  Password: “1234” | The system shows error message “Invalid e-mail or password” and returns to the login page. |
| 4 | User inputs valid e-mail and invalid password, then clicks “Login” button. | E-mail: “steve@gmail.com”  Password: “4321” | The system shows error message “Invalid e-mail or password” and returns to the login page. |
| 5 | User inputs invalid e-mail and password, then clicks “Login” button. | E-mail: “sten@gmail.com”  Password: “4321” | The system shows error message” Invalid e-mail or password” and returns to the login page. |
| 6 | User inputs empty e-mail and valid password, then clicks “Login” button. | E-mail: “”  Password: “1234” | The system shows error message “Invalid e-mail or password” and returns to the login page. |
| 7 | User inputs valid e-mail and empty password, then clicks “Login” button. | E-mail: “steve@gmail.com”  Password: “” | The system shows error message “Invalid e-mail or password” and returns to the login page. |
| 8 | User inputs empty e-mail and password, then clicks “Login” button. | E-mail: “”  Password: “” | The system shows error message “Invalid e-mail or password” and returns to the login page. |

**System Test Case 04 (STC-04):** Team leader and Collector can logout from the system on a mobile application.

**Description**

Team leader and collector can log out of the system by clicking “log out” button to destroy the session of user. This case is for logging out on a mobile application.

**Test Script**

1. User clicks “logout” button.

**Test case:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test no.** | **Description** | **Input** | **Expect output** |
| 1 | User wants to log out form the system. | User clicks “Logout” button. | The system shows the confirmation popup “Do you want to exit?” and also shows “Cancel” button and “Yes” button. |
| 2 | After user clicks “Log out” button, user confirms to log out from the system by click “Yes” button. | User clicks “Yes” button. | The system shows the login page and allow user to input E-mail and password. |
| 3 | After user clicks “Log out” button, user want to cancel to log out from the system by click “Cancel” button. | User clicks “Cancel” button. | The system hides the confirmation popup. |

**System Test Case 05 (STC-05):** New collector can send register request to the administrator on a mobile application requesting to be a member.

**Description**

New collector has to register their information on a mobile application in order to use the system. The user has to provide the name, e-mail, password, and telephone number to register to the system.

**Test Script**

1. On the registration page.

2. User completes all the provided input text boxes.

3. User clicks “Register” button.

**Test Data:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Email** | **Password** | **Tel.** |
| Peerapong Chompootepa | pcchamp@gmail.com | 542115044 | 0811668438 |

**Test case:**

| **Test No.** | **Description** | **Input** | **Expect output** |
| --- | --- | --- | --- |
| 1 | User completes all the provided input text boxes like the test data, then clicks “Register” button. | Name: “Peerapong Chompootepa”  Password: “542115044”  E-mail: “pcchamp@gmail.com”  Tel.: “0811668438” | The system shows the message “Registered Successfully”. |
| 2 | User inputs wrong e-mail format, then clicks “Register” button. | Name: “Peerapong Chompootepa”  Password: “542115044”  E-mail: “pcchamp”  Tel.: “0811668438” | The system shows the error message “E-mail must be an email format. |
| 3 | User inputs wrong format of telephone number, then clicks “Register” button. | Name: “Peerapong Chompootepa”  Password: “542115044”  E-mail: “pcchamp@gmail.com”  Tel.: “sdfghjk” | The system shows the error message shows “Telephone number must be a numeric only”. |
| 4 | User inputs blank data on name field, then clicks “Register” button. | Name: “”  Password: “542115044”  E-mail: “pcchamp@gmail.com”  Tel.: “0811668438” | The system shows the error message shows “Please fill in all required fields”. |
| 5 | User inputs blank data on password field, then clicks “Register” button. | Name: “Peerapong Chompootepa”  Password: “”  E-mail: “pcchamp@gmail.com”  Tel.: “0811668438” | The system shows the error message shows “Please fill in all required fields”. |
| 6 | User inputs blank data on e-mail field, then clicks “Register” button. | Name: “Peerapong Chompootepa”  Password: “542115044”  E-mail: “”  Tel.: “0811668438” | The system shows the error message shows “E-mail must be an email format”. |
| 7 | User inputs blank data on telephone field, then clicks “Register” button. | Name: “Peerapong Chompootepa”  Password: “542115044”  E-mail: “pcchamp@gmail.com”  Tel.: “” | The system shows the error message shows “Telephone number must be a numeric only”. |
| 8 | User want to cancel to register. | User clicks “Close” button | The system hides the popup and shows login page. |

**System Test Case 06 (STC-06):** Administrator can view list of new collector requests sorted by date on a web application.

**Description**

Administrator can view a list of the new collector requests. In the list contains such as a name, email, and telephone number.

**Test Script**

1. User clicks “New Collector Request” menu to go to new collector request page.

**Test case:**

| **Test no.** | **Description** | **Input** | **Expect output** |
| --- | --- | --- | --- |
| 1 | User can view a list of the new collector request. | User clicks “New Collector Request” button. | The system displays list of new account request contains such as a name, email, and telephone number with the “Approve” and “Decline” buttons behind each account. |

**System Test Case 07 (STC-07):** Administrator can search a new collector request by using new collector name on a web application.

**Description**

Administrator can search name of a new collector request that administrator need to find quickly by input the keyword in the search box.

**Test Script**

1. User clicks “New collector request” menu to go to new collector request page.

2. User inputs the keyword in the text box on the top right on new collector request page.

**Test Data:**

|  |
| --- |
| **Search** |
| Worrasete |

**Test case:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test no.** | **Description** | **Input** | **Expect output** |
| 1 | User inputs keyword matching with the name of new collector request in database. | Search: “Worrasete” | The system shows list of new collector request that includes the word “Worrasete” on the screen. |
| 2 | User inputs some keyword matching with the name of new collector request in database. | Search: “Worr” | The system shows list of new collector request that includes the word "Worr" on the screen. |
| 3 | User inputs keyword not matching with the name of new collector request in database. | Search: “Varrased” | The system shows message “No matching records found”. |

**System Test Case 08 (STC-08):** Administrator can select to approve or decline new member on a web application.

**Description**

Administrator can approve the account which request from the collectors by clicking the “Approve” buttons to permit them to the system otherwise administrator can click “Decline” button if it is an unacceptable account.

**Test Script**

1. User clicks “New Collector Request” menu to go to new collector request page.

2. User clicks “Approve” button.

3. User clicks “OK” button.

**Test case:**

| **Test no.** | **Description** | **Input** | **Expect output** |
| --- | --- | --- | --- |
| 1 | User wants to approve the registered account. | User clicks “Approve” button. | The system shows popup asking for confirmation “Do you want to approve?” with “Cancel” and “OK” buttons. |
| 2 | After user clicks “Approve” button, user confirms to approve the registered account. | User clicks “OK” button. | The system removes the name of selected as a member from user interface and the system updates user status in the database. |
| 3 | After user clicks “Approve” button, user cancels to approve the registered account. | User clicks “Cancel” button. | The system hides the popup and redirects to new collector list page. |
| 4 | User wants decline the registered account out of the system. | User clicks “Decline” button. | The system shows popup asking for confirmation “Do you want to decline?” with “Cancel” and “OK” buttons. |
| 5 | After user clicks “Decline” button, user confirms to decline the registered account. | User clicks “OK” button. | The system removes the name of selected as a member from user interface and deletes an account out of the database. |
| 6 | After user clicks “Decline” button, user cancels to decline the registered account. | User clicks “Cancel” button. | The system hides the popup and redirects to new collector list page. |

**System Test Case 09 (STC-09):** Administrator can create a team which contains a team name, team leader and team members on a web application.

**Description**

Administrator can create the team which he has to input the name of team. In addition, administrator has to choose the member into the team and also specify the team leader in the team then clicks save to create a new team to the system.

**Test Script**

1. User clicks “Team” menu to go to team information page.

2. User clicks “New Team” button to go to create team page.

3. User completes all input data.

4. User clicks “Save” button.

**Test data:**

|  |
| --- |
| **Team name** |
| TeamScience2 |

**Test case:**

| **Test no.** | **Description** | **Input** | **Expect output** |
| --- | --- | --- | --- |
| 1 | User completes all steps of input data, then clicks “Save” button. | Team name: “TeamScience2” | The new team will be created in the database. |
| 2 | User completes input data and inputs blank the text box of team name, then clicks “Save” button. | Team name: “” | The system shows error message “Please input team name more than 6 characters”. |
| 3 | User completes to input data but does not choose any member, then clicks “Save” button. | Team name: “TeamScience2” | The system shows error message “Please select members”. |
| 4 | User completes input data but does not choose a team leader, then clicks “Save” button. | Team name: “TeamScience2” | The system shows error message “Please select the team leader”. |
| 5 | User inputs a team name less than 6 characters, then clicks “Save” button. | Team name: “Team1” | The system shows error message “Please input team name more than 6 characters”. |
| 6 | User inputs the duplicate name of team existing in the system, then clicks “Save” button. | Team name: “Team021” | The system shows error message “Team name is already”. |
| 7 | User cancels to create a team. | User clicks “Cancel” button. | The system redirects to the team list page. |

**System Test Case 10 (STC-10):** Administrator and Team leader can view list of team sorted by team name on a web application.

**Description**

Administrator can view a list of all teams that administrator created. Team leader can view list of a team which he is the team leader. The list of team is sorted by name to show on team list page.

**Test Script**

1. User clicks “Team” menu to go to team list page.

**Test case:**

| **Test no.** | **Description** | **Input** | **Expect output** |
| --- | --- | --- | --- |
| 1 | Administrator can view a list of all teams. | Administrator clicks “Team” menu. | The system displays list of all teams with the “View”, “Edit”, and “Delete” buttons behind each row. |
| 2 | Team leader can view list of a teams. | Team leader clicks “Team” menu | The system displays list of a teams that he is a team leader with the “View” and “Edit”, buttons behind each row. |

**System Test Case 11 (STC-11):** Administrator and Team leader can search the team by using team name on a web application.

**Description**

Administrator and Team leader can search a name of team that they need to find quickly by input the keyword in the search box.

**Test Script**

1. User clicks “Team” menu to go to new team list page.

2. User inputs the keyword in the text box on the top right on team list page.

**Test Data:**

|  |
| --- |
| **Search** |
| Apple01 |

**Test case:**

| **Test no.** | **Description** | **Input** | **Expect output** |
| --- | --- | --- | --- |
| 1 | User inputs keyword matching with the name of team in database. | Search: “Apple01” | The system shows list of team that includes the word “Apple01” on the screen. |
| 2 | User inputs some keyword matching with the name of team in database. | Search: “ple” | The system shows list of team that includes the word "ple" on the screen. |
| 3 | User inputs keyword not matching with the name of team in database. | Search: “addle” | The system shows message “No matching records found”. |

**System Test Case 12 (STC-12):** Administrator and Team leader can modify the selected team information on a web application.

**Description**

Administrator can edit team name and change a team leader in the team by selecting from team members in the team. In addition, administrator can add and delete by selecting check boxes. Team leader can do as same as the administrator except changing a team leader.

**Test Script**

1. User clicks “Team” menu to go to team list page.

2. User clicks “Edit” button.

3. If administrator or team leader want to edit team name,

3.1 User inputs a name in the text box.

3.2 User clicks “Save” button.

4. If administrator want to change a team leader,

4.1 Administrator selects a radio button behind the row.

4.2 Administrator clicks “Save” button.

5. If administrator or team leader want to add new team members,

5.1 User selects members by checking on the check boxes.

5.2 User clicks “Save” button.

6. If administrator or team leader want to remove team members,

6.1 User unchecks the check boxes.

6.2 User clicks “Save” button.

**Test data:**

|  |
| --- |
| **Team name** |
| TeamScience2 |

**Test case:**

| **Test no.** | **Description** | **Input** | **Expect output** |
| --- | --- | --- | --- |
| 1 | User completes all step of input data, then clicks “Save” button. | Team name: “TeamScience2” | The new team will be created in the database. |
| 2 | User completes input data but blank the text box of name, then clicks “Save” button. | Team name: “” | The system shows error message “Please input team name more than 6 characters”. |
| 3 | User completes input data but does not choose any member and click “Save” button. | Team name: “TeamScience2” | The system shows error message “Please select members”. |
| 4 | User completes input data but does not choose a team leader. | Team name: “TeamScience2” | The system shows error message “Please select the team leader”. |
| 5 | User inputs a team name less than 6 characters. | Team name: “Team1” | The system shows error message “Please input team name more than 6 characters”. |
| 6 | User cancels to modify a team | User clicks “Cancel” button. | The system redirects to the team list page. |

**System Test Case 13 (STC-13):** Administrator and Team leader can view the selected team information includes a team name, team leader name, and list of members on a web application.

**Description**

User can view team information by clicking the team that user want to view which contains a team name, team leader name and list of team members.

**Test Script**

1. User clicks “Team” menu to go to team list page.

2. User clicks “View” button behind each team at the team list page.

**Test case:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test no.** | **Description** | **Input** | **Expect output** |
| 1 | User can view the selected team information. | User clicks “View” button. | The system shows window of a team details includes a team name, team leader name, and list of members. |

**System Test Case 14 (STC-14):** Administrator can remove a team out of the system on a web application.

**Description**

Administrator can remove a team out of the database by clicking “Delete” button to remove the team from database.

**Test Script**

1. User clicks “Team” menu to go to team information page.

2. User clicks “Delete” button behind each team at the team list page.

3. User clicks “Ok” button.

**Test case:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test no.** | **Description** | **Input** | **Expect output** |
| 1 | User wants to remove a team by click ““Delete” button. | User clicks “Delete” button. | The system shows the confirmation popup “Do you want to delete?” and also shows “Cancel” button and “OK” button. |
| 2 | After user clicks “Delete” button, user confirms to remove the team by click “OK” button. | User clicks “OK” button. | The system removes the team out of the user interface and deletes the team in the database. |
| 3 | After user clicks “Delete” button, user wants to cancel to remove the team by click “Cancel” button. | User clicks “Cancel” button. | The system hides the confirmation popup and redirects to team list page. |

**System Test Case 15 (STC-15):** Administrator, Team leader, and Collector can edit profile information which includes a name, password and telephone number on a web application.

**Description**

Administrator, Team leader, and Collector can update the personal information up to date and manage the wrong information of his profile on web application.

**Test Script**

1. User clicks “Edit Profile” menu to go to edit profile page.

5. User enters new information (i.e. name, password, telephone).

6. User clicks “Save” button.

**Test Data:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Password** | **Confirm password** | **Tel.** |
| Peerapong C | 1234 | 1234 | 0811668436 |

**Test case:**

| **Test No.** | **Description** | **Input** | **Expect output** |
| --- | --- | --- | --- |
| 1 | User completes all steps of input, then clicks “Save” button. | Name: “Peerapong C”  Password: “1234”  Confirm password: “ 1234”  Telephone number: “0811668436” | The system updates the new profile of user in the database. |
| 2 | User inputs blank data on name field, then clicks “save” button. | Name: “”  Password: “1234”  Confirm password: “ 1234”  Telephone number: “0811668436” | The system shows the error message will show “Please input name”. |
| 3 | User inputs blank data on password field, then clicks “Save” button. | Name: “Peerapong C”  Password: “”  Confirm password: “ 1234”  Telephone number: “0811668436” | The system shows the error messages “Password and confirm password not match”. |
| 4 | User inputs blank data on confirm password field, then clicks “Save” button. | Name: “Peerapong C”  Password: “1234”  Confirm password: “”  Telephone number: “0811668436” | The system shows the error messages “Password and confirm password not match”. |
| 5 | User inputs blank data on telephone field, then clicks “Save” button. | Name: “Peerapong C”  Password: “1234”  Confirm password: “ 1234”  Telephone number: “” | The system updates the new profile of user in the database. |
| 6 | User inputs the password does not match with the confirm password. | Name: “Peerapong C”  Password: “1234”  Confirm password: “ 6547”  Telephone number: “0811668436” | The system updates the new profile of user in the database. |
| 7 | User cancels to edit profile. | User clicks “Close” button. | The system redirects to project list page |

**Feature 2: Project management**

**System Test Case 16 (STC-16):** Administrator can create the project includes a project name, project description, and team on a web application.

**Description**

Administrator can create the project which administrator has to input the name of project and project description. In addition, administrator has to select the team to do the project by a combo box.

**Test Script**

1. User clicks “Project” menu to go to project information page.

2. User clicks “New Project” button to go to create project page.

3. User completes the all text boxes and selects the team.

4. User clicks “Save” button.

**Test Data:**

|  |  |  |
| --- | --- | --- |
| **Project name** | **Description** | **Assign Team** |
| AngKaew1 | ChiangMai | TeamA |

**Test case:**

| **Test no.** | **Description** | **Input** | **Expect output** |
| --- | --- | --- | --- |
| 1 | User wants to create the project by click “New Project” button. | User clicks “New Project” button. | The system provides a text box to input the project name, project description and a combo box to select the team to do the project. |
| 2 | User completes the all text boxes and selects the team, then clicks “Save” button. | Project name:“AngKaew1”  Description: “ChiangMai”  Assign Team: “TeamA” | The system creates new project into database and shows list of new project on project list page. |
| 3 | User completes the all text boxes and selects the team but inputs the project name less than 6 characters, then clicks “Save” button. | Project name:“AngK”  Description: “ChiangMai”  Assign Team: “TeamA” | The system shows error message “Please input project name more than 6 characters” on the screen. |
| 4 | User inputs blank the text box of project name, then clicks “Save” button. | Project name:“”  Description: “ChiangMai”  Assign Team: “TeamA” | The system shows error message “Please input project name” on the screen. |
| 5 | User inputs blank the text box of description, then clicks “Save” button. | Project name:“AngKaew1”  Description: “”  Assign Team: “TeamA” | The system creates new project into database and shows list of new project on project list page. |
| 6 | User deselects the team to do the project, then clicks “Save” button. | Project name:“AngKaew1”  Description: “ChiangMai”  Assign Team: “” | The system shows error message “Please select the team” on the screen. |
| 7 | User inputs the duplicate name of project existing in the system, then clicks “Save” button | Project name: “SIA0012”  Description: “ChiangMai”  Assign Team: “TeamA” | The system shows error message “Project name is already”. |
| 8 | User cancels to create the project. | User clicks “Cancel” button. | The system redirects to the project list page. |

**System Test Case 17 (STC-17):** Administrator can remove the project out of the system on a web application.

**Description**

Administrator can remove the project out of the system on a web application.

**Test Script**

1. On project list page.

1. User clicks “Delete” button behind the project that user wants to remove

**Test case:**

| **Test no.** | **Description** | **Input** | **Expect output** |
| --- | --- | --- | --- |
| 1 | User wants to remove the project. | User clicks “Delete” button. | The system shall show popup asking for confirmation “Do you want to delete?” with “OK” and “Cancel” buttons. |
| 2 | After user clicks “Delete” button, user confirms to remove the project by clicks “OK” button. | User clicks “”OK” button. | The system removes the project out of the system and shows the project remaining on project list page. |
| 3 | After user clicks “Delete” button, user wants to cancel to remove the project by clicks “Cancel” button. | User clicks “Cancel” button. | The system hides the popup and redirects to project list page. |

**System Test Case 18 (STC-18):** Administrator can view list of all projects sorted by created date of the project on a web application

**Description**

Administrator can view list of all project created by administrator. The project is sorted by date created to show on the project list page.

**Test Script**

1. User clicks “Project” menu to go to project list page.

**Test case:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test no.** | **Description** | **Input** | **Expect output** |
| 1 | User can view list of all project. | User clicks “Project” menu. | The system displays list of all project with the “View”, “Edit”, and “Delete” buttons behind each row. |

**System Test Case 19 (STC-19):** Team leader and Collector can view list of the involved projects sorted by created date of the project on a web application and mobile application.

**Description**

Team leader and Collector can view list of project assigned by administrator. The project is sorted by date created to show on project list page.

**Test Script**

1. User log in to the system.

**Test case:**

| **Test no.** | **Description** | **Input** | **Expect output** |
| --- | --- | --- | --- |
| 1 | On a web application; team leader wants to view list of project. | Login to the system | The system displays list of involved project list with the “View” and “Edit” buttons behind each row on project list page. |
| 2 | On a web application; collector wants to view list of project. | Login to the system | The system displays list of involved project list with the “View” button behind each row on project list page. |
| 3 | On a mobile application; team leader wants to view list of project. | Login to the system | The system displays list of involved project on project list page. |
| 4 | On a mobile application; collector wants to view list of project. | Login to the system | The system displays list of involved project on project list page. |

**System Test Case 20 (STC-20):** Administrator, Team leader and Collector can search the project by using project name on a web application.

**Description**

Administrator, Team leader and Collector can search the project that they need to find quickly by input the keyword in the search box.

**Test Script**

1. User clicks “Project” menu to go to project information page.

2. User inputs the keyword in the text box.

**Test Data:**

|  |
| --- |
| **Search** |
| Project001 |

**Test case:**

| **Test no.** | **Description** | **Input** | **Expect output** |
| --- | --- | --- | --- |
| 1 | User inputs keyword matching with the name of project in database. | Search: “Project001” | The system shows “Apple01” project on the screen. |
| 2 | User inputs some keyword matching with the name of project in database. | Search: “Project” | The system shows list of project that includes the word "Project" on the screen. |
| 3 | User inputs keyword not matching with the name of project in database. | Search: “Project111” | The system shows message “No matching records found”. |

**System Test Case 21 (STC-21):** Administrator can modify the project information includes editing project name, editing project description, and changing a team on a web application.

**Description**

Administrator can editing project name, editing project description and changing a team on a web application.

**Test Script**

1. User clicks “project” menu to go to project list page.

2. User clicks “Edit” button.

3. If user want to edit project name,

3.1 User inputs a project name in the text box.

3.2 User clicks “Save” button.

4. If user want to edit project description,

4.1 User inputs a project description in the text box.

4.2 User clicks “Save” button.

5. If user want to change the team,

5.1 User selects the team by combo box.

5.2 User clicks “Save” button.

**Test Data:**

|  |  |  |
| --- | --- | --- |
| **Project name** | **Project description** | **Team** |
| AngKaew2 | CMU ChiangMail | TeamB |

**Test case:**

| **Test no.** | **Description** | **Input** | **Expect output** |
| --- | --- | --- | --- |
| 1 | Use wants to create the project by click “New Project” button. | User clicks “New Project” button. | The system provides a text box to input the project name, project description and a combo box to select the team to do the project. |
| 2 | User completes the all text boxes and selects the team, then clicks “Save” button. | Project name:“AngKaew1”  Description: “ChiangMai”  Assign Team: “TeamA” | The system creates new project into database and shoe list of new project on project list page. |
| 3 | User completes the all text boxes and selects the team but inputs the project name less than 6 characters, then clicks “Save” button. | Project name:“AngK”  Description: “ChiangMai”  Assign Team: “TeamA” | The system shows error message “Please input project name more than 6 characters” on the screen. |
| 4 | User inputs blank the text box of project name, then clicks “Save” button. | Project name:“”  Description: “ChiangMai”  Assign Team: “TeamA” | The system shows error message “Please input project name” on the screen. |
| 5 | User inputs blank the text box of description, then clicks “Save” button. | Project name:“AngKaew1”  Description: “”  Assign Team: “TeamA” | The system creates new project into database and show list of new project on project list page. |
| 6 | User deselects the team to do the project, then clicks “Save” button. | Project name:“AngKaew1”  Description: “ChiangMai”  Assign Team: “” | The system shows error message “Please select the team” on the screen. |
| 7 | User cancels to create the project. | User clicks “Cancel” button. | The system redirects to the project list page. |

**System Test Case 22 (STC-22):** Administrator can view detail of selected projects information which consisting of map with assigned pins and location information on a web application.

**Description**

Administrator can view project information by clicking the project he want to view which contains a map and location information (place name, user, latitude, longitude, status, assigned date, and detail) of the selected project on a web application.

**Test Script**

1. User clicks “Project” menu to go to project list page.

2. User clicks “View” button behind each project to view the information.

**Test case:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test no.** | **Description** | **Input** | **Expect output** |
| 1 | User can view the selected project. | User clicks “View” menu. | The system shows project which contains map and location information (place name, user, latitude, longitude, status, assigned date, and detail) of the selected project on project information page. |

**System Test Case 23 (STC-23):** Team leader and Collector can view detail of selected projects information which consisting of map with assigned pins and location information of the selected project on a web application and mobile application.

**Description**

Team leader can view list of all location information of each collector of the selected project that has to assign previously. Collector can view location information of the selected project that assign by team leader. Project information contain a place name, collector name, latitude, longitude, status of collecting (Initial, Pending, Recollect, Finish), and date of update the project. In addition, map is shown together with a pin above the list on a web application and mobile application.

**Test Script**

1. User click “Project” menu to go to project information page.
2. On project information page
   1. On a web application; user clicks “View” button behind each project to view the project information.

2.2 On a mobile application; user selects the project to view the project information.

**Test case:**

| **Test no.** | **Description** | **Input** | **Expect output** |
| --- | --- | --- | --- |
| 1 | On a web application; user can view the selected project that they are a team leader. | User clicks “View” button. | The system retrieves all location information of every collectors and displays project information page that has a map with a pin and a list of the location information consisting of place name, collector name, latitude, longitude, status of collecting(Initial, Pending, Recollect, Finish), and update date. |
| 2 | On a web application; user can view the selected project that they are a collector. | User clicks “View” button. | The system retrieves only location information that was assigned by team leader of selected project and displays project information page that has a map with a pin and a list of the location information consisting of place name, collector name, latitude, longitude, status of collecting(Initial, Pending, Recollect, Finish), and update date. |
| 3 | On a mobile application; user can view the selected project that they are a team leader. | User selects the project. | The system retrieves all location information of every collectors and displays project information page that has a map with a pin and a list of the location information consisting of place name, collector name, latitude, longitude, status of collecting(Initial, Pending, Recollect, Finish), and update date. |
| 4 | On a mobile application; user can view the selected project that they are a collector. | User selects the project. | The system retrieves only location information that was assigned by team leader of selected project and displays project information page that has a map with a pin and a list of the location information consisting of place name, collector name, latitude, longitude, status of collecting(Initial, Pending, Recollect, Finish), and update date. |

**System Test Case 24 (STC-24):** Administrator can view list of water parameter of the selected member sorted by date on a web application.

**Description**

Administrator can view history of water parameter such as PH and NI including parameter name, predicted result, status, and date of update status on parameter list page on a web application.

**Test Script**

1.User clicks “Project” menu to go to project information page.

2. User clicks “View” button behind each project to view the information of project.

**Test case:**

| **Test no.** | **Description** | **Input** | **Expect output** |
| --- | --- | --- | --- |
| 1 | User can view list of water parameter. | User clicks “View” button. | The system shows list of selected parameter including parameter name, predicted result, status, and date of update status on parameter list page. |

**System Test Case 25 (STC-25):** Team leader and Collector can view list of water parameter of the selected member sorted by date on a web application and mobile application.

**Description**

Team leader and Collector can view history of water parameter such as PH and NI including parameter name, predicted result, status, date of update status on parameter list page on a web application and mobile application.

**Test Script**

1.User clicks “Project” menu to go to project list page.

2. On project list page

2.1 On a web application; user clicks “View” button behind each project to view the information of project.

2.2 On a mobile application; user selects project to view the information of project.

3. On parameter information page

3.1 On a web application; user clicks “View” button behind each member to view the list of parameter.

3.2 On a mobile application; user selects member to view the list of parameter.

**Test case:**

| **Test no.** | **Description** | **Input** | **Expect output** |
| --- | --- | --- | --- |
| 1 | On a web application; Team leader can view list of water parameter. | Team leader clicks “View” button behind each member that he want to view on project information page. | The system shows list of parameter list including parameter name, predicted result, status, and date of update status on parameter list page. |
| 2 | On a web application; Collector can view list of water parameter. | Collector clicks “View” button behind assigned location on project information page. | The system shows list of parameter list including parameter name, predicted result, status, and date of update status on parameter list page. |
| 3 | On a mobile application; Team leader can view list of water parameter. | Team leader selects a member behind each member that he want to view on project information page. | The system shows list of parameter list including parameter name, predicted result, status, and date of update status on parameter list page. |
| 4 | On a mobile application; Collector can view list of water parameter. | Collector selects assigned location on project information page. | The system shows list of parameter list including parameter name, predicted result, status, and date of update status on parameter list page. |

**Feature 3: Map location management**

**System Test Case 26 (STC-26):** Team leader can assign work location on Google map to each collector on a web application.

**Description**

Team leader can identify the location for the team members to monitor the water sampling. This method is operated on the web application by the team leader of the project only.

**Test Script**

1. User clicks “Assign” button behind the project that he is a team leader.
2. User clicks “+place” button.
3. User enters the location name and mark the position including selects the team member into that place.
4. User clicks “Save” button.

**Test data:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Place Name** | **Latitude** | **Longitude** | **Assign To** |
| Lampang | 18.785645 | 99.000115 | Worrasete T. |

* **Latitude and Longitude** will acquire from Google maps automatically when user mark the position on Google maps.
* **Assign To.** is a combo box to select the team member, user must selects the team member into that place.

**Test case:**

| **Test no.** | **Description** | **Input** | **Expect output** |
| --- | --- | --- | --- |
| 1 | User completes all the provided input text boxes and do all steps like the test data, then click “Save” button. | **Place Name: “**Lampang”  **Latitude: “**18.785645”  **Longitude: “**99.000115”  **Assign To: “**Worrasete T.” | The system adds the assignment to the team member in the database and displays the new assignment in the list below of Google Maps and also add a new marker on Google maps. |
| 2 | User completes all steps like the test data but inputs blank the name of place, then clicks “Save” button. | **Place Name: “”**  **Latitude: “**18.785645”  **Longitude: “**99.000115”  **Assign To: “**Worrasete T.” | The system displays popup message “Please input place name” on the screen. |
| 3 | User cancels to assign work. | User clicks “Cancel” button. | The system hides the assignment window and marker disappear. |

**System Test Case 27 (STC-27):** Team leader and Collector can view the direction to the selected location on Google Maps on a mobile application.

**Description**

Team leader and Collector can choose to enable or disable the direction of location that collector must to go for collecting the data.

**Test Script**

1. User selected project that collector want to view assignment on project information page.
2. User clicks “ON” button on the top right page.

**Test case:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test no.** | **Description** | **Input** | **Expect output** |
| 1 | User enables the direction of assigned location. | User clicks “ON” button on the top right page. | The system obtains the shortest direction from Google maps and displays the assignment location using a blue line on Google maps. |
| 2 | User disables the direction of assigned location. | User clicks “OFF” button on the top right page. | The system hides the direction on Google maps. |

**System Test Case 28 (STC-28):** Team leader and Collector can view location information of each assigned pin on selected project by pressing on the pin on a mobile application.

**Description**

User can view other places information of the selected project by press on pin, the information consisting of place name, assign to, and update.

**Test Script**

1. On project information page.
2. User press on pin on Google maps.

**Test case:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test no.** | **Description** | **Input** | **Expect output** |
| 1 | User can view information of other location on each pin. | User pressing on pin on Google maps. | The system displays information of selected pin consisting of place name, collector name, and date of update on Google maps. |

**Feature 4: Water parameter calculation**

**System Test Case 29 (STC-29):** Collector can add water parameter to collect the test data on a mobile application.

**Description**

Collector can add a name of water parameter to collect the test data on a mobile application. Example: pH, Ni.

**Test Script**

1. On parameter list page.
2. User clicks “+” button on the top right of the parameter list page.
3. User inputs the name of parameter that collector want to collect.
4. User clicks “Ok” button.

**Test data:**

|  |
| --- |
| **Water parameter** |
| pH |

**Test case:**

| **Test no.** | **Description** | **Input** | **Expect output** |
| --- | --- | --- | --- |
| 1 | User inputs complete the name of parameter, then clicks “Save” button. | Water parameter: “pH” | The system displays “pH” on the screen and creates new parameter shows on the parameter list page. |
| 2 | User inputs blank the name of parameter, then clicks “Save” button. | Water parameter: “” | The system shows error message “Please input parameter more than 1 character” on the screen. |
| 3 | User cancels to create a new parameter by click “Cancel” button. | User clicks “Cancel” button. | The system hides the pop-up for create parameter and redirects to the parameter list page. |

**System Test Case 30 (STC-30):** Collector can delete water parameter on a mobile application.

**Description**

User can delete water parameter to collect the test data on a mobile application. Example: pH, Ni.

**Test Script**

1. On parameter list page
2. User clicks “X” button on the right of the parameter.
3. User clicks “Yes” button.

**Test case:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test no.** | **Description** | **Input** | **Expect output** |
| 1 | User wants to delete a water parameter by click “X” button behind each water parameter. | User clicks “X” button. | The system shows the pop-up “Do you want to delete?” with “Yes” and “Cancel” button |
| 2 | After user clicks “X” button, user confirms to delete a water parameter by click “Yes” button. | User clicks “Yes” button. | The system removes water parameter out of the database and updates the information and also displays the water parameter remaining in the system on parameter list page. |
| 3 | After user clicks “X” button, user cancels to delete a water parameter by click “Cancel” button. | User clicks “Cancel” button. | The system hides the pop-up and shows parameter list page. |

**System Test Case 31 (STC-31):** Collector can choose the image from camera roll or take a new photo to collect RGB value with the test result on a mobile application.

**Description**

Collector can choose the image from camera roll or take a new photo for collecting RGB value to calculate the test result

**Test Script**

1. On parameter information page
2. User clicks the icon for chooses the image from camera roll or take a new photo.

**Test case:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test no.** | **Description** | **Input** | **Expect output** |
| 1 | User can choose the image from camera roll. | User chooses icon to selects the image from camera roll. | The system provides camera roll for selecting the image. |
| 2 | After user chooses icon to selects the image from camera roll, user selects the image on camera roll, then clicks “Save” button. | Selects the image and click “Save” button | The system redirects to parameter information page with the image for collecting RGB value consisting of date and time. |
| 3 | User can take a new photo. | User chooses icon to take a new photo. | The system provides camera window for take a new photo. |
| 4 | After user chooses icon to take a new photo, user takes a new photo, then clicks “Save” button. | Take a new photo and click “Save” button | The system redirects to parameter information page with the image for collecting RGB value. |

**System Test Case 32 (STC-32):** Collector can manage the data to let the system calculate the test result based on standard color scale on a mobile application.

**Description**

For calculation part, collector must input the value of standard color scale and touch on every band of standard color scale to compare with the test indicator. The application will send the data to developed component and returns the test result from chemical analysis to the system on mobile application.

**Test Script**

1. User touches on the first band of standard color scale of water parameter

2. User inputs the value of the first band of standard color scale.

3. User completes every band of standard color scale.

4. User touches the area of testing indicator.

5. User clicks “Mark” button.

5. User clicks "Calculate" button.

6. User clicks "Save" button.

**Test data:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Value of RGB** | **RGB** | **Test result** | **RGB of test indicator** |
| 2 | FF1493 | 6 | FF6875 |
| 4 | FF69B4 |
| 6 | FF6875 |
| 8 | FF7523 |
| 10 | FF8523 |

* RGB obtains from the system when user pressing on band of standard color scale of water parameter

**Test case:**

| **Test no.** | **Description** | **Input** | **Expect output** |
| --- | --- | --- | --- |
| 1 | User completes all step of calculation process, then clicks “Calculate” button. | Value1: “2”  RGB1: “FF1493”  Value2: “4”  RGB2: “FF69B4 ”  Value3: “6”  RGB3: “FF6875”  Value4: “8”  RGB4: “FF7523”  Value5: “10”  RGB5: “FF8523”  ValuleTest: “?”  RGBtest: “FF6875” | The system send the data to calculating component and returns test result from chemical analysis shows on user interface.  Ex. ValuleTest: “6”  RGBtest: “FF6875” |
| 2 | After test no.1, user clicks “Clear” button. | Click “Clear” button | The system removes value of test result in the database and removes out of the user interface.  Ex. ValuleTest: “”  RGBtest: “FF6875” |
| 3 | User touches on the band and inputs blank value, then clicks “+” button. | Value1: “”  RGB1: “FF1493” | The system shows error message “Please input the value”. |
| 4 | User inputs value of band less than 2 value, then clicks “Calculate” button. | Value1: “2”  RGB1: “FF1493” | The system shows error message “Please input the value more than 1 value”. |
| 5 | User completes all step of calculation process but don’t clicks “Mark” button, then clicks “Calculate” button. | Value1: “2”  RGB1: “FF1493”  Value2: “4”  RGB2: “FF69B4 ”  Value3: “6”  RGB3: “FF6875”  Value4: “8”  RGB4: “FF7523”  Value5: “10”  RGB5: “FF8523”  ValuleTest: “”  RGBtest: “” | The system shows error message “Please mark the RGB value”. |
| 6 | User completes all step of calculation process, then clicks “Cancel” button. | Value1: “2”  RGB1: “FF1493”  Value2: “4”  RGB2: “FF69B4 ”  Value3: “6”  RGB3: “FF6875”  Value4: “8”  RGB4: “FF7523”  Value5: “10”  RGB5: “FF8523”  ValuleTest: “”  RGBtest: “FF6875” | The system redirects to parameter list page. |

**Feature 5: Parameter result tracing**

**System Test Case 33 (STC-33):** Administrator can view detail of test result of selected water parameter on a web application.

**Description**

Administrator can view detail of test result of selected parameter consisting name of parameter, predicted result with the RGB value, status and image with data that used to calculate the test result.

**Test Script**

1. On parameter list page.
2. User clicks “View” button to go to parameter information page.

**Test case:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test no.** | **Description** | **Input** | **Expect output** |
| 1 | User can view the test result. | User clicks “View” button. | The system displays detail of water parameter consisting of name of parameter, predicted result with the RGB value, status and image with data that used to calculate the test result on parameter information page. |

**System Test Case 34 (STC-34):** Team leader and Collector can view detail of test result of selected water parameter on a web application and mobile application.

**Description**

Team leader can view detail of test result of selected parameter consisting name of parameter, predicted result with the RGB value, status and image with data that used to calculate the test result and also team leader can review test result in parameter information page. Collector can view detail of test result same with the team leader but collector can view only test result that collector collected.

**Test Script**

1. On parameter list page.
   1. On a web application; user clicks “View” button to go to parameter information page.
   2. On a mobile application; user selects the water parameter to go to parameter information page.

**Test case:**

| **Test no.** | **Description** | **Input** | **Expect output** |
| --- | --- | --- | --- |
| 1 | On a web application; Team leader can view the test result. | Team leader clicks “View” button. | The system displays information of test result consisting of name of parameter, predicted result with the RGB value, status and image with data that used to calculate the test result with the “Recollect” and “Finish” buttons for review the test result on parameter information page. |
| 2 | On a mobile application; Team leader can view the test result. | Team leader selects water parameter. | The system displays information of test result consisting of name of parameter, predicted result with the RGB value, status and image with data that used to calculate the test result with the “Recollect” and “Finish” buttons for review the test result on parameter information page. |
| 3 | On a web application; Collector can view the test result. | Collector clicks “View” button. | The system displays information of test result consisting of name of parameter, predicted result with the RGB value, status and image with data that used to calculate the test result on parameter information page. |
| 4 | On a mobile application; Collector can view the test result. | Collector selects water parameter. | The system displays information of test result consisting of name of parameter, predicted result with the RGB value, status and image with data that used to calculate the test result on parameter information page. |

**System Test Case 35 (STC-35):** Team leader can mark status of the selected water parameter to “Recollect” on a web application and mobile application.

**Description**

Team leader can mark for recollecting on water parameter on a mobile application when team leader disagree to the test result and want collector to test again.

**Test Script**

1. On a parameter information page.
2. User clicks “Recollect” button.

**Test case:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test no.** | **Description** | **Input** | **Expect output** |
| 1 | User updates status for recollecting on test result. | User clicks “Recollect” button. | The system redirects to parameter list page and change the status in database, then displays status “Recollect” on the screen. |
| 2 | User cancels to update status for recollecting. | User clicks “Cancel” button. | The system redirects to parameter list page. |

**System Test Case 36 (STC-36):** Team leader can mark status of the selected water parameter to “Finish” on a web application and mobile application.

**Description**

Team leader can mark to finish on water parameters on a mobile application when team leader agree with the test result.

**Test Script**

1. On parameter information page.
2. Team leader clicks “Finish” button.

**Test case:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test no.** | **Description** | **Input** | **Expect output** |
| 1 | User updates status to finish on test result. | User clicks “Finish” button. | The system redirects to parameter list page and change the status in database, then displays status “Finish” on the screen. |
| 2 | User wants to cancel to update status for finishing. | User clicks “Cancel” button. | The system redirects to parameter list page. |

**Feature 6: Messaging system**

**System Test Case 37 (STC-37):** Team leader and Collector can send the message to each other via group message within the team on a mobile application.

**Description**

Team leader and Collector can send or receive the message to each other via group message within the team.

**Test Script**

1. User clicks “Teams” menu on the top left of the page.

2.  User selected team that user want to send a message.

3. User inputs the message that user want to conversation.

4. User clicks “Send” button.

**Test data:**

|  |
| --- |
| **Message** |
| Hi, everyone please check your e-mail. |

**Test case:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test no.** | **Description** | **Input** | **Expect output** |
| 1 | User inputs the message, then clicks “Send” button. | Message: “Hi, everyone please check your e-mail.” | The system displays “Hi, everyone please check your e-mail.” on the screen. |
| 2 | User inputs blank the message, then clicks “Send” button. | Message: “” | The system displays “” on the screen. |

**Appendix A**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Users** | | | | | | |
| **user\_id** | **user\_name** | **user\_email** | **user\_pass** | **user\_tell** | **user\_status** | **is\_admin** |
| 1 | Worrasete | win@gmail.com | hahaha | 087546532 | 1 | 1 |
| 2 | Steve Job | steve@gmail.com | 1234 | Null | 1 | 0 |
| 3 | Tim Cook | timcook@gmail.com | 1234 | Null | 1 | 0 |
| 4 | Pcpenchamp | prp@gmail.com | 1234 | 0811668436 | 0 | 0 |
| 5 | Sarun Boon | sarun\_t@gmail.com | 1234 | 31654980 | 0 | 0 |

* **Test Data 1**

|  |  |  |
| --- | --- | --- |
| **Teams** | | |
| **team\_id** | **team\_name** | **team\_leader (user\_id)** |
| 1 | Apple team | 3 |
| 4 | Team 021 | 3 |
| 5 | TeamB | 2 |

* **Test Data 2**
* **Test Data 3**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Projects** | | | | | | | | | |
| **project\_id** | **project\_name** | **project\_desc** | **project\_created\_date** | **team\_id** | **team\_name** | **team\_leader (user\_id)** | **leader\_name** | **user\_id** | **user\_name** |
| 1 | SIA1 | Water quality site 1 | 2014-07-22 21:18:10 | 1 | Apple team | 3 | Tim Cook | 2 | Steve Jobb |
| 2 | SIA2 | Water quality site 2 | 2014-07-25 11:21:52 | 1 | Apple team | 3 | Tim Cook | 2 | Steve Jobb |
| 3 | SIA3 | Water quality site 3 | 2014-11-24 23:12:46 | 2 | CMU\_Camt | 2 | Steve Jobb | 3 | Tim Cook |
| 4 | SIA4 | Water quality site 4 | 2015-04-05 22:31:47 | 2 | CMU\_Camt | 2 | Steve Jobb | 3 | Tim Cook |

* **Test Data 4**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Locations** | | | | | | | |
| **project\_id** | **project\_name** | **team\_id** | **team\_name** | **team\_leader** | **leader\_name** | **location\_id** | **location\_name** |
| 1 | SIA 1 | 1 | Apple team | 3 | Tim Cook | 1 | place1 |
| 1 | SIA 1 | 1 | Apple team | 3 | Tim Cook | 2 | place2 |
| 2 | SIA 2 | 1 | Apple team | 3 | Tim Cook | 3 | place3 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Locations** | | | | | | |
| **user\_id** | **user\_name** | **status\_id** | **loc\_status\_desc** | **location\_lat** | **location\_long** | **date\_time\_statuschg** |
| 2 | Steve Job | 7 | Recollect | 18.7894 | 99.0037 | 2015-05-22 15:55:32 |
| 2 | Steve Job | 8 | Finish | 18.7882 | 98.0043 | 2015-04-17 14:31:57 |
| 3 | Tim Cook | 1 | Initial | 18.7908 | 98.0024 | 2015-04-17 14:29:29 |

* **Test Data 5**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Water** | | | | | | |
| **location\_id** | **location\_name** | **project\_id** | **user\_id** | **user\_name** | **water\_image** | **water\_rgb** |
| 1 | place1 | 1 | 2 | Steve Jobb | IMG\_20150527\_201418.jpg | 6b3421 |
| 1 | place2 | 1 | 2 | Steve Jobb | IMG\_20150218\_004718.jpg | 524d18 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Water** | | | | | |
| **water\_parameter** | **water\_id** | **water\_value** | **water\_date** | **water\_status** | **loc\_status\_desc** |
| PH | 1 | 0.0 | 2015-06-03 21:23:24 | 8 | Finish |
| NI | 2 | 2.357905586284194 | 2015-06-04 13:36:02 | 7 | Recollect |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Result** | | | | | |
| **result\_id** | **water\_id** | **result\_parameter** | **result\_rgb** | **result\_date** | **result\_status** |
| 1 | 1 | 1 | FFFFFF | 2015-05-21 21:28:33 | 1 |
| 2 | 1 | 5 | 666666 | 2015-05-21 21:28:41 | 1 |
| 3 | 1 | 10 | 000000 | 2015-05-21 21:28:52 | 1 |

* **Test Data 6**