|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Document History** | |  |  |  |
| **Version Number** | **Record Date** | **Prepared/**  **Modified By** | **Reviewed By** | **Change Details** |
| 0.1 | 14/04/2014 | Tanawat, Nattakit | PF,NC | * Draft |
| 0.8 | 11/05/2014 | Tanawat, Nattakit | PF,NC | * Draft |
| 0.9 | 28/05/2014 | Tanawat, Nattakit | PF,NC | * Draft |
| 1.0 | 15/07/2014 | Tanawat, Nattakit | PF,NC | * Draft |

Document Designed by

Tanawat Sitthitan  
Nattakit Chaiwongsri  
July 31, 2014   
  
Document Approved by

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
Mr. Phudinan Singkhamfu

August 27, 2014

Content

[Chapter 1 | Introduction 3](#_Toc394577815)

[1.1 Purpose 3](#_Toc394577816)

[1.2 Scope 3](#_Toc394577817)

[1.3 Objective 3](#_Toc394577818)

[1.4 Dependency Document 3](#_Toc394577819)

[1.5 Environment Test 3](#_Toc394577820)

[1.5.1 Notebook 4](#_Toc394577821)

[1.5.2 Server Software 4](#_Toc394577822)

[1.5.3 Mobile 4](#_Toc394577823)

[1.5.4Test Procedure 4](#_Toc394577824)

[Chapter 2 | Test Deliverable 5](#_Toc394577825)

[Unit test plan 5](#_Toc394577826)

# Chapter 1 | Introduction

This test approach document describes the appropriate strategies, process, workflows and methodologies used to plan, organize, execute and manage testing of promotion provider.

## 1.1 Purpose

The purpose of Test Plan document describes the testing plan for the promotion provider in this document consists of the test cases which have to test in each progress period. The system testing covers the URS.

## 1.2 Scope

This test plan document created for promotion provider to measure user requirements, system requirements and to identify the test which will be performed.

## 1.3 Objective

The objective of promotion provider application testing is to ensure the project can satisfy all of user’s requirements and all of functions are consistently and reliability

## 1.4 Dependency Document

The Software Test Plan Document is depended on the following document.

* Software Project Management Plan version 1.2
* Software Requirement Specification version 1.2
* Software Design Document version 1.2
* Test Record version 1.2
* Traceability Record version 1.2

## 1.5 Environment Test

This test plan requires the environment set preparation as following before implement testing part, by the test team.

### 1.5.1 Notebook

* Test on Android emulator API 10 version 2.3
* Test on Android device version 2.3

### 1.5.2 Server Software

* Test on Chrome version 26.0.1410.64

### 1.5.3 Mobile

* iPhone (iOS7)

## 1.5.4Test Procedure

1. Test follows each test case.
2. Description column is how to test in each test case.
3. Next column is input column; we will input the value in “Input” column.
4. If the result is a match with expected output, the test result is pass, but if the result is not matched with expect output, the test result is fail then leave a comment at the test record.
5. Record the result which are “Pass” and “Fail”.

# Chapter 2 | Test Deliverable

## Unit test plan

**UT-TC01: SQLConnector:insertDataIntoDatabase($table, $columnNames, $columnValues)**

***Description***

This unit test is for adding data into database

***Responsibility***

Tanawat Sitthitan, Nattakit Chaiwongsri

***Test Date***

29 July 2014

**Expect Result**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test No.** | **Class name** | **Test item** | **Input** | **Expected Result** |
| 1 | SQLConnector | insertDataIntoDatabase($table, $columnNames, $columnValues) | $table = Accounts  $columnNames = {email, password, account\_type, name, join\_date, status}  $columnValues = {  [warmup@wap.com](mailto:warmup@wap.com), 81dc9bdb52d04dc20036dbd8313ed055, 1, 2014-07-09, 1  } | Input data are in Accounts table in columns which are email = warmup@wap.com, password=81dc9bdb52d04dc20036dbd8313ed055, account\_type = 1, name, join\_date = 2014-07-09 and status=1. |
| 2 | SQLConnector | insertDataIntoDatabase($table, $columnNames, $columnValues) | $table = Account  $columnNames = {email, password, account\_type, name, join\_date, status}  $columnValues = {  [warmup@wap.com](mailto:warmup@wap.com), 81dc9bdb52d04dc20036dbd8313ed055, 1, 2014-07-09, 1  } | Nothing change in database |

**UT-TC02: SQLConnector.php:disableARow($tableName, $condition)**

***Description***

This unit test is for changing status from 1 to 2 in a row in database table

***Responsibility***

Tanawat Sitthitan, Nattakit Chaiwongsri

***Test Date***

29 July 2014

**Expect Result**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test No.** | **Class name** | **Test item** | **Input** | **Expected Result** |
| 1 | SQLConnector | disableARow($tableName, $condition) | $tableName = Accounts  $condition = 100 | A account that has id at 100 change status to 2 |
| 2 | SQLConnector | disableARow($tableName, $condition) | $tableName = Accounts  $condition = 102 | A account that has id at 102 change status to 2 |

**UT-TC03: SQLConnector.php:enableARow($tableName, $condition)**

***Description***

This unit test is for changing status from 2 to 1 in a row in database table

***Responsibility***

Tanawat Sitthitan, Nattakit Chaiwongsri

***Test Date***

29 July 2014

**Expect Result**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test No.** | **Class name** | **Test item** | **Input** | **Expected Result** |
| 1 | SQLConnector | disableARow($tableName, $condition) | $tableName = Accounts  $condition = 100 | A account that has id at 100 change status to 1 |
| 2 | SQLConnector | disableARow($tableName, $condition) | $tableName = Accounts  $condition = 102 | A account that has id at 102 change status to 1 |

**UT-TC04: SQLConnector.php:isDuplicate($tableName, $conditionName, $conditionValue)**

***Description***

This unit test is for checking existing data in database table

***Responsibility***

Tanawat Sitthitan, Nattakit Chaiwongsri

***Test Date***

29 July 2014

**Expect Result**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test No.** | **Class name** | **Test item** | **Input** | **Expected Result** |
| 1 | SQLConnector | isDuplicate($tableName, $conditionName, $conditionValue) | $tableName = Accounts  $conditionName = email  $conditionValue = [warmup@wap.com](mailto:warmup@wap.com) | 1 |
| 2 | SQLConnector | isDuplicate($tableName, $conditionName, $conditionValue) | $tableName = Accounts  $conditionName = email  $conditionValue = [warmupcafe@wap.com](mailto:warmupcafe@wap.com) | 0 |

**UT-TC05: SQLConnector.php:getDataFromDatabase($tableName)**

***Description***

This unit test is for retrieving data from database table

***Responsibility***

Tanawat Sitthitan, Nattakit Chaiwongsri

***Test Date***

29 July 2014

**Expect Result**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test No.** | **Class name** | **Test item** | **Input** | **Expected Result** |
| 1 | SQLConnector | getDataFromDatabase($tableName) | $tableName = Accounts | All data in Accounts table |
| 2 | SQLConnector | getDataFromDatabase($tableName) | $tableName = Promotions | All data in Accounts Promotions table |

**UT-TC06: SQLConnector.php:updatePromotionTable(Promotion $promotion)**

***Description***

This unit test is for updating data in promotion table

***Responsibility***

Tanawat Sitthitan, Nattakit Chaiwongsri

***Test Date***

29 July 2014

**Expect Result**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test No.** | **Class name** | **Test item** | **Input** | **Expected Result** |
| 1 | SQLConnector | updatePromotionTable(Promotion $promotion) | promotion object that contain account\_id = 100, name = Free bar, description = Only for member, img =“”, shared =“”, start\_date = 2014-07-07, end\_date = 2014-07-25, status=1, promotion\_id =1 | promotion at id = 1 change data to account\_id = 100, name = Free bar, description = Only for member, img =“”, shared =“”, start\_date = 2014-07-07, end\_date = 2014-07-25, status=1, |
| 2 | SQLConnector | updatePromotionTable(Promotion $promotion) | promotion object that contain account\_id = 100, name = Free bar, description = Only for member, img =“”, shared =“”, start\_date = 2014-07-07, end\_date = 2014-07-25, status=1, promotion\_id =2 | promotion at id = 2 change data to account\_id = 100, name = Free bar, description = Only for member, img =“”, shared =“”, start\_date = 2014-07-07, end\_date = 2014-07-25, status=1, |

**UT-TC07: AccountDAOImpl.php:getAccountByEmail($email)**

***Description***

This unit test is for getting a account by email

***Responsibility***

Tanawat Sitthitan, Nattakit Chaiwongsri

***Test Date***

29 July 2014

**Expect Result**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test No.** | **Class name** | **Test item** | **Input** | **Expected Result** |
| 1 | AccountDAOImpl | getAccountByEmail($email) | email = [admin1@wap.com](mailto:admin1@wap.com) | Object of admin1 |
| 2 | AccountDAOImpl | getAccountByEmail($email) | email = “” | null |

**UT-TC08: PromotionDAOImpl:getPromotionByName($name)**

***Description***

This unit test is for getting a promotion by using promotion name

***Responsibility***

Tanawat Sitthitan, Nattakit Chaiwongsri

***Test Date***

29 July 2014

**Expect Result**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test No.** | **Class name** | **Test item** | **Input** | **Expected Result** |
| 1 | PromotionDAOImpl | getPromotionByName($name) | name = free bar | object of promotion that has name = free bar |

**UT-TC09: PromotionDAOImpl:getPromotionById($id)**

***Description***

This unit test is for getting a promotion by using promotion id

***Responsibility***

Tanawat Sitthitan, Nattakit Chaiwongsri

***Test Date***

29 July 2014

**Expect Result**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test No.** | **Class name** | **Test item** | **Input** | **Expected Result** |
| 1 | PromotionDAOImpl | getPromotionById($id) | 1 | object of promotion that has id = 1 |
| 2 | PromotionDAOImpl | getPromotionById($id) | 2 | object of promotion that has id = 2 |
| 3 | PromotionDAOImpl | getPromotionById($id) | 100 | null |

**UT-TC10: RequestSignupInfoDAOImpl:getRequesSignupById($id)**

***Description***

This unit test is for getting a register request by using id

***Responsibility***

Tanawat Sitthitan, Nattakit Chaiwongsri

***Test Date***

29 July 2014

**Expect Result**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test No.** | **Class name** | **Test item** | **Input** | **Expected Result** |
| 1 | RequestSignupInfoDAOImpl | getRequesSignupById($id) | 1 | Object of RequestSignupInfo that has id =1 |
| 2 | RequestSignupInfoDAOImpl | getRequesSignupById($id) | 999 | null |

**UT-TC11: RequestSignupInfoDAOImpl:getRequstByStatus($status)**

***Description***

This unit test is for getting requests by using status

***Responsibility***

Tanawat Sitthitan, Nattakit Chaiwongsri

***Test Date***

29 July 2014

**Expect Result**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test No.** | **Class name** | **Test item** | **Input** | **Expected Result** |
| 1 | RequestSignupInfoDAOImpl | getRequstByStatus($status) | 1 | List of RequestSignupInfo that has status =1 |
| 2 | RequestSignupInfoDAOImpl | getRequstByStatus($status) | 2 | List of RequestSignupInfo that has status =2 |
| 3 | RequestSignupInfoDAOImpl | getRequstByStatus($status) | 3 | null |

**UT-TC12: ShopInformationDAOImpl:getShopInformationById($id)**

***Description***

This unit test is for changing password for account

***Responsibility***

Tanawat Sitthitan, Nattakit Chaiwongsri

***Test Date***

29 July 2014

**Expect Result**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test No.** | **Class name** | **Test item** | **Input** | **Expected Result** |
| 1 | ShopInformationDAOImpl | getShopInformationById($id) | 100 | Object of ShopInformation that has id=100 |
| 2 | ShopInformationDAOImpl | getShopInformationById($id) | 1 | null |
| 3 | ShopInformationDAOImpl | getShopInformationById($id) | 103 | Object of ShopInformation that has id=103 |

**UT-TC13: AccountController:changePassword($password, $id)**

***Description***

This unit test is for changing password for account

***Responsibility***

Tanawat Sitthitan, Nattakit Chaiwongsri

***Test Date***

29 July 2014

**Expect Result**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test No.** | **Class name** | **Test item** | **Input** | **Expected Result** |
| 1 | AccountController | changePassword($password, $id) | password = 1234  id = 1 | Account at id 1  password change to 1234 |
| 2 | AccountController | changePassword($password, $id) | password = zxcv  id = 2 | Account at id 2  password change to zxcv |

**UT-TC14: AccountController:doLogin:($email, $password)**

***Description***

This unit test is for changing password for account

***Responsibility***

Tanawat Sitthitan, Nattakit Chaiwongsri

***Test Date***

29 July 2014

**Expect Result**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test No.** | **Class name** | **Test item** | **Input** | **Expected Result** |
| 1 | AccountController | doLogin:($email, $password) | email = admin1@wap  password = 1234 | return true |
| 2 | AccountController | doLogin:($email, $password) | email = admin1@wap  password = 2222 | return false |