**Web-based Ordering & Ingredient Estimating for Bakery Manufacturer**

Test Plan

By

**Miss. Nontra Mahachanont 542115026**

**Mr. Parinya Panyanak 542115034**

Department of Software Engineering

College of Arts, Media and Technology

Chiang Mai University

Project Advisor

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Mr.Phudinan Singkhamfu**

**Document History**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Document Name** | **Version** | **Status** | **Date** | **Viewable** | **Reviewer** | **Responsible** |
| **Documents** | | | | | | |
| Web-based Ordering & Ingredient Estimating for Bakery Manufacturer – Test Plan\_V.0.1.docx | V.0.1  - Add Chapter 1  - Add Chapter 2 | Draft | 07-06-2014 | NM, PP, PS | NM, PP | NM, PP |
| Web-based Ordering & Ingredient Estimating for Bakery Manufacturer – Test Plan\_V.0.4.docx | V.0.4  - Add Chapter 3  - Add Chapter 4 | Draft | 14-06-2014 | NM, PP, PS | NM, PP | NM, PP |
| Web-based Ordering & Ingredient Estimating for Bakery Manufacturer – Test Plan\_V.0.5.docx | V.0.5  - Add Development Tool Review  - Adjust Abstract  - Add Chapter3  - Quality Standards | Draft | 17-02-2014 | NM, PP, PS | NM, PP | NM, PP |
| Web-based Ordering & Ingredient Estimating for Bakery Manufacturer – Test Plan\_V.0.7.docx | V.0.7  - Add Chapter4  - Project Plan  - Adjust Technology Review  - Adjust Schedule & Milestone  - Adjust Project Name | Draft | 18-02-2014 | NM, PP, PS | NM, PP | NM, PP |
| Web-based Ordering & Ingredient Estimating for Bakery Manufacturer – Test Plan\_V.0.8.docx | V.0.8  - Add Chapter5  - References  - Adjust Chapter 1  - Adjust Chapter 2  - Adjust Chapter 4  - Adjust Project Name  - Fix inappropriate word | Draft | 03-03-2014 | NM, PP, PS | NM, PP | NM, PP |
| Web-based Ordering & Ingredient Estimating for Bakery Manufacturer – Test Plan\_V.0.9.docx | V.0.9  - Adjust Feature  - Adjust Project Name  - Adjust System architecture  - Fix inappropriate word | Draft | 21-03-2014 | NM, PP, PS | NM, PP | NM, PP |
| Web-based Ordering & Ingredient Estimating for Bakery Manufacturer – Test Plan\_V.1.0.docx | V.1.0  - Adjust Feature  - Adjust System architecture  - Adjust limits  - Fix inappropriate word | Released | 25-03-2014 | NM, PP, PS | NM, PP | NM, PP |

**\*NM = Nontra Mahachanont**

**\*PP = Parinya Panyanak**

**\* PS = Phudinan Singkhamfu**

**Table of contents**

[**Chapter One | Introduction** 4](#_Toc391906866)

[1.1 Objectives 4](#_Toc391906867)

[1.2 Scope 4](#_Toc391906868)

[1.3 Acronyms and Definitions 4](#_Toc391906869)

[1.3.1 Acronyms 4](#_Toc391906870)

[1.3.2 Definitions 5](#_Toc391906871)

[**Chapter Two| Test Plan and Test Procedure** 6](#_Toc391906872)

[2.1 Test Objectives 6](#_Toc391906873)

[2.2 Scope of Testing 6](#_Toc391906874)

[2.3 Test Duration 6](#_Toc391906875)

[2.4 Test Responsibility 7](#_Toc391906876)

[2.5 Result of Testing 7](#_Toc391906877)

[2.6 Test Environment 7](#_Toc391906878)

[2.6.1 Hardware 7](#_Toc391906879)

[**Chapter Three | Unit Testing of Web-based Ordering & Ingredient Estimating for Bakery Manufacturer** 8](#_Toc391906880)

[**Chapter Four | System Testing of Web-based Ordering & Ingredient Estimating for Bakery Manufacturer** 9](#_Toc391906881)

# **Chapter One | Introduction**

## 1.1 Objectives

The objectives of the test plan of Web-based Ordering & Ingredient Estimating for Bakery Manufacturer are used to create test plan of the unit testing. It makes sure that the defects are discovered and fixed all case in progress I. The user requirements system is covered by system testing. The unit testing covers all of implemented methods in Web-based Ordering & Ingredient Estimating for Bakery Manufacturer project.

## 1.2 Scope

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

## 1.3 Acronyms and Definitions

### 1.3.1 Acronyms

SRS Software Requirement Specification

URS User Requirement Specification

SDD Software Design Document

UI User Interface

UTC Unit Testing Case

STC System Testing Case

### 1.3.2 Definitions

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

IEEE Institute for Electrical and Electronics Engineers. Biggest global interest group for engineers of different branches and computer scientists. [IEEE90]

Requirement (1) A condition or capability needed by a user to solve a problem or achieve an objective. (2) A condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed document. (3) A documented representation of a condition or capability as in definition (1) or (2). [IEEE90]

Specification Precise description of an activity or work product which reserves 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 bed one, how it will be done) must not be mixed. [IEEE90]

White box testing A software testing method in which the internal structure/design/implementation of the item being tested is known to the tester. The tester chooses inputs to exercise paths through the code and determines the appropriate outputs. Programming know-how and the implementation knowledge is essential. White box testing is testing beyond the user interface and into the nitty-gritty of a system.

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 software 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 specified requirements.

Null Define values represent missing unknown data.

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

## 2.1 Test Objectives

The objectives of testing Web-based Ordering & Ingredient Estimating for Bakery Manufacturer project are:

1. The defects are detected.

2. The defects are fixed.

3. Functionalities and user interface covered all requirements that user want.

4. All functionalities and features that are define and feature that are define in progress 1, must be follow in project plan

## 2.2 Scope of Testing

Web-based Ordering & Ingredient Estimating for Bakery Manufacturer will test by press button testing techniques that are unit testing in web application side.

## 2.3 Test Duration

|  |  |
| --- | --- |
| **Progress** | **Date and Duration** |
| **Progress I** | **Start:** 10 June 2014  **End:** 05 July 2014  **Duration:** 25 Days |

## 

## 2.4 Test Responsibility

|  |  |
| --- | --- |
| **Item** | **Responsibility** |
| **Unit Test** | Parinya Panyanak |
| **System Test** | Parinya Panyanak |

## 2.5 Result of Testing

1. Actual Output: The actual output that are performed by each test case.

2. Pass/Fail criteria:

2.1 Pass: The result of actual result is same with expected result.

2.2 Fail: The result of actual result is not same with expected result.

## 2.6 Test Environment

### 2.6.1 Hardware

* Computers

**- Name:** Sony VAIO-VAIO

**Processor:** Intel® Pentium® Processor T4300 (2.10 GHz)

**Memory:** 2.00 GB

**Graphics:** Intel® Graphics Media Accelerator HD

**Operating System:** Window® 7 Home Basic (64-bit)

**- Name:** Sony VAIO

**Processor:** Intel® Core (TM) I3-3120M (2.5 GHz)

**Memory:** 4.00 GB

**Graphics:** Radeon Graphics System

**Operating System:** Window® 8 Based Processor (64-bit)

**2.6.2 Software**

* Microsoft Visual Studio Ultimate
* Selenium

# **Chapter Three | Unit Testing of Web-based Ordering & Ingredient Estimating for Bakery Manufacturer**

**Class:** AccountController

**Unit Test Case-01(UTC-01):** public ActionResult Login(LoginModel model, string returnUrl)

**Description:** This test case used to check procedure of Login method.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **no.** | **description** | **input** | | **Expect output** |
| **Id** | **password** |
| 1 | Input valid id and password | [tung@dd.com](mailto:tung@dd.com) | 123456 | Login to home page |
| 2 | Input valid id but, invalid password | [tung@dd.com](mailto:tung@dd.com) | sdsdsdsd | Error message |
| 3 | Input invalid id and valid password | [tt@hotmail.com](mailto:tt@hotmail.com) | 123456 | Error message |
| 4 | Input invalid both id and password | [gg@gmail.com](mailto:gg@gmail.com) | sdsdsdsd | Error message |
| 5 | Do not input both id and password | - | - | Error message |
| 6 | Do not input id but input password | [tugn@dd.com](mailto:tugn@dd.com) | - | Error message |
| 7 | Do not input password but, input id | - | sdsdsdsd | Error message |

**Class:** UserController

**Unit Test Case-01(UTC-02):** public ActionResult Index ()

**Description:** This test case used to check correctly http to go to Index page.

|  |  |  |
| --- | --- | --- |
| **no.** | **description** | **Expect output** |
|
| 1 | To return Valid http get of index page | Login to index page |
| 2 | To return Invalid http get of index page | Error to return index page |

**Class:** UserController

**Unit Test Case-01(UTC-03):** public ActionResult Details (int id)

**Description:** This test case used to check http to go to Details page with correctly user.

|  |  |  |
| --- | --- | --- |
| **no.** | **description** | **Expect output** |
|
| 1 | To return Valid http get of Details page | go to details page |
| 2 | To return Invalid http get of Details page | Error to go to details page |
| 3 | To return Valid http get Details page, but invalid user | go to details page and show details of user |
| 4 | To return Valid http get Details page and valid user | go to details page and show details of valid user |
| 5 | To return Invalid http get Details page, but valid user | Error to go to details page |
| 6 | To return Invalid http get Details page, and invalid user | Error to go to details page |

**Class:** UserController

**Unit Test Case-04(UTC-04):** public ActionResult UserProfile(string id)

**Description:** This test case used to check http to go to UserProfile page with correctly user.

|  |  |  |
| --- | --- | --- |
| **no.** | **description** | **Expect output** |
|
| 1 | To return Valid http get of Details page | go to UserProfile page |
| 2 | To return Invalid http get of Details page | Error to go to UserProfile page |
| 3 | To return Valid http get Details page, but invalid user | go to UserProfile page and show details of user |
| 4 | To return Valid http get Details page and valid user | go to UserProfile page and show details of valid user |
| 5 | To return Invalid http get Details page, but valid user | Error to go to UserProfile page |
| 6 | To return Invalid http get Details page, and invalid user | Error to go to UserProfile page |

**Class:** UserController

**Unit Test Case-05(UTC-05):** public ActionResult Create()

**Description:** This test case used to check http to go to Create page.

|  |  |  |
| --- | --- | --- |
| **no.** | **description** | **Expect output** |
|
| 1 | To return valid http get to create page | go to create page |
| 2 | To return invalid http get to create page | Error to return index page |

**Class:** UserController

**Unit Test Case-06(UTC-06):** public ActionResult Create (User user)

**Description:** This test case used to check the procedure of creating new user.

|  |  |  |
| --- | --- | --- |
| **no.** | **description** | **Expect output** |
|
| 1 | In put valid information to form | Create new user and go to index |
| 2 | In put invalid information to form | Error will be show |
| 3 | Does not input information | Error will be show |

**Class:** UserController

**Unit Test Case-07(UTC-07):** public ActionResult Edit (int id = 0)

**Description:** This test case used to check http get to call Edit page.

|  |  |  |
| --- | --- | --- |
| **no.** | **description** | **Expect output** |
|
| 1 | To return Valid http get of edit page | go to edit page |
| 2 | To return Invalid http get of edit page | Error to return to edit page |

**Class:** UserController

**Unit Test Case-08(UTC-08):** public ActionResult Edit (User user)

**Description:** This test case used to check edit method that can update information correctly.

|  |  |  |
| --- | --- | --- |
| **no.** | **description** | **Expect output** |
|
| 1 | Valid http get of index page | Login to index page |
| 2 | Invalid http get of index page | Error to return index page |

**Class:** UserController

**Unit Test Case-09(UTC-09):** public ActionResult Delete(int id = 0)

**Description:** This test case used to check procedure of Login method.

|  |  |  |
| --- | --- | --- |
| **no.** | **description** | **Expect output** |
|
| 1 | Valid http get of index page | Login to index page |
| 2 | Invalid http get of index page | Error to return index page |

**Class:** UserController

**Unit Test Case-10(UTC-10):** public ActionResult DeleteConfirmed(int id)

**Description:** This test case used to check procedure of Login method.

**Class:** ProductController

**Unit Test Case-11(UTC-11):** public ActionResult Index()

**Description:** This test case used to check procedure of Login method.

**Class:** ProductController

**Unit Test Case-12(UTC-12):** public ActionResult Details(int id = 0)

**Description:** This test case used to check procedure of Login method.

**Class:** ProductController

**Unit Test Case-13(UTC-13):** public ActionResult Create()

**Description:** This test case used to check procedure of Login method.

**Class:** ProductController

**Unit Test Case-14(UTC-14):** public ActionResult Create(Product product)

**Description:** This test case used to check procedure of Login method.

**Class:** ProductController

**Unit Test Case-15(UTC-15):** public ActionResult Edit(int id = 0)

**Description:** This test case used to check procedure of Login method.

**Class:** ProductController

**Unit Test Case-16(UTC-16):** public ActionResult Edit(Product product)

**Description:** This test case used to check procedure of Login method.

**Class:** ProductController

**Unit Test Case-17(UTC-17):** public ActionResult Delete(int id = 0)

**Description:** This test case used to check procedure of Login method.

**Class:** ProductController

**Unit Test Case-18(UTC-18):** public ActionResult DeleteConfirmed(int id)

**Description:** This test case used to check procedure of Login method.

**Test Data:**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Name** | **Detail** | **Image** | **lat** | **lng** | **StartDate** | **EndDate** | **StartTime** | **EndTime** |
|  |  |  |  |  |  |  |  |  |

**Test Cases:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test No.** | **Description** | **Input** | **Expect Output** |
| 1 |  |  |  |

# **Chapter Four | System Testing of Web-based Ordering & Ingredient Estimating for Bakery Manufacturer**

**Test Case:**

**Description:**

**Test Procedure:**

**Test Data:**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Name** | **Detail** | **Image** | **lat** | **lng** | **StartDate** | **EndDate** | **StartTime** | **EndTime** |
|  |  |  |  |  |  |  |  |  |

**Test Cases:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test No.** | **Description** | **Input** | **Expect Output** |
| 1 |  |  |  |