|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | **Description: Logo_FPT_University_doc** | **MINISTRY OF EDUCATION AND TRAINING** | | | **FPT UNIVERSITY** | | |
| Capstone Project |
| Information System In Logistic Company  Software Test Documentation |
| |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | |  |  |  |  | | --- | --- | --- | --- | | **Group 6** | | | | | **Group Members** | Nguyễn Hoàng Việt Khánh |  | 60142 | | Nguyễn Thị Yên Thịnh |  | 60153 | | Nguyễn Đỗ Vượng |  | 60267 | | An Ngọc Anh |  | 60277 | | **Supervisor** | Lâm Hữu Khánh Phương | | | | **Capstone Project code** | SMDH | | | | |
| * Ho Chi Minh City, Mar / 2013 - |

# Table of Contents

[Table of Contents 1](#_Toc342604345)

[5.1 Introduction 2](#_Toc342604351)

[*5.1.1* System Overview 2](#_Toc342604352)

[*5.1.2* Test Approach 2](#_Toc342604353)

[5.2 Test Plan 3](#_Toc342604354)

[5.2.1 Functions 3](#_Toc342604355)

[*5.2.2* *Features not to be tested* 5](#_Toc342604356)

[5.2.2.1 Performance test 5](#_Toc342604357)

[*5.2.3* Testing Tools and Environment 5](#_Toc342604358)

[5.3 Test Cases 6](#_Toc342604359)

[*5.3.1* *Functions* 6](#_Toc342604360)

[5.3.1.1 Login 6](#_Toc342604361)

[5.3.1.2 Role ‘Vendor’ 6](#_Toc342604362)

5.3.1.3 Role 'Hub'…………………………………………………………………………………………....7

[5.3.1.4 Role ‘Office Staff’ 10](#_Toc342604363)

[5.3.1.5 Role ‘Customer’ 16](#_Toc342604364)

[5.4 Checklist 17](#_Toc342604365)

[*5.4.1* Checklist validation 17](#_Toc342604366)

[*5.4.2* Submission Checklist 17](#_Toc342604367)

[5.5 Other material (if any) 18](#_Toc342604368)

[*5.5.1* Checklist validation 18](#_Toc342604369)



## Introduction

### System Overview

This document describes the approach and methodologies used by the testing group to plan, organize and manage the testing of this system. It describes implementation details of test cases or technical details of how the product features should work.

The system will be tested is also the system our team are currently developing. This system developed based on MVC3 framework. It is a website providing Home Delivery service for online and offline shops.

### Test Approach

The testing for this project will consist of Unit, System/Integration (combined) and Acceptance test levels. It is hoped that there will be at least one full time independent test person for system/integration testing. Most testing will be done by the test lead with the development team’s participation.

Unit Testing will be done by the developers and will be approved by the implementing team leader. All unit test information (sample, output, data printouts, and defect information…) will be managed by developers and it must be showed when team required.

System/Integration Testing will be performed by the test lead and implementing team leader with assistance from the individual developers as required.

Acceptance Testing will be performed by the test lead and development team leader. The acceptance test will be done for a period of 2 weeks after completion of the System/Integration test process. Program will enter into Acceptance test after all critical and major defects have been corrected. A program may have one major defect as long as it does not obstruct testing of the program (There is a work around for the error). Prior to completion of acceptance testing all open critical and major defects must be corrected.

## Test Plan

The following is a list of areas to be focus on during testing of website:

### Functions

#### Log in

Test all functions in Login module including:

* Login
* Logout
* Change Password

#### Role ‘Staff’

Test all functions of role ‘Office Staff’ including:

**Pricing:**

* Approve request (set price)
* Repricing

**Collection Plan:**

* + - Create Auto Collection Plan
    - Cancel Collection Plan
    - View Collection Plan Detail
    - Assign Delivery Men to Collection Plan
    - Mark as Finished Collection Plan

**Delivery Plan:**

* + - Create Auto Delivery Plan
    - Cancel Delivery Plan
    - View Delivery Plan Detail
    - Assign Delivery Men to Delivery Plan
    - Mark as Finished Delivery Plan

**Return Plan:**

* Create Auto Return Plan
* Cancel Return Plan
* View Return Plan Detail
* Assign Delivery Men to Return Plan
* Mark as Finished Return Plan

#### Role ‘Customer’

Test all functions of role ‘Customer’ including:

**Product:**

* + - Create New Product
    - Edit Product
    - Delete Product

**Order Management:**

* Approve Order
* Reject Order
* Add new Order

**Request Management:**

* Add order to request
* Submit request to Tiktak
* Approve Pricing
* Cancel request
* Reject Pricing
* Ask to repricing

#### Role ‘Hub staff’

Test all functions of role ‘Hub staff’ including:

**Order tracking:**

* View Order by status = *In Hub*
* View Order by status = *Delivered*
* View Order by status = *Waiting for Return*
* View Order by status = *Returned*

**Delivery Order to Buyer:**

* View receiver information
* Mark Order as finished
* Mark Order as Waiting for Return

#### Role ‘Buyer’

Test all functions of role ‘Buyer’ including:

**Order:**

* Make an Order
* Check Order info

### *Features not to be tested*

#### Performance test

There was not enough time so we cannot optimize source code and improve performance. We can only ensure the system work well but not at its best performance.

### Testing Tools and Environment

* Windows 7 operating system
* Google Chrome 19 or higher
* Microsoft Visual Studio 2010 SP1
* Microsoft SQL Server 2008 R2

## Test Cases

### *Functions*

#### Login

#### Role ‘Customer’

##### **Product Management**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test case No** | **Test case** | **Input** | **Expected Result** | **Status** | **Note** |
| **Add new product** | | | | | |
| TC\_1 | Create product successful | 1, Open the main page of Product 2, click Add New button on the top page 3, Input new product info 4, Click Create button | 4.Redirect to the main page of Product  - The new product is shown in data table and save to database | Passed |  |
| TC\_2 | Cancel button | 1, Open the main page of Product 2, click Add New button on the top page 3, Input new product info 4, Click Cancel button | 4. Redirect to the main page of Product  - Nothing is changed | Passed |  |
| TC\_3 | Create product without input | 1, Open the main page of Product 2, click Add New button on the top page 3, No input new product info 4, Click Create button | 4. Don't allow redirect to the main page of Product  - Focus into the first Required field. | Passed |  |
| TC\_4 | Create product with empty value | 1, Open the main page of Product 2, click Add New button on the top page 3, Let input is empty 4, Click Create button | 4. All required fields must have value. | Passed |  |
| TC\_5 | Create product only with Require fields input | 1, Open the main page of Product 2, click Add New button on the top page 3, Input only into Rquire fields 4, Click Create button | 4. Redirect to the main page of Product  - The new product is shown in data table and save to database | Passed |  |
| TC\_6 | Create product only with one of Require fields input is blank | 1, Open the main page of Product 2, click Add New button on the top page 3, Let one of require fields is blank \* Other required fields are valid 4, Click Create button | 4. Show validate message. Required field must have value | Passed |  |
| **Edit existed product** | | | | | |
| TC\_7 | Edit existed product with no change | 1, Open the main page of Product 2, Click Edit icon on the record user want to modify 3, Don't change  4, Click Save button | 4. Redirect to the main page of Product  - Nothing is changed | Passed |  |
| TC\_8 | Edit existed product with valid value | 1, Open the main page of Product 2, Click Edit icon on the record user want to modify 3, Change some valid values 4, Click Save button | 4. Redirect to the main page of Product - The product is updated in the data table and save to database | Passed |  |
| TC\_9 | Edit existed product with invalid value | 1, Open the main page of Product 2, Click Edit icon on the record user want to modify 3, Change some invalid values 4, Click Save button | 4. Show message: Value is invalid and ask user to input value again | Passed |  |
| TC\_10 | Edit existed product with Cancel button | 1, Open the main page of Product 2, click Edit icon on the record user want to modify 3, Input the values 4, Click Cancel button | Redirect to the main page of Product - Nothing is changed | Passed |  |
| **Delete existed product** | | | | | |
| TC\_11 | Delete existed product with Cancel button | 1, Open the main page of Product 2, Click Delete icon on the record user want to delete 3, Click Cancel button | 2. Show confirm message  3.  Redirect to the main page of Product - Nothing is changed | Passed |  |
| TC\_12 | Delete existed product successfully | 1, Open the main page of Product 2, click Delete icon on the record user wants to delete 3, Click OK button | 2. Show confirm message  3. Redirect to the main page of Product - The order is removed from the data table. Database is updated | Passed |  |

##### Order Management

#### Role ‘Hub’

#### Role ‘Office staff’

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test case No** | **Test case** | **Input** | **Expected Result** | **Status** | **Type** | **Test Date** | **PIC** | **Note** |
| **COLLECTION PLAN** | | | | | | | | |
| TC\_1 | Group all requests | 1, Open the main page of Collection plan 2, Choose *Group All Requests* 3, Click *Group* button | 3, All requests are grouped by latitude, longitude | N/A | Normal |  |  |  |
| TC\_2 | Group <no.> request(s) | 1, Open the main page of Collection plan 2a, Choose *Group <no.> Request(s)* 2b, Input the number user want to group requests 3, Click *Group* button | 2b, Input the positive number 3, Requests are grouped by each 3 requests which has the same latitude, longitude | N/A | Normal |  |  | group 3 requests |
| TC\_3 | Add request(s) to create plan | 1, Open the main page of Collection plan 2, Choose the requests to create plan 3, Click *Add* button | 3, Request(s) are copied to Selected Request box | N/A | Normal |  |  |  |
| TC\_4 | Remove request(s) from Selected Request box | 1, Open the main page of Collection plan 2, Choose the requests to remove from Selected Request box to Request box 3, Click *Remove* button | 3, Request(s) are removed to Request box | N/A | Normal |  |  |  |
| TC\_5 | Enter plan no. to create plan | 1, Open the main page of Collection plan 2, Input plan no. to create plan | 2, the plan no. must be a positive number | N/A | Normal |  |  | ex: input 3 plans |
| TC\_6 | Create new plan(s) successfully | \*Pre-condition: pass TC\_1(or TC\_2), TC\_3, TC\_5 1, Click *Solve* button | 1, Move to *Edit Plan(s)* screen - Plan(s) are created successfully | N/A | Normal |  |  |  |
| TC\_7 | View plan(s) info which are created (include: request(s) and map) | \*Pre-condition: pass TC\_06 1, click on plan user want to view/edit | 1, - Request(s) of this plan are displayed in the data table  - Map will be displayed the way DM will be going to | N/A | Normal |  |  |  |
| TC\_8 | Delete request of plan | \* Pre-condition: pass TC\_07 1, click *Delete* button on request user want to delete 2a, Click *OK* button 2b, Click *Cancel* button | 1, Confirm message will be displayed 2a, Redirect to *Edit Plan(s)* screen - The request is removed from the data table. Map is also updated | N/A | Normal |  |  |  |
| TC\_9 | Move request to another plan | \* Pre-condition: pass TC\_07 1, click *Move to another plan* button on request user want to remove from this plan 2, Choose plan user want to move the request to 3a, Click *OK* button 3b, Click *Cancel* button | 1, *Move to another plan* screen will be displayed - Combobox has plans list 3a, Redirect to *Edit Plan(s)* screen - The request is removed from the data table. Map is also updated  3b, Redirect to *Edit Plan(s)* screen - Nothing is changed | N/A | Normal |  |  |  |
| TC\_10 | Save modified plan(s) to database | \*Pre-condition: pass TC\_8 or TC\_9 1, click *Save* button | 1, Redirect to *Plan Management* screen - The table will be displayed the plan(s) are created | N/A | Normal |  |  |  |
| TC\_11 | Search *New* status | 1, Open *Plan Management* screen 2, Choose status = *New* 3, Click *Filter* button | 3, All plans have *New* status will be displayed in the data table | N/A | Normal |  |  |  |
| TC\_12 | Search *All* status | 1, Open *Plan Management* screen 2, Choose status = *All* 3, Click *Filter* button | 3, All plans will be displayed in the data table | N/A | Normal |  |  |  |
| TC\_13 | Search *Assigned* status | 1, Open *Plan Management* screen 2, Choose status = *Assigned* 3, Click *Filter* button | 3, All plans have *Assigned* status will be displayed in the data table | N/A | Normal |  |  |  |
| TC\_14 | Search *Canceled* status | 1, Open *Plan Management* screen 2, Choose status = *Canceled* 3, Click *Filter* button | 3, All plans have *Canceled* status will be displayed in the data table | N/A | Normal |  |  |  |
| TC\_15 | Search *Finished* status | 1, Open *Plan Management* screen 2, Choose status = *Finished* 3, Click *Filter* button | 3, All plans have *Finished* status will be displayed in the data table | N/A | Normal |  |  |  |
| TC\_16 | Search *None* status | 1, Open *Plan Management* screen 2, Choose status = *None* 3, Click *Filter* button | 3, All plans will be displayed in the data table | N/A | Normal |  |  |  |
| TC\_17 | Search multiple status | 1, Open *Plan Management* screen 2, Choose status = *New|Assigned|Canceled|Finished* 3, Click *Filter* button | 3, All plans have *New|Assigned|Canceled|Finished* status will be displayed in the data table | N/A | Normal |  |  |  |
| TC\_18 | Search by *Date* | 1, Open Plan Management screen 2, Input *From Date* and *To Date* value3, Click *Filter* button | 2, The date must be valid 3, All orders mapped search condition will be displayed in the data table | N/A | Normal |  |  |  |
| TC\_19 | Search by *From Date* | 1, Open Plan Management screen 2, Input *From Date - To Date* is blank3, Click *Filter* button | 2, The *From Date* must be valid 3, All orders mapped search condition will be displayed in the data table | N/A | Normal |  |  |  |
| TC\_20 | Search by *To Date* | 1, Open *Plan Management* screen 2, Input *To Date - From Date* is blank3, Click *Filter* button | 2, The *To Date* must be valid 3, All orders mapped search condition will be displayed in the data table | N/A | Normal |  |  |  |
| TC\_21 | View detail of each plan | 1, Open *Plan Management* screen 2, Click *View Detail* button on the plan | 2, *View Detail* screen will be displayed | N/A | Normal |  |  |  |
| TC\_22 | Assign Delivery men to plan | \*Pre-condition: Pass TC\_21 1, Check the DM to assign 2, Click *OK* button | 2, Redirect to *View Detail* screen - DM(s) are assigned to the plan | N/A | Normal |  |  |  |
| **DELIVERY PLAN** | | | | | | | | |
| TC\_1 | Group all orders | 1, Open the main page of Delivery plan 2, Choose Group All orders 3, Click Group button | 3, All orders are grouped by latitude, longitude | N/A | Normal |  |  |  |
| TC\_2 | Group <no.> order(s) | 1, Open the main page of Delivery plan 2a, Choose Group <no.> order(s) 2b, Input the number user want to group orders 3, Click Group button | 2b, Input the positive number 3, orders are grouped by each 3 orders which has the same latitude, longitude | N/A | Normal |  |  | group 3 orders |
| TC\_3 | Add order(s) to create plan | 1, Open the main page of Delivery plan 2, Choose the orders to create plan 3, Click Add button | 3, order(s) are copied to Selected order box | N/A | Normal |  |  |  |
| TC\_4 | Remove order(s) from Selected order box | 1, Open the main page of Delivery plan 2, Choose the orders to remove from Selected order box to order box 3, Click Remove button | 3, order(s) are removed to order box | N/A | Normal |  |  |  |
| TC\_5 | Enter plan no. to create plan | 1, Open the main page of Delivery plan 2, Input plan no. to create plan | 2, the plan no. must be a positive number | N/A | Normal |  |  | ex: input 3 plans |
| TC\_6 | Create new plan(s) successfully | \*Pre-condition: pass TC\_1(or TC\_2), TC\_3, TC\_5 1, Click *Solve* button | 1, Move to *Edit Plan(s)* screen - Plan(s) are created successfully | N/A | Normal |  |  |  |
| TC\_7 | View plan(s) info which are created (include: order(s) and map) | \*Pre-condition: pass TC\_06 1, click on plan user want to view/edit | 1, - order(s) of this plan are displayed in the data table  - Map will be displayed the way DM will be going to | N/A | Normal |  |  |  |
| TC\_8 | Delete order of plan | \* Pre-condition: pass TC\_07 1, click Delete button on order user want to delete 2a, Click OK button 2b, Click Cancel button | 1, Confirm message will be displayed 2a, Redirect to Edit Plan(s) screen - The order is removed from the data table. Map is also updated | N/A | Normal |  |  |  |
| TC\_9 | Move order to another plan | \* Pre-condition: pass TC\_07 1, click Move to another plan button on order user want to remove from this plan 2, Choose plan user want to move the order to 3a, Click OK button 3b, Click Cancel button | 1, *Move to another plan* screen will be displayed - Combobox has plans list 3a, Redirect to Edit Plan(s) screen - The order is removed from the data table. Map is also updated  3b, Redirect to Edit Plan(s) screen - Nothing is changed | N/A | Normal |  |  |  |
| TC\_10 | Save modified plan(s) to database | \*Pre-condition: pass TC\_8 or TC\_9 1, click *Save* button | 1, Redirect to *Plan Management* screen - The table will be displayed the plan(s) are created | N/A | Normal |  |  |  |
| TC\_11 | Search *New* status | 1, Open *Plan Management* screen 2, Choose status = *New* 3, Click *Filter* button | 3, All plans have *New* status will be displayed in the data table | N/A | Normal |  |  |  |
| TC\_12 | Search *All* status | 1, Open *Plan Management* screen 2, Choose status = *All* 3, Click *Filter* button | 3, All plans will be displayed in the data table | N/A | Normal |  |  |  |
| TC\_13 | Search *Assigned* status | 1, Open *Plan Management* screen 2, Choose status = *Assigned* 3, Click *Filter* button | 3, All plans have *Assigned* status will be displayed in the data table | N/A | Normal |  |  |  |
| TC\_14 | Search *Canceled* status | 1, Open *Plan Management* screen 2, Choose status = *Canceled* 3, Click *Filter* button | 3, All plans have *Canceled* status will be displayed in the data table | N/A | Normal |  |  |  |
| TC\_15 | Search *Finished* status | 1, Open *Plan Management* screen 2, Choose status = *Finished* 3, Click *Filter* button | 3, All plans have *Finished* status will be displayed in the data table | N/A | Normal |  |  |  |
| TC\_16 | Search *None* status | 1, Open *Plan Management* screen 2, Choose status = *None* 3, Click *Filter* button | 3, All plans will be displayed in the data table | N/A | Normal |  |  |  |
| TC\_17 | Search multiple status | 1, Open *Plan Management* screen 2, Choose status = *New|Assigned|Canceled|Finished* 3, Click *Filter* button | 3, All plans have *New|Assigned|Canceled|Finished* status will be displayed in the data table | N/A | Normal |  |  |  |
| TC\_18 | Search by *Date* | 1, Open Plan Management screen 2, Input *From Date* and *To Date* value3, Click *Filter* button | 2, The date must be valid 3, All orders mapped search condition will be displayed in the data table | N/A | Normal |  |  |  |
| TC\_19 | Search by *From Date* | 1, Open Plan Management screen 2, Input *From Date - To Date* is blank3, Click *Filter* button | 2, The *From Date* must be valid 3, All orders mapped search condition will be displayed in the data table | N/A | Normal |  |  |  |
| TC\_20 | Search by *To Date* | 1, Open *Plan Management* screen 2, Input *To Date - From Date* is blank3, Click *Filter* button | 2, The *To Date* must be valid 3, All orders mapped search condition will be displayed in the data table | N/A | Normal |  |  |  |
| TC\_21 | View detail of each plan | 1, Open *Plan Management* screen 2, Click *View Detail* button on the plan | 2, *View Detail* screen will be displayed | N/A | Normal |  |  |  |
| TC\_22 | Assign Delivery men to plan | \*Pre-condition: Pass TC\_21 1, Check the DM to assign 2, Click *OK* button | 2, Redirect to *View Detail* screen - DM(s) are assigned to the plan | N/A | Normal |  |  |  |

#### Role ‘Buyer’

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test case No** | **Test case** | **Input** | **Expected Result** | **Status** | **Type** | **Note** |
| TC\_01 | Add product to Cart | 1. Open the main page 2. Click *Add* button on the product record | 2. The selected product will be moved to Cart area - Total price is updated | Passed | Normal |  |
| TC\_02 | Remove product from Cart | \*Pre-condition: pass TC\_01 1. Click *Remove* button on the product user doesn't need to buy | 1. The product will be removed from Cart. - Total price will be eliminated amount of removed product | Failed | Normal |  |
| TC\_03 | *OK* button with *Direct Delivery* | \*Pre-condition: pass TC\_01 1. Move to Customer Info area 2. Input the customer info, such as: Name: An Ngoc Anh Phone: 0912345678 Email: anhan@gmail.com Delivery type: Direct 3. Click OK button | 1. Focus on Customer Info area 2. Input must be valid - After chose Direct delivery: Address info area will be displayed. Then, user will input their address. 3. Show the confirm message | Failed | Normal |  |
| TC\_04 | *OK* button with *Buffer Delivery* | \*Pre-condition: pass TC\_01 1. Move to Customer Info area 2. Input the customer info, such as: Name: An Ngoc Anh Phone: 0912345678 Email: anhan@gmail.com Delivery type: Buffer delivery 3. Click OK button | 1. Focus on *Customer Info* area 2. Input must be valid - After chose Buffer delivery. Hub info area will be displayed. Then, user must choose Hub where receive  3. Show the confirm message and passcode | Failed | Normal |  |
| TC\_05 | *Cancel* button | \*Pre-condition: pass TC\_01 1. Move to *Customer Info* area 2. Input the customer info, such as: Name: An Ngoc Anh Phone: 0912345678 Email: anhan@gmail.com Delivery type: Buffer delivery 3. Click *Cancel* button | 1. Focus on *Customer Info* area 2. Input must be valid  3. Nothing happens | Failed | Normal |  |
| TC\_06 | *Name* field is blank | \*Pre-condition: pass TC\_01 1. Move to *Customer Info* area 2. Input Info as the following condition: - Name = "" | 2. Display (\*) behind textbox to remind user fill in his name |  |  |  |
| TC\_07 | *Phone number* is blank | \*Pre-condition: pass TC\_01 1. Move to *Customer Info* area 2. Input Info as the following condition: - Phone = "" | 2. Display (\*) behind textbox to remind user fill in his phone |  |  |  |
| TC\_08 | Invalid *phone number (special character)* | \*Pre-condition: pass TC\_01 1. Move to *Customer Info* area 2. Input Info as the following condition: - Phone = "!@$@$@@" | 2. show message: Phone number must be positive, whole number |  |  |  |
| TC\_09 | Invalid *phone number (character)* | \*Pre-condition: pass TC\_01 1. Move to *Customer Info* area 2. Input Info as the following condition: - Phone = "adsfds" | 2. show message: Phone number must be positive, whole number |  |  |  |
| TC\_10 | *Email* is blank | \*Pre-condition: pass TC\_01 1. Move to *Customer Info* area 2. Input Info as the following condition: - Email = "" | 2. Display (\*) behind textbox to remind user fill in his email |  |  |  |
| TC\_11 | Invalid *email* | \*Pre-condition: pass TC\_01 1. Move to *Customer Info* area 2. Input Info as the following condition: - Email = "adsfds" | 2. Show warning: Email must be in the format: [aA-zZ0-9]@[aA-zZ0-9].[aA-zZ0-9] |  |  |  |
| TC\_12 | Display address on map (*Direct delivery)* | \*Pre-condition: pass TC\_01 1. Move to Customer Info area 2. Input the customer info, such as: Name: An Ngoc Anh Phone: 0912345678 Email: anhan@gmail.com Choose Delivery type: Direct 3. Input buyer address | 2. show address area 3. Map will displayed buyer address |  |  |  |
| TC\_13 | Display address on map (*Buffer delivery)* | \*Pre-condition: pass TC\_01 1. Move to Customer Info area 2. Input the customer info, such as: Name: An Ngoc Anh Phone: 0912345678 Email: anhan@gmail.com Choose Delivery type: Buffer 3. Choose Hub type and Hub | 2. show Hub area 3. Map will displayed hub address |  |  |  |

## Checklist

N/A

### Checklist validation

N/A

### Submission Checklist

N/A

## Other material (if any)

N/A

### Checklist validation

N/A