**Software**

**Requirements**

**Specification**

**for**

**<Binh Tay Market App>**

**Prepared by <Group\_NgocThu>**

**<16/11/2021>**

|  |  |  |
| --- | --- | --- |
| Version | Created Day | By |
| 1.0 | 23/10/2021 | Group\_NgocThu |
| 2.0 | 30/10/2021 | Group\_NgocThu |
| 3.0 | 07/11/2021 | Group\_NgocThu |
| 4.0 | 14/11/2021 | Group\_NgocThu |

Table of Contents

[1 Introduction 6](#_Toc87913625)

[*1.1* Purpose 6](#_Toc87913626)

[*1.2* Document Conventions 6](#_Toc87913627)

[*1.3* Intended Audience and Reading Suggestions 6](#_Toc87913628)

[*1.4* Product Scope 6](#_Toc87913629)

[*1.5* References 6](#_Toc87913630)

[2 Overall Description 6](#_Toc87913631)

[*2.1* Product Perspective 6](#_Toc87913632)

[*2.2* Product Functions 7](#_Toc87913633)

[*2.3* User Classes and Characteristics 8](#_Toc87913634)

[*2.4* Operating Environment 8](#_Toc87913635)

[3 External Interface Requirements 9](#_Toc87913636)

[*3.1* User Interfaces 9](#_Toc87913637)

[*3.2* Hardware Interfaces 9](#_Toc87913638)

[4 System Features 9](#_Toc87913639)

[4.1 Login 9](#_Toc87913640)

[4.1.1 Overview 9](#_Toc87913641)

[4.1.2 Actors 9](#_Toc87913642)

[4.1.3 Use-case diagrams 9](#_Toc87913643)

[4.1.4 Workflows 10](#_Toc87913644)

[4.1.5 External interfaces: 10](#_Toc87913645)

[4.2 Register form 12](#_Toc87913646)

[4.2.1 Overview 12](#_Toc87913647)

[4.2.2 Actors 12](#_Toc87913648)

[4.2.3 Use-case diagrams 12](#_Toc87913649)

[4.2.4 Workflows 12](#_Toc87913650)

[4.2.5 External interfaces 13](#_Toc87913651)

[4.3 Search 14](#_Toc87913652)

[4.3.1 Overview 14](#_Toc87913653)

[4.3.2 Actors 14](#_Toc87913654)

[4.3.3 Use case diagram 14](#_Toc87913655)

[4.3.4 Workflows 14](#_Toc87913656)

[4.3.5 External Interfaces 15](#_Toc87913657)

[4.4 Update profile 16](#_Toc87913658)

[4.4.1 Overview 16](#_Toc87913659)

[4.4.2 Actors 16](#_Toc87913660)

[4.4.3 Use-case diagrams 16](#_Toc87913661)

[4.4.4 Workflow 16](#_Toc87913662)

[4.4.5 External Interfaces 17](#_Toc87913663)

[4.5 Track order progress 18](#_Toc87913664)

[4.5.1 .Overview 18](#_Toc87913665)

[4.5.2 Use-case diagram 19](#_Toc87913666)

[4.5.3 Workflows 19](#_Toc87913667)

[4.5.4 External interfaces: 20](#_Toc87913668)

[4.6 Cancel order: 20](#_Toc87913669)

[4.6.1 Overview 20](#_Toc87913670)

[4.6.2 Actors 21](#_Toc87913671)

[4.6.3 Use-case diagram 21](#_Toc87913672)

[4.6.4 Workflows 21](#_Toc87913673)

[4.6.5 External interfaces 22](#_Toc87913674)

[4.7 View/ Update Shopping Cart 22](#_Toc87913675)

[4.7.1 Overview 22](#_Toc87913676)

[4.7.2 Actors 23](#_Toc87913677)

[4.7.3 Use-case diagrams 23](#_Toc87913678)

[4.7.4 Workflows 23](#_Toc87913679)

[4.7.5 External interfaces 24](#_Toc87913680)

[4.8 View product details 25](#_Toc87913681)

[4.8.1 Overview 25](#_Toc87913682)

[4.8.2 Actors 25](#_Toc87913683)

[4.8.3 Use-case diagrams 25](#_Toc87913684)

[4.8.4 Workflows 26](#_Toc87913685)

[4.8.5 External interfaces 28](#_Toc87913686)

[4.9 Report saleman’s behaviour 28](#_Toc87913687)

[4.9.1 Overview 28](#_Toc87913688)

[4.9.2 Actors 28](#_Toc87913689)

[4.9.3 Use-case diagrams 29](#_Toc87913690)

[4.9.4 Workflows 29](#_Toc87913691)

[4.9.5 External interfaces 30](#_Toc87913692)

[4.10 Pay The Bill 32](#_Toc87913693)

[4.10.1 Overview 32](#_Toc87913694)

[4.10.2 Actors 32](#_Toc87913695)

[4.10.3 Use-case diagrams 32](#_Toc87913696)

[4.10.4 Workflows 32](#_Toc87913697)

[4.10.5 External interfaces 34](#_Toc87913698)

[4.11 3D technology for dress rehearsal 39](#_Toc87913699)

[4.11.1 Overview 39](#_Toc87913700)

[4.11.2 Actors 39](#_Toc87913701)

[4.11.3 Use-case diagrams 39](#_Toc87913702)

[4.11.4 Workflows 39](#_Toc87913703)

[4.11.5 External interfaces 40](#_Toc87913704)

[4.12 Add/ remove product 41](#_Toc87913705)

[4.12.1 Overview 42](#_Toc87913706)

[4.12.2 Actors 42](#_Toc87913707)

[4.12.3 Use-case diagrams 42](#_Toc87913708)

[4.12.4 Workflows 42](#_Toc87913709)

[4.13 Confirm order 43](#_Toc87913710)

[4.13.1 Overview 43](#_Toc87913711)

[4.13.2 Actors 43](#_Toc87913712)

[4.13.3 Use-case diagrams 44](#_Toc87913713)

[4.13.4 Workflows 44](#_Toc87913714)

[5 Other Nonfunctional Requirements 45](#_Toc87913715)

[*5.1* Performance Requirements 45](#_Toc87913716)

[*5.2* Safety Requirements 45](#_Toc87913717)

[*5.3* Security Requirements 45](#_Toc87913718)

[*5.4* Software Quality Attributes 46](#_Toc87913719)

[6 Other Requirements 46](#_Toc87913720)

# Introduction

## Purpose

The purpose of this document is to build an online sales system based on AI technology connected to Binh Tay market

## Document Conventions

|  |  |
| --- | --- |
| Term | Definition |
| AI | Artificial Intelligence |
| 3D | Three-dimensional |

## Intended Audience and Reading Suggestions

This is a specification document for Binh Tay Market App, the document is made under the guidance of the lecturer. This document is intended for customers, developers and testers.

## Product Scope

The purpose of this sales system is to allow buyers to easily access and buy products from Binh Tay market with convenient support from AI, sellers can easily post available products at the market to increase their popularity. accessible to customers who do not go to the market to buy directly.

## References

*IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.*

*Software Requirement Specification with example:*

[*https://krazytech.com/projects/sample-software-requirements-specificationsrs-report-airline-database?fbclid=IwAR3CGxy2LYeVlkXCSN6P9oC849128wGLcQ7NO83Z9-c4byl0U78nW01f3oc*](https://krazytech.com/projects/sample-software-requirements-specificationsrs-report-airline-database?fbclid=IwAR3CGxy2LYeVlkXCSN6P9oC849128wGLcQ7NO83Z9-c4byl0U78nW01f3oc)

*Writing a software requirements specification document:*

[*https://medium.com/@enisinanaj/writing-a-software-requirements-specification-document-97d622805aef*](https://medium.com/@enisinanaj/writing-a-software-requirements-specification-document-97d622805aef)

# Overall Description

## Product Perspective

## Product Functions

* Search & filter
* Anti-counterfeiting AI scanning technology
* 3D technology for trying on clothes online
* Locating the address
* Provide facial recognition images for drone delivery
* Online payment
* AI confirm orders automatically
* Multi-platform support

## User Classes and Characteristics

Diagram

Description automatically generated

## Operating Environment

* Android
  + RAM: 2GB or higher
  + Android version 4.0.3 or higher
  + More than 300MB of free memory space
* IOS:
  + Version 8.0 and above
  + Free space of 400MB or more
* Windows

# External Interface Requirements

## User Interfaces

* Front-end software: React Native
* Back-end software: .NET Core
* Database software: MySQL

## Hardware Interfaces

* Windows
* IOS
* Android

# System Features

## Login

### Overview

* This screen is used to login to BinhTayMarketApp

### Actors

* All project members can login to BinhTayMarketApp

### Use-case diagrams

Diagram

Description automatically generated

#### Pre-condition

* User must open BinhTayMarketApp

#### Post-condition

* User will be able to use BinhTayMarketApp

#### Triggers

* Clicks on the “Login as” button on the login form

### Workflows

|  |  |  |
| --- | --- | --- |
| Scenario | Actor | System |
| 1. User successfully login | 1.1. Go to this login form | 1.2. Display login form |
| 1.3 Enter username, password and click login button or click login with social media account (facebook, google, twitter) | 1.4 Display homepage |
| 2. User inputs invalid data | 2.1 Refer to step 1.1 | 2.2 Refer to step 1.2 |
| 2.3 Refer to step 1.3 | 2.4 Dialog error messages of invalid fields will be displayed at the center of screen |

### External interfaces:

#### Prototype:

Graphical user interface, application

Description automatically generated

#### Screen description:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Field name | Control type | Required | Data type | Default value | Description |
| 1 | Username | Textbox | Yes | Text | Blank | Username contains minimum 6 characters, maximum 40 characters |
| 2 | Password | Textbox | Yes | Text | Blank | Password contains minimum 6 characters, maximum 40 characters |
| 3 | Login as customer | Link label | Yes | Text | N/A | Select the login status as a customer |
| 4 | Login as salesman | Label | Yes | Text | N/A | Select the login status as a salesman |
| 5 | Forgot password | Link label | Yes | Text | N/A | Link to change password form |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Button | | | | | | |
| 1 | Login with Facebook | button | Yes | N/A | N/A | Login with Facebook account |
| 2 | Login with Gmail | Button | Yes | N/A | N/A | Login with Google account |
| 3 | Login with Twitter | button | Yes | N/A | N/A | Login with Twitter account |
| 4 | Sign up | button | Yes | N/A | N/A | Login with username and password |

## Register form

### Overview

* This screen is used to sign up account

### Actors

* Salesman and Customer

### Use-case diagrams

Diagram

Description automatically generated

#### Pre-condition

* User must open BinhTayMarketApp

#### Post-condition

* User will be able to BinhTayMarketApp

#### Triggers

#### Clicks on the “Sign up” button on the login form

### Workflows

|  |  |  |
| --- | --- | --- |
| Scenario | Actor | System |
| 1. User successfully sign up | 1.1. Go to this Register form | 1.2. Display Register form |
| 1.3. Enter all textbox, choose gender in combobox gender, then add a front of ID photo and behind ID-card photo, then click sign up button | 1.4 Show success message |
| 2. User inputs invalid data | 2.1 Refer to step 1.1 | 2.2 Refer to step 1.2 |
| 2.3 Refer to step 1.3 | 2.4 Dialog error messages of invalid fields will be displayed at the center of screen |

### External interfaces

#### Prototype:

Graphical user interface, text, application

Description automatically generatedGraphical user interface, application

Description automatically generated

#### Screen description:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Field name | Control type | Required | Data type | Default value | Description |
| 1 | Firstname | Textbox | Yes | Text | Blank | Maximum 50 characters |
| 2 | Lastname | Textbox | Yes | Text | Blank | Maximum 50 characters |
| 3 | Phone number | Textbox | Yes | Text | Blank | Integer 0-9  9-11 characters |
| 4 | Email | Textbox | Yes | Text | Blank | Must contain @ |
| 5 | Birthday | Datetime picker | Yes | Date | Date | Format date dd/MM/yyyy |
| 6 | Gender | Combo box | Yes | Text | Male | Select gender |
| Picture box | | | | | | |
| 7 | Front | Picture box | Yes | Image | Blank | Contains the front of the ID card |
| 8 | Behind | Picture box | Yes | Image | Blank | Contains the back of the ID card |
| Button | | | | | | |
| 9 | Next | Button | Yes | N/A | N/A | Click to next step |
| 10 | Sign up | Button | Yes | N/A | N/A | Click to sign up |

## Search

### Overview

* This screen is used to search products

### Actors

* Salesman and customer

### Use case diagram

Diagram

Description automatically generated

#### Pre-condition

* User must click search box

#### Post-condition

* Products will display on the screen

#### Triggers

#### Clicks on the “Search” button

### Workflows

|  |  |
| --- | --- |
| Actor | System |
| 1.1. Go to this search form | 1.2. Display search form, search history and recommended keyword |
| 1.3. Enter tagname, categries, product name | 1.4 Show the respective products |

### External Interfaces

#### Protype

Graphical user interface, text

Description automatically generated

#### Screen description:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Field name | Control type | Required | Data type | Default value | Description |
| 1 | Searchbox | Textbox | Yes | Text | Blank | Search content |
| 2 | Voice recognition | Icon | No | N/A | N/A | Convert speech to text in search input |
| 3 | History | List | No | Text | Text | History search list |
| 4 | Most searched products | List | Yes | Text | Text | Most searched products |
| Button | | | | | | |
| 5 | Search | Button | Yes | N/A | N/A | Click to execute search progress |
| 6 | Cart | Button | Yes | N/A | N/A | Click to show cart |

## Update profile

### Overview

* This screen is used to update profile

### Actors

* Salesman and Customer

### Use-case diagrams

A picture containing diagram

Description automatically generated

4.3.3.1. Pre-condition

* User click on “Profile” button

4.3.3.2. Post-condition

* Update profile form will be dispayed

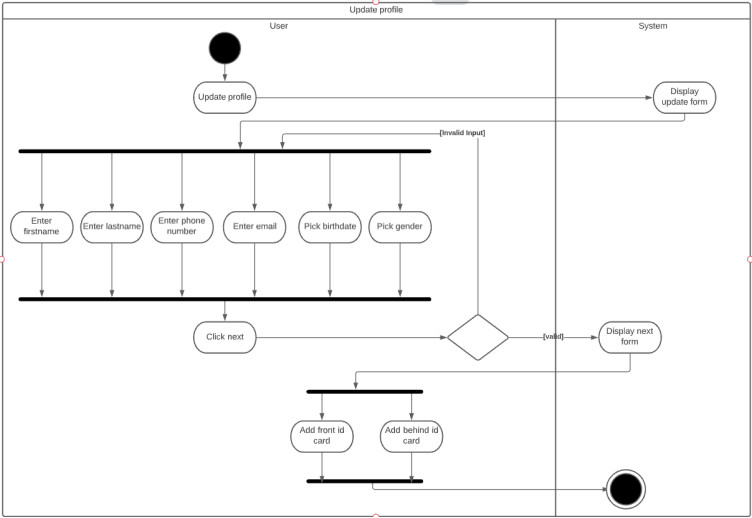
4.3.3.3. Triggers

* Clicks on the “Done” button on the update profile form

### Workflow

|  |  |
| --- | --- |
| Actor | System |
| 1.1. Go to this Register form | 1.2. Display register form |
| 1.3. Enter all textbox, choose gender in combobox gender, then add a front of ID photo and behind ID-card photo to corresponding picture box, then click sign up button | 1.4 Show success message |

### External Interfaces



#### Prototype:

Graphical user interface, application

Description automatically generatedGraphical user interface, application

Description automatically generated

#### Screen description:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Field name | Control type | Required | Data type | Default value | Description |
| 1 | Firstname | Textbox | Yes | Text | Blank | Maximum 50 characters |
| 2 | Lastname | Textbox | Yes | Text | Blank | Maximum 50 characters |
| 3 | Phone number | Textbox | Yes | Text | Blank | Contains 9-11 characters |
| 4 | Email | textbox | Yes | Text | Blank | Must contain @ |
| 5 | Birthday | datetime picker | Yes | Date | Date | Format date dd/MM/yyyy |
| 6 | Gender | Combo box | Yes | Text | Male | Select gender |
| Picture box | | | | | | |
| 7 | Front | picture box | Yes | Image | Blank | Front image if the ID card |
| 8 | Behind | picture box | Yes | Image | Blank | Behind image if the ID card |
| Button | | | | | | |
| 9 | Next | button | Yes | N/A | N/A | Next step |
| 10 | Done | button | Yes | N/A | N/A | Complete update profile |

## Track order progress

### .Overview

* This screen is used to Track order progress

### Use-case diagram

A picture containing text

Description automatically generated

#### Pre-condition

User must be opened BinhTayMarketApp

#### Post-condition

User was login to BinhTayMarketApp and placed order

#### Triggers

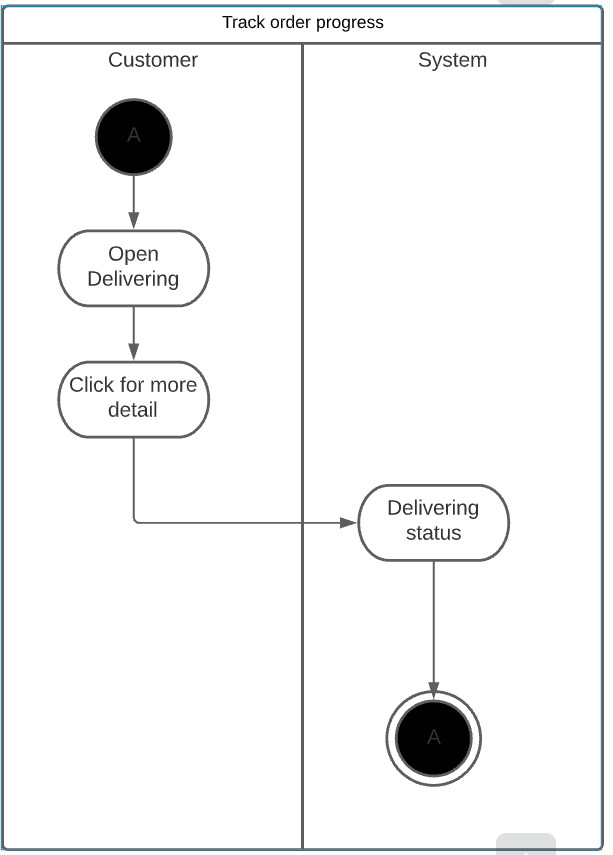
Clicks on the “Click more detail” button to track order progress

### Workflows

|  |  |  |
| --- | --- | --- |
| Scenario | Actor | System |
| 1. User successfully  update | 1.1. Go to this Register form | 1.2. Display update form |
| 1.3. Enter all textbox, choose gender in combobox gender, then add a front of ID photo and behind ID-card photo to corresponding picture box, then click done button | 1.4 Show success message |
| 2. User inputs invalid data | 2.1 Refer to step 1.1 | 2.2 Refer to step 1.2 |
| 2.3 Refer to step 1.3 | 2.4 Validate fields and valid format  If there are still errors, dialog error messages of invalid fields will be displayed at the center of screen |

### External interfaces:

#### Prototype:



Graphical user interface, application

Description automatically generatedGraphical user interface, application

Description automatically generated

#### Screen description:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Field name | Control type | Required | Data type | Default value | Description |
| 1 | Click for more detail | link label | Yes | Text | Blank | Linking to track progress  screen |
| 2 | Google map | textbox | Yes | Text | map | Shipper's location |
| 3 | Delivery information | textarea | Yes | Text | Text | includes progress made and progress in progress |

## Cancel order:

### Overview

* This screen is used to Cancel order

### Actors

* Customer

### Use-case diagram

A picture containing text

Description automatically generated

#### Pre-condition

User must be opened BinhTayMarketApp

#### Post-condition

User was login to BinhTayMarketApp and placed a order

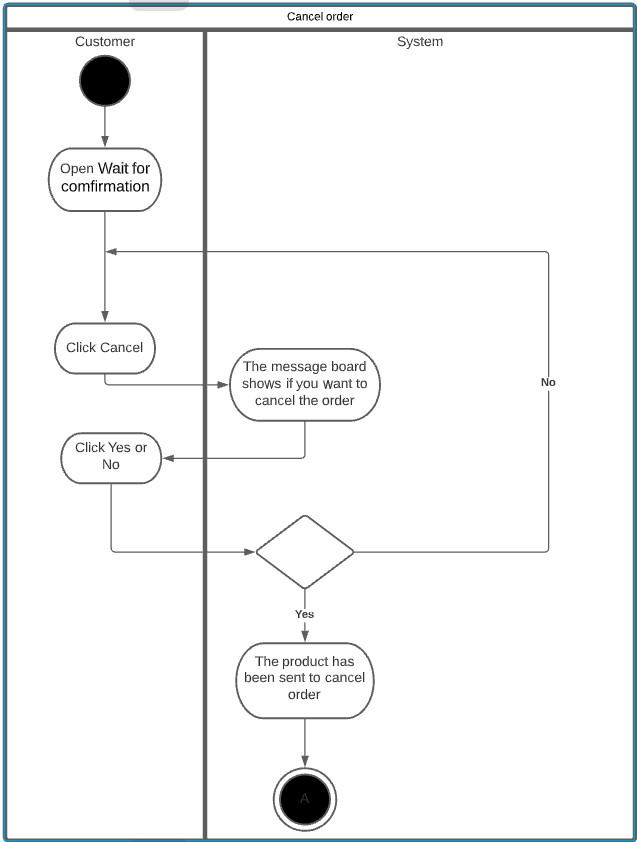
#### Triggers

Clicks on the “Cancle” button to Cancel order

### Workflows

|  |  |  |
| --- | --- | --- |
| Scenario | Actor | System |
| 1. User successfully sign up | 1.1. Go to “Wait for comfirmation” screen | 1.2. Display “Wait for comfirmation” screen, show detail order form |
| 1.3. Click cancle buttom | 1.4 show message that you want to cancel the order or not |
|  | 1.5 Click yes | 1.6 The order will be canceled |
| 2. User click no | 2.1 Refer to step 1.1 | 2.2 Refer to step 1.2 |
| 2.3 Refer to step 1.3 | 2.4 Refer to step 1.4 |
| 2.5 Refer to step 1.5 | 2.6 Refer to step 1.6 |

### External interfaces



#### Prototype:

Graphical user interface, application

Description automatically generatedGraphical user interface, application

Description automatically generated

#### Screen description:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Field name | Control type | Required | Data type | Default value | Description |
| Defect detail | | | | | | |
| 1 | Check | Check  box | Yes | N/A | Checked |  |
| 2 | Product image | Picture Box | Yes | Text | Blank | Maximum 50 characters |
| 3 | Quantity | Input | Yes | Number | 1 | Integer 0-9 |
| 4 | Price | label | Yes | Text | N/A |  |

## View/ Update Shopping Cart

### Overview

This screen is used to view/update cart

### Actors

Customer

### Use-case diagrams

A picture containing diagram

Description automatically generated

#### Pre-condition

User must be logged in

#### Post-condition

Notice of successful update shopping cart

#### Triggers

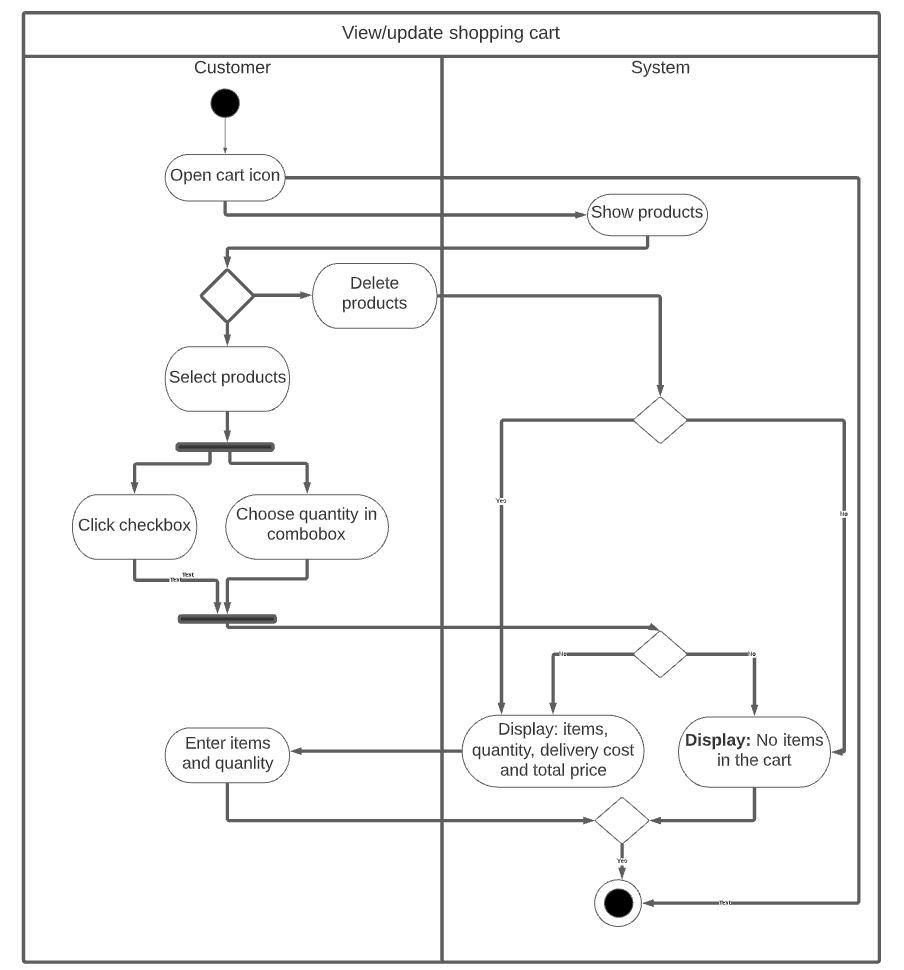
#### Clicks on the button

### Workflows

|  |  |  |
| --- | --- | --- |
| **Scenario** | **Actor** | **System** |
| 1. User successfully view/update shopping cart | 1.1. Go to view/update cart form | 1.2. Display view/update shopping cart form |
| 1.3 Select item and quantity | 1.4 Update from user input |
| 2. User inputs invalid data | 2.1 Refer to step 1.1 | 2.2 Refer to step 1.2 |
| 2.3 Refer to step 1.3 | 2.4 Validate mandatory fields and valid format  If there are still errors, dialog error messages of invalid fields will be displayed at the center of screen |

### External interfaces

#### Prototype:



Graphical user interface, application

Description automatically generated

#### Screen description:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Field name | Control type | Required | Data type | | Default value | Description |
| Defect detail | | | | | | | |
| 1 | Empty box | checkbox | Yes | | Text | Blank | choose each item |
| 2 | Increase or decrease | Num.Stepper | Yes | | Text | 1 | increase or decrease the amount |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Button | | | | | | |
| 1 | Back | button | Yes | N/A | N/A | Back to product details page |
| 2 | Purchase | button | Yes | N/A | N/A | Pay the bill |

## View product details

### Overview

This screen is used view product details

### Actors

Customer

### Use-case diagrams

Text

Description automatically generated

#### Pre-condition

* User must be logged in

#### Post-condition

* Notice of successful update shopping cart

#### Triggers

* Click on any products

### Workflows

|  |  |  |
| --- | --- | --- |
| **Scenario** | **Actor** | **System** |
| 1. User successfully view/update shopping cart | 1.1. Go to this screen | 1.2. Display view/update shopping cart form |
| 1.3 Select item and quantity | 1.4 Update from user input |
| 2. User inputs invalid data | 2.1 Refer to step 1.1 | 2.2 Refer to step 1.2 |
| 2.3 Refer to step 1.3 | 2.4 Validate mandatory fields and valid format  If there are still errors, dialog error messages of invalid fields will be displayed at the center of screen |

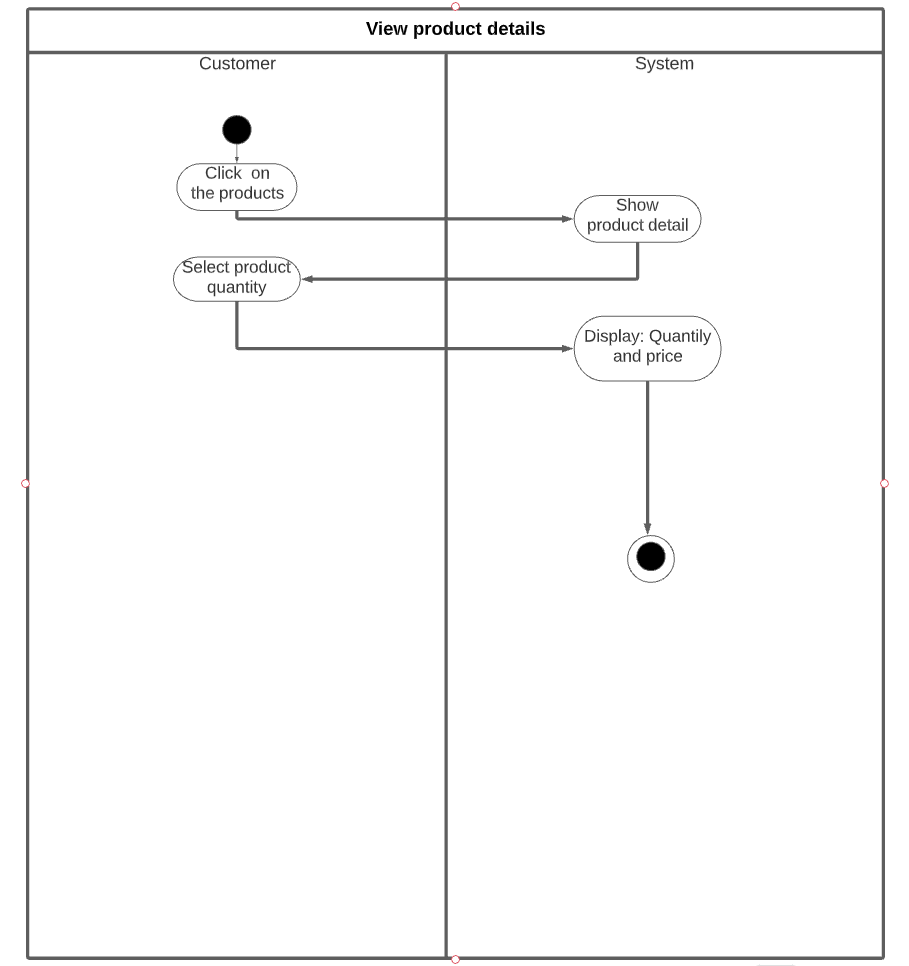
#### Screen description:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1. No | Field name | Control type | Required | Data type | Default value | Description |
| Defect detail | | | | | | |
| 1 | Image | image | Yes | image | Blank | product images |
| 2 | Size | Combobox | Yes | Text | Blank | size |
| 3 | Price | Label | Yes | Number | Blank | product price |
| 4 | Address | Combobox | Yes | Text | Blank | receiver's address |
| 5 | Delivery cost | Label | Yes | Number | Blank | Delivery cost |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Button | | | | | | |
| 1 | Back | button | Yes | N/A | N/A | Back to product details page |
| 2 | Bell | button | Yes | N/A | N/A | Get notification |
| 3 | Shopping cart | button | Yes | N/A | N/A | View Customer’s shopping cart |
| 4 | Try on | button | Yes | N/A | N/A | try on clothes |
| 5 | Add to cart | button | Yes | N/A | N/A | Add item to cart |
| 6 | Buy Now | button | Yes | N/A | N/A | Go to the payment page |
| 7 | View shop | button | Yes | N/A | N/A | View shop |

### External interfaces

#### Prototype:



Graphical user interface, application

Description automatically generated

## Report saleman’s behaviour

### Overview

This screen is used to report saleman’s behaviour

### Actors

Customer

### Use-case diagrams

A picture containing graphical user interface

Description automatically generated

#### Pre-condition

User must be logged in

#### Post-condition

Notice of successful report

#### Triggers

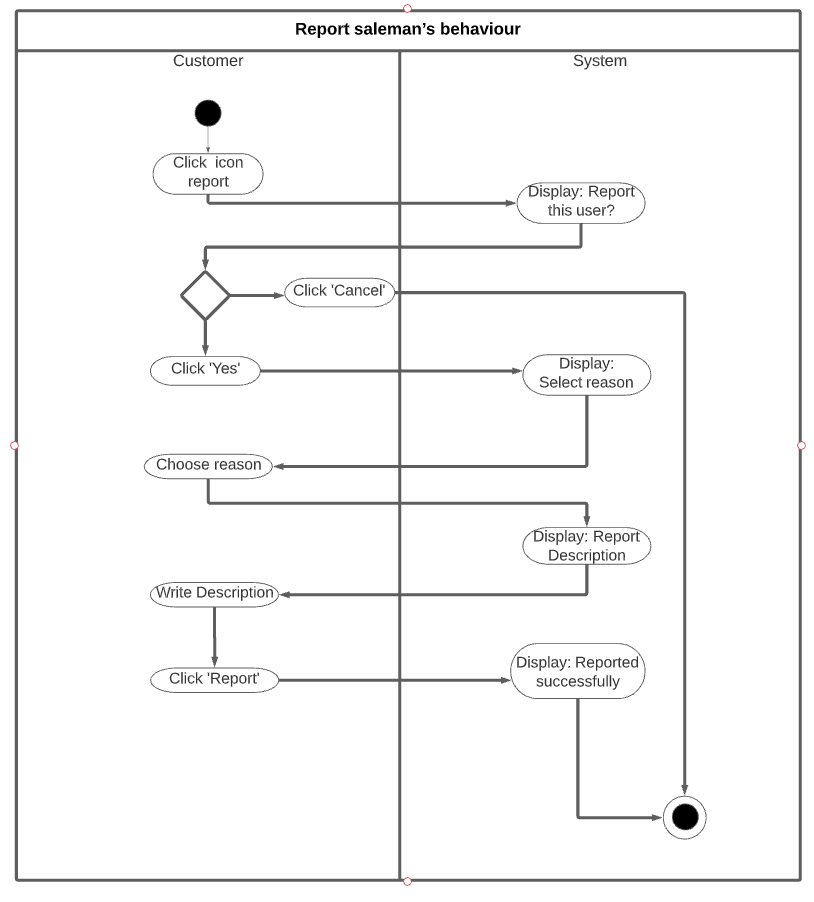
 In the saleman page, press the button to report

### Workflows

|  |  |  |
| --- | --- | --- |
| **Scenario** | **Actor** | **System** |
| 1. User successfully view/update shopping cart | 1.1. Go to “saleman page” and press the button | 1.2. Show notice “Report this user?” |
| 1.3 Select “Yes” | 1.4 Display “Report form” |
| 1.5 Select reason to report | 1.6 Display “Report description” |
| 1.7 Write content and press button “Report” | 1.8 Show notice “Report successfully” |
| 2. User inputs invalid data | 2.1 Refer to step 1.1 | 2.2 Refer to step 1.2 |
| 2.3 Refer to step 1.3 | 2.4 Validate mandatory fields and valid format  If there are still errors, dialog error messages of invalid fields will be displayed at the center of screen |

### External interfaces

#### Prototype:



Shape

Description automatically generatedGraphical user interface, text, application, chat or text message

Description automatically generated

Text

Description automatically generated Graphical user interface

Description automatically generated

#### Screen description:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Field name | Control type | Required | Data type | | Default value | Description |
| Defect detail | | | | | | | |
| 1 | Select Reason | Label | Yes | | Text | Blank | choose a reason to report |
| 2 | Report Description | Textbox | Yes | | Text | Blank | Maximum 50 characters  Detail the content of the report |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Button | | | | | | |
| 1 | Back | button | Yes | N/A | N/A | Back to saleman’s page |
| 2 | Report | button | Yes | N/A | N/A | Send report |

## Pay The Bill

### Overview

Pay the products in the cart

### Actors

System

### Use-case diagrams

A picture containing graphical user interface

Description automatically generated

#### Pre-condition

User must be logged in

#### Post-condition

Notice of successful bill payment

#### Triggers

Click on on the Defect View

### Workflows

|  |  |  |
| --- | --- | --- |
| **Scenario** | **Actor** | **System** |
| 1. User successfully paid | 1.1. Go to this screen | 1.2. Display “Cart” screen. |
| 1.3 User can increase or decrease quantity of products and Tick the product want to buy and click “Buy” button. | 1.4. Validate mandatory fields and valid format.  1.5. All fields are valid, and Display “Purchase” screen. |
| 2. User inputs invalid data | 2.1 Refer to step 1.1 | 2.2 Refer to step 1.2 |
| 2.3 Refer to step 1.3 | 2.4 Validate mandatory fields and valid format  If there are still errors, dialog error messages of invalid fields will be displayed at the center of screen |
| 3.User Purchase product | 3.1 Go to “Purchase” screen | 3.2 Display “Purchase” screen |
| 3.2 User can edit the address | 3.3 Display “Address” screen |
| 3.4 User can increase or decrease the number of products and Tick the products you want to pay and click “Next” button. | 3.5 All fields are valid, and Display “Order Payment” screen. |
| 1. User Edit Address | 4.1 Go to “Address” screen | 4.2 Display “Address” screen. |
| 4.3 Fill in mandatory fields and optional fields and click “Done” button | 4.4 Validate mandatory fields and valid format.  All fields are valid, the new Address will be saved to the database and returned to the “Purchase” screen. |
| 1. User Order Payment | 5.1 Go to “Order Payment” screen | 5.2 Display “Order Payment” screen |
| 5.2 Fill in mandatory fields and click “Next” button | 5.3 Validate mandatory fields and valid format.  All fields are valid, the data will be saved to the database and appears successful order message displayed at the center of screen |

|  |
| --- |
|  |

### External interfaces

#### Prototype:

Cart Homepage

Graphical user interface, application

Description automatically generated

Button



LinkLabel



Purchase

Graphical user interface, application

Description automatically generated

Button



Linklabel



Edit Address

Graphical user interface, application

Description automatically generated

Combobox

Text

Description automatically generatedText

Description automatically generatedText

Description automatically generated

Button



Order Payment

Graphical user interface, application

Description automatically generated

Textbox

Shape, rectangle

Description automatically generated

A picture containing graphical user interface

Description automatically generated

DateTimePicker

A picture containing text

Description automatically generated

Button



## 3D technology for dress rehearsal

### Overview

Pay the products in the cart

### Actors

System

### Use-case diagrams

A picture containing graphical user interface

Description automatically generated

#### Pre-condition

User must be logged in

#### Post-condition

3D body image

#### Triggers

In “Product clothes” screen , user click “try on” button

### Workflows

|  |  |  |
| --- | --- | --- |
| **Scenario** | **Actor** | **System** |
| 1. User select product | 1.1. Go to “Cloth detail” screen | 1.2. Display “Cloth detail” screen. |
| 1.3 Fill in mandatory fields and optional fields and click “try on” button | 1.4. Validate mandatory fields and valid format and display image 3D |
| 1.5 Click on “Add to cart” button | 1.6 Add product to card and save to database |
| 1.6 Click on “Buy now” | 1.7 Display “Edit Address” screen.  1.8 Enter your shipping address |
| 1.9 Click on “View Shop” | 1.10 Back to product catalog |
| 2. User inputs invalid data | 2.1 Refer to step 1.1 | 2.2 Refer to step 1.2 |
| 2.3 Refer to step 1.3 | 2.4 Validate mandatory fields and valid format  If there are still errors, dialog error messages of invalid fields will be displayed at the center of screen |

### External interfaces

#### Prototype:

**3D technology for dress rehearsal**

Application

Description automatically generated

Button

A picture containing shape

Description automatically generated 

Combobox

Text

Description automatically generated with medium confidence Diagram

Description automatically generated

LinkLabel



## Add/ remove product

### Overview

Add new product or remove product

### Actors

Salesman

### Use-case diagrams

Diagram

Description automatically generated with medium confidence

#### Pre-condition

Salesman must be logged in

#### Post-condition

Products are added or removed from the database

#### Triggers

.In “My Shop” screen. Click on “Add new products”

### Workflows

|  |  |  |
| --- | --- | --- |
| **Scenario** | **Actor** | **System** |
| 1.User successfully add new product | 1.1. Go to “Add new products” screen | 1.2. Display “Add new products” screen. |
| 1.3 user can attach file and QR code by clicking icon | 1.4. Submit to “Attach image” screen  After user attached image, return to “Add new products” screen with current status and values of all fields and control except change in attachment area. |
| 1.5 Click on “Add” button | 1.6 Add product to products catalog and save to database |
| 2. User inputs invalid data | 2.1 Refer to step 1.1 | 2.2 Refer to step 1.2 |
| 2.3 Refer to step 1.3 | 2.4 Validate mandatory fields and valid image  If there are still errors, dialog error messages of invalid fields will be displayed at the center of screen |

## Confirm order

### Overview

Handling single orders

### Actors

Salesman

### Use-case diagrams

A picture containing diagram

Description automatically generated

#### Pre-condition

Salesman must be logged in

#### Post-condition

Unprocessed orderss will be removed from the database

#### Triggers

In “My Shop” screen. Click on “wait for confirmation”

### Workflows

|  |  |  |
| --- | --- | --- |
| **Scenario** | **Actor** | **System** |
| 1. User select product | 1.1. Go to “wait for confirmation” screen | 1.2. Display “wait for confirmation” screen. |
| 1.3 Fill in mandatory fields and optional fields and click “Confirm” button | 1.4. |
| 1.5 Click on “Add to cart” button | 1.6 Add product to card and save to database |
| 1.6 Click on “Buy now” | 1.7 Display “Edit Address” screen.  1.8 Enter your shipping address |
| 1.9 Click on “View Shop” | 1.10 Back to product catalog |
| 2. User inputs invalid data | 2.1 Refer to step 1.1 | 2.2 Refer to step 1.2 |
| 2.3 Refer to step 1.3 | 2.4 Validate mandatory fields and valid format  If there are still errors, dialog error messages of invalid fields will be displayed at the center of screen |

# Other Nonfunctional Requirements

## Performance Requirements

* The application loads to the homepage within 3 seconds
* The application responds to the user's actions within 2 seconds

## Safety Requirements

* Database does not lose data
* Regular backups
* Encrypted customer information
* Do not show customer password unless customer clicks show password
* Only display the last 4 digits of the customer's bank card account

## Security Requirements

* Two-factor authentication
* Send notification of customer's phone number/email when another device logs in to the account
* The app automatically asks for the password when the customer opens the app after 5 minutes of inactivity
* Login with password, fingerprint, face recognition
* The system always uses Secure Sockets Layer (SSL) in all transactions containing sensitive customer information.

## Software Quality Attributes

* Availability: available goods are sufficient to meet customer needs
* Correctness: accurate display and transmission information: which seller's order information is only transmitted to that seller, the drone recognizes the face correctly before delivering to the customer
* Usability: users can easily use the system without any further complicated instructions

# Other Requirements