Software Design Document

**Document History**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Document Name** | **Version** | **Status** | **Date** | **Viewable** | **Reviewer** | **Responsible** |
| iWish - SDD V.[0.1].docx | **Add**   * Introduction * Class Diagram * Class Description * DB * UI * SDW * SDA | Draft | 20-05-2015 | VC, SK, KS | VC, SK | VC, SK |
| iWish - SDD V.[1.0].docx | **Edit**   * Introduction * Class Diagram * Class Description * DB * UI * SDW * SDA | Release | 22-07-2015 | VC, SK, KS | VC, SK | VC, SK |
| iWish - SDD V.[1.5].docx | **Edit**   * Introduction | Release | 23-08-2015 | VC, SK, KS | VC, SK | VC, SK |
| iWish - SDD V.[2.0].docx | **Add**   * SDA * Class diagram * Class description * UI | Draft | 10-10-2015 | VC, SK, KS | VC, SK | VC, SK |
| iWish - SDD V.[2.0].docx | **Edit**   * Class description | Release | 21-10-2015 | VC, SK, KS | VC, SK | VC, SK |

**\*VC = Visava Chumnuam**

**\*SK = Saksorn Khongsirirat**

**\*KS = Kittitouch Suteeca**

**Table of Contents**

1. Introduction 4

1.1 Project Overview 4

1.1.1 Purpose 4

1.1.2 Scope 4

1.1.3 Acronyms …………………………………………………………………………………………....4

2. System Architecture 5

3. Class Diagram 6

4. Class Description 8

5. Database Design 30

6. Sequence Diagram 31

7. User Interface 45

# **Introduction**

**1.1 Purpose**

The purpose of software design is creating the iWish’s system. The system has Android application and Web application. This document contains about Class Diagram, Database, User Interface and Sequence Diagram.

**1.2 Scope**

The application will be develop by Android studio.

* Focus on only one supermarket.
* Provide management system products in database.
* Provide android application for manage wish list.
* Convenient features for android application site.
  1. **Acronyms and Definition**

**Acronyms**

SDA Sequence Diagram Android

SDW Sequence Diagram Web

SK Saksorn Khongsirirat

SRSA System Requirement Specification Android

SRSW System Requirement Specification Web

UIA User Interface Android

UIW User Interface Web

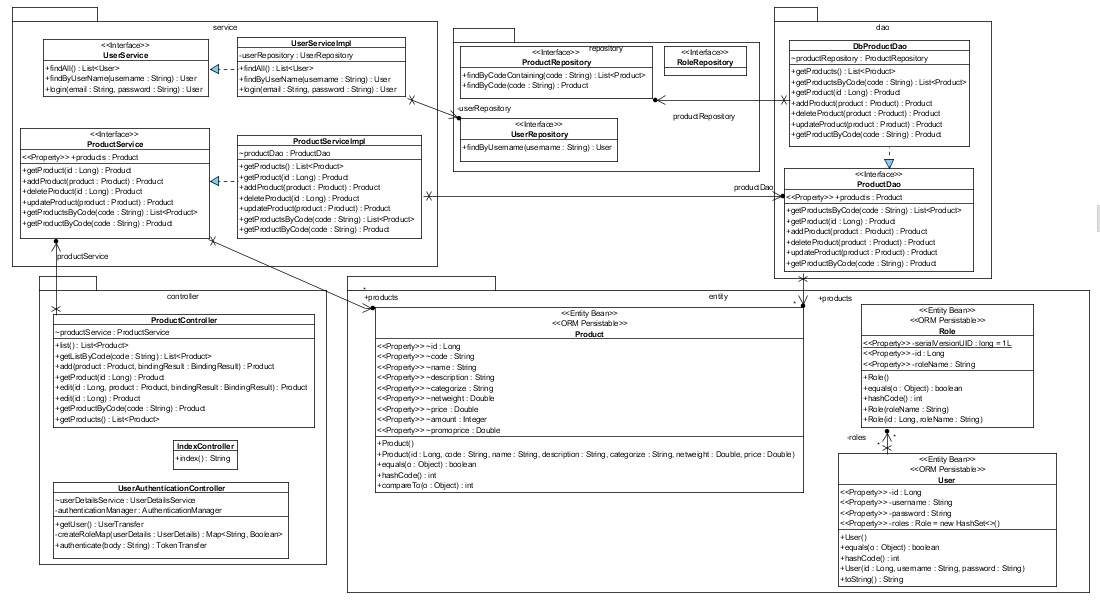
VC Visava Chumnuam

1. **System Architecture**

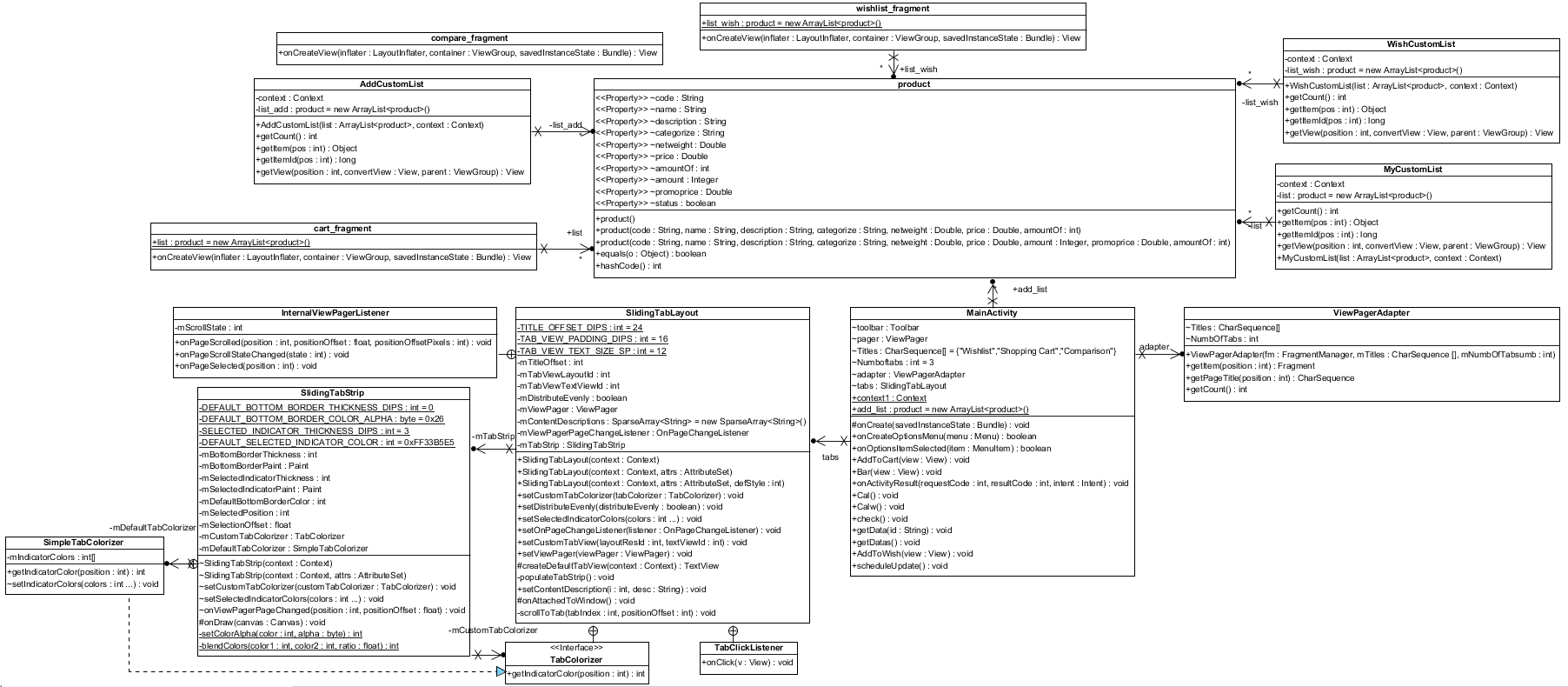
****

**Figure 23: Class Diagram Web**

1. **Class Diagram**

****

**Figure 21: Class Diagram Web**

****

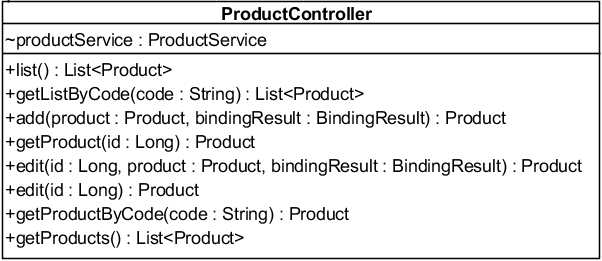
**Figure 22: Class Diagram Android**

1. **Class Description**

**Web Application**

**Controller package**

* 1. Class name: ProductController



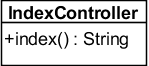
**Property:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | productService | Object of ProductService, used for control feature about product. | **Type:** ProductService |

**Method:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | list | Method for send all product in the system to client on web. | **Return:** List<Product> |
| 2 | getListByCode | Method for send list of product that similar to code in the system to client on web. | **Return:** List<product>  **Param:** String |
| 3 | add | Method for receive product from client’s product’s information form to add product in the system. | **Return:** Product  **Param:** Product, BindingResult |
| 4 | getProduct | Method for send specific product in the system to show to client on web. | **Return:** Product  **Param:** Long |
| 5 | edit | Method for receive new information to update product in the system. | **Return:** Product  **Param:** Long, Product, BindingResult |
| 6 | edit | Method for receive delete product that client request from the system. | **Return:** Product  **Param:** Long |
| 7 | getProductByCode | Method for send specific product that same code in the system to client on app. | **Return:** Product  **Param:** String |
| 8 | getProductsByCode | Method for send all product in the system to client on app. | **Return:** List<Product> |

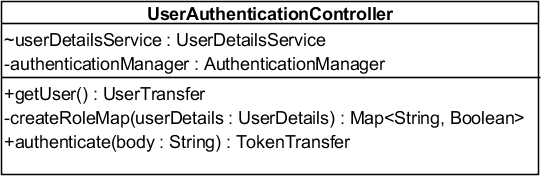
* 1. Class name: IndexController



**Method:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | index | Method for redirect to index page | **Return:** String |

* 1. Class name: UserAuthenticationController



**Property:**

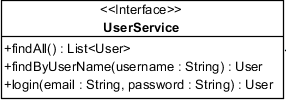
|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | userDetailsService | Object of UserDetailsService, used for call feature about the detail of user. | **Type:** UserDetailsService |
| 2 | authenticationManager | Object of AuthenticationManager, used for Authentication feature. | **Type:** AuthenticationManager |

**Method:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | getUser | Method for get logged in user. | **Return:** UserTransfer |
| 2 | createRoleMap | Method for create map of role of logged user. | **Return:** UserDetails  **Param:** Map<String, Boolean> |
| 3 | authenticate | Method for authenticate user. | **Return:** TokenTransfer  **Param:** String |

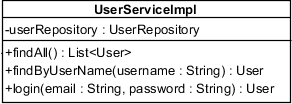
**Service package**

* 1. Class name: UserService



\*Used for declare feature that relate to user.

* 1. Class name: UserServiceImpl



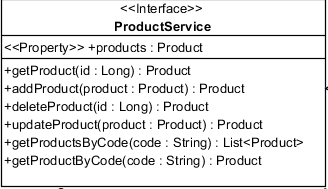
**Property:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | userRepository | Object of UserRepository, used for call feature of database about user. | **Type:** UserRepository |

**Method:**

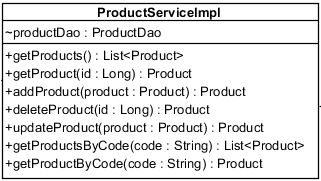
|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | findAll | Method for get list contains information of all user in the system. | **Return:** List<User> |
| 2 | findByUserName | Method for get user’s information in the system. | **Return:** User  **Param:** String |
| 3 | login | Method for login to the system. | **Return:** User  **Param:** String, String |

* 1. Class name: ProductService



\*Used for declare feature that relate to product.

* 1. Class name: ProductServiceImpl



**Property:**

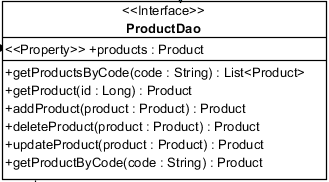
|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | productDAO | Object of ProductDAO, used for call feature relate to database about Product. | **Type:** productDAO |

**Method:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | getProducts | Method for do something before get all product in the system. | **Return:** List<Product> |
| 2 | getProductsByCode | Method for do something before get list of product that similar to code in the system. | **Return:** Product  **Param:** String |
| 3 | getProduct | Method for do something before get product in the system. | **Return:** Product  **Param:** Long |
| 4 | addProduct | Method for do something before add product to the system. | **Return:** Product  **Param:** Product |
| 5 | deleteProduct | Method for do something before delete product from the system. | **Return:** Product  **Param:** Product |
| 6 | updateProduct | Method for do something before update product’s detail to the system. | **Return:** Product  **Param:** Product |
| 7 | getProductByCode | Method for do something before get specific product that same code in the system. | **Return:** Product  **Param:** String |

**Dao Package**

* 1. Class name: ProductDao



\*Used for declare feature about product that relate to database.

* 1. Class name: DbProductDao



**Property:**

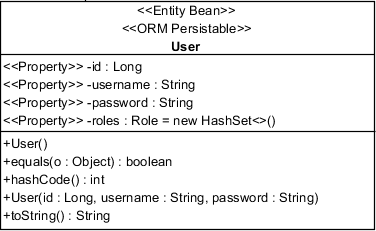
|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | productRepository | Object of productRepository used for call feature of database about product. | **Type:** ProductRepository |

**Method:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | getProducts | Method for get all product in the system. | **Return:** List<Product> |
| 2 | getProductsByCode | Method for get list of product that similar to code in the system. | **Return:** Product  **Param:** String |
| 3 | getProduct | Method for get product in the system. | **Return:** Product  **Param:** Long |
| 4 | addProduct | Method for add product to the system. | **Return:** Product  **Param:** Product |
| 5 | deleteProduct | Method for delete product from the system. | **Return:** Product  **Param:** Product |
| 6 | updateProduct | Method for update product’s detail to the system. | **Return:** Product  **Param:** Product |
| 7 | getProductByCode | Method for get specific product that same code in the system. | **Return:** Product  **Param:** String |

**Entity package**

* 1. Class name: User



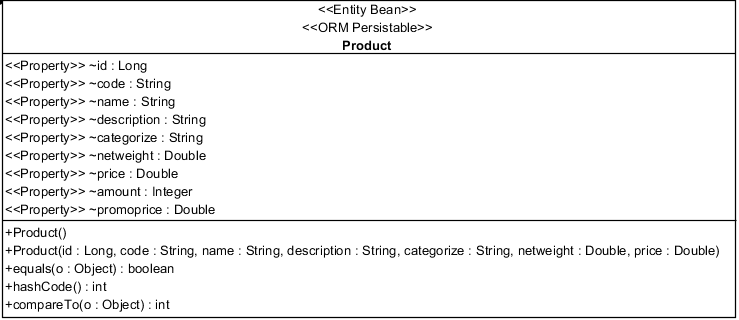
**Property:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | id | Store the id of the user | **Type:** Long |
| 2 | username | Store the user name of the user | **Type:** String |
| 3 | password | Store the password of the user | **Type:** String |
| 4 | roles | Store the role of the user | **Type:** String |

**Method:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | User | Method for create default user. |  |
| 2 | equal | Method for check same user in the system. | **Return:** boolean  **Param:** Object |
| 3 | hashCode | Method for check same user in the system. | **Return:** int |
| 4 | User | Method for create user following argument. | **Param:** Long, String, String |
| 5 | toString | Method for get user’s information. | **Return:** String |

* 1. Class name: Product



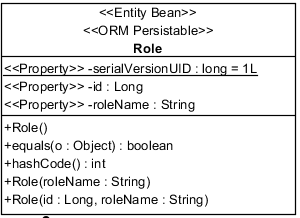
**Property:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | id | Store the id of the product | **Type:** Long |
| 2 | code | Store the code of barcode of the product | **Type:** String |
| 3 | name | Store description of the product | **Type:** String |
| 4 | description | Store the description of the product | **Type:** String |
| 5 | categorize | Store the categorize type of the product | **Type:** String |
| 6 | netweight | Store the net weight/net volume of the product | **Type:** Double |
| 7 | price | Store the price of the product | **Type:** Double |
| 8 | amount | Store the amount that required for promotion price of the product | **Type:** Integer |
| 9 | promoprice | Store the promotion price of the product | **Type:** Double |

**Method:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | Product | Method for create default product. |  |
| 2 | Product | Method for create product following argument. | **Param:** Long, String, String, String, String, Double, Double |
| 3 | equal | Method for check same product in the system. | **Return:** boolean  **Param:** Object |
| 4 | hashCode | Method for check same product in the system. | **Return:** int |
| 5 | compareTo | Method for sorting products in the system. | **Return:** int  **Param:** Object |

* 1. Class name: Product



**Property:**

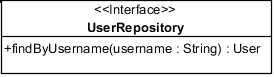
|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | SerialVersionUID | Store the universal version of the class | **Type:** Long |
| 1 | id | Store the id of the role | **Type:** Long |
| 2 | roleName | Store the name of role | **Type:** String |

**Method:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | role | Method for create default role. |  |
| 2 | equal | Method for check same role in the system. | **Return:** boolean  **Param:** Object |
| 3 | hashCode | Method for check same role in the system. | **Return:** int |
| 4 | role | Method for create role following argument. | **Param:** String |
| 5 | role | Method for create role following argument. | **Param:** Long, String |

**Repository package**

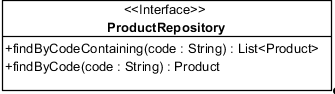
* 1. Class name: UserRepository



**Method:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | findByUsername | Method for generate SQL language for getting user by username. | **Return:** User  **Param:** String |

* 1. Class name: ProductRepository



**Method:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | findByCodeContaining | Method for generate SQL language for getting list of products that similar to product’s code. | **Return:** List<Product>  **Param:** String |
| 2 | findByCode | Method for generate SQL language for getting specific product by product’s code. | **Return:** Product  **Param:** String |

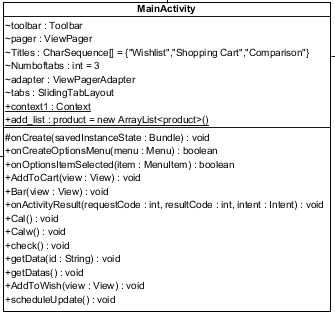
* 1. Class name: UserRepository



\*Used for generate SQL language to create role table in the database.

**Android Application**

* 1. Class name: MainActivity



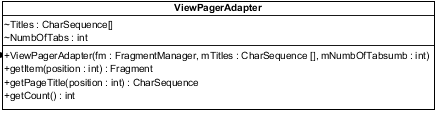
**Property:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | toolbar | Object of Toolbar, used for call feature of Toolbar. | **Type:** Toolbar |
| 2 | pager | Object of ViewPager, used for call feature of ViewPager. | **Type:** ViewPager |
| 3 | Titles | Store the title of page. | **Type:** CharSequence[] |
| 4 | Numboftabs | Store number of page. | **Type:** int |
| 5 | adapter | Object of ViewPagerAdapter, used for call feature of ViewPagerAdapter. | **Type:**ViewPagerAdapter |
| 6 | tabs | Object of SlidingTabLayout, used for call feature of SlidingTabLayout. | **Type:** SlidingTabLayout |
| 7 | Context1 | Object of Context, used for call feature of Context. | **Type:** Context |
| 8 | Add\_list | Store all products that in the system (webservice). | **Type:** ArrayList<product> |

**Method:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | OnCreate | Method for create whole application. | **Return:** void  **Param:** Bundle |
| 2 | OnCreateOptionsMenu | Method for create menu of application. | **Return:** boolean  **Param:** Menu |
| 3 | OnOptionsItemSelected | Method for assign work to menu of application. | **Return:** boolean  **Param:** MenuItem |
| 4 | AddToCart | Method for show add product methods to shopping cart and add by input product’s id. | **Return:** void  **Param:** View |
| 5 | Bar | Method for add product to cart by barcode scanner. | **Return:** void  **Param:** View |
| 6 | onActivityResult | Method for receive product’s id from barcode. | **Return:** void  **Param:** int, int, Intent |
| 7 | Cal | Method for calculate total price for shopping cart. | **Return:** void |
| 8 | Calw | Method for calculate total price for wish list. | **Return:** void |
| 9 | check | Method for check item on wish list and shopping cart. | **Return:** void |
| 10 | getData | Method for get product’s information by product’s id. | **Return:** void  **Param:** String |
| 11 | getDatas | Method for get product’s information by product’s id. | **Return:** void |
| 12 | AddToWish | Method for show all products to select to wish list. | **Return:** void  **Param:** View |
| 13 | scheduleUpdate | Method for update information every second. | **Return:** void |

* 1. Class name: ViewPagerAdapter



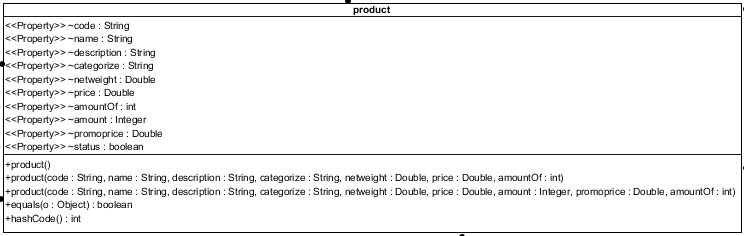
**Property:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | Titles | Store the title of page. | **Type:** CharSequence[] |
| 2 | Numboftabs | Store number of page. | **Type:** int |

**Method:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | ViewPagerAdapter | Method for construct ViewPagerAdapter. | **Param:** FragmentManager, CharSequence[], int |
| 2 | getItem | Method for show page in each tab. | **Return:** Fragment  **Param:** int |
| 3 | getPageTitle | Method for get title of page. | **Return:** CharSequence  **Param:** int |
| 4 | getCount | Method for get amount of page. | **Return:** int |

* 1. Class name: Product



**Property:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | code | Store the code of barcode of the product. | **Type:** String |
| 2 | name | Store description of the product. | **Type:** String |
| 3 | description | Store the description of the product. | **Type:** String |
| 4 | categorize | Store the categorize type of the product. | **Type:** String |
| 5 | netweight | Store the net weight/net volume of the product. | **Type:** Double |
| 6 | price | Store the price of the product. | **Type:** Double |
| 7 | amountOf | Store the amount of product in the list. | **Type:** int |
| 8 | amount | Store the amount that required for promotion price of the product. | **Type:** Integer |
| 9 | promoprice | Store the promotion price of the product. | **Type:** Double |
| 10 | status | Store the status of available of product. | **Type:** boolean |

**Method:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | Product | Method for create default product. |  |
| 2 | Product | Method for create product following argument. | **Param:** String, String, String, String, Double, Double, int |
| 3 | Product | Method for create product following argument. | **Param:** String, String, String, String, Double, Double, Integer, Double int |
| 4 | equal | Method for check same product in the system. | **Return:** boolean  **Param:** Object |
| 5 | hashCode | Method for check same product in the system. | **Return:** int |

**Fragment Class**

* 1. Class name: cart\_fragment



**Property:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | list | Store the products in shopping cart. | **Type:** ArrayList<product> |

**Method:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | onCreateView | Method for create shopping cart page. | **Return:** View  **Param:** LayoutInflater, ViewGroup, Bundle |

* 1. Class name: wishlist\_fragment



**Property:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | list\_wish | Store the products in wish list. | **Type:** ArrayList<product> |

**Method:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | onCreateView | Method for create wish list page. | **Return:** View  **Param:** LayoutInflater, ViewGroup, Bundle |

* 1. Class name: compare\_fragment

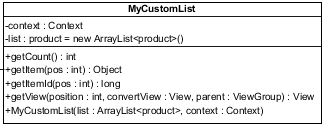


**Method:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | onCreateView | Method for create comparison page. | **Return:** View  **Param:** LayoutInflater, ViewGroup, Bundle |

**Custom List Class**

* 1. Class name: MyCustomList



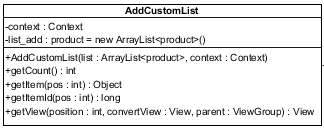
**Property:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | context | Store the context. | **Type:** Context |
| 2 | list | Store list of products in shopping cart. | **Type:** ArrayList<product> |

**Method:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | getCount | Method for get size of the list. | **Return:** Fint |
| 2 | getItem | Method for show page in each tab. | **Return:** Object  **Param:** int |
| 3 | getItemId | Method for get product’s id in list. | **Return:** long  **Param:** int |
| 4 | getView | Method for create list and assign work to its element. | **Return:** View  **Param:** int, View, ViewGroup |
| 5 | MyCustomList | Method for construct MyCustomList. | **Param:** ArrayList<product>, Context |

* 1. Class name: AddCustomList



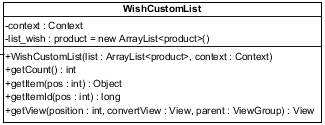
**Property:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | context | Store the context. | **Type:** Context |
| 2 | list\_add | Store list of all products in the system (webservice). | **Type:** ArrayList<product> |

**Method:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | getCount | Method for get size of the list. | **Return:** Fint |
| 2 | getItem | Method for show page in each tab. | **Return:** Object  **Param:** int |
| 3 | getItemId | Method for get product’s id in list. | **Return:** long  **Param:** int |
| 4 | getView | Method for create list and assign work to its element. | **Return:** View  **Param:** int, View, ViewGroup |
| 5 | MyCustomList | Method for construct MyCustomList. | **Param:** ArrayList<product>, Context |

* 1. Class name: WishCustomList



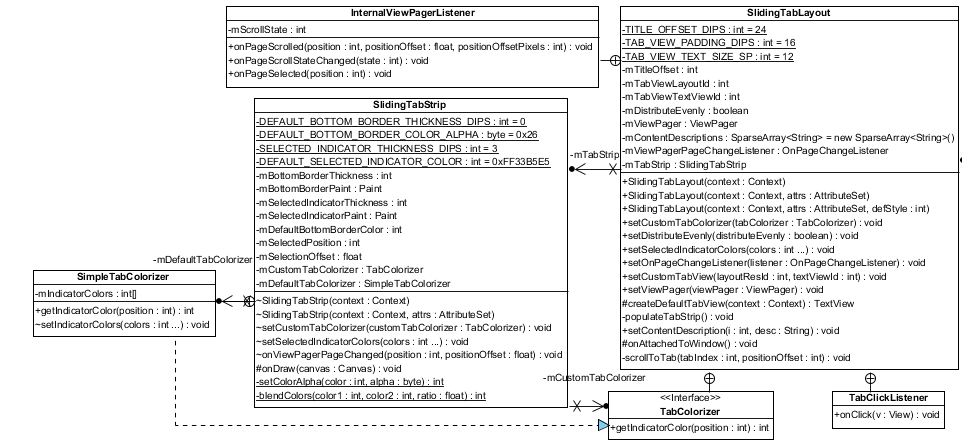
**Property:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | context | Store the context. | **Type:** Context |
| 2 | list\_wish | Store list of all products in wish list | **Type:** ArrayList<product> |

**Method:**

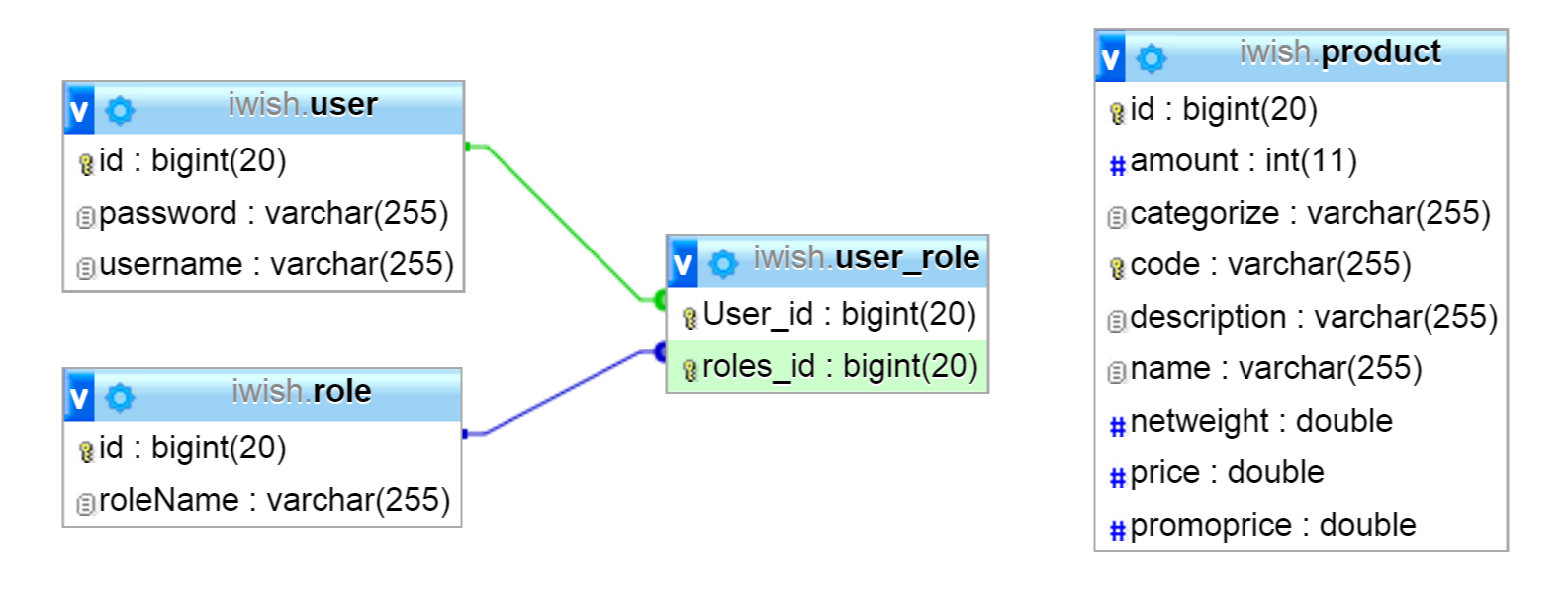
|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Remark** |
| 1 | getCount | Method for get size of the list. | **Return:** Fint |
| 2 | getItem | Method for show page in each tab. | **Return:** Object  **Param:** int |
| 3 | getItemId | Method for get product’s id in list. | **Return:** long  **Param:** int |
| 4 | getView | Method for create list and assign work to its element. | **Return:** View  **Param:** int, View, ViewGroup |
| 5 | MyCustomList | Method for construct MyCustomList. | **Param:** ArrayList<product>, Context |

**Off the shelf Class**

****

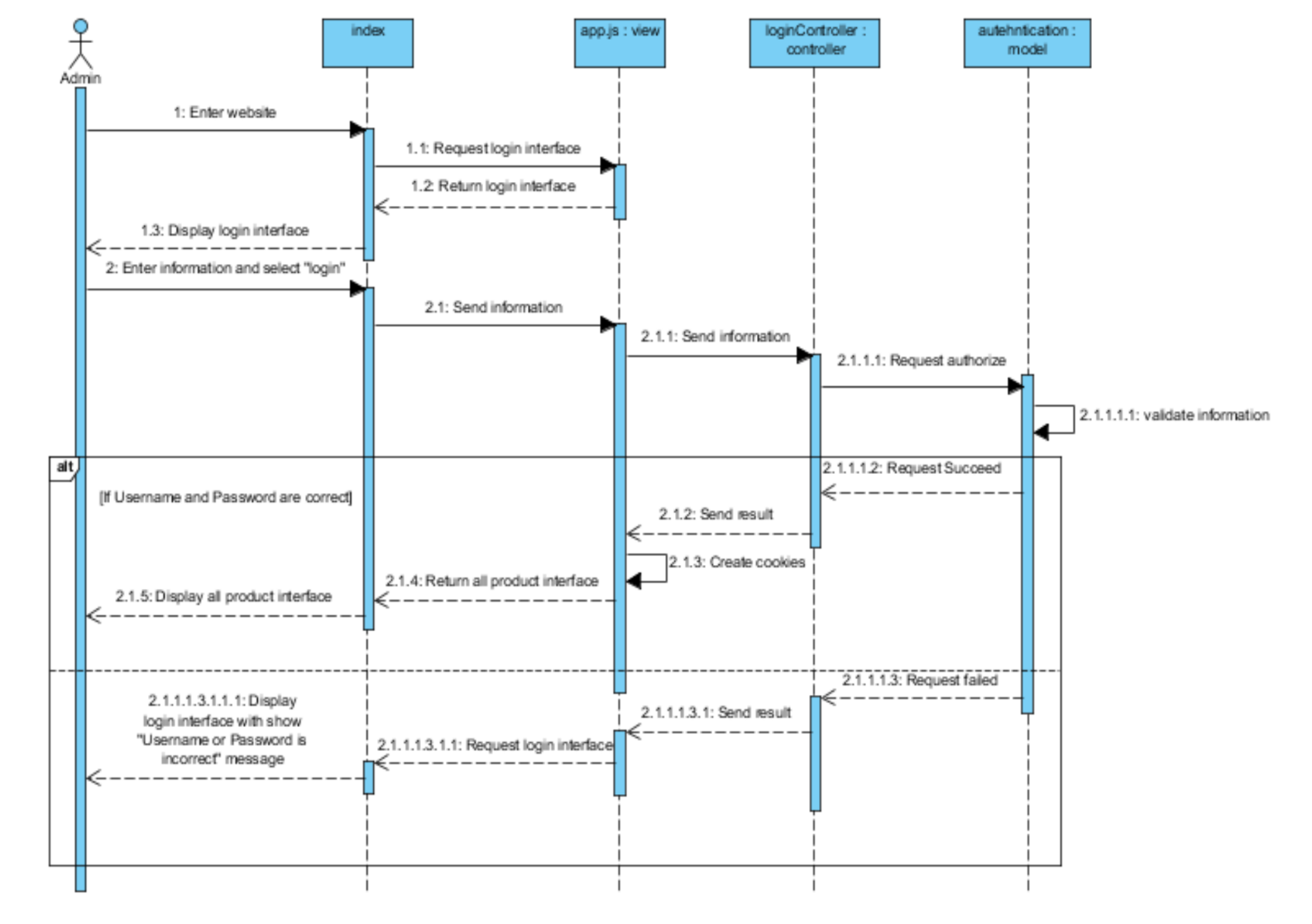
\*These class are open-source from android that allow the application follow the basic of android’s material design and can change page or tab by sliding.

1. **Database Design**

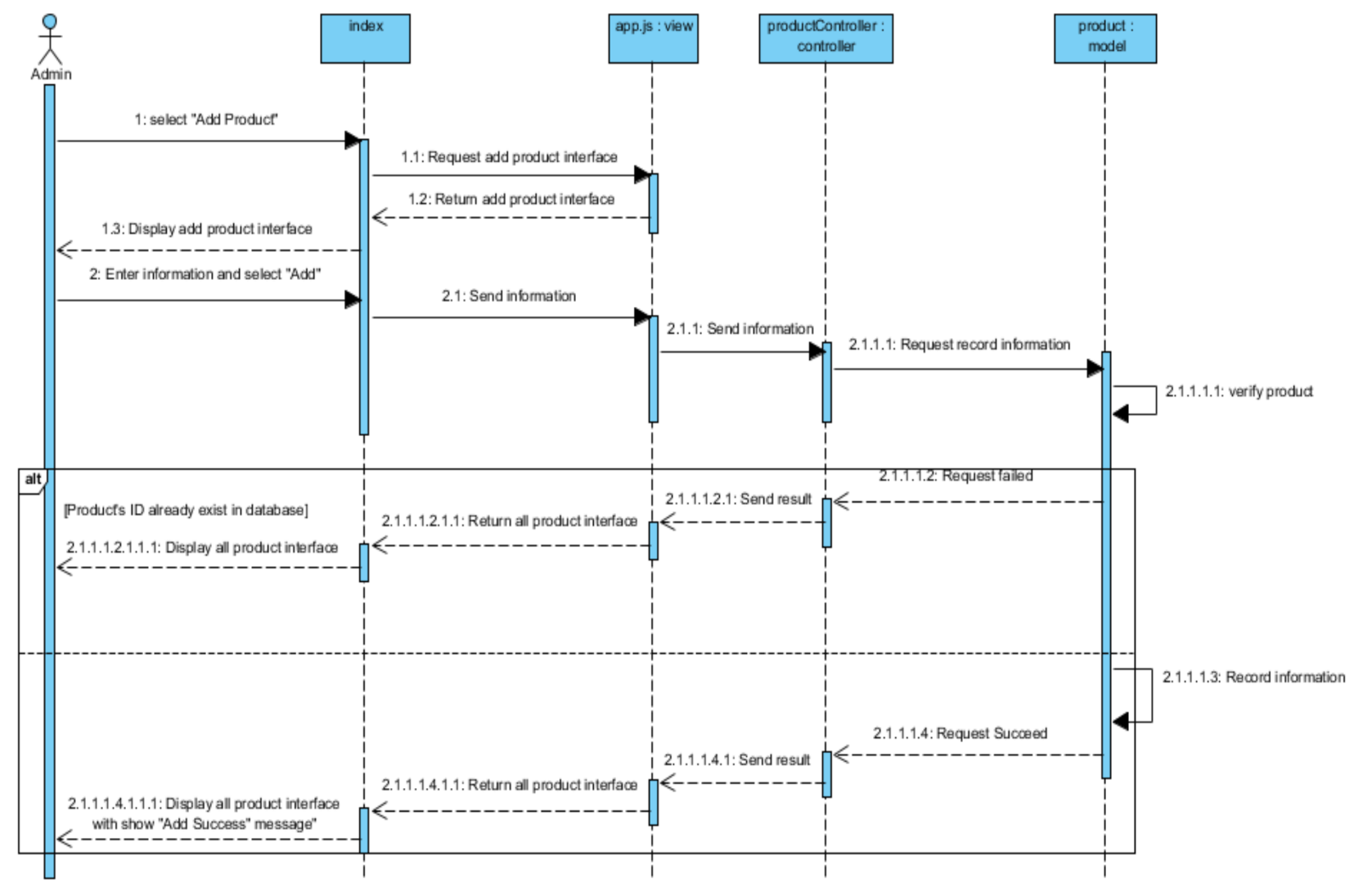
****

**Figure 26: Database Design**

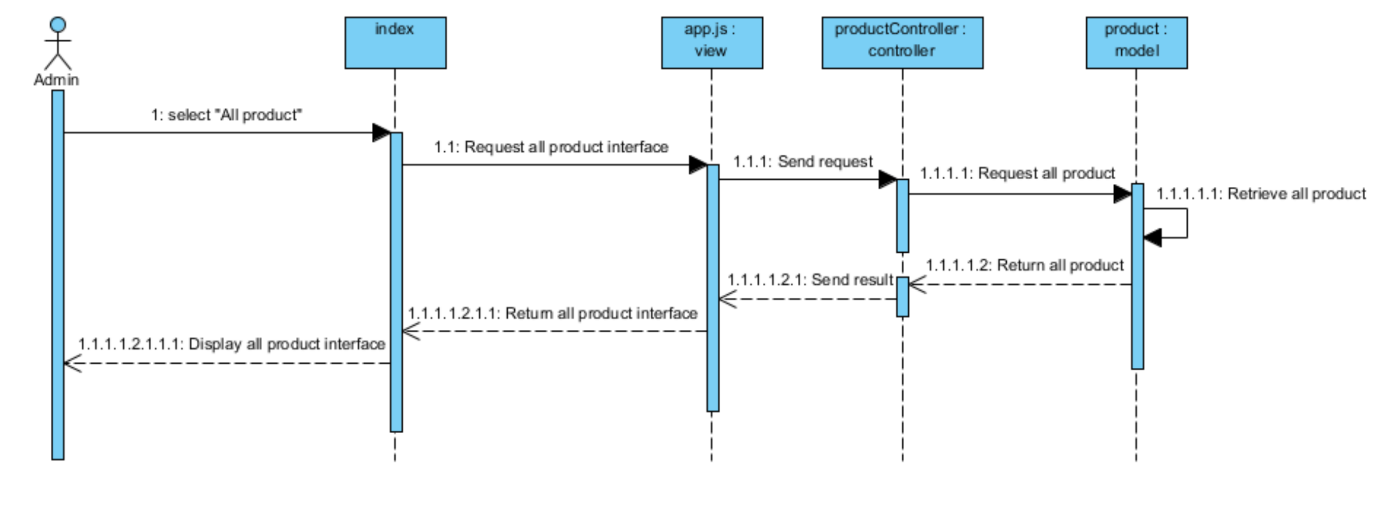
1. **Sequence Diagram Web Application**



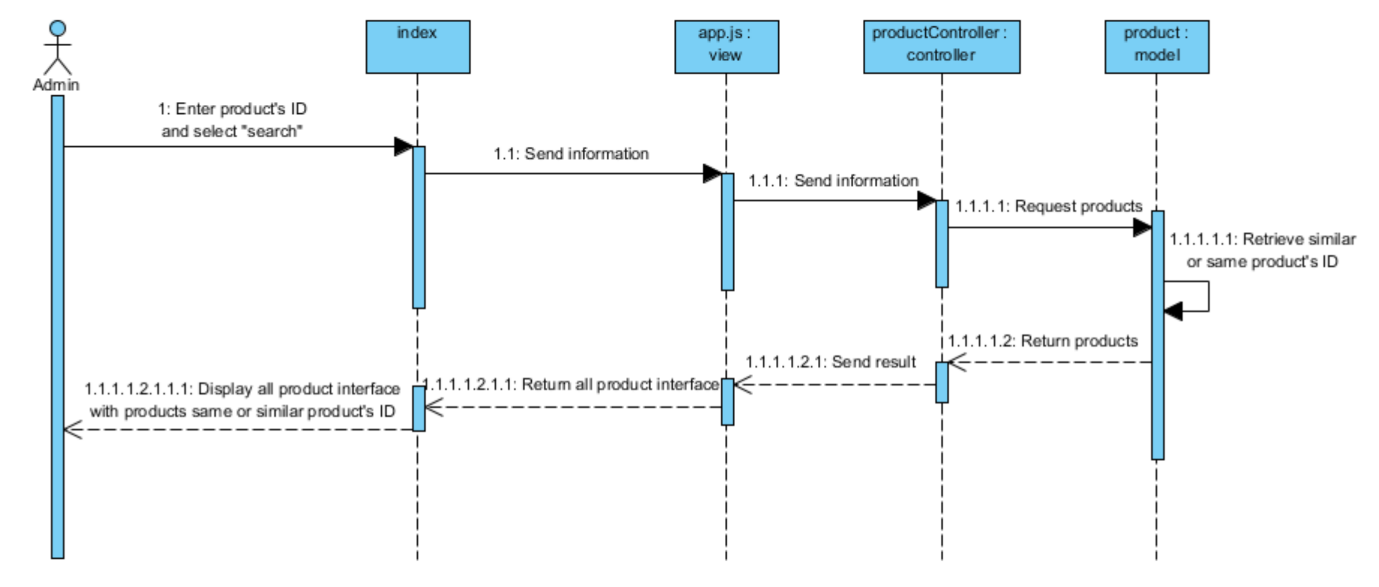
**Figure 27: SDW-01 Login**



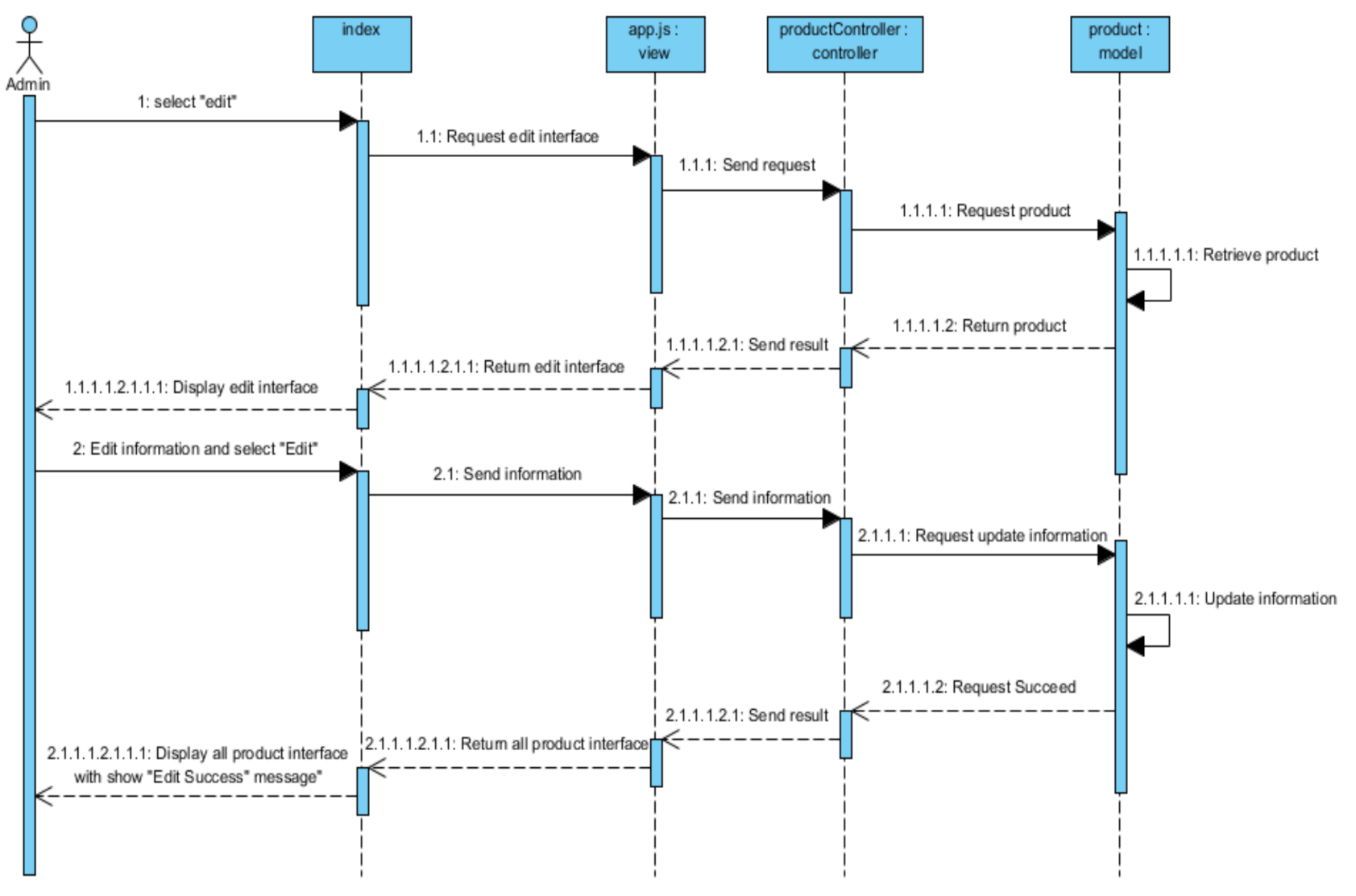
**Figure 28: SDW-02 Add New Product**

****

**Figure 29: SDW-03 View Product**

****

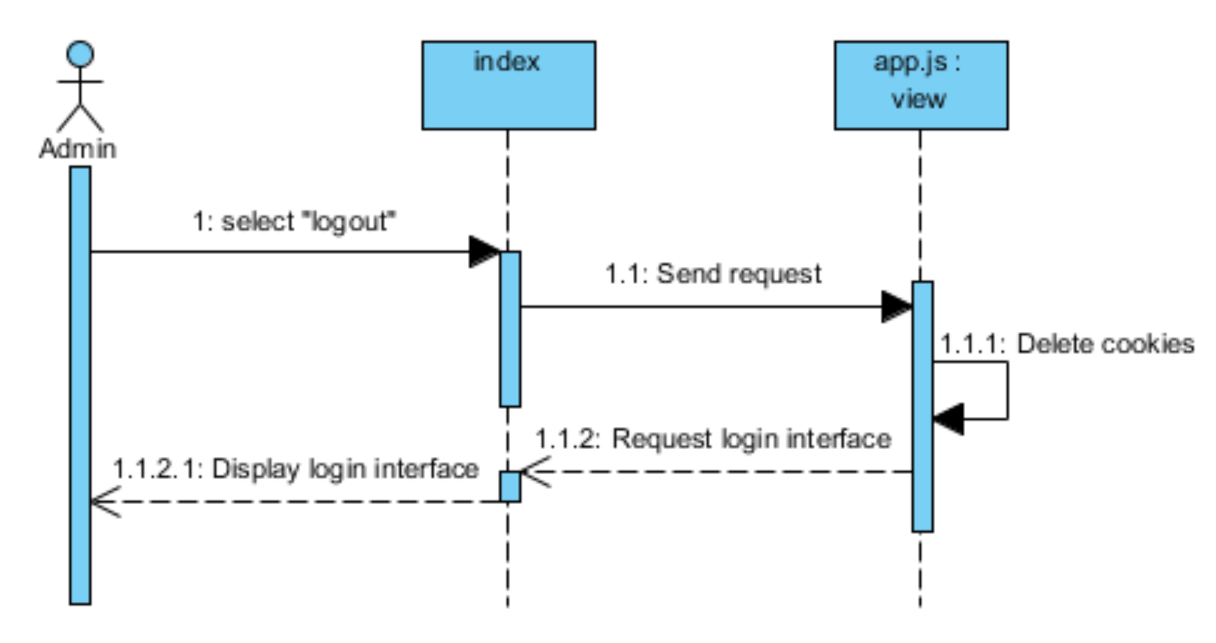
**Figure 30: SDW-04 Search Product by ID**



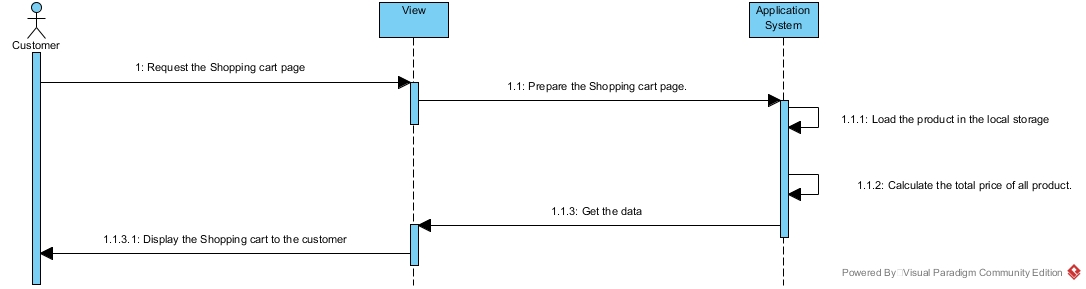
**Figure 31: SDW-05 Edit Product’s Information**



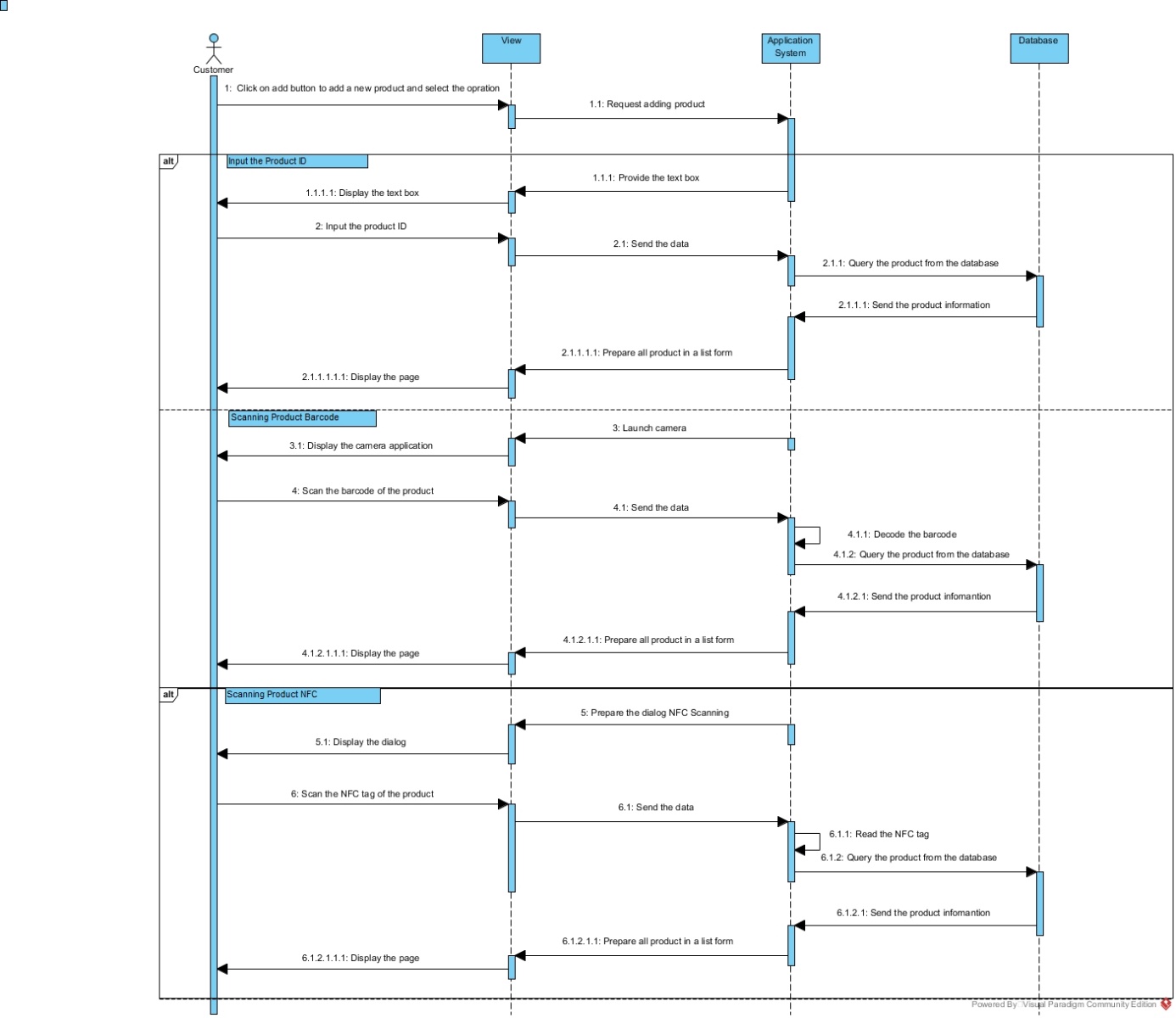
**Figure 32: SDW-06 Delete Product**



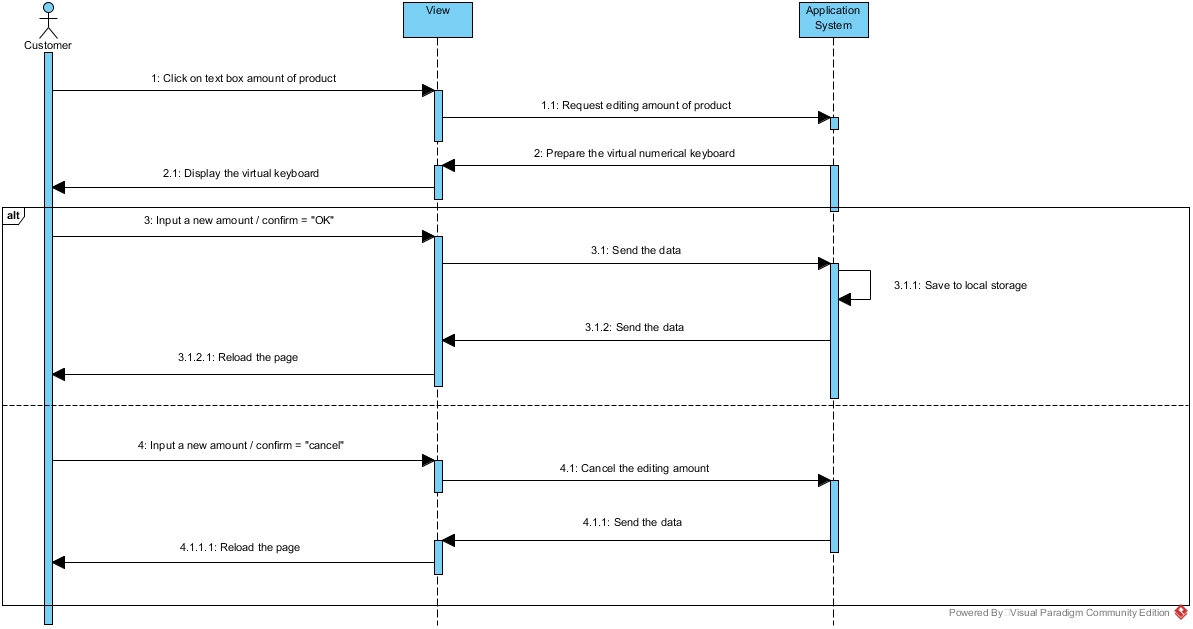
**Figure 33: SDW-07 Log Out**



**Figure 34: SDA-01 View Product in Shopping Cart**



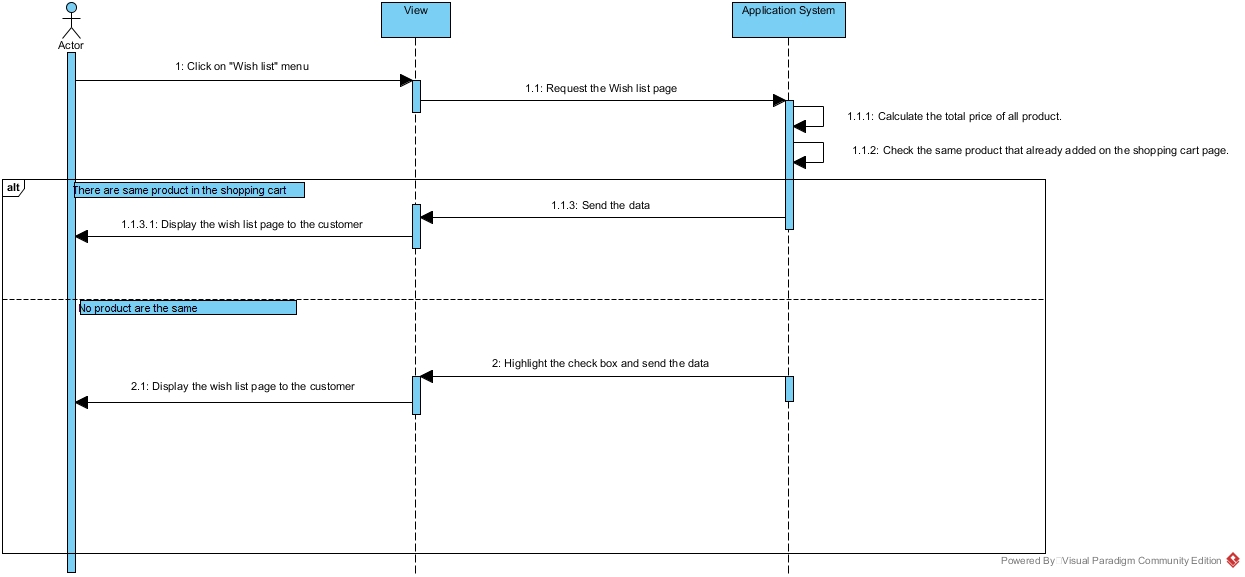
**Figure 35: SDA-02 ADD Product in Shopping Cart**



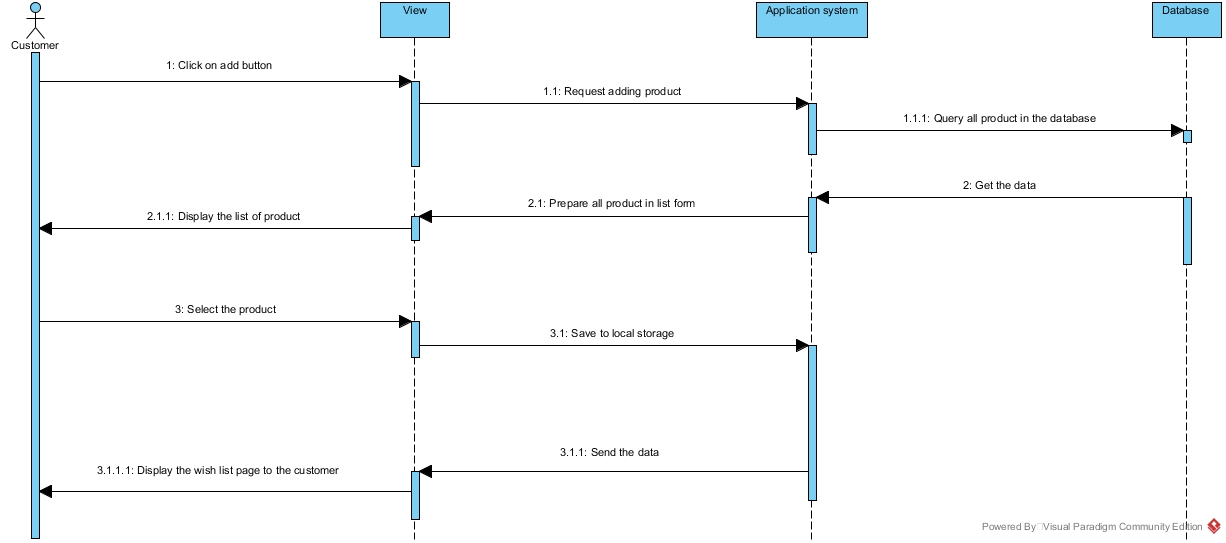
**Figure 36: SDA-03 Edit Amount of Product**

## C:\Users\Wayne\Downloads\Project\Pic\Sequence Delete.jpg

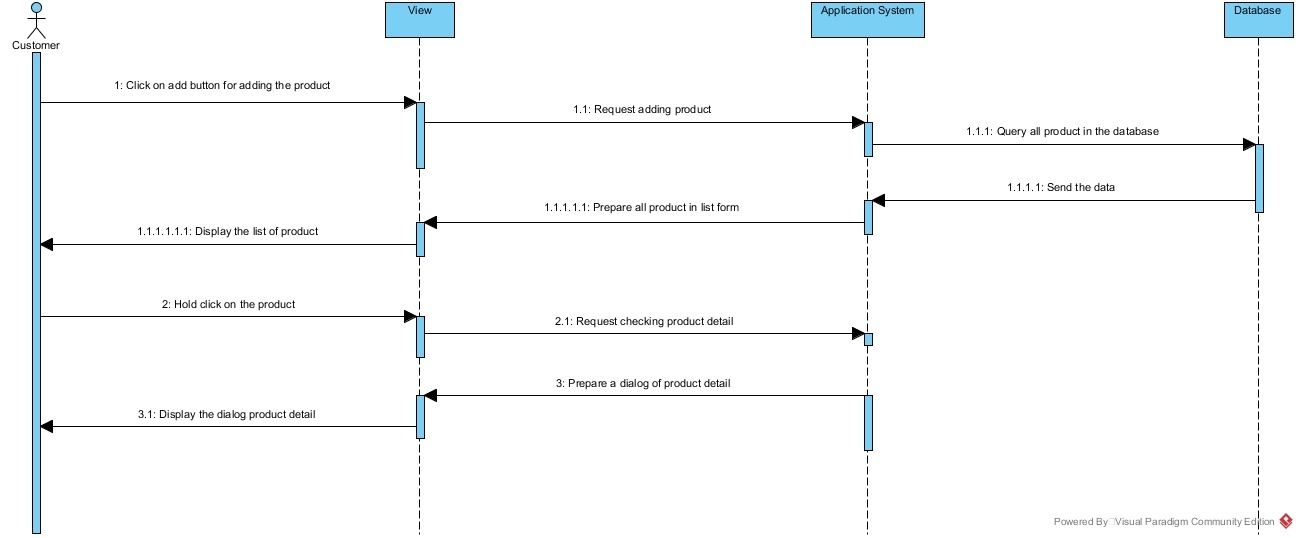
**Figure 37: SDA-04 Delete Product**



**Figure 38: SDA-05 View the Wish list page**

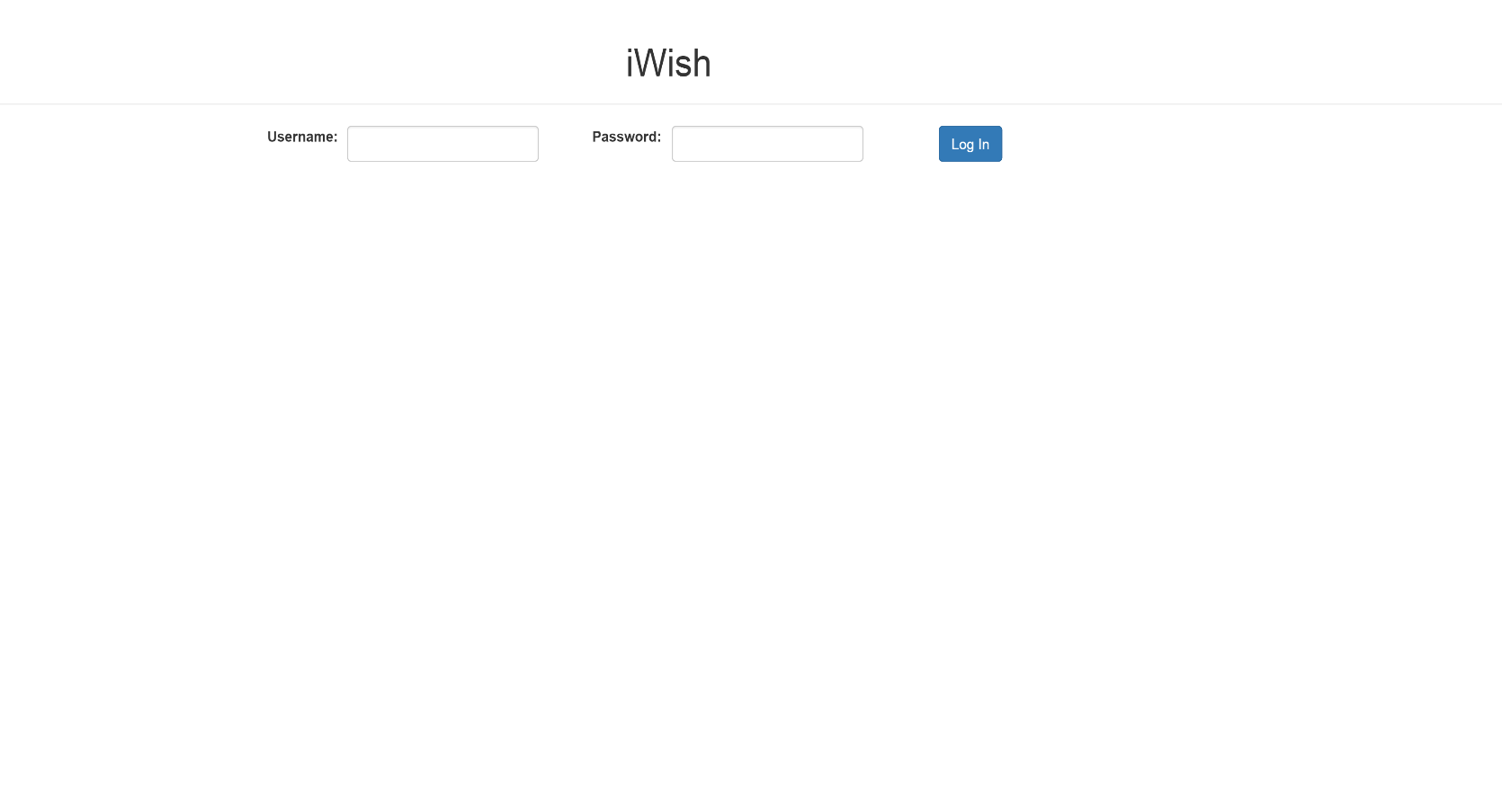


**Figure 39: SDA-06 Add a new product to the Wish list page**

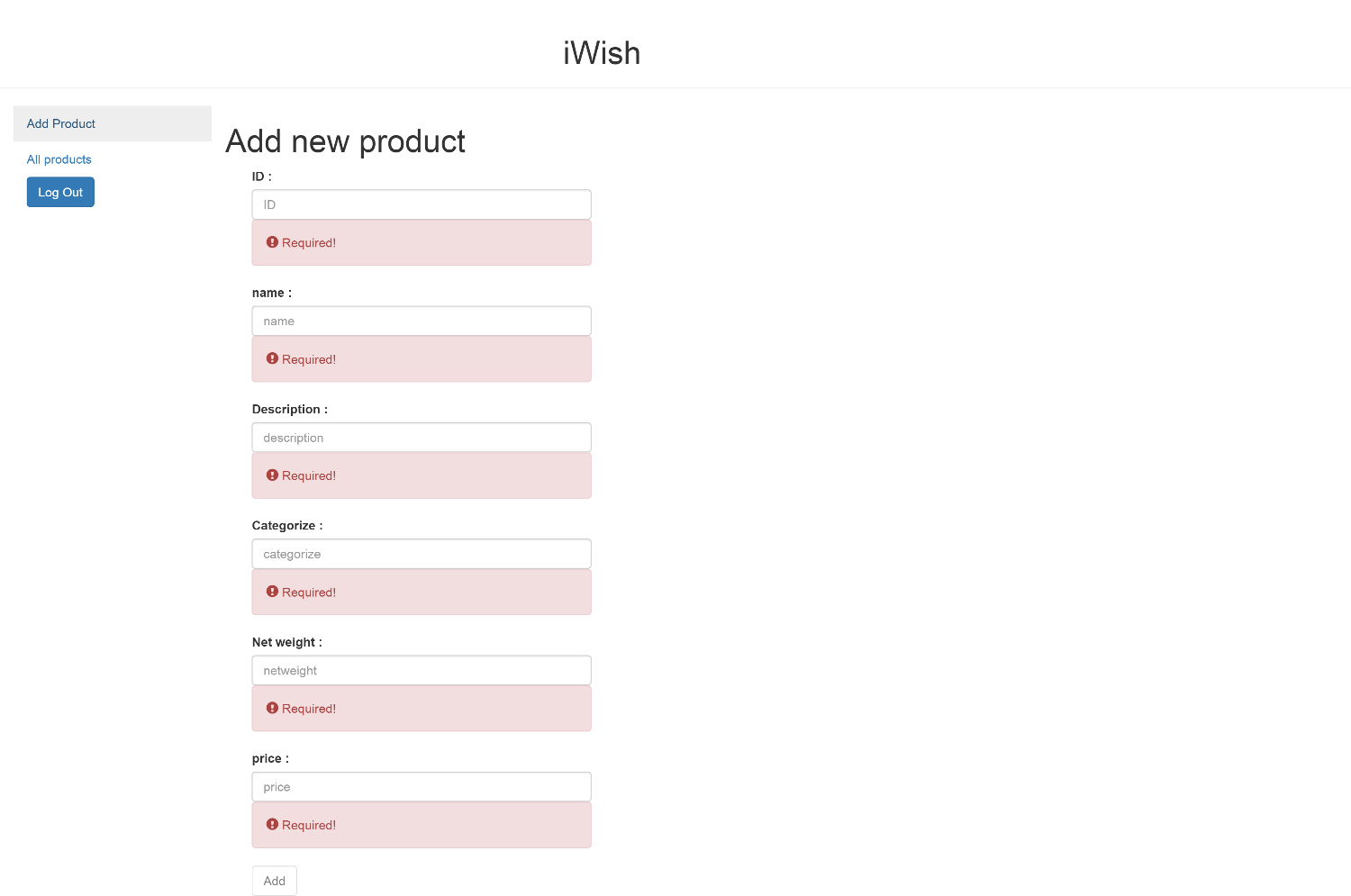


**Figure 40: SDA-07 Check product detail**

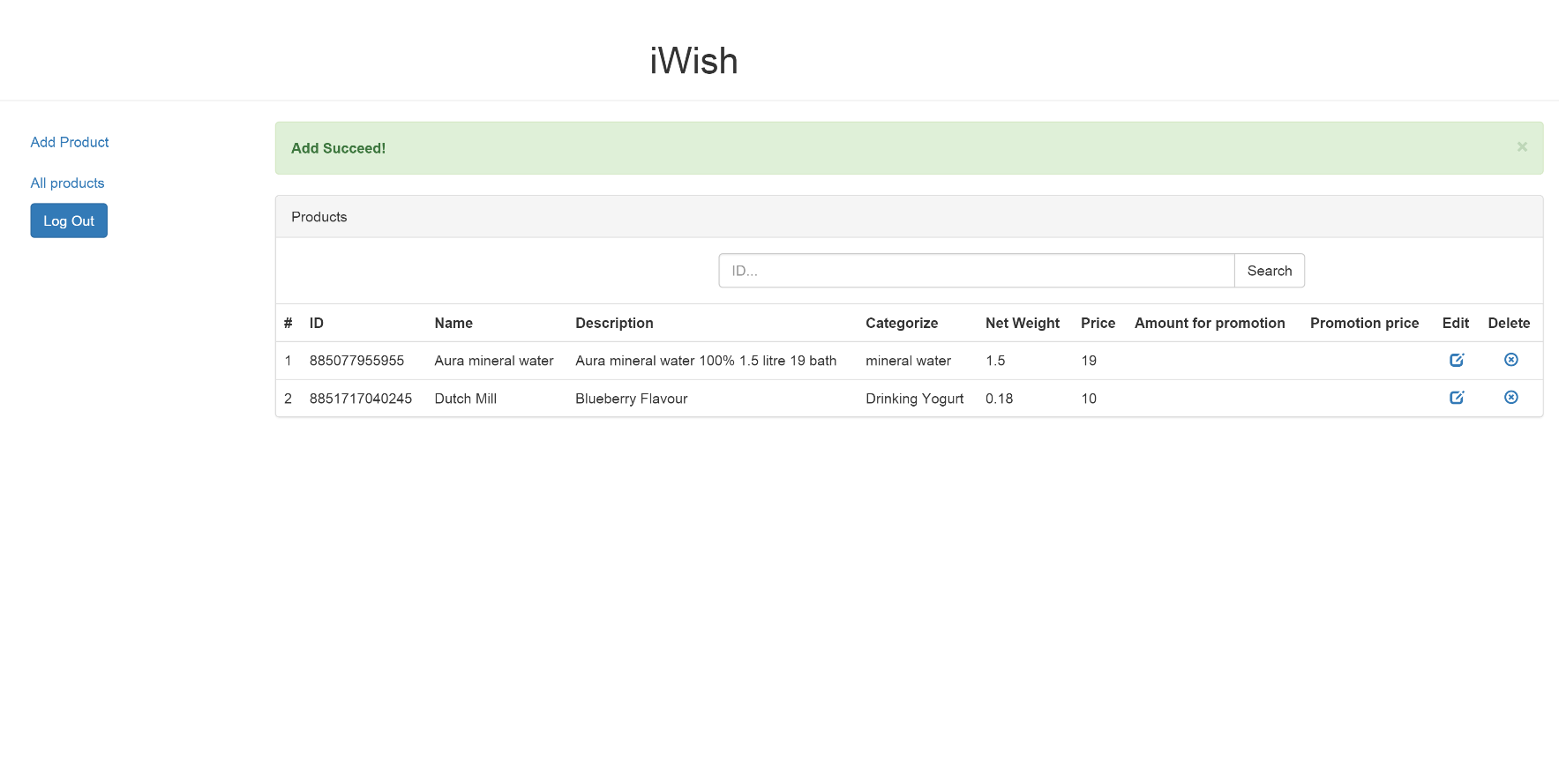
1. **UI (Web Application)**

****

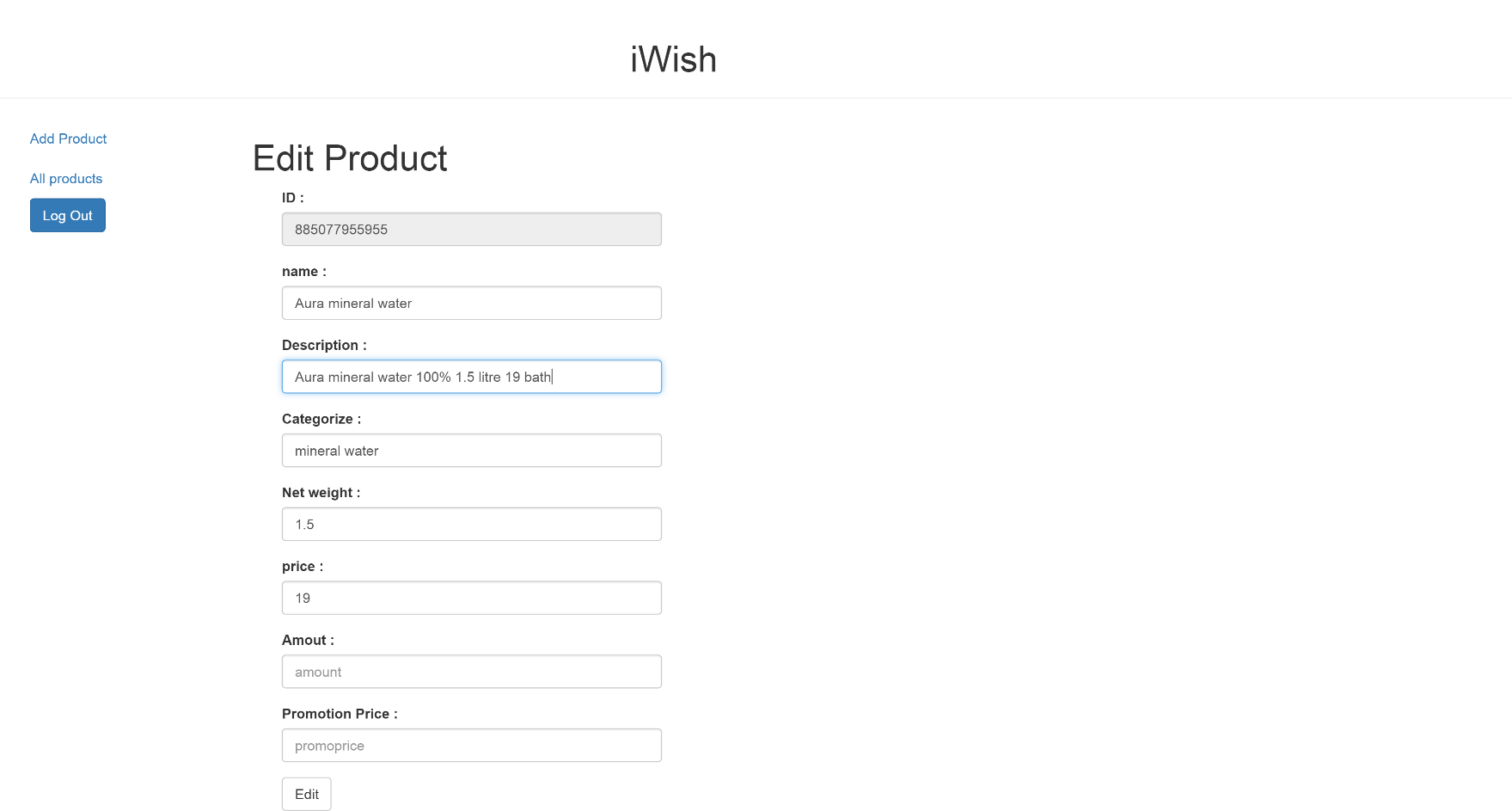
**Figure 41: UIW-01 Login**

****

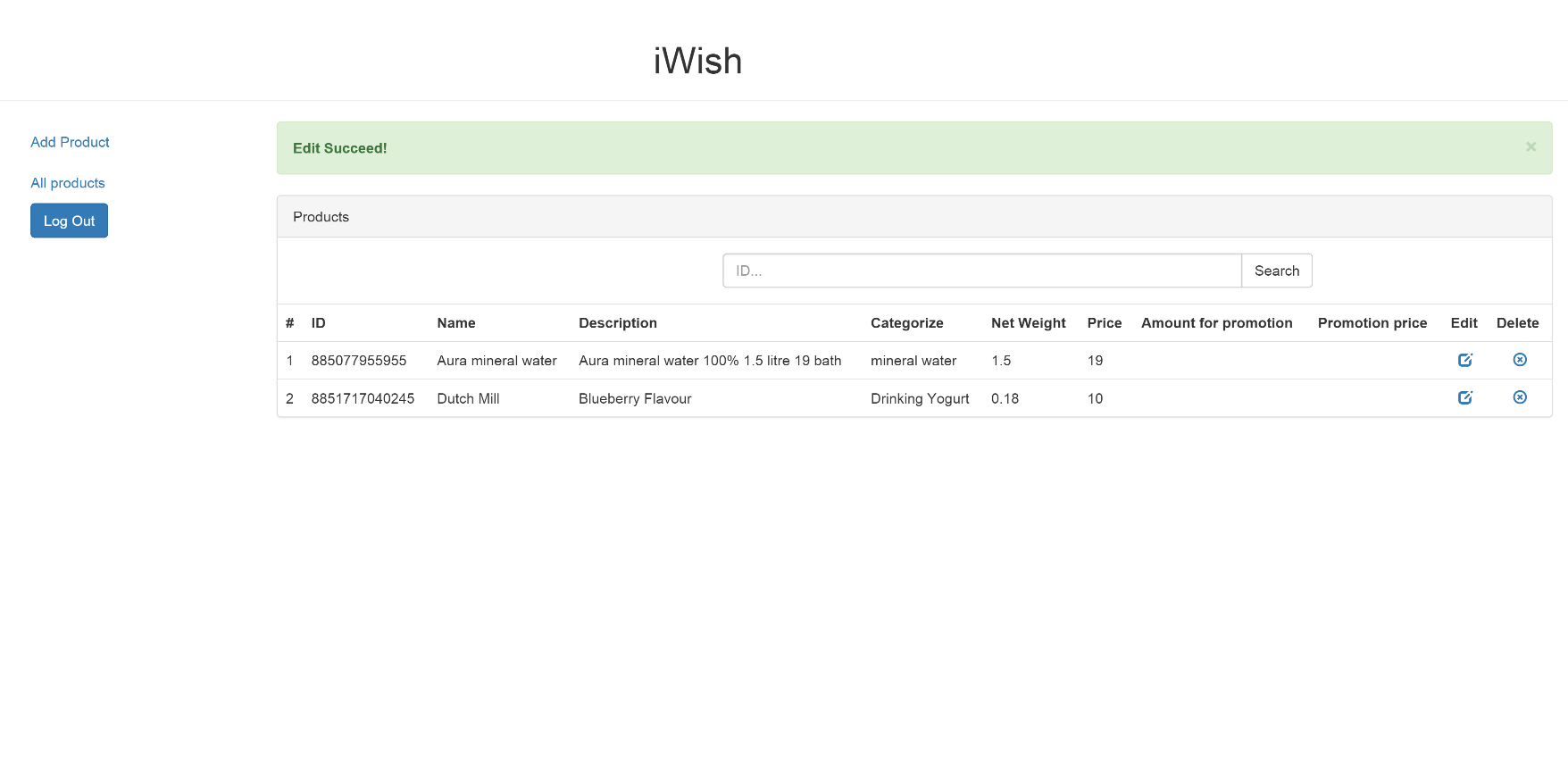
**Figure 42: UIW-02 Add Product**

****

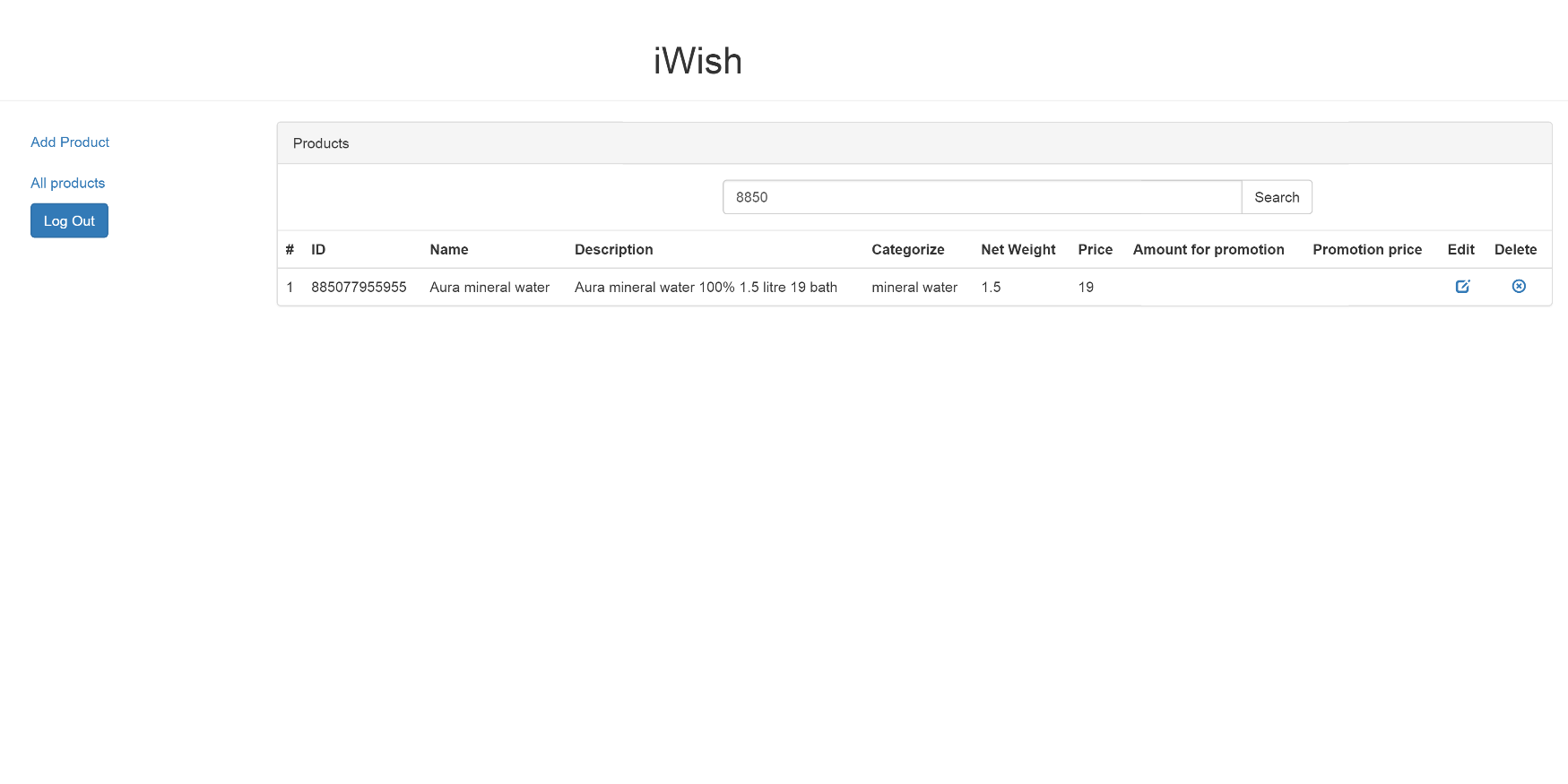
**Figure 43: UIW-03 Add Product Success**

****

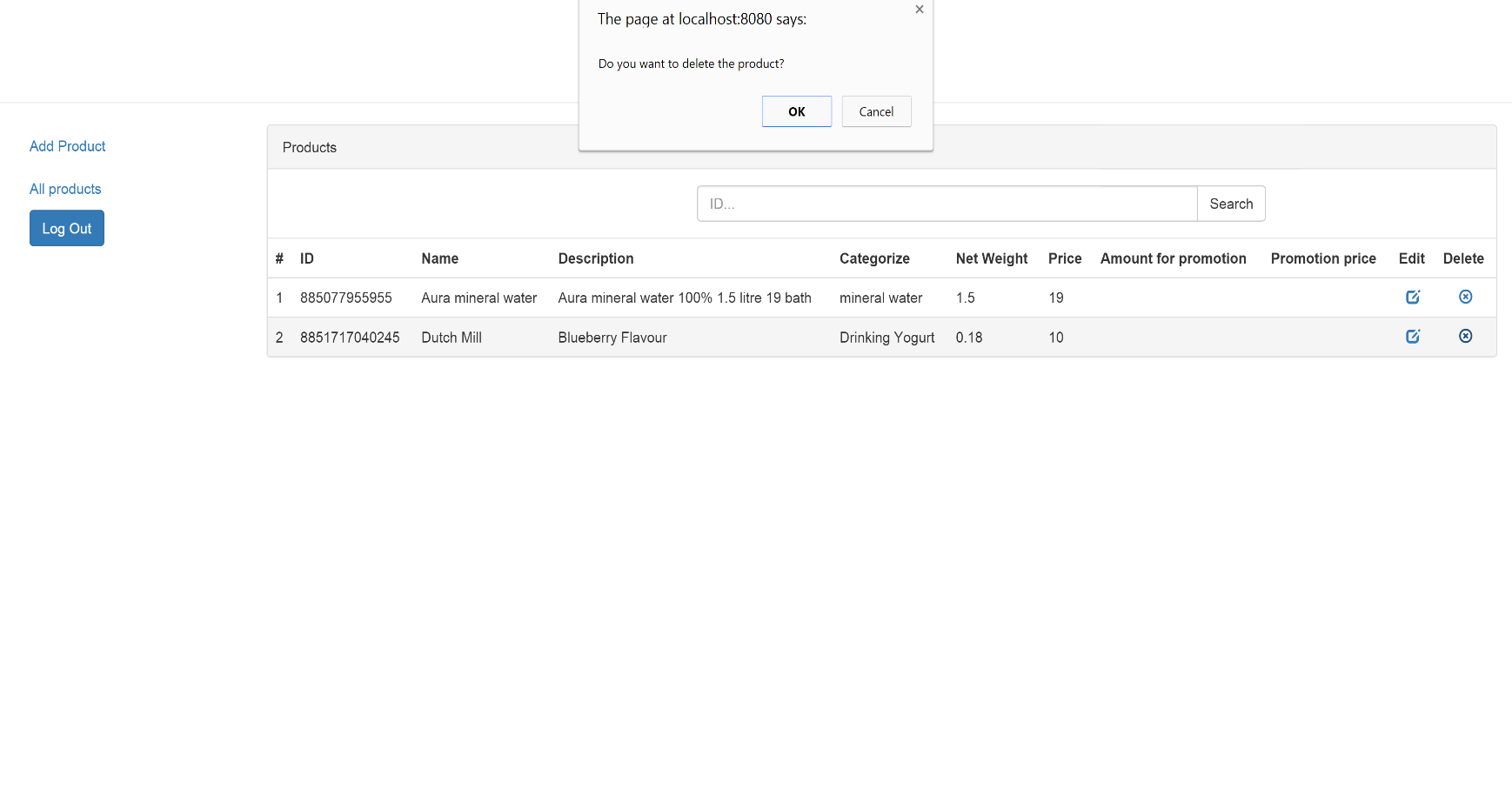
**Figure 44: UIW-04 Edit Information**

****

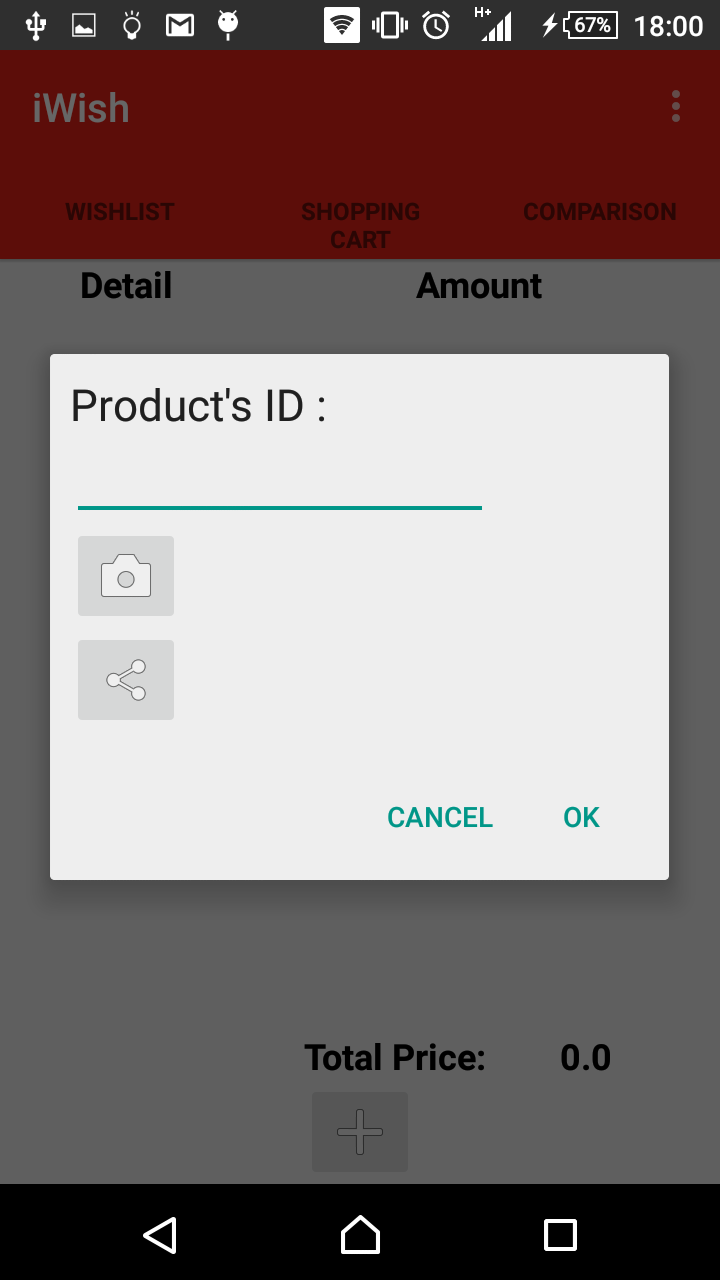
**Figure 45: UIW-05 Edit Success**

****

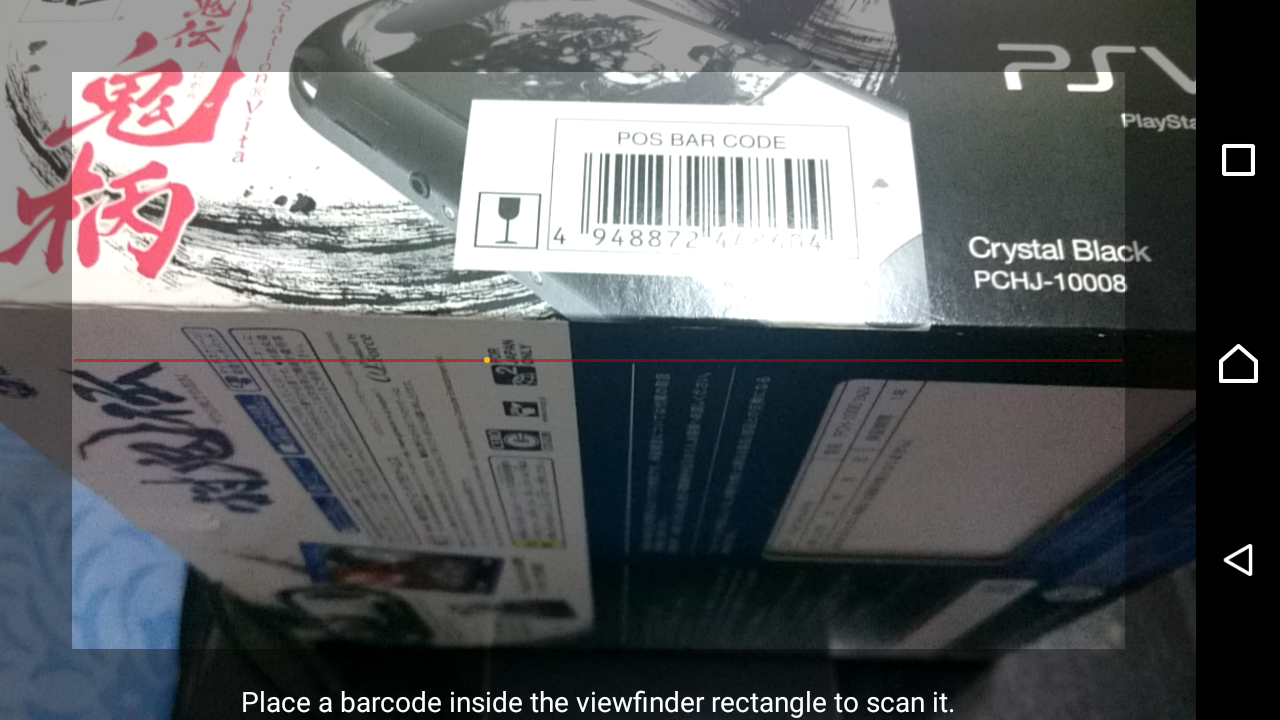
**Figure 46: UIW-06 Search Product**

****

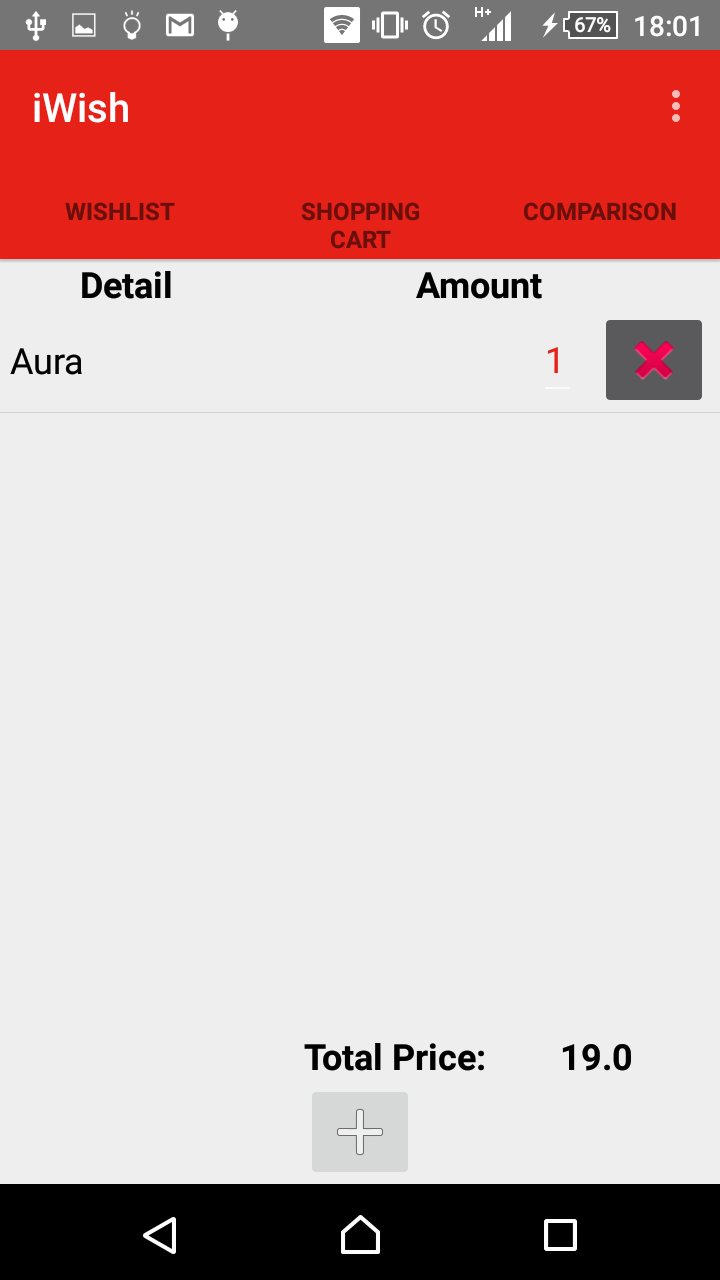
**Figure 47: UIW-06 Delete**

****

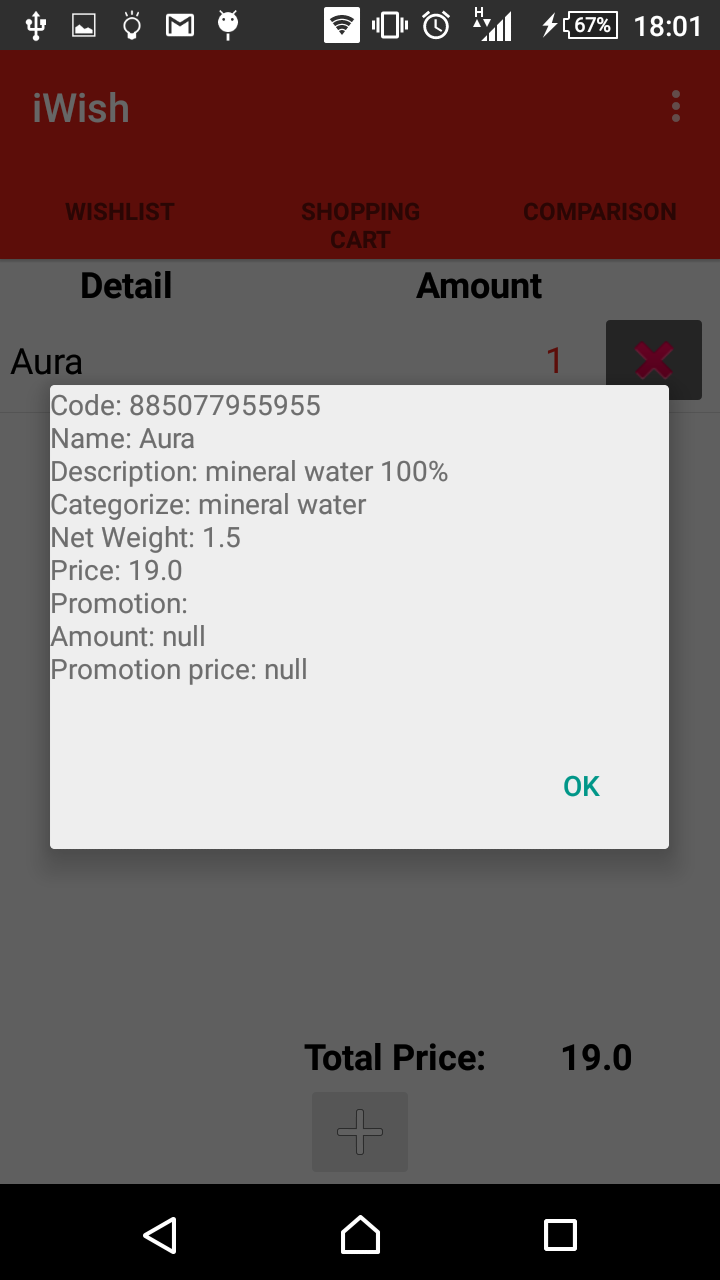
**Figure 48: UIA-01 Adding Dialog**

****

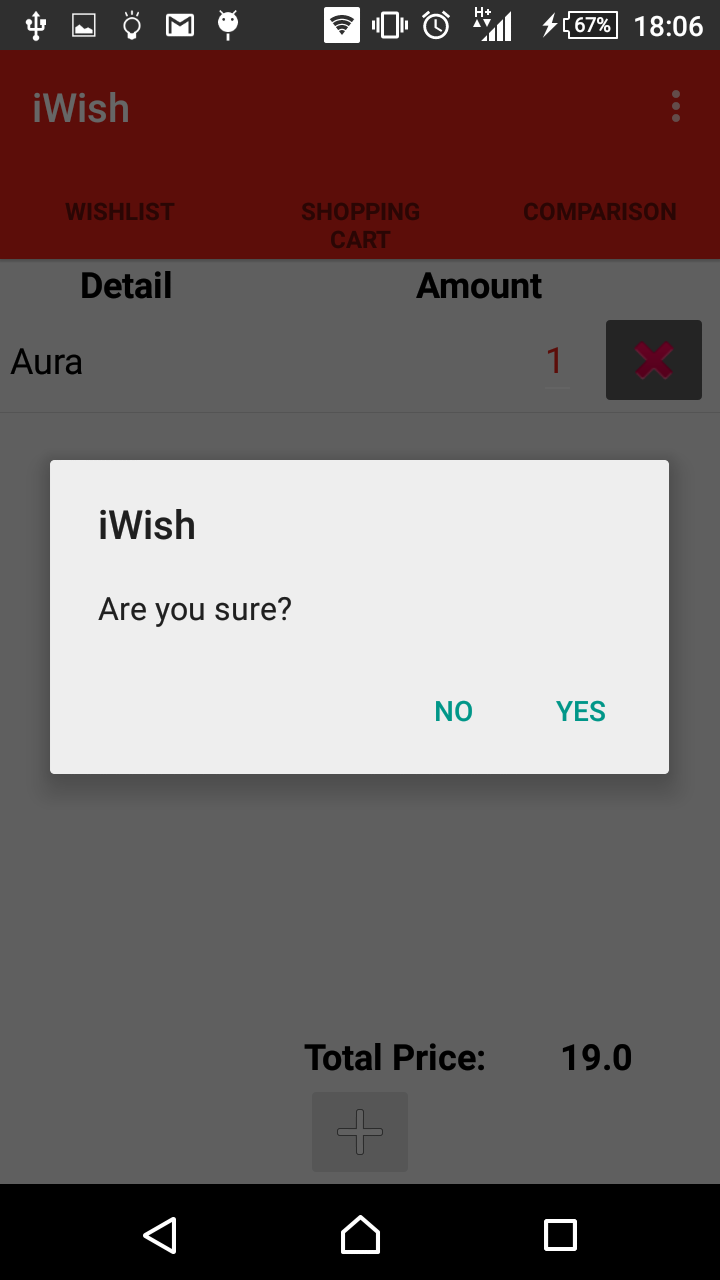
**Figure 49: UIA-02 Barcode Scanner**

****

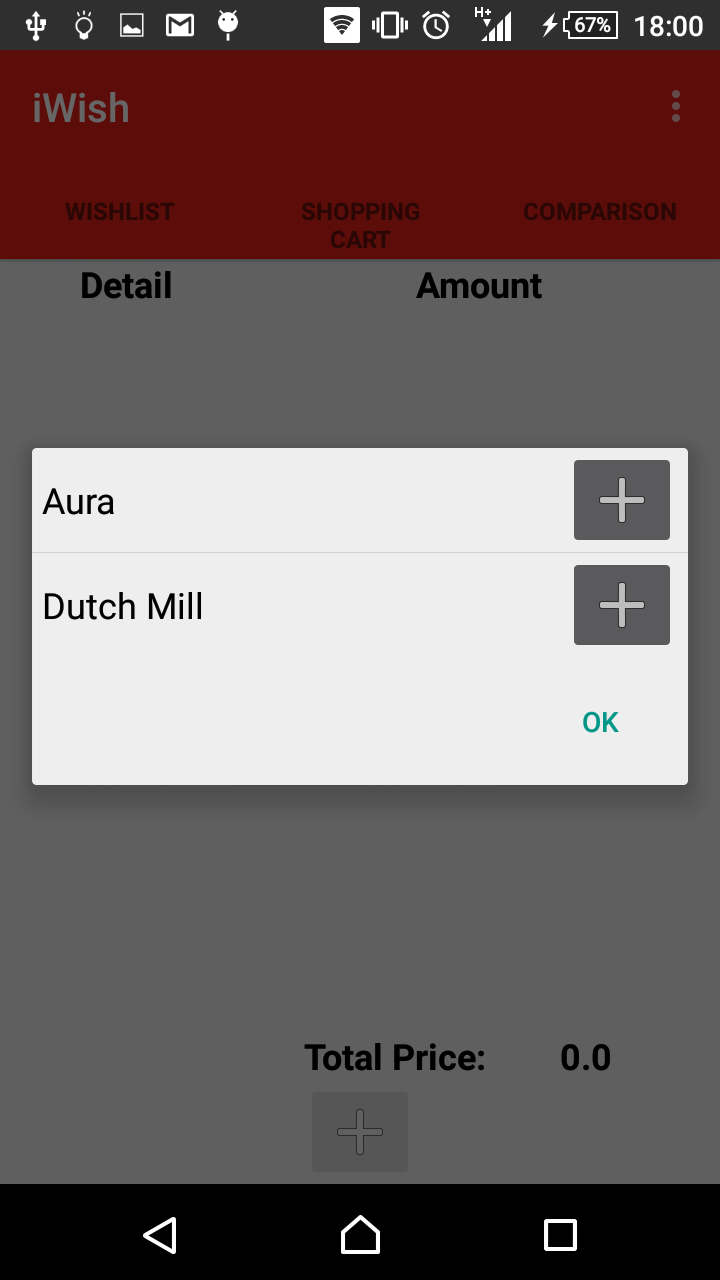
**Figure 50: UIA-03 View Shopping Cart**

****

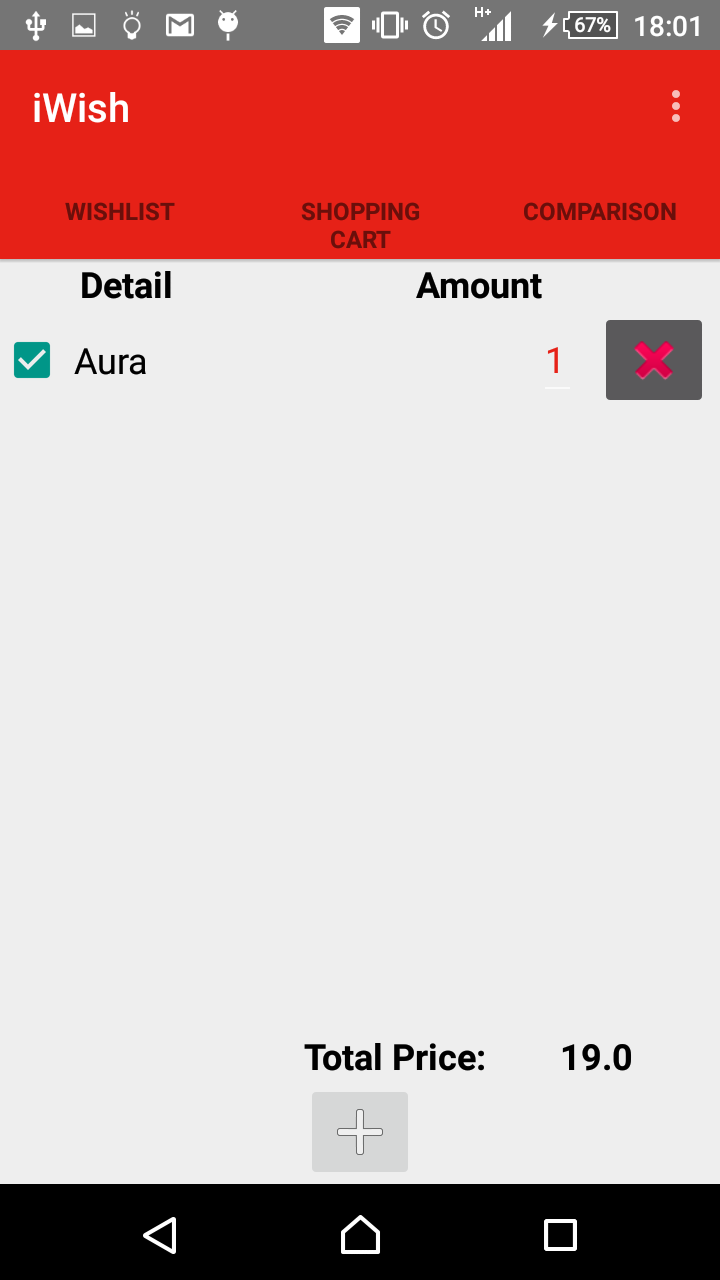
**Figure 51: UIA-04 View Details**

****

**Figure 52: UIA-05 Delete**

****

**Figure 53: UIA-06 Add to wish list**

****

**Figure 54: UIA-07 View wish list**