|  |
| --- |
| **System Design Specification** |

###### **Topic: Online Retail Application**

###### **Swim Wear a E-Commerce online Shopping**

###### **A Web Application**

**Development Team: 404**

**Member:**

**Đỗ Hùng Cường**

**Nguyễn Duy Vinh**

**Lê Thị Minh Phương**

**Nguyễn Thị Thủy Tiên**

**Lưu Minh Tín**

**Nguyễn Hồng Hà**

December 22, 2016

Table of Contents

**Introduction………………………………………………………………………………….. 2**

**System Overview…………………………………………………………………………****.. 3**

**System Architecture……………………………………………………………………... 4**

**User Interface…………………………………………………………………...……….... 5**

**Database System…….………………………………………………………****............... 6**

**Entity Relationship Diagram………………………………………………………….. 9**

**Class Diagram…………………………………………………………..…………….…….. 10**

**Use Case Diagram.…………………………………………………………………………. 12**

**Activity Diagram.…………………………………………………………………………… 13**

**Sequence Diagram ………………………………………………………………………… 26**

**Reference…………….………………………………………………………………………… 38**

### [Introduction](http://www.enteract.com/~bradapp/docs/sdd.html" \l "TOC_SEC4)

This document is designed to be a reference for any person wishing to implement or any person interested in the architecture of the E-Commerce online Shopping web application. This document describes the system architecture and sub-architecture that associate with the website interfaces, database schemas, and the motivations behind the chosen design. Both high-level and low-level designs are included in this document.

This document should be read by an individual with a technical background and has experience reading use case diagram (UC), class diagram, entity relationship diagram (ERD), sequence diagrams, activity diagram, and development experience in object oriented programming and event driven programming.

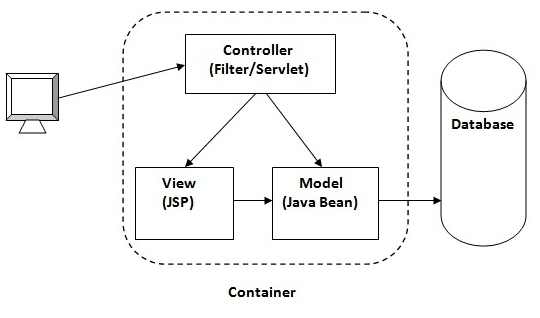
### [System Overview](http://www.enteract.com/~bradapp/docs/sdd.html#TOC_SEC5)

### C:\Users\USER.USER-PC\Desktop\New Bitmap Image.bmp

Swim Wear a E-Commerce online Shopping is a management website that provides swimwear for customers and it also helps the administrator can easily manages the shop. The customers and the administrator have to login to the system to access the data, purchase products, view transaction, or update the database. The password will be encrypt by the system. The database will contains user database and system database.

### [System Architecture](http://www.enteract.com/~bradapp/docs/sdd.html" \l "TOC_SEC12)

#### 



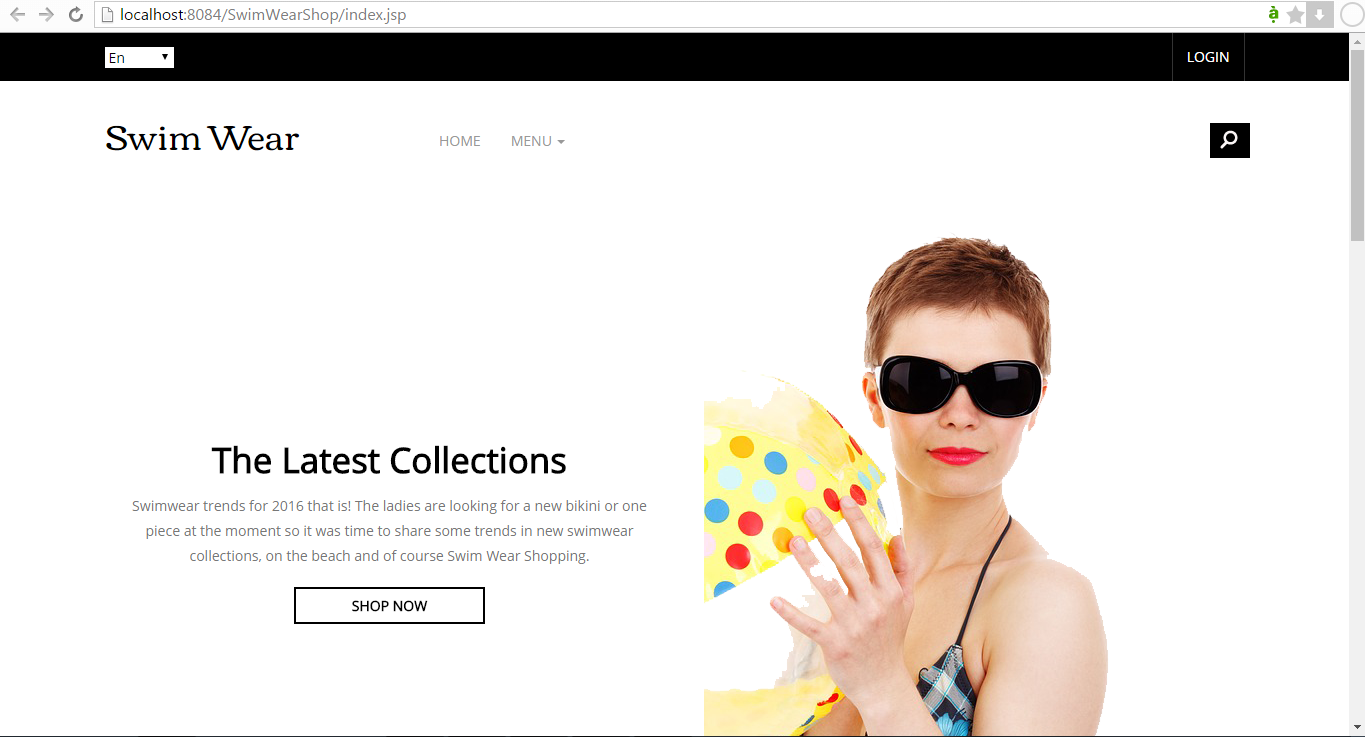
Swim Wear a E-Commerce online Shopping will apply model-view-controller pattern (MVC) with the user interface (view) is designed using jsp page, the controller will apply servlet and the model will be the java bean. A user always sees the view and communicates with the controller. The data retrieved from the database system will be used to assign into object of java bean (model) and handled by the controller then show the result in jsp page for user.

1. Major functions performed by the system: Interactions between customer and system, as well as between admin, staff with the system and its database
2. The architecture of the system in non-technical terms: Web-based, MVC design pattern
3. User access mode: graphical user interface
4. Responsible organization: Team 404 – Software engineering subject
5. System name or title: Swim Wear a E-Commerce online Shopping
6. System code: java jdk platform
7. System category:
8. *Customer interface:* performs clearly defined functions for which there is a readily identifiable security consideration and need
9. *Admin system:* provides general ADP for all system user
10. Operational status:Operational & Under development

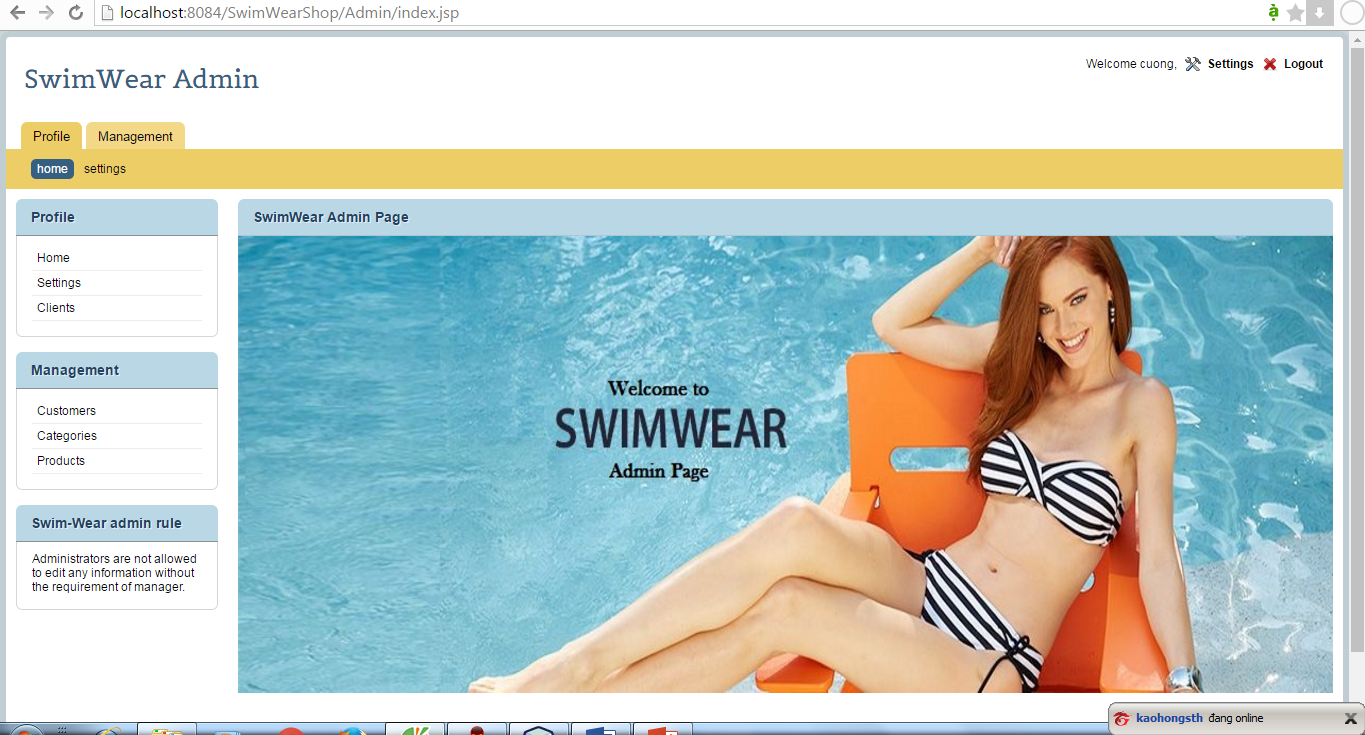
### User Interface

Using jsp (HTML), javascript, jquery

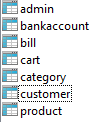
For Customer

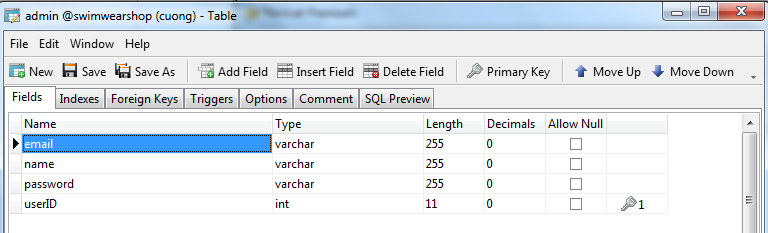


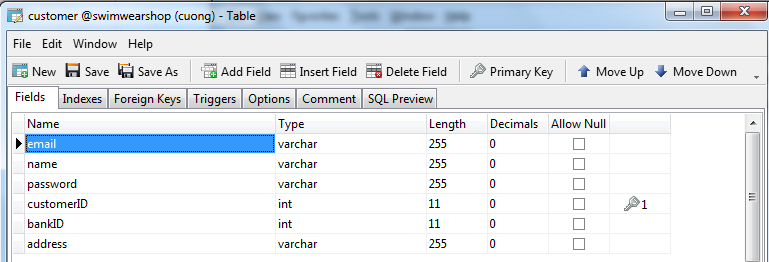
For Administrator

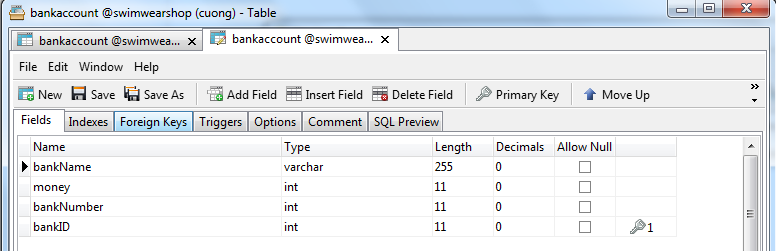


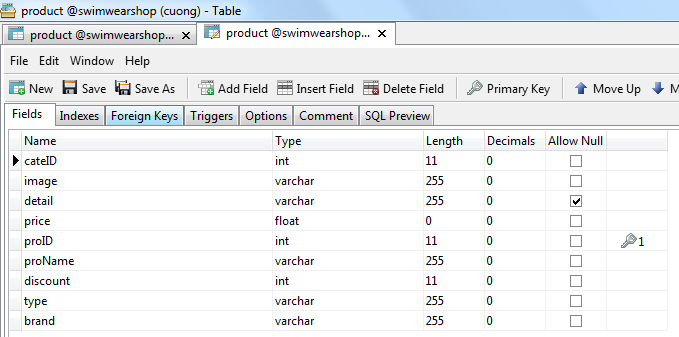
### Database System

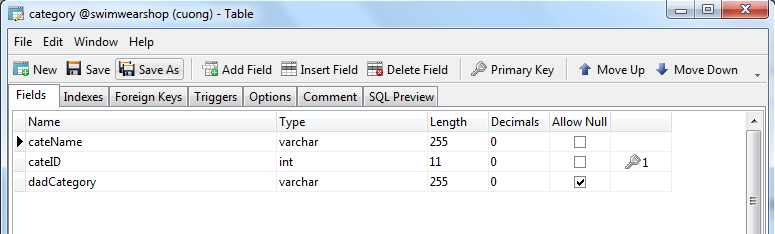


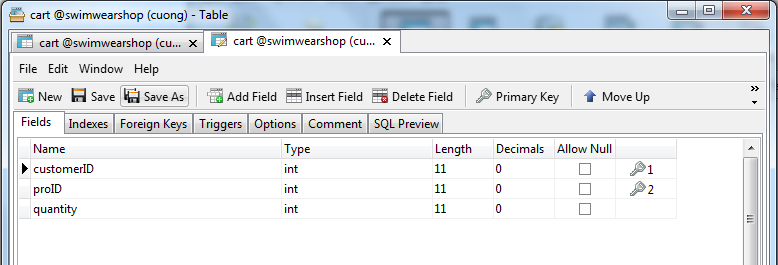


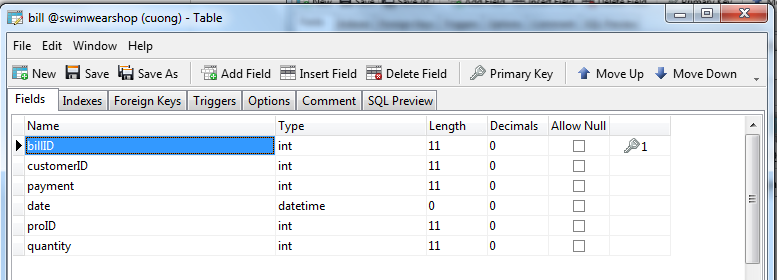












There are 7 tables available in database system.

+ Each admin has userID , email, name, password

+ Each customer has customer\_ID, name, address, bankID, email, password

+ Each bankaccount has bankID, bankName, accountBalance

+ The product contains product\_iD, product\_Name, type, brand, details, price, discount, category\_ID, image

+ Each category has categoryID, categoryName, dadcategory

+ Each Cart has productID, customerID, quantity

+Each Bill has billID, customerID, productID, quantity, payment, date

### Entity Relationship diagram



### Class diagram



Basing on the entity relationship diagram and class diagram :

+ Each customer can has only one bankaccount

+ Each customer can add many product to Cart that contain the products that he/she has ordered.

+ The bill which includes many list orders in cart is generated when customer pay for this orders.

+ Each category contains many products

### Use Case Diagram



Basing on the use case diagram :

+ The guest can only view products, search item, and register new account to become a customer.

+ The customers, admins can login/logout, edit profile, view products.

+ The customers can search products, add/remove products from cart, pay the bill and view their transaction.

+ The admins can update products to database system, the can also manage the customers.

### Activity Diagram



The guests can create an account to become a customer, they can access register page and input information such as email, password, name, address, bank account to register. The system will check the inputted value and update the new account to the database system if the guest input correctly. Then system will return login page for guest to login to the system.



The users (admins/ customers) can login into the system. When users access login, system will displays this page with the textbox and password field for user to input username and password. The username and password inputted are compared with the information from database system, if username and password submitted is correctly then system will sends request to the homepage, else system will asks user to input again.

User click on button Logout

**Start**

Logout

System return to Login page

**End**

The current user account cookie in this browser is set to null

Only the users (admins/ customers) who had logged into the system can logout. When users click on logout button. The system will sets the current user account in user’s browser equals null. Then system sends request to the login page.

User Access Profile Page

User Input Infor to edit

Set New User Infor

Update New Infor to Database System

check

Start

incorrect

correct

Display New Profile for Customer

End

System display profile page with all of the user information

Edit profile

Only the users (admin/ customer) who had logged into the system can edit their profile. When users access edit profile page, system will displays this page with the password field and some textbox for user to input the information that they want to edit such as: username and password, address. The information are checked by the system, if information submitted is correctly then system will sets new user information for this current customer and also updates this new information to database system, then display the new profile for users, else system will asks user to input again.



The users can view products and their information by access product page and select the product they want to view, then system will get the list products from the database and show for user.

User click search button and input a String to search

**Start**

search item

Return search value

Error page

Error

**End**

System check all item in database system

No error

System displays the item(s) that is found for customer

System compare the input of customer with the item available

The users can search items by inputting a String to search and click on “search” button. The system will checks the item in database system to find out which items have some similarity with the String that customers inputted. Then system will return the search value and displays for the customers. The search value can be the list of search item or “not thing” is system cannot find the similarity with the String that customers inputted.



Only the customers can add products to their cart. When customers access set product page and select an item, input quantity then click on “+” button, system will insert this selected item into the list ordered, then system will updates the cart in the database system.



Only the customers can remove products from their shopping cart, they can access their shopping bag page and click the delete button, actually click on the symbol “X” beside the orders they want to remove. Then system will delete this order from the database system.



Only the customers can pay for the bill to get the products they had ordered. When customers access shopping cart page, system will displays the list of orders that contains all the products and their quantities that customer had added to cart. Then if customers click on “Pay by account balance” button then the system will check the account balance of the customer is enough to pay the bill or not, if not show message “not enough money” , else update the transaction to the database system.



Only the admins can manage the users in the system. The admin can access customer management page ,then system get the list customers from database system and display all the customers for admin, then admin can view transaction of any customer they like.



Only the admins can add new product to database system. The admin can access product management page then system get the list products from database system and display all the products for admin, then admin can click on button “add new product” and input all information of a products to add. Then the system will check the inputted value, if everything is correct then system will inserts the product to database system, else admin have to input again.

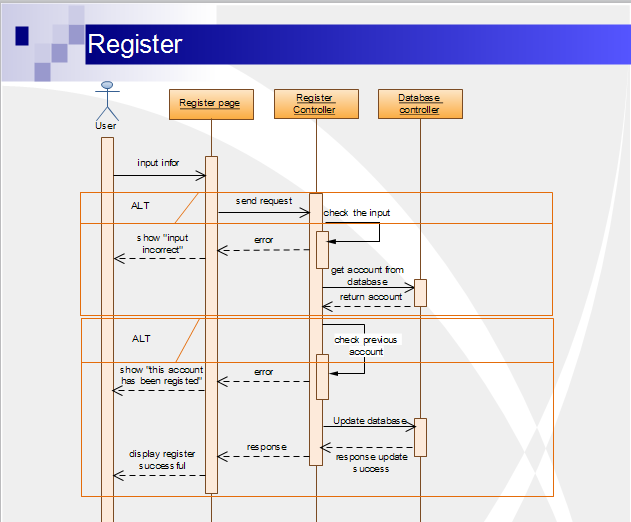


Only the admins can delete products from database system. The admin can access product management page then system get the list products from database system and display all the products for admin, then admin can click on button “delete”, the system will delete the product from database system.

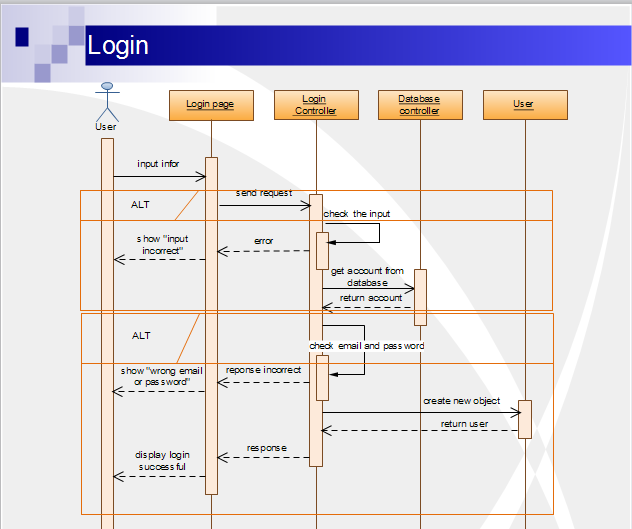


Only the admins can edit products. The admin can access product management page then system get the list products from database system and display all the products for admin, then admin can select the product to input the information of a products to edit. Then the system will check the inputted value, if everything is correct then system will update new information of the product to database system, else admin have to input again.

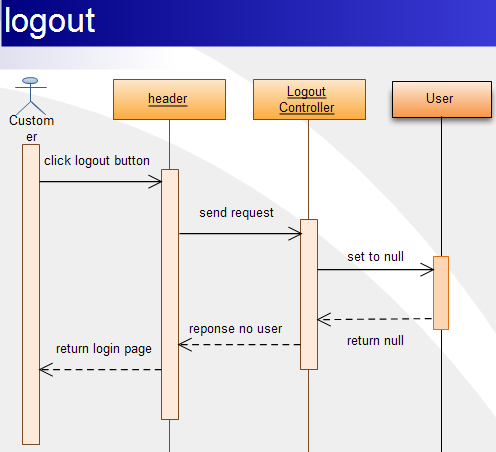
### Sequence Diagram



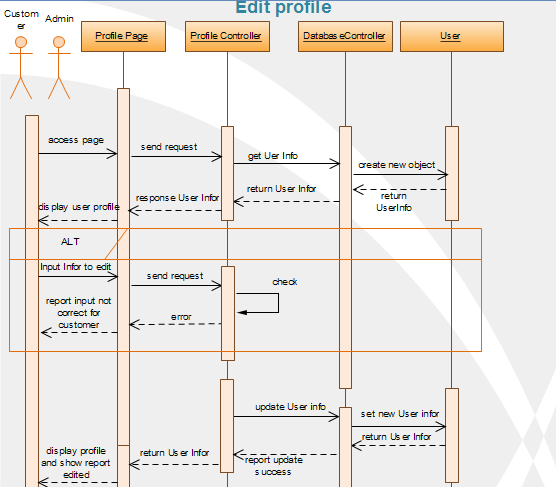
User input the information to register and the system will send request to register controller page, then check the input, if incorrect show “input incorrect” else system will get the accounts in the database and use check the current input of user is the same as one of these previous account or not. If something error, system show “this account has been registered”, else update the account to the database system and display for user.



User input the information to login and the system will send request to login controller to check the input, if incorrect show “input incorrect” else system will get the accounts in the database and use check the current input of user is the same as one of these previous account or not. If the email and password inputted correct then system will create new object of class user and display login successfully else system ask to input again.

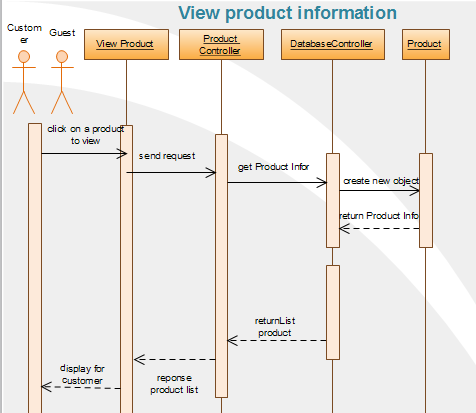


User click logout button and the system will send request to logout controller, then set the current user local on this browser equals null then return login page.

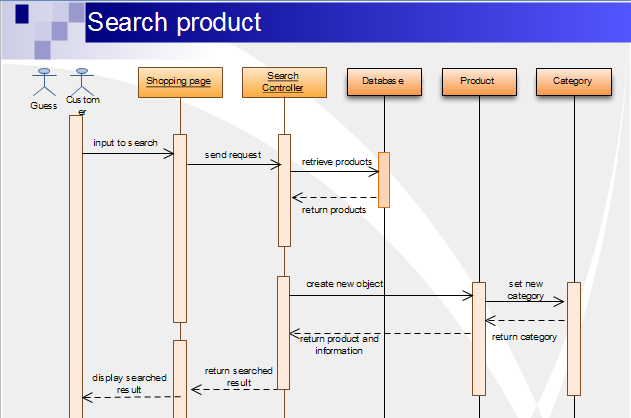


When user access the profile page, system will send request to profile controller then get all the user from database.

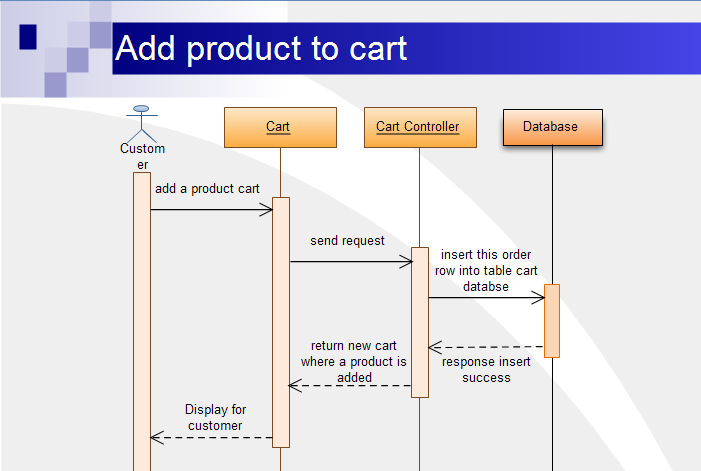
User input the information to edit and the system will send request to profile controller page, then check the input, if incorrect show “input incorrect” else system will update the information of current account of customer to the database and display “edit successfully”



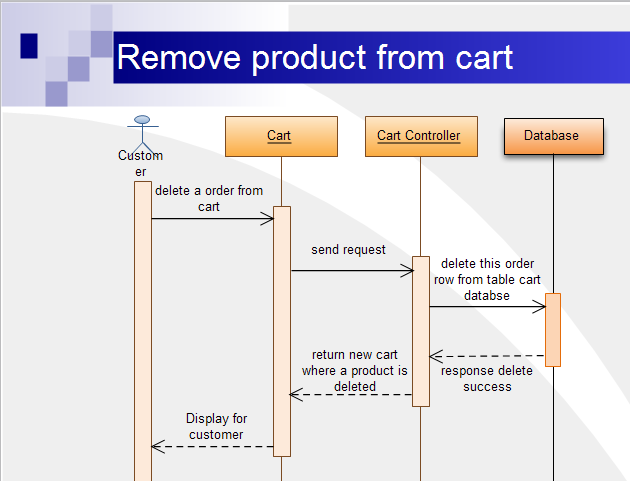
When user click a product to view, system will send request to product controller to get all the information of this product from database. Then return the list of product’s information and display for user.



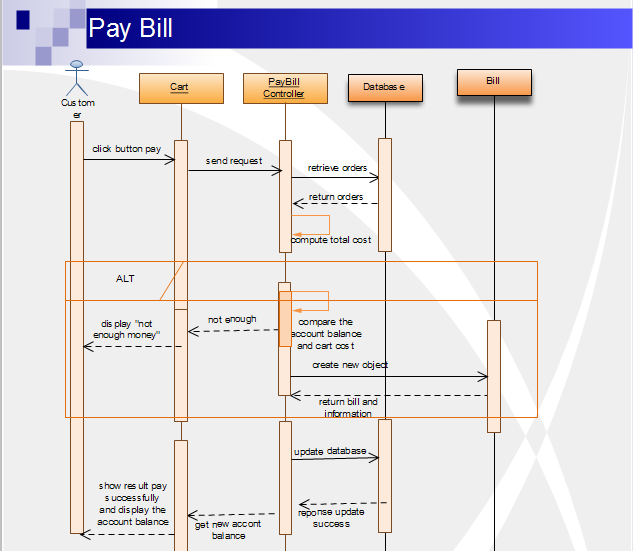
User input a String to search and system send request to Search Controller to retrieve the products from database to find out the search value and display for user. System also return the search value by product information and category.



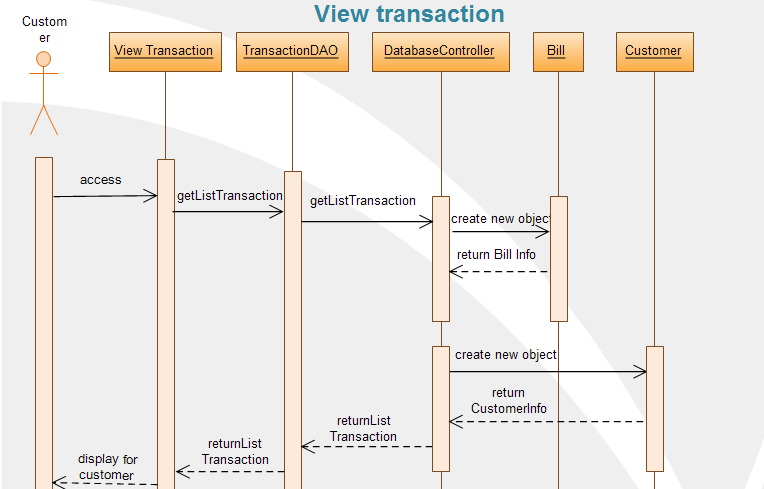
When customer add product to cart, the system send request to cart controller , then insert the product to the database system then return the new cart which already contains the product added ad display for user.



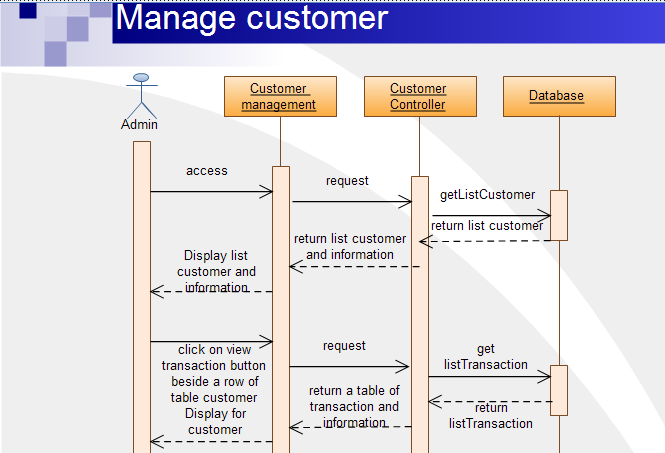
When customer remove from cart, the system send request to cart controller , then delete the product from the database system then return the new cart which already contains the product removed and display for customer.



When customer click button pay for the cart, the system send request to pay bill controller, then retrieve the orders from the database to compute the total cost of the cart. System will compare this total payment with account balance of the customer, if not enough money, system display “not enough money” else update the transaction to database system and show result the update successfully, compute and display the new account balance after paying the bill for customer.

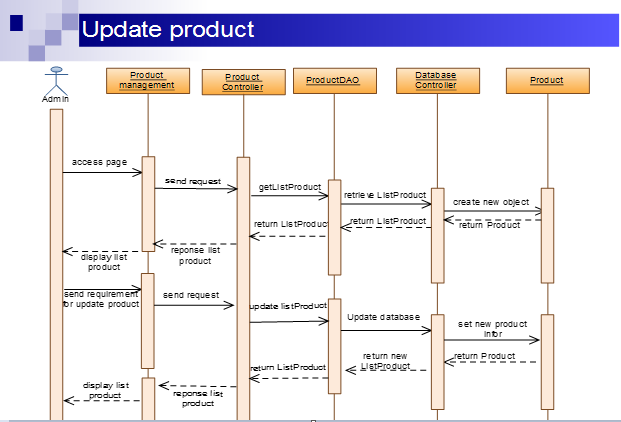


When customer access view transaction page, system will get all the transaction of the customer from database system and return the this list transaction then display for the customer.



When admin goes to customer management page, the system will send request to Customer controller and get the list of customers form database system then show for admin.

Admin can click on view transaction button match with a customer and system will send request to customer controller page to get the list transaction of this customer and return a table of transaction to display for admin.



When admin goes to product management page, the system will send request to product controller and get the list of customers form database system then show for admin.

Admin sends requirement to update the product, system will send request to product controller and update the product list to database system. Then return the new list of products and display this list for admin.

### Reference

* Team Discussion
* Vision & Scope
* Software Requirement Specification
* Project Plan document
* Software Engineer, by Dick Hamlet and Joe Maybee
* Software Design Template, by Brad Appleton