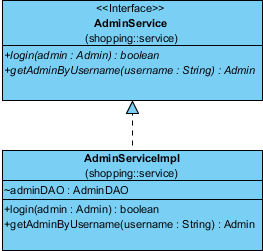
Method Design

1. Service Classes
   1. Admin Service for Admin Entity



* + 1. Method-01 getAdminByUsername(username : String) : Admin

This method is used to retrieve the admin object data from database by username.

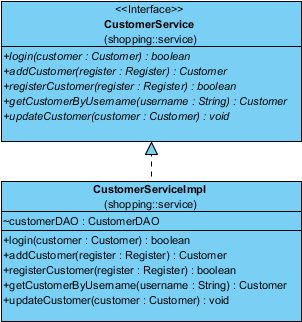
|  |  |
| --- | --- |
| Method Name | getAdminByUsername |
| Parameter Lists | String username |
| Return Type | Admin |
| Implementation description | Retrieve the admin object data from database by username.  If the admin with the input username exists in database, method returns the admin object.  Otherwise, method returns null. |

* + 1. Method-02 login(admin : Admin) : Boolean

This method is used to receive and validate the username and password customers input for login.

|  |  |
| --- | --- |
| Method Name | Login |
| Parameter Lists | Admin admin |
| Return Type | Boolean |
| Implementation description | Receive and store the input data of username and password for admin login as the Admin object.  Check whether the admin object data exists in database or not.  If the admin object exists in the database, which means the input of username and password are matched, method returns true.  Otherwise, method returns false. |

* 1. Customer Service for Customer Entity



* + 1. Method-03 getCustomerByUsername(username : String) : Customer

This method is used to retrieve the customer object data from database by username.

|  |  |
| --- | --- |
| Method Name | getCustomerByUsername |
| Parameter Lists | String username |
| Return Type | Customer |
| Implementation description | Retrieve the customer object data from database by username.  If the customer with the input username exists in database, method returns the customer object.  Otherwise, method returns null. |

* + 1. Method-04 registerCustomer(register : Register) : Boolean

This method is used to receive the input data of customer for registration and validate the registration data.

|  |  |
| --- | --- |
| Method Name | registerCustomer |
| Parameter Lists | Register register |
| Return Type | Boolean |
| Implementation description | Receive and store the input data of customer for registration as a Register object.  Retrieve the customer object data in database by the username of Register object that customer inputs.  If there exists the customer with the username input, method sets the error message “username is existed. ” and returns false.  Else if the password is not matched with the re password, method sets the error message “password and repassword are not matched. ” and returns false.  Else method returns true. |

* + 1. Method-05 addCustomer(register : Register) : Customer

This method is used to add a new valid customer object data into the database.

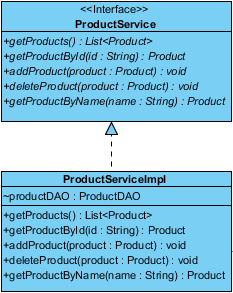
|  |  |
| --- | --- |
| Method Name | addCustomer |
| Parameter Lists | Register register |
| Return Type | Customer |
| Implementation description | Receive and store the registration data of customer in a Register object.  Convert the Register object into matched Customer object.  Save the Customer object in the database.  Return the Customer object. |

* + 1. Method-06 updateCustomer(customer : Customer) : void

This method is used to save the updated customer object data in the database.

|  |  |
| --- | --- |
| Method Name | updateCustomer |
| Parameter Lists | Customer customer |
| Return Type | void |
| Implementation description | Receive the customer object data and store it in database. |

* 1. Product Service for Product Entity



* + 1. Method-07 getProducts() : List<Product>

This method is used to retrieve all the Product object data from the database.

|  |  |
| --- | --- |
| Method Name | getProducts |
| Parameter Lists | None |
| Return Type | List<Product> |
| Implementation description | Retrieve all the Product object data from the database.  If there are products in the database, method returns a list of products.  Otherwise, method returns null. |

* + 1. Method-08 getProductById(id : String) : Product

This method is used to retrieve the Product object data from database by product id.

|  |  |
| --- | --- |
| Method Name | getProductById |
| Parameter Lists | String id |
| Return Type | Product |
| Implementation description | Receive the product id input.  Retrieve the Product object from database by input id.  If the there is a matched Product object with the input id in the database, method returns the product.  Otherwise, method returns null. |

* + 1. Method-09 getProductByName(name : String) : Product

This method is used to retrieve the Product object data from database by product name.

|  |  |
| --- | --- |
| Method Name | getProductById |
| Parameter Lists | String name |
| Return Type | Product |
| Implementation description | Receive the product name input.  Retrieve the Product object from database by input name.  If the there is a matched Product object with the input name in the database, method returns the product.  Otherwise, method returns null. |

* + 1. Method-10 addProduct(product : Product) : void

This method is used to add a new valid Product object data into the database.

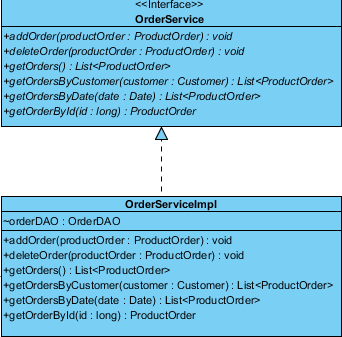
|  |  |
| --- | --- |
| Method Name | addProduct |
| Parameter Lists | Product product |
| Return Type | void |
| Implementation description | Receive the Product object data for product addition and save it in database. |

* + 1. Method-11 deleteProduct(product : Product) : void

This method is used to delete a selected product object data from the database.

|  |  |
| --- | --- |
| Method Name | deleteProduct |
| Parameter Lists | Product product |
| Return Type | void |
| Implementation description | Receive the selected Product object data in the database and delete it from database. |

* 1. Order Service for ProductOrder Entity



* + 1. Method-12 getOrders() : List<ProductOrder>

This method is used to retrieve all the ProductOrder object data from the database.

|  |  |
| --- | --- |
| Method Name | getOrders |
| Parameter Lists | None |
| Return Type | List<ProductOrder> |
| Implementation description | Retrieve all the ProductOrder object data from the database.  If there are orders in the database, method returns a list of orders.  Otherwise, method returns null. |

* + 1. Method-13 getOrdersByCustomer(customer : Customer) : List<ProductOrder>

This method is used to retrieve the ProductOrder object data from database by Customer object.

|  |  |
| --- | --- |
| Method Name | getOrdersByCustomer |
| Parameter Lists | Customer customer |
| Return Type | List<ProductOrder> |
| Implementation description | Receive the Customer object.  Retrieve the ProductOrder object from database by Customer object.  If there are matched ProductOrder objects with the input customer in the database, method returns a list of orders.  Otherwise, method returns null. |

* + 1. Method-14 getOrderById(id : long) : ProductOrder

This method is used to retrieve the ProductOrder object data from database by order id.

|  |  |
| --- | --- |
| Method Name | getOrderById |
| Parameter Lists | Long id |
| Return Type | List<ProductOrder> |
| Implementation description | Receive the order id.  Retrieve the ProductOrder object from database by order id.  If there are matched ProductOrder objects with the input order id in the database, method returns a list of orders.  Otherwise, method returns null. |

* + 1. Method-15 getOrdersByDate(date : Date) : List<ProductOrder>

This method is used to retrieve the ProductOrder object data from database by order id.

|  |  |
| --- | --- |
| Method Name | getOrdersByDate |
| Parameter Lists | Date date |
| Return Type | List<ProductOrder> |
| Implementation description | Receive the Date object.  Retrieve the ProductOrder object from database by date.  If there are matched ProductOrder objects with the input date in the database, method returns a list of orders.  Otherwise, method returns null. |

* + 1. Method-16 addOrder(productOrder : ProductOrder) : void

This method is used to add a new valid ProductOrder object data into the database.

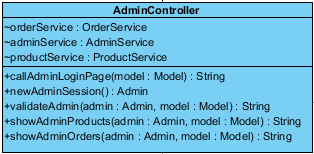
|  |  |
| --- | --- |
| Method Name | addOrder |
| Parameter Lists | ProductOrder productOrder |
| Return Type | Void |
| Implementation description | Receive the ProductOrder object data for order addition and save it in database. |

* + 1. Method-17 deleteOrder(productOrder : ProductOrder) : void

This method is used to delete a selected ProductOrder object data from the database.

|  |  |
| --- | --- |
| Method Name | deleteOrder |
| Parameter Lists | ProductOrder productOrder |
| Return Type | void |
| Implementation description | Receive the selected ProductOrder object data in the database and delete it from database. |

1. Controller Classes
   1. Admin Controller



* + 1. Method-18 newAdminSession() : Admin

This method is used to create a new session for Admin.

|  |  |
| --- | --- |
| Method Name | newAdminSession |
| Parameter Lists | None |
| Return Type | Admin |
| Implementation description | Create a new Admin object as an adminSession and return it. |

* + 1. Method-19 callAdminLoginPage(model : Model) : String

This method is used to call the login page for Admin to receive the input of username and password.

|  |  |
| --- | --- |
| Method Name | callAdminLoginPage |
| Parameter Lists | Model model |
| Return Type | void |
| Implementation description | Receive the request to call the login page for Admin.  Return the String which refers to the login page for Admin in the View.  Receive the input of username and password for Admin login and store it in modelAttribute. |

* + 1. Method-20 validateAdmin(admin : Admin, model : Model) : String

This method is used to validate the input of username and password for Admin login.

|  |  |
| --- | --- |
| Method Name | validateAdmin |
| Parameter Lists | Admin admin , Model model |
| Return Type | String |
| Implementation description | Receive the request to validate the input of username and password for Admin login.  Call the login method from adminService to validate the username and password.  Return the String which refers to the page after admin logs into the system, successful page or error message.  If the username and password are valid and matched, method retrieves the admin data from database and stores it in adminSession. |

* + 1. Method-21 showAdminProducts(admin : Admin, model : Model) : String

This method is used to call the page to show products list in the view of an Admin.

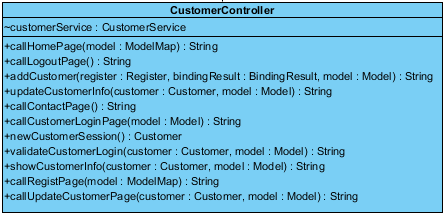
|  |  |
| --- | --- |
| Method Name | showAdminProducts |
| Parameter Lists | Admin admin , Model model |
| Return Type | String |
| Implementation description | Receive the request to show products list in the view of an Admin.  Receive the admin data from adminSession.  Retrieve all the product data from database and store them in modelAtrribute.  Return the String which refers to the page to show the products list. |

* + 1. Method-22 showAdminOrders(admin : Admin, model : Model) : String

This method is used to call the page to show shopping history list in the view of Admin.

|  |  |
| --- | --- |
| Method Name | showAdminOrders |
| Parameter Lists | Admin admin , Model model |
| Return Type | String |
| Implementation description | Receive the request to show shopping history in the view of an Admin.  Receive the admin data from adminSession.  Retrieve all the order data as shopping history from database and store them in modelAtrribute.  Return the String which refers to the page to show the shopping history list. |

* 1. Customer Controller



* + 1. Method-23 newCustomerSession() : Customer

This method is used to create a new session for Admin.

|  |  |
| --- | --- |
| Method Name | newCustomerSession |
| Parameter Lists | None |
| Return Type | Customer |
| Implementation description | Create a new Customer object as a customerSession and return it. |

* + 1. Method-24 callHomePage(model : ModelMap) : String

This method is used to call the home page.

|  |  |
| --- | --- |
| Method Name | callHomePage |
| Parameter Lists | Model model |
| Return Type | String |
| Implementation description | Receive the request to show the home page.  Return the String which refers to the home page. |

* + 1. Method-25 callCustomerLoginPage(model : Model) : String

This method is used to call the page to receive the username and password for customer to login.

|  |  |
| --- | --- |
| Method Name | callCustomerLoginPage |
| Parameter Lists | Model model |
| Return Type | String |
| Implementation description | Receive the request to show the page to receive the username and password for customer to login.  Receive the input of username and password for customer login and store it in modelAttribute.  Return the String which refers to the page for customer login. |

* + 1. Method-26 validateCustomerLogin(customer : Customer, model : Model) : String

This method is used to validate the input of username and password for customer login.

|  |  |
| --- | --- |
| Method Name | validateCustomerLogin |
| Parameter Lists | Customer customer, Model model |
| Return Type | String |
| Implementation description | Receive the request to validate the input of username and password for customer login.  Receive the input of username and password for customer login.  Call the method login from CustomerService to validate the input data.  If the input data can be logged into the system, the matched customer data will be retrieved from the database and added to customerSession.  Return to the String which refers to the page after customer logs into the system.  Otherwise, method returns to the page showing the failed login. |

* + 1. Method-27 showCustomerInfo(customer : Customer, model : Model) : String

This method is used to call the page to show the personal information of customer.

|  |  |
| --- | --- |
| Method Name | validateCustomerLogin |
| Parameter Lists | Customer customer, Model model |
| Return Type | String |
| Implementation description | Receive the request to show the personal information of customer.  Retrieve the customer data from database by the customerSession.  Return to the String which refers to the page showing the personal information of customer. |

* + 1. Method-28 callLogoutPage() : String

This method is used to call the page after logout.

|  |  |
| --- | --- |
| Method Name | callLogoutPage |
| Parameter Lists | None |
| Return Type | String |
| Implementation description | Receive the request to show the page after logout.  Return the String which refers to the page after logout. |

* + 1. Method-29 callRegistPage(model : ModelMap) : String

This method is used to call the page for customer registration.

|  |  |
| --- | --- |
| Method Name | callRegistPage |
| Parameter Lists | ModelMap model |
| Return Type | String |
| Implementation description | Receive the request to call the page for customer registration.  Return to the String which refers to the page receiving the registration data. |

* + 1. Method-30 addCustomer(register : Register, bindingResult : BindingResult, model : Model) : String

This method is used to add a valid customer into the database.

|  |  |
| --- | --- |
| Method Name | addCustomer |
| Parameter Lists | Register register, BindingResult bindingResult, Model model |
| Return Type | String |
| Implementation description | Receive the request to add a valid customer into the database.  Receive the Register object and call the registerCustomer method from CustomerService to validate the registration data and register it.  If registration is successful, method returns to the String which refers to show successful registration.  Otherwise, method returns to the page showing failure of registration. |

* + 1. Method-31 callUpdateCustomerPage(customer : Customer, model : Model) : String

This method is used to call the page for customer to update their information.

|  |  |
| --- | --- |
| Method Name | callUpdateCustomerPage |
| Parameter Lists | Customer customer, Model model |
| Return Type | String |
| Implementation description | Receive the request to call the page for customer to update their information.  Return to the String which refers to the page receiving the updated data. |

* + 1. Method-32 updateCustomerInfo(customer : Customer, model : Model) : String

This method is used to update the customer information as the edition.

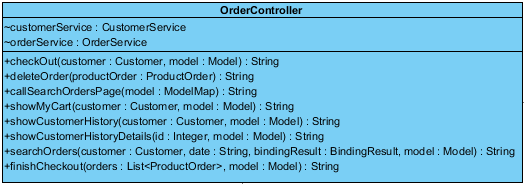
|  |  |
| --- | --- |
| Method Name | updateCustomerInfo |
| Parameter Lists | Customer : customer, Model : model |
| Return Type | String |
| Implementation description | Receive the request to update the customer information as the edition.  Receive the edition data of the customer and call the method updateCustomer from CustomerService to update the customer information in database.  Store the updated customer in the customerSession.  Return to the String which refers to the page to show the successful update. |

* + 1. Method-33 callContactPage() : String

This method is used to call the page for contact.

|  |  |
| --- | --- |
| Method Name | callContactPage |
| Parameter Lists | None |
| Return Type | String |
| Implementation description | Receive the request to call the page for contact.  Return to the String which refers to the page for contact. |

* 1. Order Controller



* + 1. Method-34 showMyCart(customer : Customer, model : Model) : String

This method is used to show the unfinished orders in the cart of the customer.

|  |  |
| --- | --- |
| Method Name | showMyCart |
| Parameter Lists | Customer customer, Model model |
| Return Type | String |
| Implementation description | Receive the request to show the unfinished orders in the cart of the customer.  Call the getOrdersByCustomer method from OrderService to retrieve the orders data of the customer.  Create a list to store the unfinished orders.  Return to the String which refers to the page to show unfinished orders list in the cart of the customer. |

* + 1. Method-35 deleteOrder(productOrder : ProductOrder) : String

This method is used to delete the selected order from the orders list in the cart of customer.

|  |  |
| --- | --- |
| Method Name | deleteOrder |
| Parameter Lists | ProductOrder productOrder |
| Return Type | String |
| Implementation description | Receive the request to delete the selected order from the orders list in the cart of customer.  Retrieve the selected order data from database and delete it.  Return to the String which refers to the unfinished orders list in the cart of the customer. |

* + 1. Method-36 showCustomerHistory(customer : Customer, model : Model) : String

This method is used to call the page to show the shopping history of the customer.

|  |  |
| --- | --- |
| Method Name | showCustomerHistory |
| Parameter Lists | Customer customer, Model model |
| Return Type | String |
| Implementation description | Receive the request to call the page to show the shopping history of the customer.  Retrieve all the orders data of the customer from database and add them to modelAttribute.  Return to the String which refers to the cart page showing the shopping history. |

* + 1. Method-37 showCustomerHistoryDetails(id : Integer, model : Model) : String

This method is used to call the page to show the details of selected shopping history of the customer.

|  |  |
| --- | --- |
| Method Name | showCustomerHistoryDetails |
| Parameter Lists | Integer id, Model model |
| Return Type | String |
| Implementation description | Receive the request to show the details of selected shopping history of the customer.  Retrieve the order (history) details of the selected order from database and add it to modelAttribute.  Return to the String which refers to the page showing the details of selected shopping history of the customer. |

* + 1. Method-38 callSearchOrdersPage(model : ModelMap) : String

This method is used to call the page for admin to search orders (shopping history) of customers.

|  |  |
| --- | --- |
| Method Name | callSearchOrdersPage |
| Parameter Lists | ModelMap model |
| Return Type | String |
| Implementation description | Receive the request to call the page for searching the orders (shopping history) by date or customer.  Return to the String which refers to the page receiving the data for searching orders (shopping history) |

* + 1. Method-39 searchOrders(, date : Date, bindingResult : BindingResult, model : Model) : String

This method is used to search orders (shopping history) by customer or date.

|  |  |
| --- | --- |
| Method Name | searchOrders |
| Parameter Lists | Customer customer, Date date, BindingResult bindingResult, Model model |
| Return Type | String |
| Implementation description | Receive the request to search the orders (shopping history) by date or customer.  Call the getOrdersByCustomer and getOrdersByDate method from OrderService to retrieve all the matched orders data from database.  Add them to modelAtrribute.  Return to the String which refers to the page showing the search result. |

* + 1. Method-40 checkOut(customer : Customer, model : Model) : String

This method is used to check out in the cart of customer.

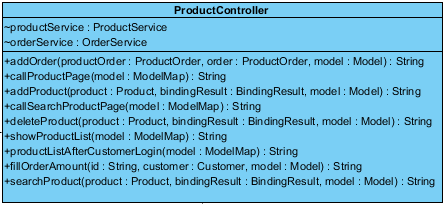
|  |  |
| --- | --- |
| Method Name | checkOut |
| Parameter Lists | Customer customer, Model model |
| Return Type | String |
| Implementation description | Receive the request to check out in the cart of customer.  Retrieve all the unfinished orders (shopping history) of the customer.  Calculate the total prices of orders in the cart.  Return to the String which refers to the page showing the information of checkout. |

* + 1. Method-41 finishCheckout(orders : List< ProductOrder>, model : Model) : String

This method is used to call the page showing the checkout result.

|  |  |
| --- | --- |
| Method Name | finishCheckout |
| Parameter Lists | List< ProductOrder> orders, Model model |
| Return Type | String |
| Implementation description | Receive the request to call the page showing the checkout result.  Retrieve all the unfinished orders (shopping history) of the customer.  Change their status from false to true to commit the order is finished.  Return to the String which refers to the page showing the checkout result. |

* 1. Product Controller



* + 1. Method-42 showProductList(model : ModelMap) : String

This method is used to call the page to show products list.

|  |  |
| --- | --- |
| Method Name | showProductList |
| Parameter Lists | ModelMap model |
| Return Type | String |
| Implementation description | Receive the request to call the page to show products list.  Retrieve all the product object data from database and add them to modelAttribute.  Return to the String which refers to the page to show products list. |

* + 1. Method-43 productListAfterCustomerLogin(model : ModelMap) : String

This method is used to call the page to show products list in the view of a customer.

|  |  |
| --- | --- |
| Method Name | productListAfterCustomerLogin |
| Parameter Lists | ModelMap model |
| Return Type | String |
| Implementation description | Receive the request to call the page to show products list in the view of a customer.  Retrieve all the product object data from database and add them to modelAttribute.  Return to the String which refers to the page to show products list in the view of a customer. |

* + 1. Method-44 fillOrderAmount(id : String, customer : Customer, model : Model) : String

This method is used to call the page to fill out the amount of the product of the order.

|  |  |
| --- | --- |
| Method Name | fillOrderAmount |
| Parameter Lists | String id, Customer customer, Model model |
| Return Type | String |
| Implementation description | Receive the request to call the page to fill out the amount of the product of the order.  Retrieve the product object data from database and add it to modelAttribute.  Create a new Productorder object, and set it with the product, and customerSession.  Add this order both in database and modelAttribute.  Return to the String which refers to the page to fill out the amount of the product of the order. |

* + 1. Method-45 addOrder(productOrder : ProductOrder, order : ProductOrder, model : Model) : String

This method is used to add a new order in customer’s cart with the input amount of product.

|  |  |
| --- | --- |
| Method Name | addOrder |
| Parameter Lists | ProductOrder productOrder, ProductOrder order,Model model |
| Return Type | String |
| Implementation description | Receive the request to add a new order into the customer’s cart.  Retrieve the order object data from database and set the amount with the input amount.  Update the order data in database.  Return to the String which refers to the page showing the orders list of the customer. |

* + 1. Method-46 callProductPage(model : ModelMap) : String

This method is used to call the page to add a new product for admin.

|  |  |
| --- | --- |
| Method Name | callProductPage |
| Parameter Lists | ModelMap model |
| Return Type | String |
| Implementation description | Receive the request to call the page to add a new product for admin.  Return to the String which refers to the page receiving the data for new product addition. |

* + 1. Method-47 addProduct(product : Product, bindingResult : BindingResult, model : Model) : String

This method is used to add a valid product into database.

|  |  |
| --- | --- |
| Method Name | addProduct |
| Parameter Lists | Product product, BindingResult bindingResult, Model model |
| Return Type | String |
| Implementation description | Receive the request to add a valid product into database.  Validate the product object.  Call the method addProduct from ProductService.  If the product addition is successful, method returns to the page showing products list.  Otherwise, method returns to the page showing the failed addition. |

* + 1. Method-48 deleteProduct(product : Product, bindingResult : BindingResult, model : Model) : String

This method is used to delete a selected Product object data from the database.

|  |  |
| --- | --- |
| Method Name | deleteProduct |
| Parameter Lists | Product product, BindingResult bindingResult, Model model |
| Return Type | String |
| Implementation description | Receive the selected Product object data in the database and delete it from database.  If the deletion is successful, method returns to the String which refers to the page showing successful deletion.  Otherwise, method returns to the String which refers to the failed message. |

* + 1. Method-49 callSearchProductPage(model : ModelMap) : String

This method is used to call the page which receives the data for searching the product.

|  |  |
| --- | --- |
| Method Name | callSearchProductPage |
| Parameter Lists | ModelMap model |
| Return Type | String |
| Implementation description | Receive the request to call the page for searching product.  Return to the String which refers to the page receiving the data for searching product. |

* + 1. Method-50 searchProduct(product : Product, bindingResult : BindingResult, model : Model) : String

This method is used to search product by id or name.

|  |  |
| --- | --- |
| Method Name | searchProduct |
| Parameter Lists | Product product, BindingResult bindingResult, Model model |
| Return Type | String |
| Implementation description | Receive the request to search product by id or name.  Call the getProductById and getProductByName method from ProductService to retrieve the product data from database.  Return to the String which refers to the page showing the search result. |