**Class: ShoppingCartDAO**

**Unit Test Case 01 (UTC-01):**

Public shoppingCart getShoppingCart(Customer customer)

**Description**

Test the getShoppingCart(Customer customer) method to see that it can get a customer’s shopping cart with the parameter, and it cannot get a customer’s shopping cart if input wrong parameter.

**Test cases**

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Description | Input | Expected Result |
| 1 | The method can get a customer’s shopping cart by using the current customer | Current customer | pass |
| 2 | The method can get a customer’s shopping cart by using another the customer | Another customer | fail |

**Unit Test Case 02 (UTC-02):**

Public ShoppingCart addProduct(Product product)

**Description**

Test the addProduct(Product product) method to see that it can Put the product that customer chose to the customer’s shopping cart or not.

**Test cases**

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Description | Input | Expected Result |
| 1 | The method can Put the product that customer chose to the customer’s shopping cart. | The product that customer chose | pass |

**Unit Test Case 03(UTC-03):**

Public ShoppingCart DeleteProduct(Product product)

**Description**

Test the DeleteProduct(Product product) method to see that it can Delete the product that inside the current customer shopping cart or not.

**Test cases**

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Description | Input | Expected Result |
| 1 | The method can Delete the product that inside the current customer shopping cart. | the product that has be inside the current customer shopping cart | pass |
| 2 | The method cannot Delete the product that inside the current customer shopping cart. | the product that isn’t inside the current customer shopping cart | fail |

**Unit Test Case 04(UTC-04):**

Public getProducts(ShoppingCart shoppingCart)

**Description**

Test the List<Product> getProducts(ShoppingCart shoppingCart) method to see that it can get the all product that inside the shopping cart or not.

**Test cases**

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Description | Input | Expected Result |
| 1 | The method can get the all product that inside the shopping cart. | The current customer’s currently shopping cart | pass |
| 2 | The method cannot get the all product that inside the shopping cart. | Other customer’s currently shopping cart | fail |
| 3 | The method cannot get the all product that inside the shopping cart. | the current customer’s other shopping cart | fail |

**Unit Test Case 05 (UTC-05):**

Public ShoppingCart clearShoppingCart(ShoppingCart shoppingCart)

**Description**

Test the clearShoppingCart(ShoppingCart shoppingCart) method to see that it can delete all products in the shopping cart or not.

**Test cases**

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Description | Input | Expected Result |
| 1 | The method can delete all products in the shopping cart. | The current customer’s currently shopping cart | pass |
| 2 | The method cannot delete all products in the shopping cart. | Other customer’s currently shopping cart | fail |
| 3 | The method cannot delete all products in the shopping cart. | the current customer’s other shopping cart | fail |

**Class: PRODUCTDAO**

**Unit Test Case 06 (UTC-06):**

Public Product getProductByName(String productName)

**Description**

Test the getProductByName(String productName) method to see that it can get a product that have same name with the parameter or not.

**Test cases**

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Description | Input | Expected Result |
| 1 | The method can get a product that have same name with correct parameter | productName=”chi 2” | pass |
| 2 | The method cannot get a product that have same name with the wrong parameter. | productName=”&abc123” | fail |

**Unit Test Case 07 (UTC-07):**

Public Product getProductByID(String productID)

**Description**

Test the getProductByID(String productID) method to see that it can get a product that have same id with the parameter or not.

**Test cases**

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Description | Input | Expected Result |
| 1 | The method can get a product that have same id with correct parameter | productID=002 | pass |
| 2 | The method cannot get a product that have same id with the wrong parameter. | productID=”abc123” | fail |
| 3 | The method cannot get a product that have same id with the wrong parameter. | productID=”abcd” | fail |
| 4 | The method cannot get a product that have same id with the wrong parameter. | productID=”$abcd” | fail |

**Unit Test Case 08 (UTC-08):**

Public Product addProduct(Product product)

**Description**

Test the addProduct(Product product) method to see that it can add a new product to the e-shopping website system or not.

**Test cases**

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Description | Input | Expected Result |
| 1 | The method can add a new product to the e-shopping website system | The product that not inside the system | pass |
| 2 | The method cannot add a new product to the e-shopping website system | The product that inside the system | fail |

**Unit Test Case 09 (UTC-09):**

Public boolean DeleteProduct(Product product)

**Description**

Test the DeleteProduct(Product product) method to see that it can delete a product in the e-shopping website system

**Test cases**

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Description | Input | Expected Result |
| 1 | The method can delete a product in the e-shopping website system | The product that inside the system | pass |
| 2 | The method cannot delete a product in the e-shopping website system | The product that not inside the system | fail |

**Unit Test Case 10 (UTC-10):**

Public product updateProduct(Product product)

**Description**

Test the updateProduct(Product product) method to see that it can change a product of e-shopping website system

**Test cases**

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Description | Input | Expected Result |
| 1 | The method can change a product of e-shopping website system | The product that inside the system | pass |
| 2 | The method cannot change a product in the e-shopping website system | The product that not inside the system | fail |

**Class: ShoppingHistoryDAO**

**Unit Test Case 11 (UTC-11):**

Public ShoppingHistory getShoppingHistoryById(String customerID)

**Description**

Test the getShoppingHistoryById(String customerID) method to see that it can get the shopping history belong to special customer that customer id same as the parameter.

**Test cases**

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Description | Input | Expected Result |
| 1 | The method can get the shopping history belong to special customer that customer id same as the parameter. | customerID=002 | pass |
| 2 | The method cannot get the shopping history belong to special customer that customer id same as the parameter. | customerID=”abc123” | fail |
| 3 | The method cannot get the shopping history belong to special customer that customer id same as the parameter. | customerID=”abcd” | fail |
| 4 | The method cannot get the shopping history belong to special customer that customer id same as the parameter. | customerID=”$abcd” | fail |

**Unit Test Case 11 (UTC-11):**

Public List<ShoppingHistory> getShoppingHistories()

**Description**

Test the getShoppingHistories() method to see that it can get all of customers shopping histories in the e-shopping website

**Test cases**

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Description | Input | Expected Result |
| 1 | The method can get all of customers shopping histories in the e-shopping website. | - | pass |

**Unit Test Case 12 (UTC-12):**

Public ShoppingHistory AddShoppingHistory(ShoppingCart shoppingCart)

**Description**

Test the AddShoppingHistory(ShoppingCart shoppingCart) method to see that it can make a shopping cart that have check out to be a shopping history with the customer name and customer id

**Test cases**

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Description | Input | Expected Result |
| 1 | The method can make a shopping cart that have check out to be a shopping history with the customer name and customer id | The Shopping cart that the customer has check out | pass |
| 2 | The method cannot make a shopping cart that have check out to be a shopping history with the customer name and customer id | The Shopping cart that the customer didn’t check out | fail |

**Class: Register**

**Unit Test Case 13 (UTC-13):**

Public Boolean validateUsername(String username)

**Description**

Test the validateUsername(String username) method to see that it can Validate the format of username before user register a new account or not.

**Test cases**

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Description | Input | Expected Result |
| 1 | The method can Validate the format of username before user register a new account by using email format | Username=”54511@gmail.com” | Return true |
| 2 | The method can Validate the format of username before user register a new account by using not email format | Username=”54511g.com” | Return false |

**Unit Test Case 14 (UTC-14):**

Public Boolean validatePassword(String password)

**Description**

Test the validatePassword(String password) method to see that it can Validate the format of user password before the user register a new account.

**Test cases**

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Description | Input | Expected Result |
| 1 | The method can Validate the format of user password before the user register a new account with password format is 5-10 characters | password =”12345678” | Return true |
| 2 | The method can Validate the format of user password before the user register a new account with password format less than 5 characters | password=”1234” | Return false |
| 3 | The method can Validate the format of user password before the user register a new account with password format more than 10 characters | password=”1234678901” | Return false |

**Unit Test Case 15 (UTC-15):**

Public Boolean validateRepassword(String rePassword)

**Description**

Test the validateRepassword(String rePassword) method to see that it can Validate repassword is same as password or not.

**Test cases**

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Description | Input | Expected Result |
| 1 | The method can Validate repassword is same as password by using same password | password =”12345678” | Return true |
| 2 | The method can Validate repassword is same as password by using different password | password =”123456” | Return false |

**Class: Login**

**Unit Test Case 16 (UTC-16):**

Public Boolean validateUsername(String username)

**Description**

Test the validateUsername(String username) method to see that it can Validate the format of username before check the username match with user password or not.

**Test cases**

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Description | Input | Expected Result |
| 1 | The method can Validate the format of username before check the username match with user password by using email format | Username=”54511@gmail.com” | Return true |
| 2 | The method can Validate the format of username before check the username match with user password by using not email format | Username=”54511g.com” | Return false |

**Unit Test Case 14 (UTC-14):**

Public Boolean validatePassword(String password)

**Description**

Test the validatePassword(String password) method to see that it can Validate the format of user password check the username match with user password.

**Test cases**

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Description | Input | Expected Result |
| 1 | The method can Validate the format of user password check the username match with user password with password format is 5-10 characters | password =”12345678” | Return true |
| 2 | The method can Validate the format of user password check the username match with user password with password format less than 5 characters | password=”1234” | Return false |
| 3 | The method can Validate the format of user password check the username match with user password with password format more than 10 characters | password=”1234678901” | Return false |