EAD Project

SP IT Online Web Store



Done by: Poon Hao Jun (1448920)

Robert Teo Tze Hurn (1430714)

From: Dism/FT/2A/01

Lecturer: Mr Low Jin Kiat

Date: 10 Aug 2015

**Table Of Contents**

|  |  |  |
| --- | --- | --- |
|  |  | **Page** |
| **Term 1** |  |  |
| Submitting a comment (processComment.jsp) |  | **done** |
| Add Product (insertproduct.jsp) |  | **done** |
| Add Category (insertcategory.jsp) |  | **done** |
| Search Product (searchproduct.jsp) |  | **done** |
| Delete Product (deleteProduct.jsp) |  | **done** |
| Update Product (updateProduct.jsp) |  | **done** |
| View Comment (viewcomments.jsp) |  | **done** |
| Delete Comment (deleteComment.jsp) |  | **done** |
|  |  |  |
| **Member A Task** |  |  |
| Add Product To Cart (indProductPage.jsp) |  | **done** |
| View Product From Cart (displaycartController) |  | **done** |
| Update Product in Cart (upCartController) |  | **done** |
| Delete Product From Cart (delCartController) |  | **done** |
| Show Order (placeorder.jsp) |  | **done** |
| Confirm Order (recentorderController) |  | **done** |
| Recent Order |  | **done** |
|  |  |  |
| **Member B Task** |  |  |
| Register Member (createMember.jsp) |  | **done** |
| Login Role (login.jsp) |  | **done** |
| Update Member Profile (updateProfile.jsp) |  | **done** |
| View Sales Report (salesreport.jsp) |  | **TBC** |
| Logout (logout.jsp) |  | **done** |
|  |  |  |
| **Advance Features** |  |  |
| Generate Discount Code (coupons.jsp) |  | **done** |
| Encrypt & Decrypt Password (mixSmash.java) |  | **TBC** |
| Forget Password (forgetpass.jsp) |  | **done** |
| Send Email (SendMail.java) |  | **done** |
| Hosting SPIT |  | **TBC** |

Term 1

-----------------

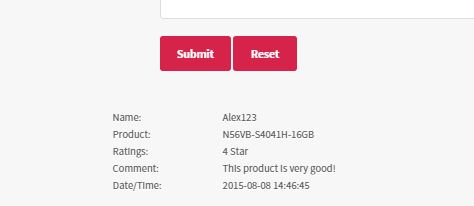
**In this section** - the basic public features of the website will be covered, as well as the various controls that an administrator is able to perform.

**Submitting a comment (indProductPage.jsp)**



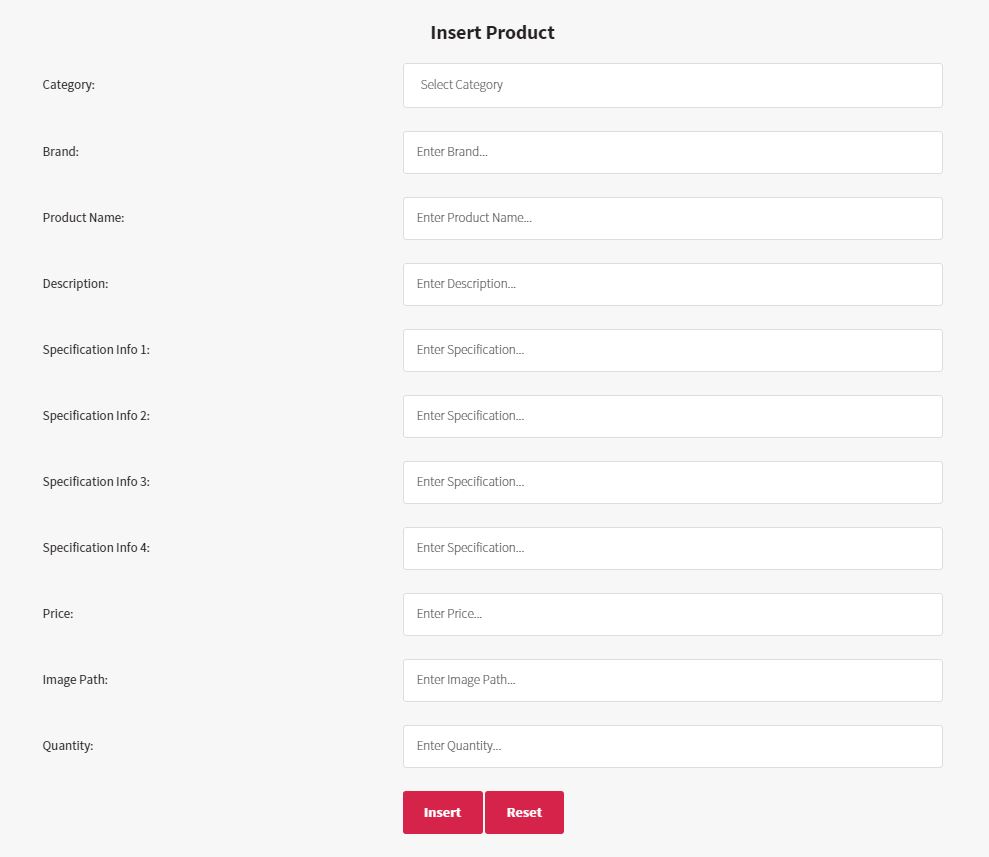
|  |
| --- |
| processComment.jsp |

**Commenting results (processComment.jsp)**



**Add Product (insertproduct.jsp)**

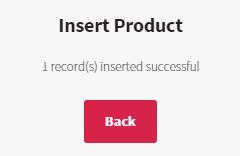
Admin wants to insert a new product:



Admin types in its specifications, etc. :



|  |
| --- |
| processinsertPro.jsp |

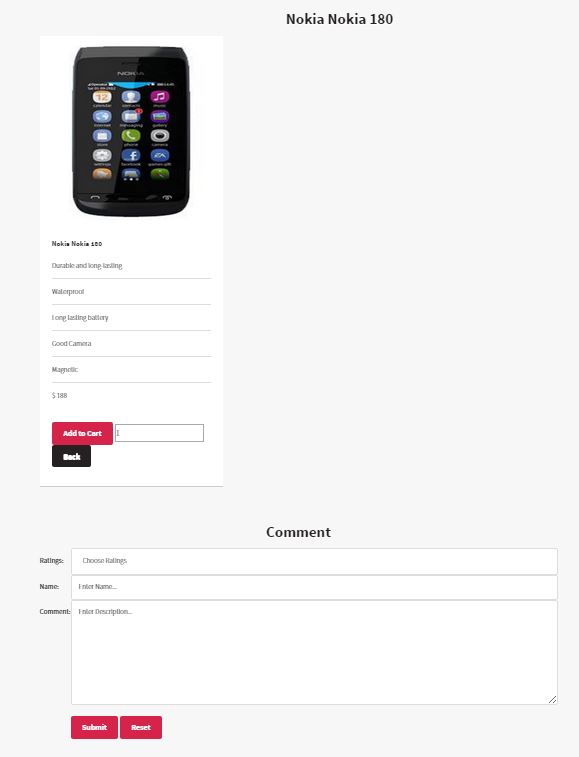


**Add Product Results**



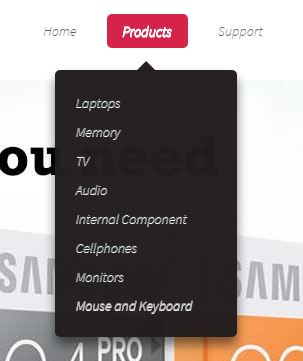
|  |
| --- |
| The specs that were previously filled, as well as the image have all appeared in the products page for cellphones as well. |

**The Product can be viewed individually too**



**Add Category (insertCategory.jsp)**

**The navigation bar currently:**



**Admin inserts a new category called Battery:**



|  |
| --- |
| **processinsCategory.jsp** |



**Add Category Results**



|  |
| --- |
| New category “Battery” has been added to the navigation bar! |

In the products page (product.jsp), battery category will appear too, with the image the administrator had inserted:





**When the administrator wishes to insert a new product, there will be the new category for him to select as well, depending on the category of the product:**



**Search Product (searchproduct.jsp)**

Admin wishes to search for Samsung products:



**Search Results (delupproduct.jsp)**



The Results will show all the products that contain “samsung” in it, whatever the category of the product.

**Delete Product (deleteProduct.jsp)**

Using the search engine to find the Nokia phone that was earlier inserted, the Nokia phone can now be deleted as well:



|  |
| --- |
| deleteProduct.jsp |

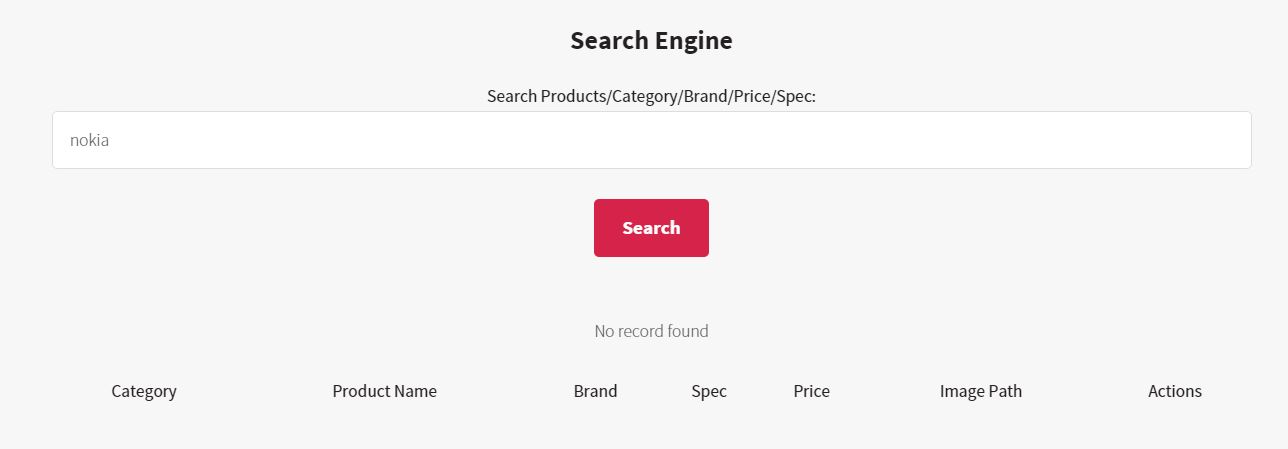


**Delete Product Results**



|  |
| --- |
| The Nokia Phone no longer exists here. |

Using the search engine, the Nokia phone can also no longer be found:



**Update Product**

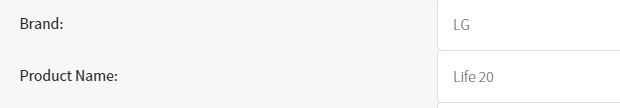
The administrator will update the samsung galaxy note 4:



|  |
| --- |
| updateProduct.jsp |



The administrator shall change Samsung Galaxy Note 4 to LG Life 20:

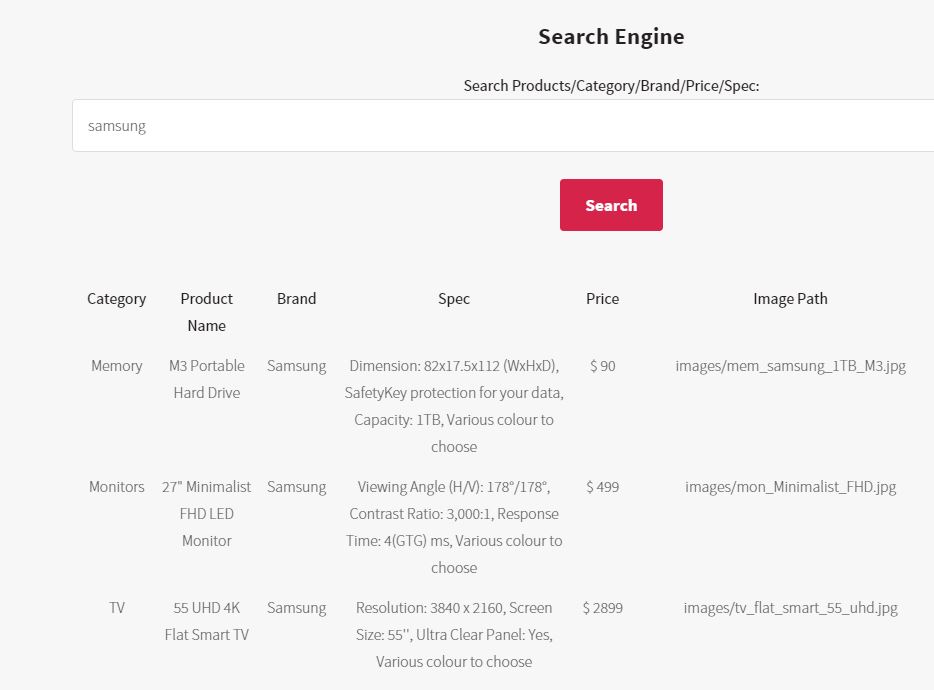






**Update Results**

When the administrator searches “samsung” on the search engine, there is one less record, which is due to the update on the Galaxy Note 4:



Searching for “LG”, however, will show the phone that was updated:



Do take note of the image path, which states “galaxy note 4”.

In the cellphones section, the phone’s name can be seen to be changed to “LG Life 20”:



**View Comments (viewcomments.jsp)**

The administrator searches for comments for “samsung” products:



(delviewcomment.jsp)



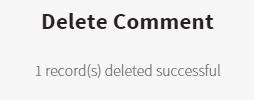
|  |
| --- |
| Comments shown are only meant for samsung products. |

**Delete Comments (deleteComment.jsp)**

The administrator will delete this comment:

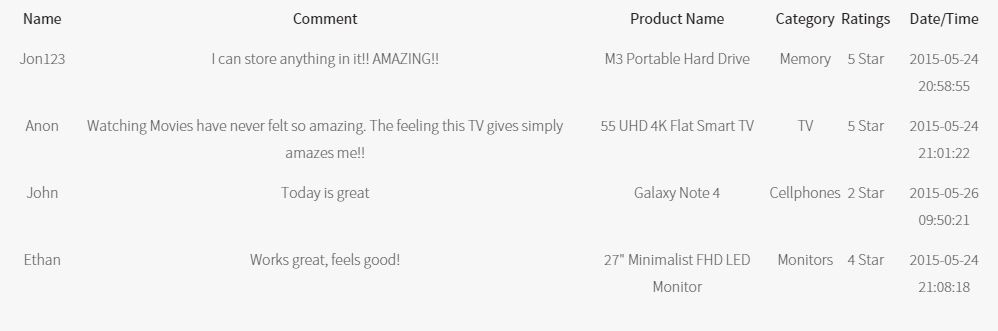


The administrator finds that comment using the viewing comments search engine:

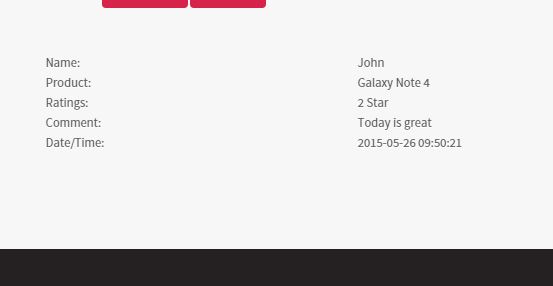
 

**Delete Comments Results**

The comment by Jonas can no longer be viewed:



In the individual page of the Galaxy Note 4, his comment is also no longer visible:



Member A

-----------------

**In this section** - It will be shown how a potential customer makes an online purchase.

**Add Product to Cart (indProductPage.jsp)**

A customer, who must be a member, will put the quantity of the product he wishes to add to cart:



|  |
| --- |
| add2CartController |

**View Product from Cart (displayCartController)**

|  |
| --- |
| add2CartController |



|  |
| --- |
| displayCartController |



The product added to the cart will be displayed, along with the quantity the customer wants:



**Update Product in Cart (upCartController)**



|  |
| --- |
| upCartController |



|  |
| --- |
| displayCartController |



Product’s quantity will be updated alongside its total price value.

**Delete Product from Cart (delCartController)**

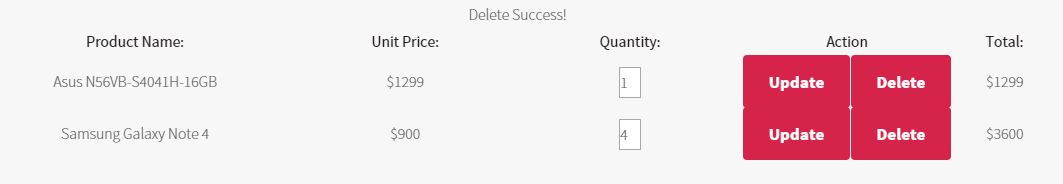


|  |
| --- |
| delCartController |



|  |
| --- |
| displayCartController |





The product that was deleted can no longer be found in the shopping cart.

**Show Order (placeorder.jsp)**



|  |
| --- |
| placeOrder.jsp |

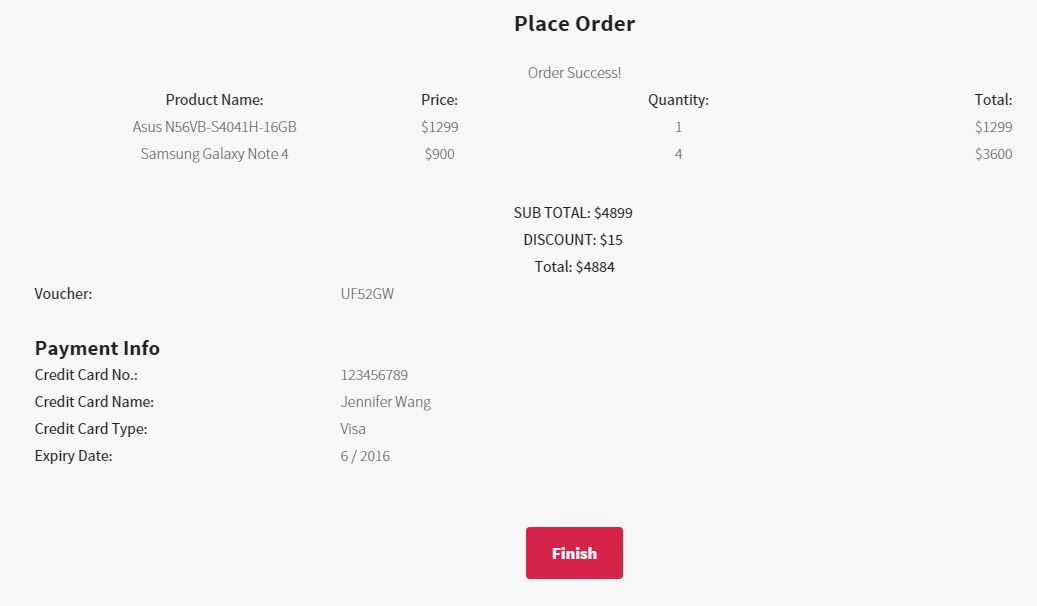


**Confirm Order (recentorderController)**



|  |
| --- |
| placeorderController |





**Recent Order (recentorderController)**



|  |
| --- |
| recentorderController |





|  |
| --- |
| The recent order by Jennifer Wang |

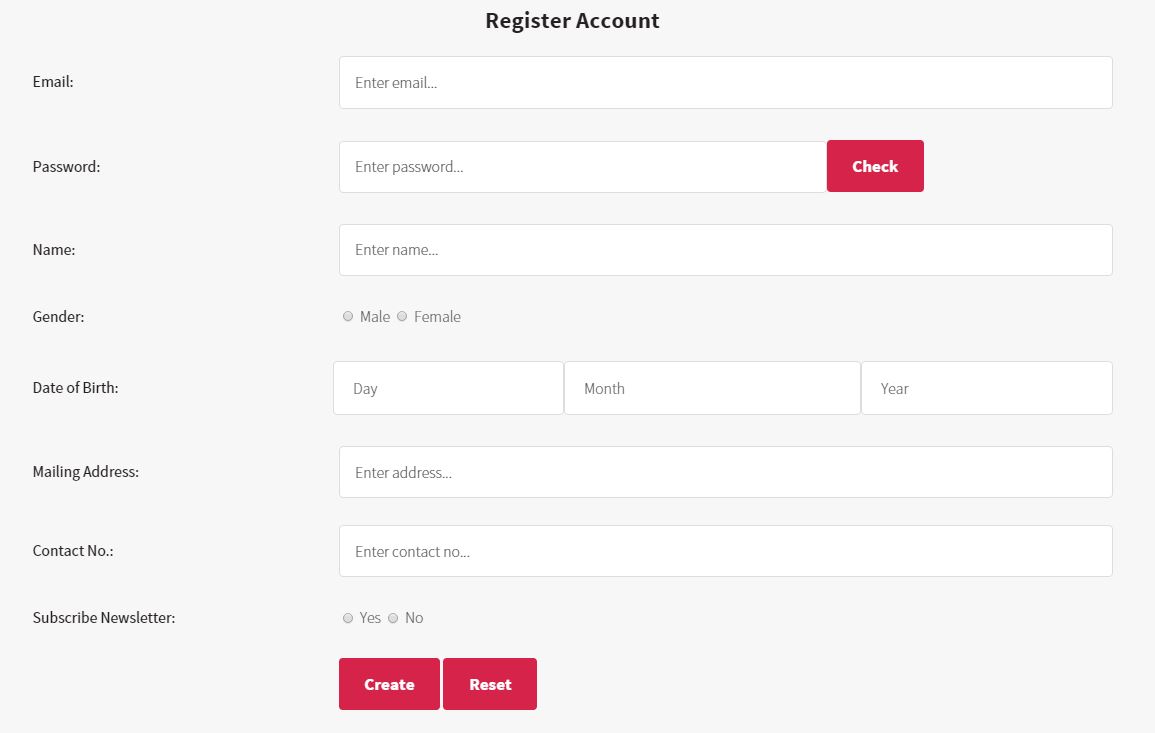
Member B

-----------------

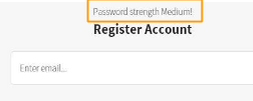
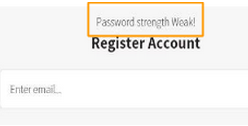
**In this section** - the login features of the website will be covered, as well as the viewing of the sales report. How members can create their accounts and how they can navigate through the website will also be touched on.

**Register Member (createMember.jsp)**

The form that a potential member has to fill:



Ability to check for weak, medium or strong passwords by clicking the check button beside the password field:







|  |
| --- |
| RegisterController |



**Login Role (login.jsp)**



|  |
| --- |
| LoginController |





The name that the member entered earlier appears on the member page as well.

**Update Member Profile (updateProfile.jsp)**

Member Sly Poon intends to change his name:





|  |
| --- |
| updateProfileController |



Member Page no longer welcomes “Sly Poon”, but welcomes “Handsome Boy 92” instead:



**View Sales Report (salesreport.jsp)**

Login into admin to see the sales report





|  |
| --- |
| **salesreport.jsp** |



Admin can select their choice for monthly report or yearly report

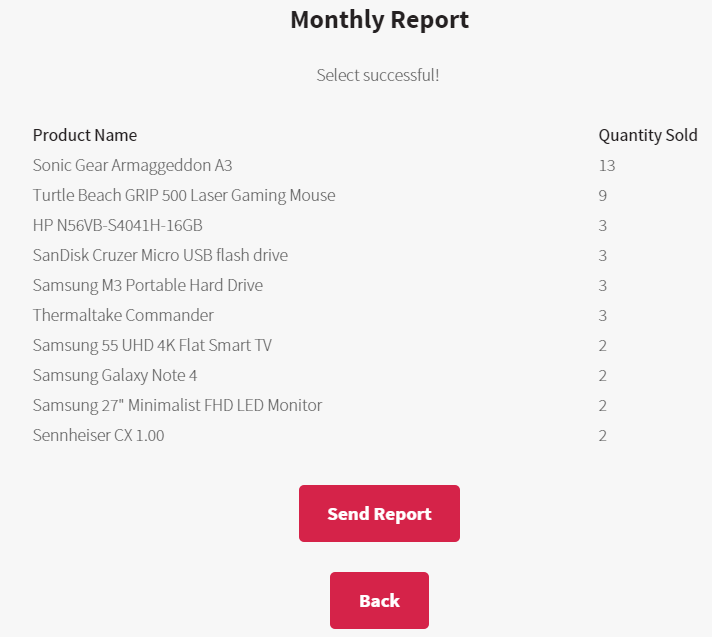




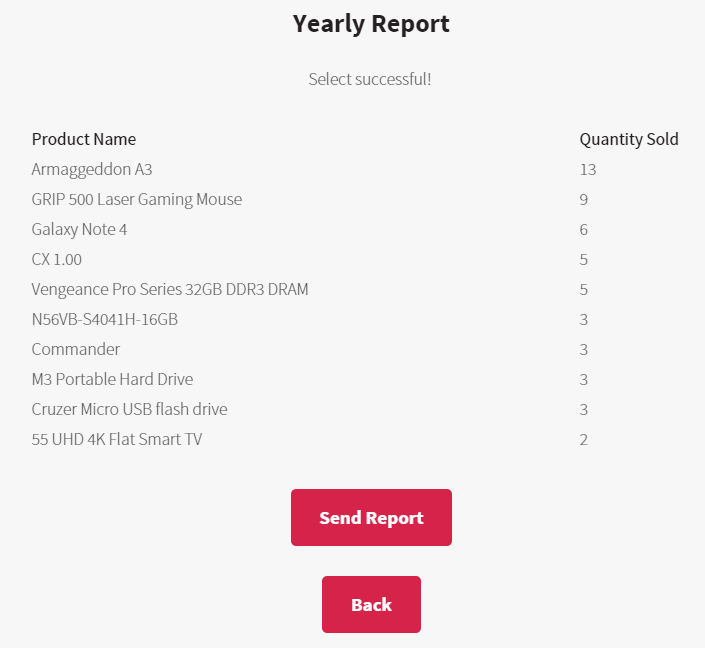
|  |
| --- |
| **salesreportController** |



|  |
| --- |
| **viewsalesreport.jsp** |



**or**



If admin would like to send report than it will call the sendreport Controller, or admin would like to go back will redirect admin back to salesreport.jsp.

**Log Out (logoutController)**



|  |
| --- |
| logoutController |



Redirect to login.jsp will display the “Logout Successful” message.

Advance features

-----------------

**In this section** - the various additional features to Enhance the member experience in the webstore and the security of the password will be covered.

**Generate Discount Code (coupons.jsp)**



|  |
| --- |
| VoucherController |



|  |
| --- |
| A different code will be generated. |

In the scenario where the customer tries to generate the code a second time in one day:



|  |
| --- |
| The voucher code remains the same. |

**Encrypt and Decrypt Password (mixSmash.java)**

Encrypting of password starts when a member signup an account in with the web store. The plain text password will be encrypted using the mixSmash.java encrypt method in the security package. The key and IV bytes is stored in the base.java.

Instead of storing the encrypted password in password attribute at the member table in the database. It stores the encrypted password in the crypt table and convert the encrypted password into a string, and store it with the member details which doesn’t match the actual password the member input.

During member input their credentials in the login.jsp, their email will be matched in the crypt table. If matched it will return the encrypted password, and pass the encrypted password to the decrypt method in mixSmash.java, converting it back into plaintext and match with the member input password.

Same concept for the forget password, when member enters their email address the password will be decrypted into plaintext from the crypt table in the database, and send it to the member email address as stated in the database.

**Forget Password (forgetpass.jsp)**



|  |
| --- |
| forgetpass.jsp |



Member is required to enter his email before the password can be sent to him. 

|  |
| --- |
| forgetPassController |

|  |
| --- |
| forgetPassController |





When the member checks their email:



**Send Email (SendMail.java)**

Email authentication session and the sending of message methods will be located in mail package called SendMail.java.

Similar to setupDBconn, SendMail.java has a method called requestSession that presents the email address and password of our webstore email to the SMTP server to request a session to send an email. For forgetting of passwords, when the member enters their email address, it will call the method sendforgetPass in the SendMail.java after the password has been decrypted. The content in their email will be the member decrypted passwords, so that they will be able to log into their account.

When a member first signup with the webstore a welcome message will be sent to them, for admin when they would like to see the sales report and get a copy of the report, it can be also sent to the admin email after clicking on send report button located at viewsalesreport.jsp.