

Iteration 2: Design and implementation of supermarket software

Student:

Dev Sutreja

Course:

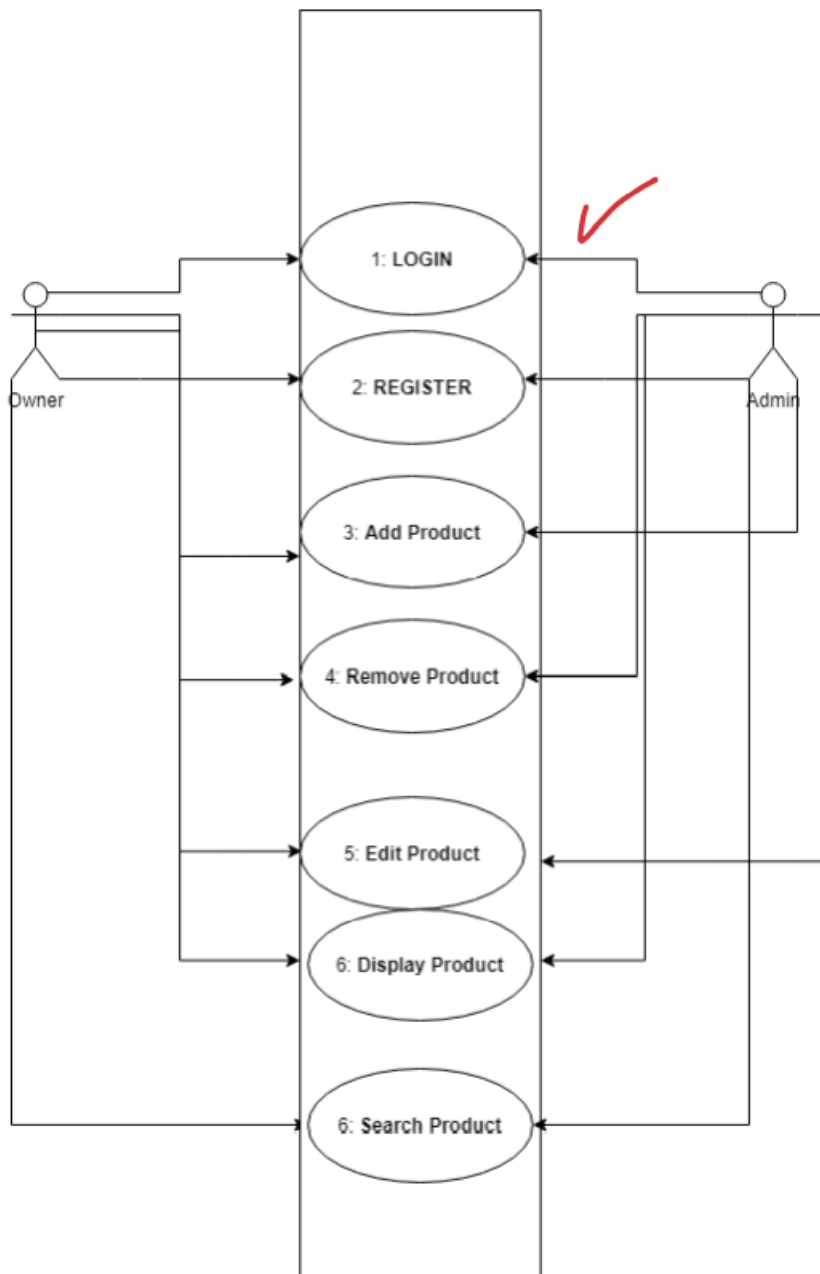
Object-oriented programming

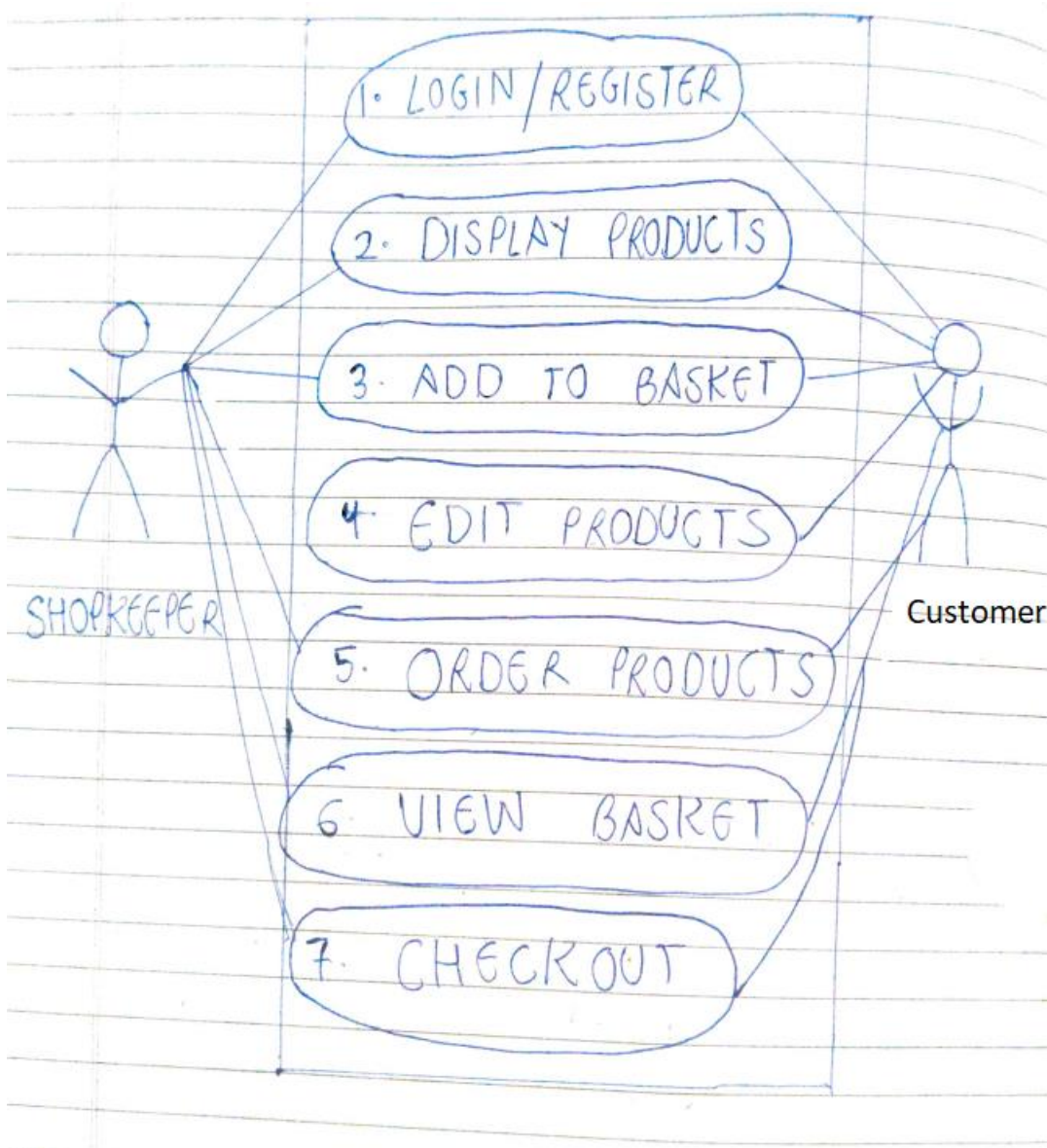
INTRODUCTION/WORKING

The following report is about the GUI I made which involves two software, Banking and Supermarket software. The user logs in/registers, selects products and adds them to the basket and finally decided what products to checkout with. After clicking on checkout, the user is directed to the Login page of the Banking software.

In order to display products, I initially decided to use text area, but I as solved the problem, I realized that I wont be able to select particular products therefore I was left with JList and JTable. I therefore used JTable to display products. The number of lines in my code increased significantly because I wasn't able to use "Abstract Table Model" for the JTable due to some errors in netbeans. I was then forced to use the Default Table model, and was then able to implement the problems with it.

Overall Use-Case diagram

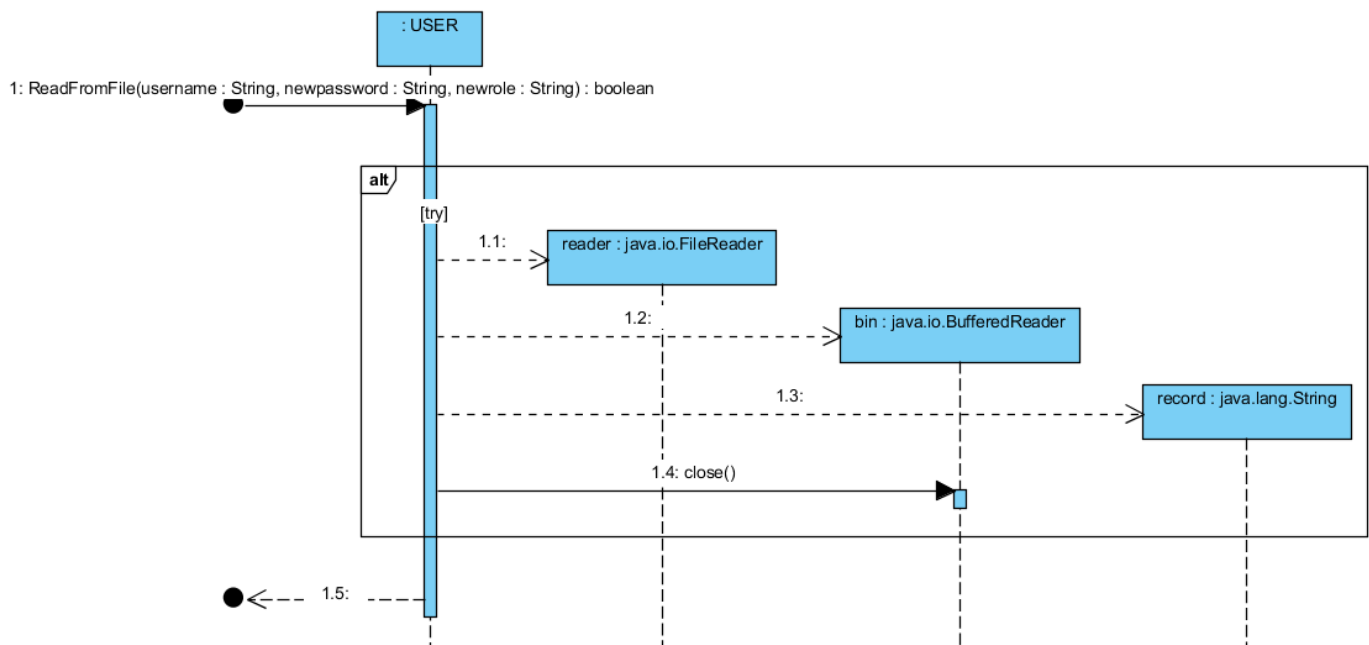


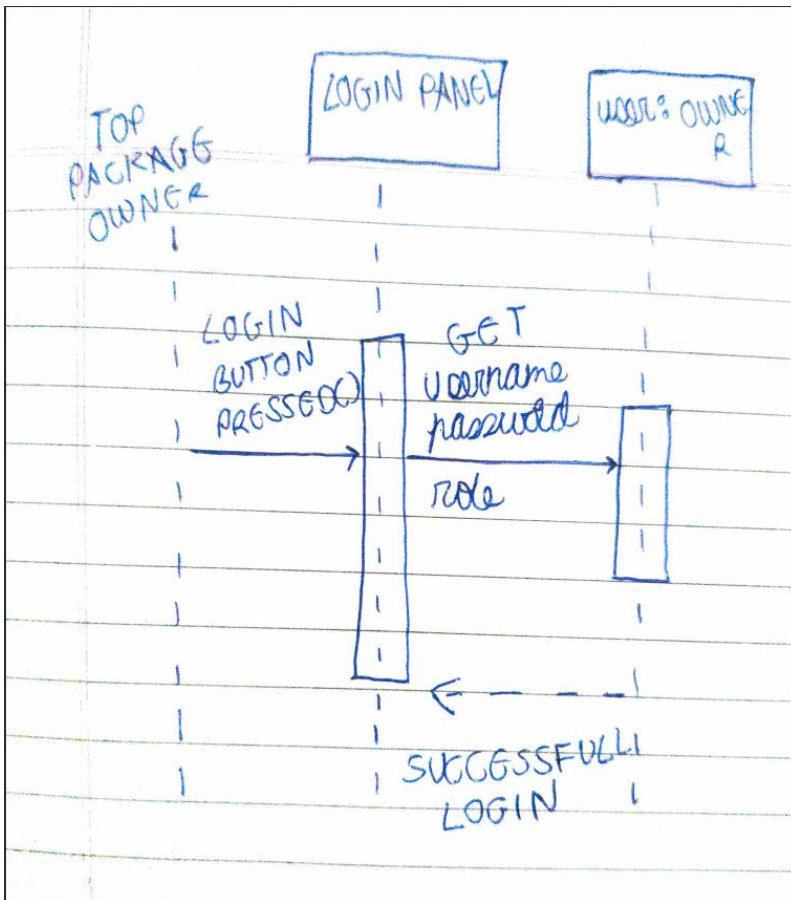


Overall Class diagram(zoom in)

Sequence diagram:

a) For Login

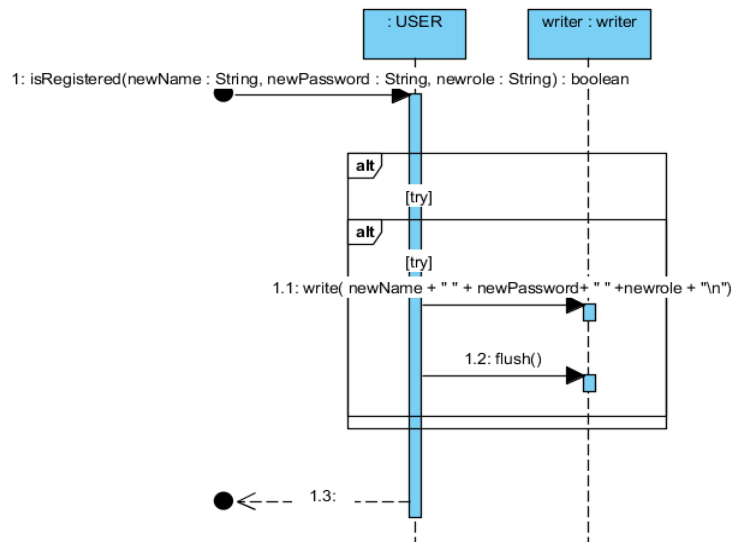




| | |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------|
| USE CASE ID | 1a |
| NAME | Log in |
| ACTORS | Shopkeeper, Customer |
| Goal | Allow the shopkeeper to login if Username and Password are correctly typed. |
| Preconditions | Shopkeeper has to register before logging in |
| Postcondition | Shopkeeper is logged in Correctly |
| Scenario | User inputs username, password and role. If all the inputs match to the inputs. Other tabs are then added only if logged in successfully. |

Verbs: Log in, register

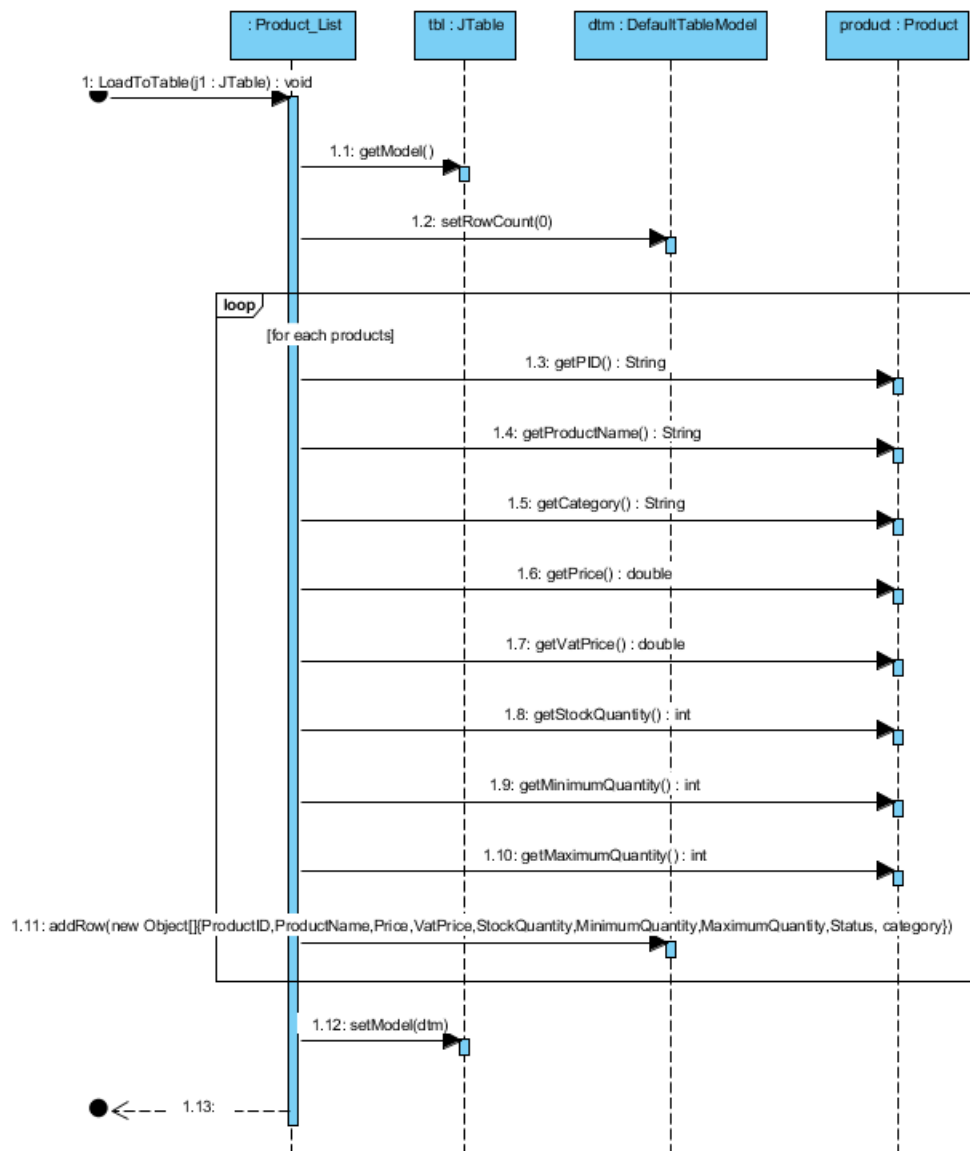
Nouns: Shopkeeper, Customer, Username, Password

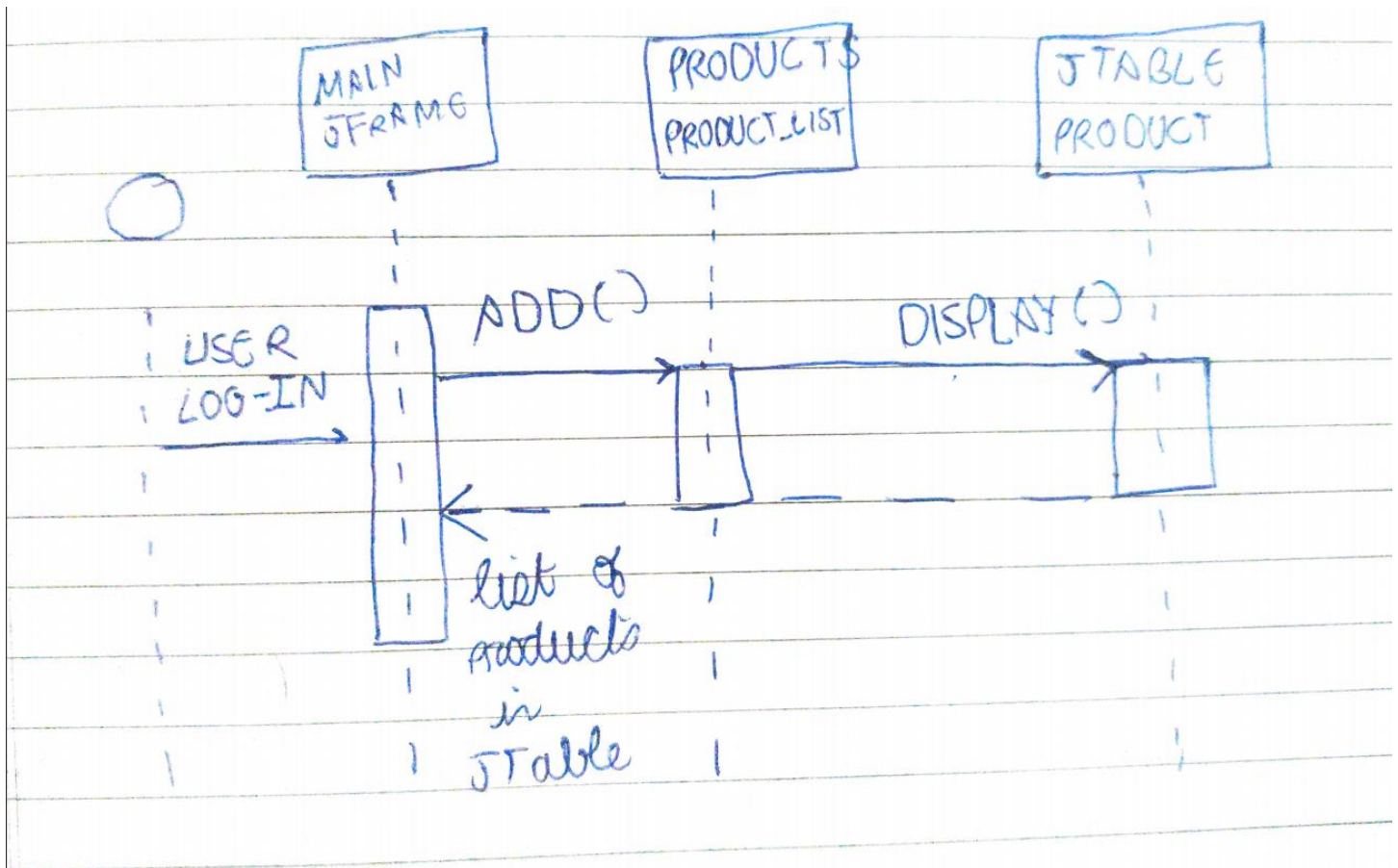


| | |
|---------------|----------------------------------------------------------------------------------------------------------------------|
| USE CASE ID | 1b |
| NAME | Register |
| ACTORS | Shopkeeper, Customer |
| Goal | Allow the shopkeeper to Register by Username and Password and role typed correctly. |
| Preconditions | None |
| Postcondition | Shopkeeper is Registered Correctly |
| Scenario | User inputs username, password and role. These details are then saved in the file, allowing him to log in next time. |

b) For Displaying Products:

| | |
|---------------|-------------------------------------------------|
| USE CASE ID | 2 |
| NAME | Display Products in JTable |
| ACTORS | Shopkeeper, Customer |
| Goal | User sees the list of all products in a JTable |
| Preconditions | Information should be added before in Edits tab |
| Postcondition | User sees the list of all products in a JTable |
| Scenario | List of products are displayed |



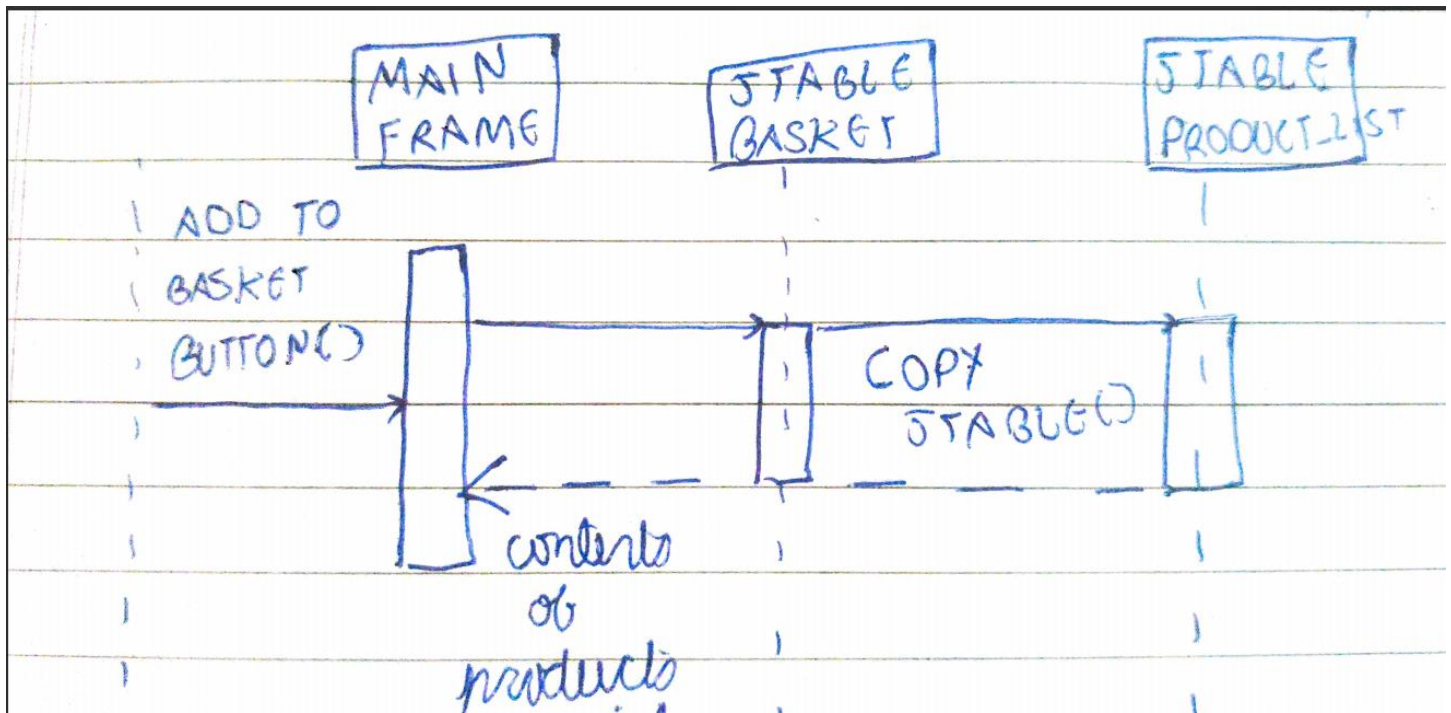
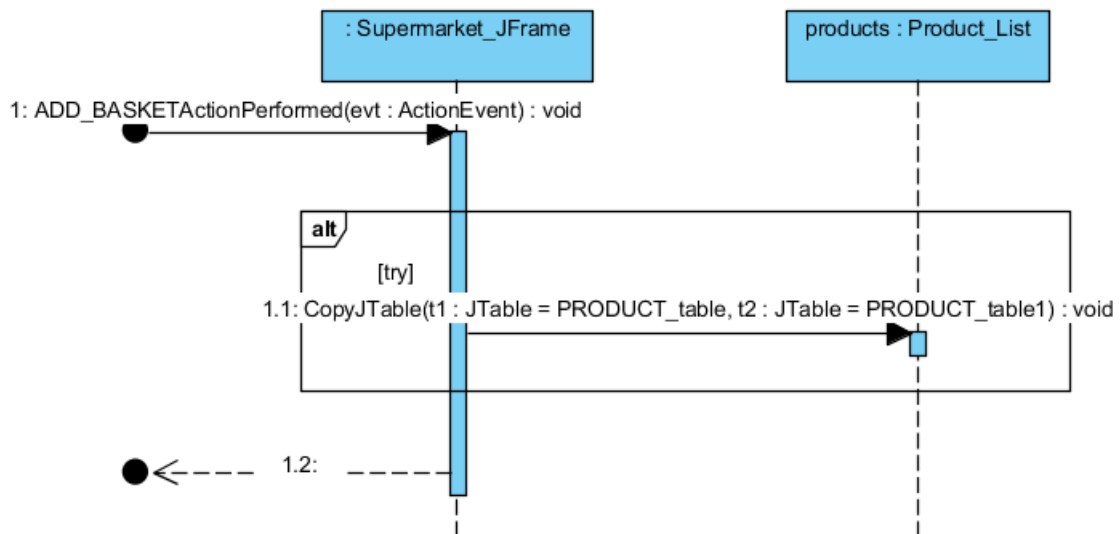


c) Add products to Basket:

| | |
|---------------|------------------------------------------------------------------------------------------------------------|
| USE CASE ID | 3 |
| NAME | Add selected Products to Basket |
| ACTORS | Shopkeeper, Customer |
| Goal | Add products to the Basket tab |
| Preconditions | Products should be selected from the table |
| Postcondition | User sees the selected products in the Basket tab |
| Scenario | User selects products from one table, which are then transferred to another table which is for the Basket. |

Verbs: Add, select, transfer

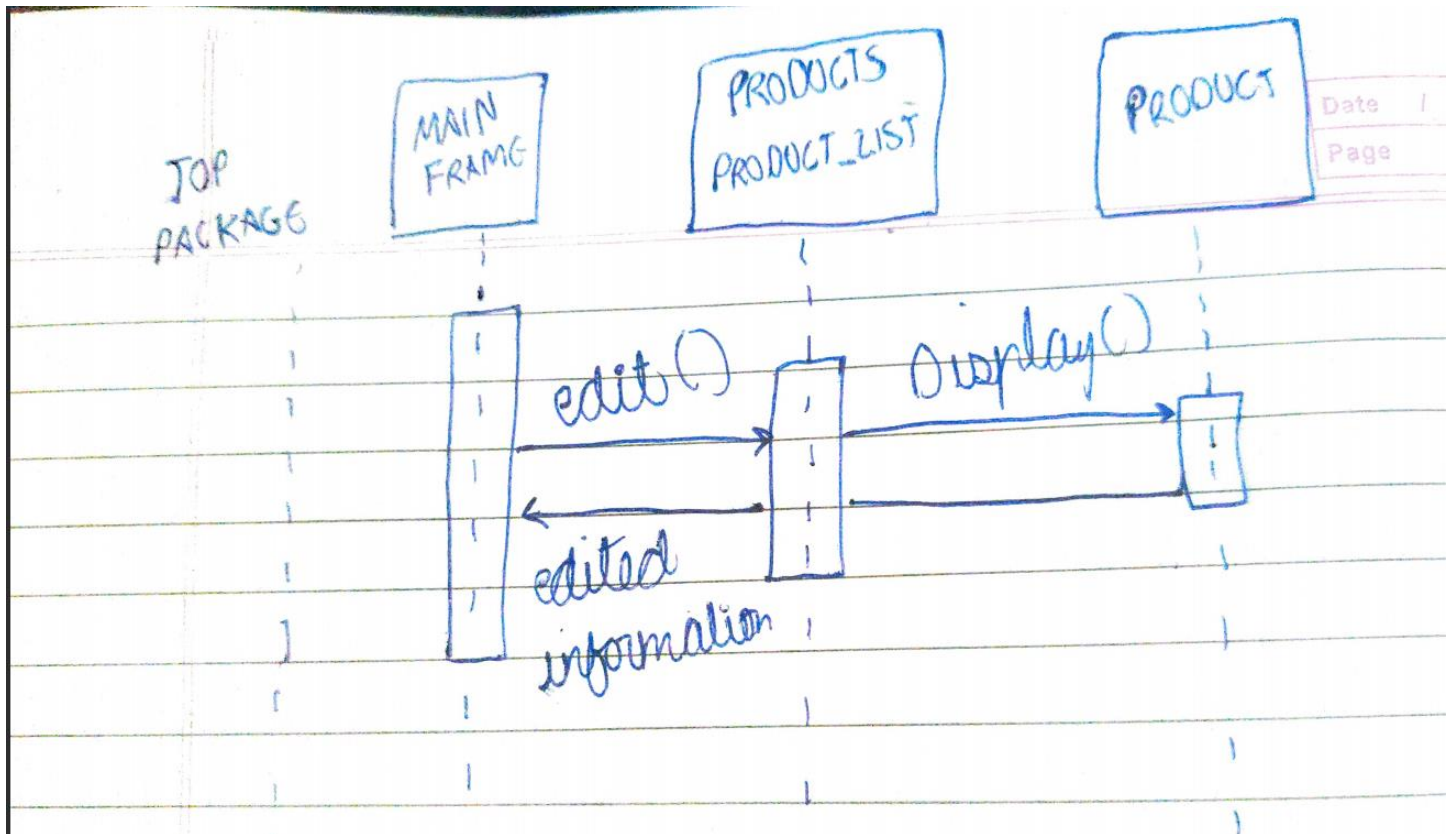
Nouns: Shopkeeper, Customer, Products, Jtable



d) To Edit the products:

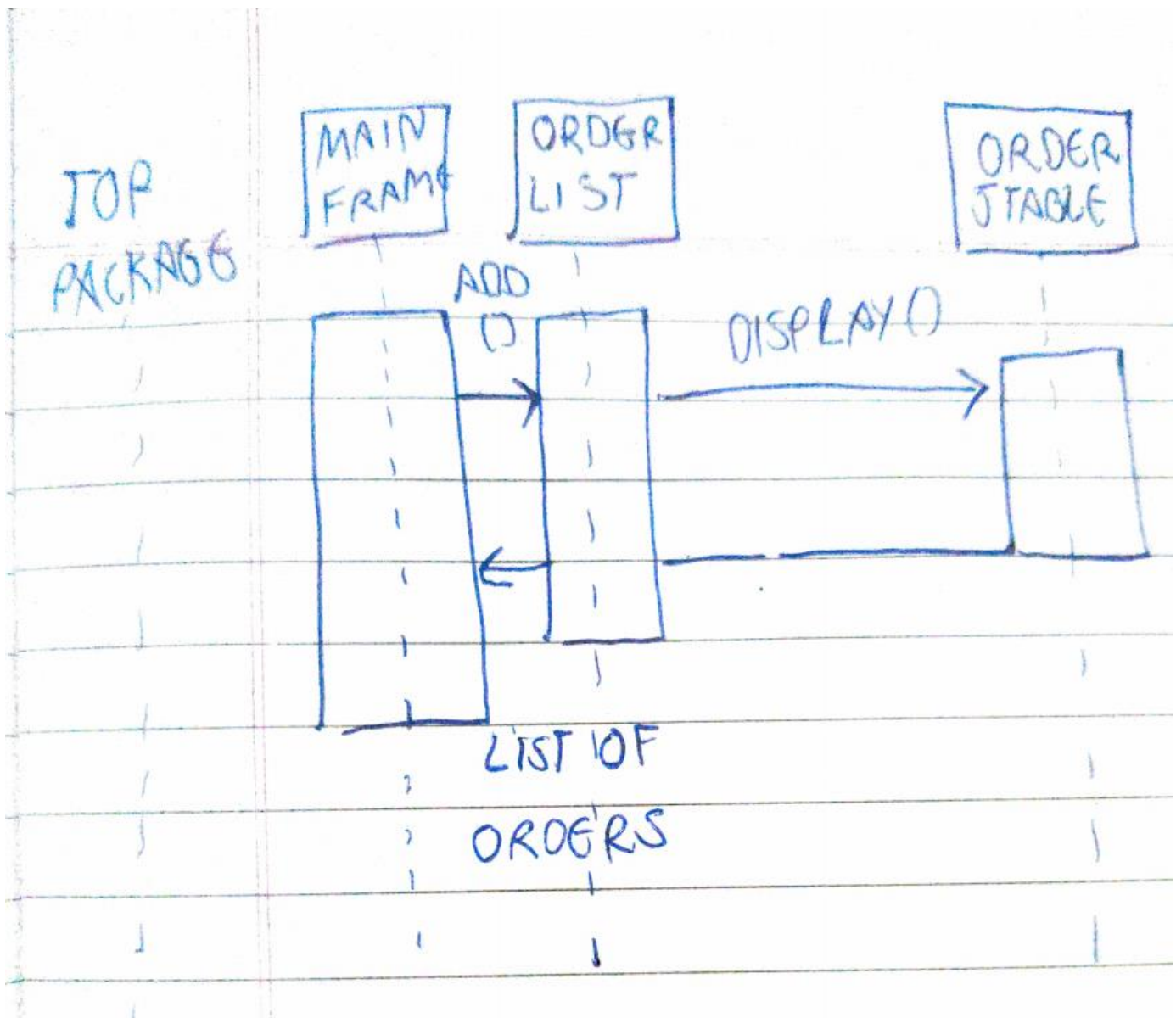
| | |
|-------------|------------------------------------|
| USE CASE ID | 4 |
| NAME | Edit information about the product |

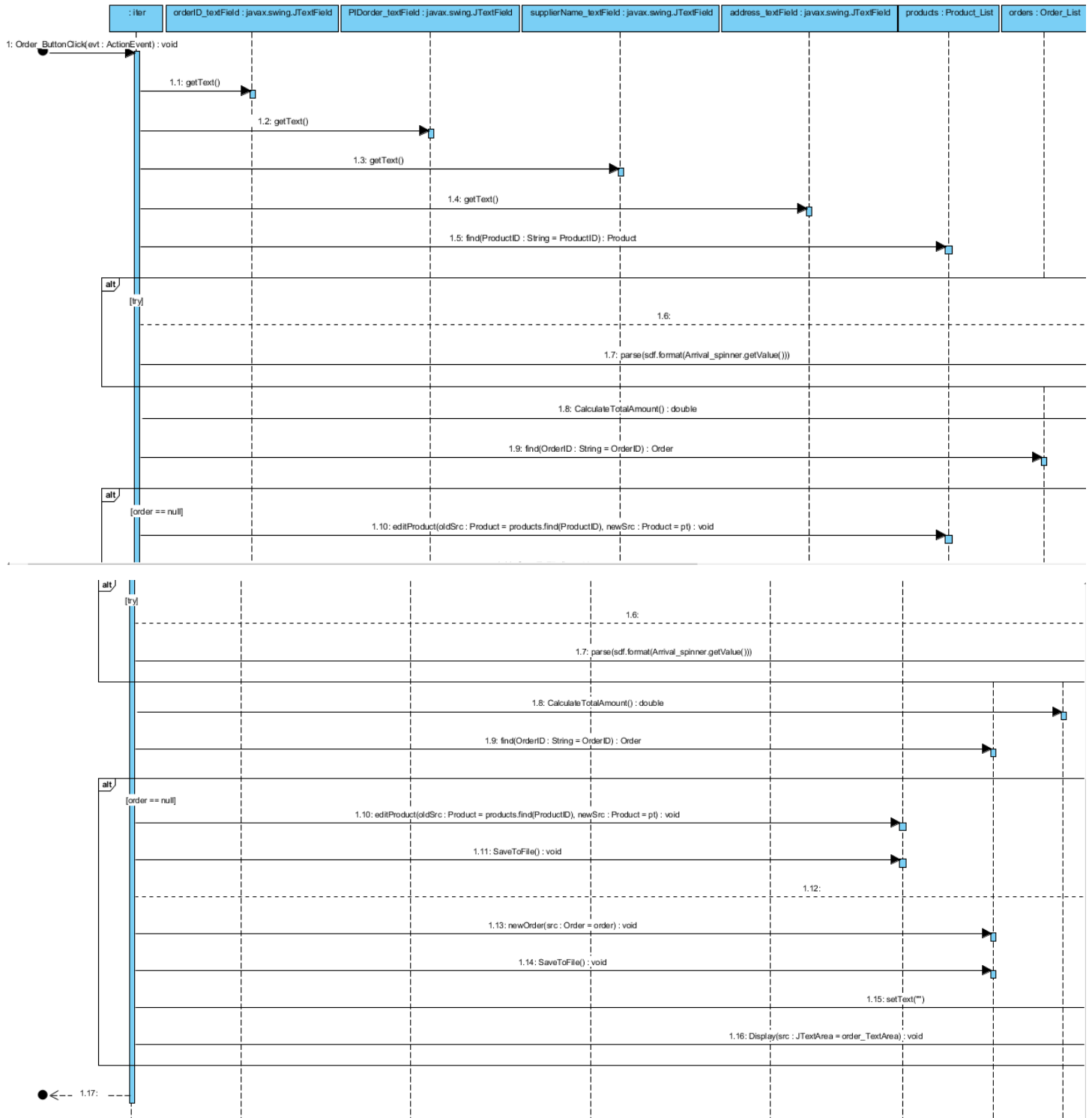
| | |
|---------------|-------------------------------------------------------------------|
| ACTORS | Shopkeeper, Customer |
| Goal | Successfully edit product information |
| Preconditions | Products should be added before editing |
| Postcondition | User is able to edit the product information |
| Scenario | User edits information from the Edit tab and saves into the file. |



f) To Order the products:

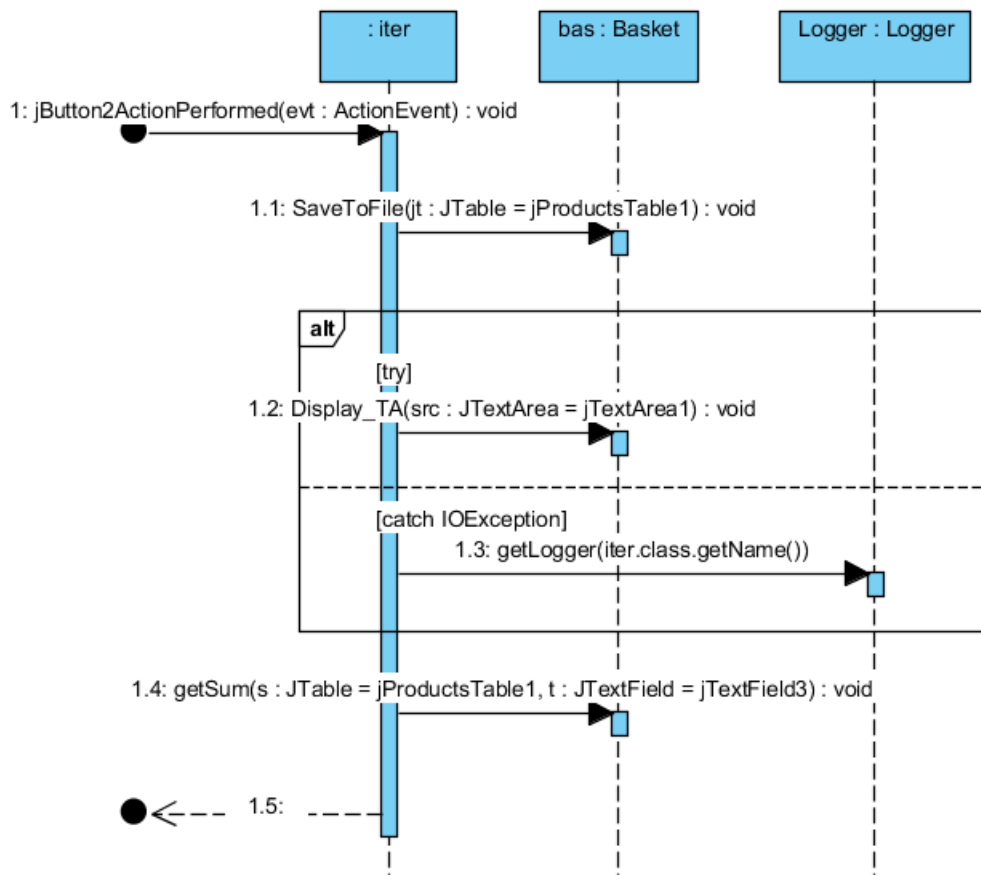
| | |
|---------------|-------------------------------------------------------------------|
| USE CASE ID | 5 |
| NAME | To order products |
| ACTORS | Shopkeeper, Customer |
| Goal | Successfully edit product information |
| Preconditions | Products should be added before editing |
| Postcondition | User is able to edit the product information |
| Scenario | User edits information from the Edit tab and saves into the file. |





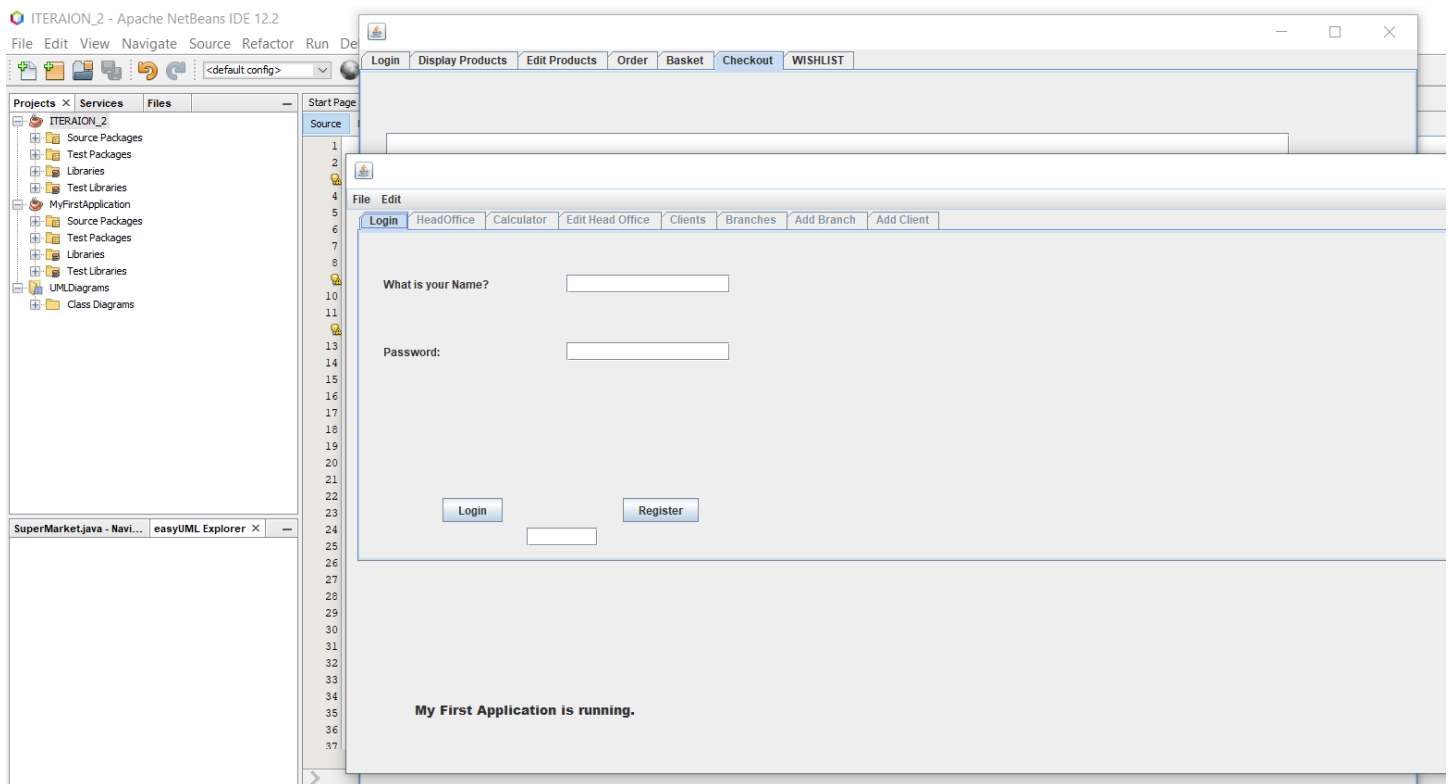
f) To View the products Basket:

| | |
|---------------|--------------------------------------------------------------|
| USE CASE ID | 6 |
| NAME | To view products Basket |
| ACTORS | Shopkeeper, Customer |
| Goal | To view the Basket of selected products |
| Preconditions | Products should be selected before |
| Postcondition | User is able to view the basket containing selected products |
| Scenario | User can add/delete the products in Basket |



g) To Confirm the products Basket:

| | |
|---------------|-----------------------------------------------------------------|
| USE CASE ID | 7 |
| NAME | Checkout |
| ACTORS | Shopkeeper, Customer |
| Goal | To confirm the Basket of selected products |
| Preconditions | Products should be selected before |
| Postcondition | User is able to confirm the basket containing selected products |
| Scenario | User can confirm the products in Basket |



TESTING:

➔ USE-CASE 1: Login and Register

a) Registering the user first:

The screenshot shows a web application window titled "Login". The interface has a light gray background. On the left side, there are three labels: "USERNAME:", "PASSWORD:", and "ROLE:". To the right of "USERNAME:" is a text input field containing the text "Dev". To the right of "PASSWORD:" is a password input field with three black dots. To the right of "ROLE:" is a dropdown menu with "Admin" selected. At the bottom left, there is a message: "The Dev is registered Please log in now". At the bottom right, there are three buttons: "LOGIN", "REGISTER", and "CLOSE". The "REGISTER" button is highlighted with a blue border.

Username: Dev

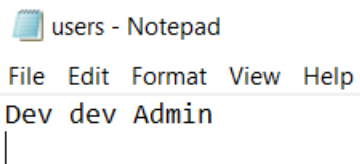
Password: [Masked]

Role: Admin

The Dev is registered Please log in now

Buttons: LOGIN, REGISTER, CLOSE

Now, checking the file for registered users:

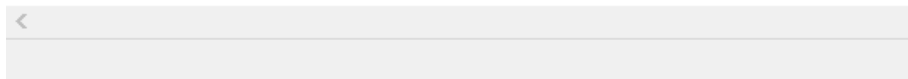


users - Notepad

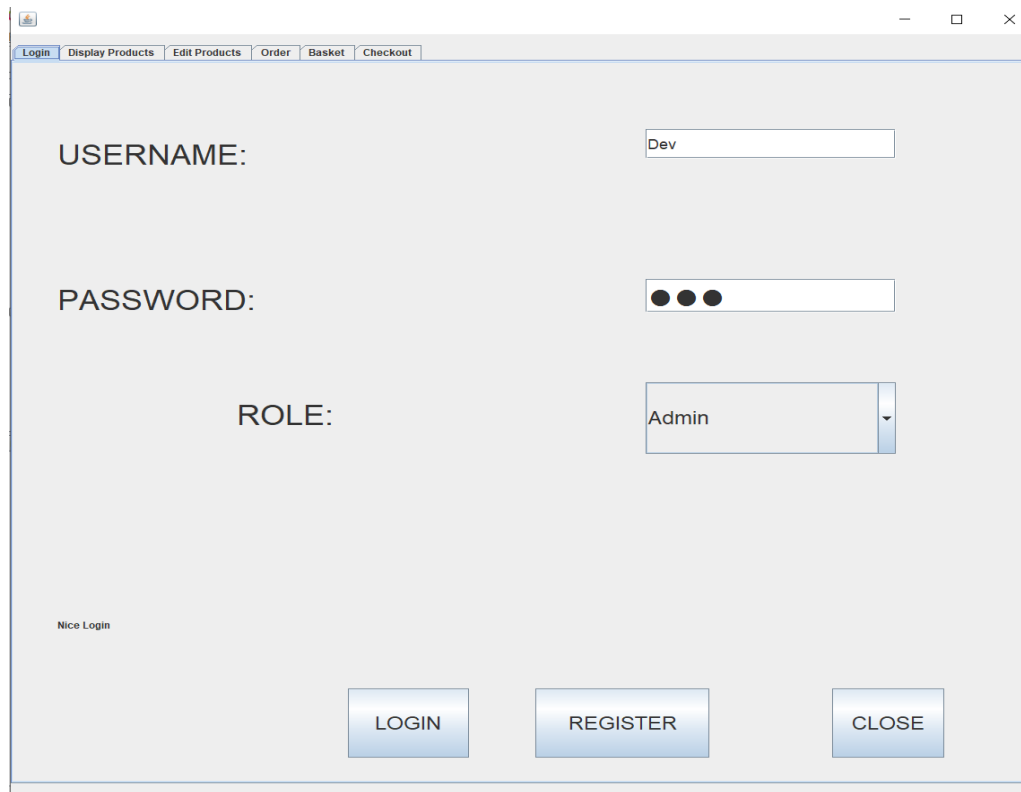
File Edit Format View Help

Dev dev Admin

|



Now, logging the user with correct and incorrect passwords.



The screenshot shows a web application window with a title bar containing a maximize button, a close button, and a window icon. The window has a menu bar with the following items: Login, Display Products, Edit Products, Order, Basket, and Checkout. The main content area is light gray and contains the following elements:

- USERNAME:** A text input field containing the value "Dev".
- PASSWORD:** A text input field with three black dots representing a masked password.
- ROLE:** A dropdown menu with "Admin" selected.
- Nice Login** text at the bottom left.
- Three buttons at the bottom: **LOGIN**, **REGISTER**, and **CLOSE**.

Username: Dev

Password: [Masked]

Role: Admin

Cant Login

LOGIN REGISTER CLOSE

Therefore,

USE CASE 1: Login and Register

The expected outcome and output obtain match, therefore the use case works!

➔ USE CASE 2: Display Products:

| ID | Name | Price | VAT Price | Items in Stock | Minimum allowed | Maximum allowed | Stock Status | Category |
|-----|-----------------|--------|-----------|----------------|-----------------|-----------------|--------------|-------------|
| 1 | Fruity | 2.0 | 2.0 | 50 | 10 | 30 | Enough Stock | Eatery |
| 2 | Doritos | 10.0 | 5.0 | 50 | 30 | 100 | Enough Stock | Eatery |
| 3 | Camera | 50.0 | 600.0 | 500 | 20 | 900 | Enough Stock | Electronics |
| 4 | Pen | 1.0 | 10.0 | 60 | 100 | 1000 | Re-Order | Stationary |
| 5 | Allenware | 900.0 | 50.0 | 90 | 90 | 900 | Re-Order | Electronics |
| 6 | Sweater | 700.0 | 1505.0 | 500 | 70 | 700 | Enough Stock | Clothes |
| 7 | Bible | 800.0 | 1720.0 | 30 | 40 | 400 | Re-Order | Books |
| 8 | Sleeping Beauty | 900.0 | 1935.0 | 100 | 20 | 200 | Enough Stock | Books |
| 9 | Jeans | 1000.0 | 2150.0 | 8 | 10 | 100 | Re-Order | Clothes |
| 10 | Biscuits | 1100.0 | 2365.0 | 200 | 15 | 150 | Enough Stock | Eatery |
| 11 | Puff | 2.0 | 4.3 | 100 | 30 | 200 | Enough Stock | Drink |
| 100 | Laptop | 35.0 | 75.25 | 50 | 10 | 100 | Enough Stock | Electronics |

We can also filter the categories, and the table shows the filtered values:

Login

Display Products

Edit Products

Order

Basket

Checkout

| ID | Name | Price | VAT Price | Items in Stock | Minimum allowed | Maximum allowed | Stock Status | Category |
|-----|-----------|-------|-----------|----------------|-----------------|-----------------|--------------|-------------|
| 3 | Camera | 50.0 | 600.0 | 500 | 20 | 900 | Enough Stock | Electronics |
| 5 | Allenware | 900.0 | 50.0 | 90 | 90 | 900 | Re-Order | Electronics |
| 100 | Laptop | 35.0 | 75.25 | 50 | 10 | 100 | Enough Stock | Electronics |

Category:

Electronics

Add to Basket

MORE INFORMATION

After this, we select the products that we want into the basket. This is done by clicking at the products all at once (press control) or by selecting them individually and clicking the Add to basket button.

We can check the files which have these products saved in them:

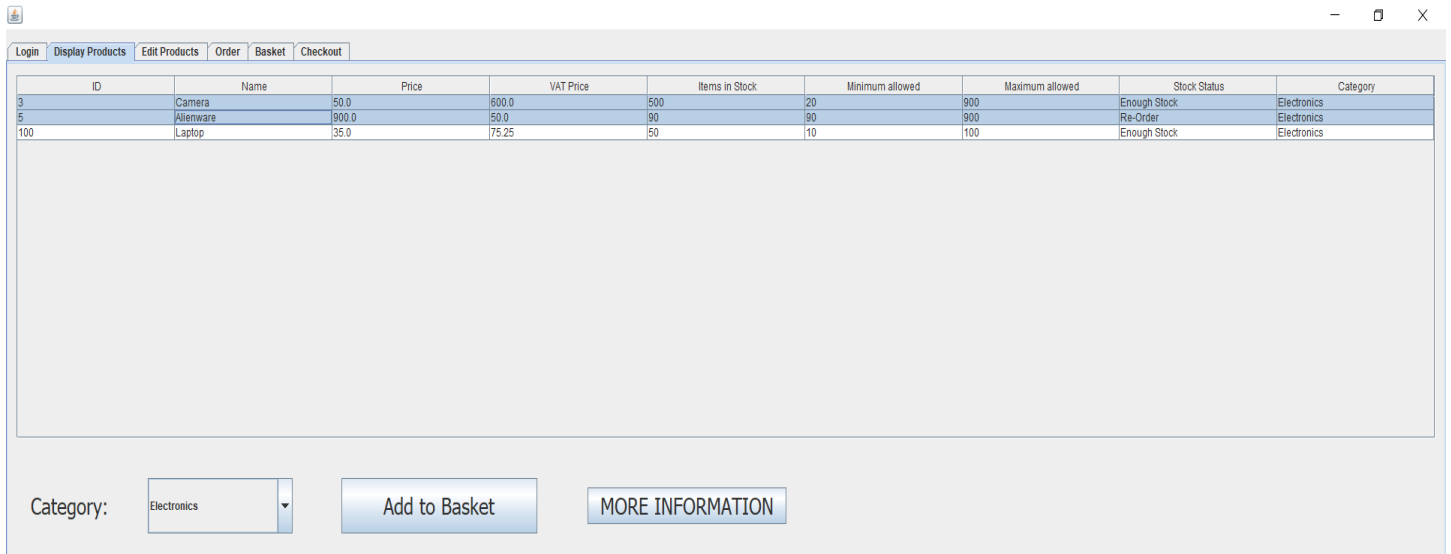
Product_List - Notepad

File Edit Format View Help

Product ID=1
Product Name=Fruity
Weight=50g
Price=2.0
VAT Price=2.0
Expiry Date=01/01/2022
Estimated Order Date=01/01/2021
Images=
Category=Eatery
Stock Quantity=50
Minimum Quantity=10
Maximum Quantity=30
Last Order Date=01/01/2021
Manufactured By= agro
Manufactured Date=12/12/2020
X-X-X-X-X-X-X-X-X-X-X-X-X-X-X
Product ID=2
Product Name=Doritos
Weight=5g
Price=10.0
VAT Price=5.0
Expiry Date=01/02/2020
Estimated Order Date=02/02/2021
Images=
Category=Eatery
Stock Quantity=50
Minimum Quantity=30
Maximum Quantity=100
Last Order Date=02/02/2021
Manufactured By=Dori
Manufactured Date=02/02/2020
X-X-X-X-X-X-X-X-X-X-X-X-X-X-X
Product ID=3
Product Name=Camera
Weight=2
Price=50.0
VAT Price=600.0
Expiry Date=02/1/2021
Estimated Order Date=2/2/2021
Images=
Category=Electronics
Stock Quantity=500

We see that, all the products are filtered according to the Categories selected and all the products on the file are seen in the JTable. Therefore, the use case 2, display products works since the expected outcome and the actual outcome match.

Use Case 3: Add to Basket



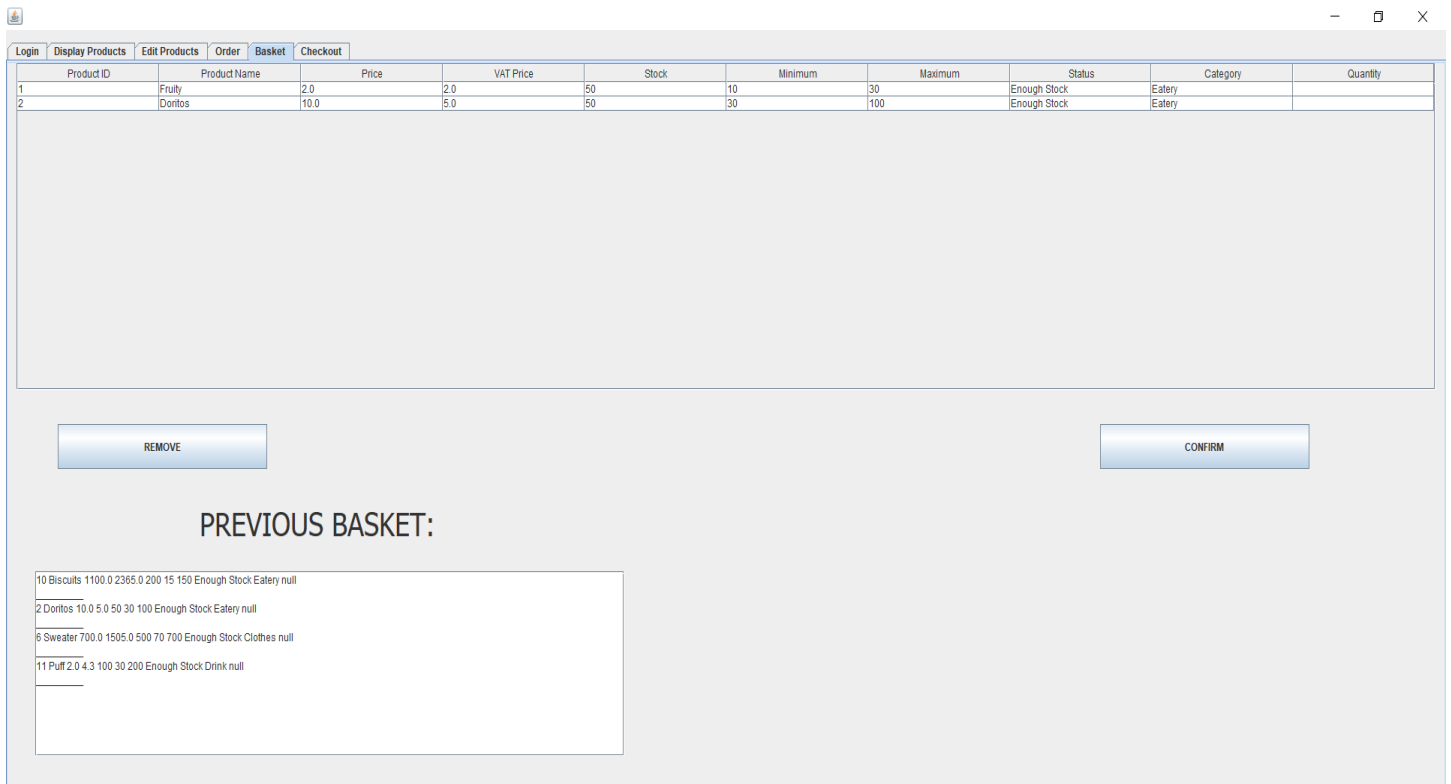
Login Display Products Edit Products Order **Basket** Checkout

| ID | Name | Price | VAT Price | Items in Stock | Minimum allowed | Maximum allowed | Stock Status | Category |
|-----|-----------|-------|-----------|----------------|-----------------|-----------------|--------------|-------------|
| 3 | Camera | 50.0 | 600.0 | 500 | 20 | 900 | Enough Stock | Electronics |
| 5 | Alienware | 900.0 | 50.0 | 90 | 90 | 900 | Re-Order | Electronics |
| 100 | Laptop | 35.0 | 75.25 | 50 | 10 | 100 | Enough Stock | Electronics |

Category: Electronics Add to Basket MORE INFORMATION

We select the following products and click the Add to basket button:

The expected outcome is to see the selected products in the basket.



Login Display Products Edit Products Order **Basket** Checkout

| Product ID | Product Name | Price | VAT Price | Stock | Minimum | Maximum | Status | Category | Quantity |
|------------|--------------|-------|-----------|-------|---------|---------|--------------|----------|----------|
| 1 | Fruity | 2.0 | 2.0 | 50 | 10 | 30 | Enough Stock | Eatery | |
| 2 | Doritos | 10.0 | 5.0 | 50 | 30 | 100 | Enough Stock | Eatery | |

REMOVE CONFIRM

PREVIOUS BASKET:

10 Biscuits 1100.0 2365.0 200 15 150 Enough Stock Eatery null

2 Doritos 10.0 5.0 50 30 100 Enough Stock Eatery null

6 Sweater 700.0 1505.0 500 70 700 Enough Stock Clothes null


11 Puff 2.0 4.3 100 30 200 Enough Stock Drink null

We can see that all the products that we added, can be seen in the Basket.

We can either remove the product from wish list or we can proceed to the checkout.

Use Case 4: Edit Products

In order to edit the products, we first have to search for the products via their Product ID.

— □ ×

LoginDisplay ProductsEdit ProductsOrderBasketCheckoutWISHLIST

ID:1

Name:Fruity

Price:2.0

Weight:50g

Category:Eatery

Expiry Date:01/01/2022

Estimated Order Date:01/01/2021

Stock Quantity:50

Minimum Allowed:10

Maximum Quantity:30

Manufactured By:agro

Manufactured Date:17/02/2021

LAST DATE:12/12/2020

EditSearchAddAttach ImagesjButton1

Product ID=1
Product Name=Fruity
Weight=50g
Price=2.0
VAT Price=2.0
Exp Date=01/01/2022
Estimated Order Date=01/01/2021
Images=
Category=Eatery
Stock Quantity=50
Minimum Quantity=10
Maximum Quantity=30
Last Order Date=01/01/2021
Manufactured By= agro
Manufactured Date=12/12/2020

Now, In order to edit the details, we change them in the text field and then press the edit button:

| Login | Display Products | Edit Products | Order | Basket | Checkout | WISHLIST |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------|--------------------|------------------------------------------|--------|----------|----------|
| ID: | <input type="text" value="1"/> | Name: | <input type="text" value="Fruity EDIT"/> | | | |
| Price: | <input type="text" value="2.002020"/> | Weight: | <input type="text" value="50g"/> | | | |
| Category: | <input type="text" value="Eatery"/> | Expiry Date: | <input type="text" value="01/01/2022"/> | | | |
| Estimated Order Date: | <input type="text" value="01/01/2021"/> | Stock Quantity: | <input type="text" value="50"/> | | | |
| Minimum Allowed: | <input type="text" value="100"/> | Maximum Quantity: | <input type="text" value="30"/> | | | |
| Manufactured By: | <input type="text" value="agro_EDITED"/> | Manufactured Date: | <input type="text" value="17/02/2021"/> | | | |
| | | Last date: | <input type="text" value="12/12/2020"/> | | | |
| <input type="button" value="Edit"/> <input type="button" value="Search"/> <input type="button" value="Add"/> <input type="button" value="Attach Images"/> <input type="button" value="jButton1"/> | | | | | | |
| <pre>Product ID=1 Product Name=Fruity Weight=50g Price=2.0 VAT Price=2.0 Exp Date=01/01/2022 Estimated Order Date=01/01/2021 Images= Category=Eatery Stock Quantity=50 Minimum Quantity=10 Maximum Quantity=30 Last Order Date=01/01/2021 Manufactured By= agro Manufactured Date=12/12/2020</pre> | | | | | | |

We can also add this new product information to a new Product like following:

— □

Login Display Products **Edit Products** Order Basket Checkout WISHLIST

ID:

Name:

Price:

Weight:

Category:

Expiry Date:

▲▼

Estimated Order Date:

▲▼

Stock Quantity:

Minimum Allowed:

Maximum Quantity:

Manufactured By:

Manufactured Date:

▲▼

LAST DATE

Edit

Search

Add

Attach Images

jButton1

Now, in order to see if the Add button works, we see the display products tab again.

— □ ×

Login **Display Products** Edit Products Order Basket Checkout WISHLIST

| ID | Name | Price | VAT Price | Items in Stock | Minimum allowed | Maximum allowed | Stock Status | Category |
|-----|-----------------|--------|-----------|----------------|-----------------|-----------------|--------------|-------------|
| 1 | Fruity | 2.0 | 4.3 | 50 | 10 | 30 | Enough Stock | Eatery |
| 2 | Doritos | 10.0 | 5.0 | 50 | 30 | 100 | Enough Stock | Eatery |
| 3 | Camera | 50.0 | 600.0 | 500 | 20 | 900 | Enough Stock | Electronics |
| 4 | Pen | 1.0 | 10.0 | 60 | 100 | 1000 | Re-Order | Stationary |
| 5 | Alienware | 900.0 | 50.0 | 90 | 90 | 900 | Re-Order | Electronics |
| 6 | Sweater | 700.0 | 1505.0 | 500 | 70 | 700 | Enough Stock | Clothes |
| 7 | Bible | 800.0 | 1720.0 | 30 | 40 | 400 | Re-Order | Books |
| 8 | Sleeping Beauty | 900.0 | 1935.0 | 100 | 20 | 200 | Enough Stock | Books |
| 9 | Jeans | 1000.0 | 2150.0 | 8 | 10 | 100 | Re-Order | Clothes |
| 10 | Biscuits | 1100.0 | 2365.0 | 200 | 15 | 150 | Enough Stock | Eatery |
| 11 | Puff | 2.0 | 4.3 | 100 | 30 | 200 | Enough Stock | Drink |
| 100 | Laptop | 35.0 | 75.25 | 50 | 10 | 100 | Enough Stock | Electronics |
| 25 | Fruity | 2.0 | 4.3 | 50 | 10 | 30 | Enough Stock | Eatery |

Category:


▼

Add to Basket

MORE INFORMATION

We can see that the Add, Edit and the Search button work correctly therefore the expected output is attained. However, the image attaching doesn't work therefore it is NOT implemented as expected.

We can check the changes made in the file.

 Product_List - Notepad

File Edit Format View Help

Category=Drink
Stock Quantity=100
Minimum Quantity=30
Maximum Quantity=200
Last Order Date=15/12/2020
Manufactured By=McDonalds
Manufactured Date=15/12/2020
X-X-X-X-X-X-X-X-X-X-X-X-X-X-X
Product ID=100
Product Name=Laptop
Weight=500g
Price=35.0
VAT Price=75.25
Exp Date=13/02/2021
Estimated Order Date=13/02/2021
Images=
Category=Electronics
Stock Quantity=50
Minimum Quantity=10
Maximum Quantity=100
Last Order Date=13/02/2021
Manufactured By=sony
Manufactured Date=13/02/2021
X-X-X-X-X-X-X-X-X-X-X-X-X-X-X
Product ID=25
Product Name=Fruity
Weight=50g
Price=2.0
VAT Price=4.3
Exp Date=01/01/2022
Estimated Order Date=01/01/2021
Images=
Category=Eatery
Stock Quantity=50
Minimum Quantity=10
Maximum Quantity=30
Last Order Date=17/02/2021
Manufactured By= agro
Manufactured Date=12/12/2020
X-X-X-X-X-X-X-X-X-X-X-X-X-X-X

USE CASE 5: Ordering Products

The same procedure applies for ordering products:

The screenshot shows a web application window with a menu bar at the top containing: Login, Display Products, Edit Products, Order (highlighted), Basket, Checkout, and WISHLIST. The main area contains several input fields and buttons:

- Order ID:** A text box containing the value '1'.
- Product ID:** A text box containing the value 'P2'.
- Supplier Address:** A text box containing the value 'AKOta'.
- Supplier Name:** A text box containing the value 'Dev'.
- Arrival Date:** A date picker showing '02/02/2020'.
- Search:** A blue button with the text 'Search'.
- Order:** A large blue button with the text 'Order'.

At the bottom of the window, there is a text area displaying the following information:

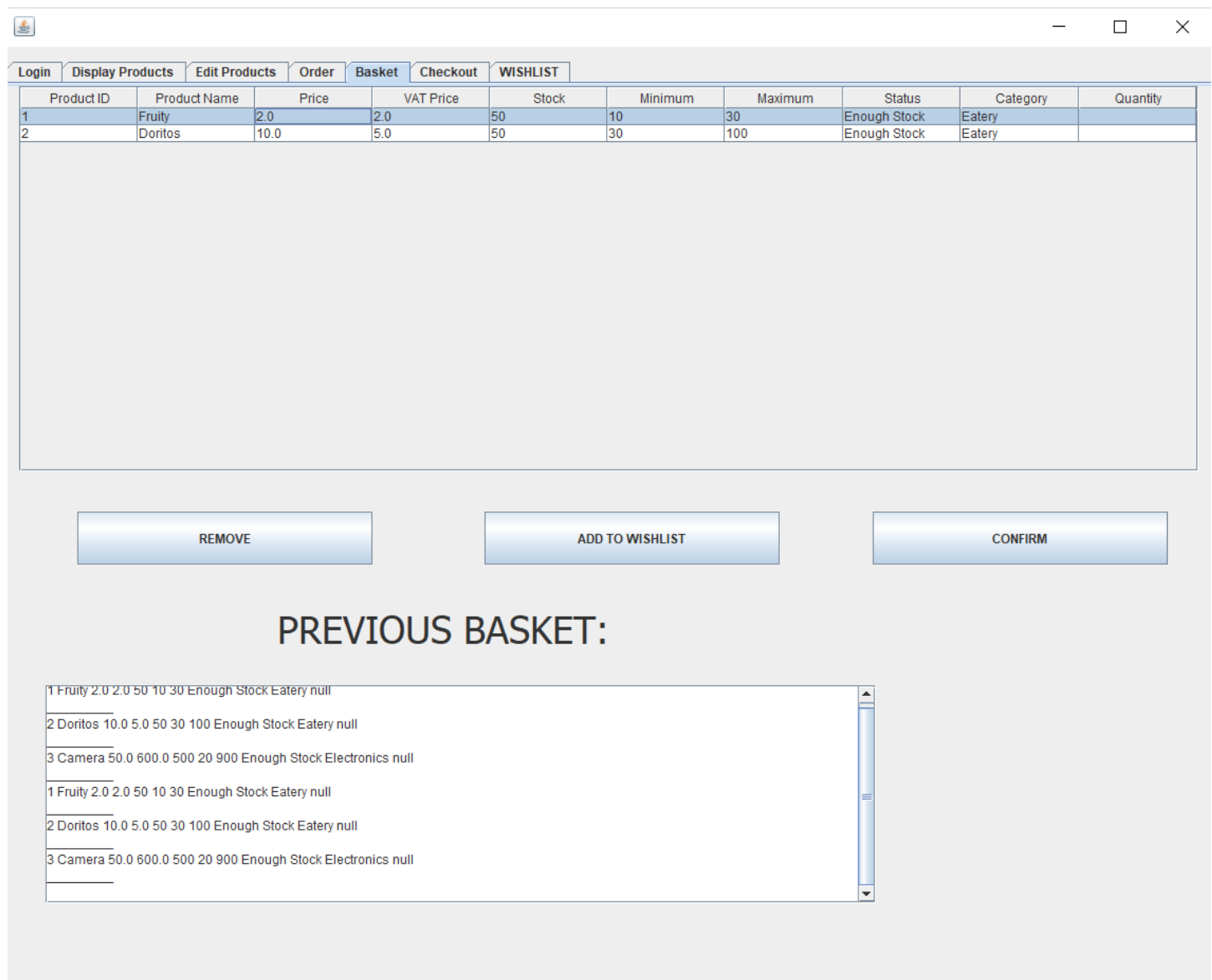
```
Order ID=1
Product ID=P2
Total Amount=92000.0
Supplier Name=Dev
Supplier Address=AKOta
Delivery Date=02/02/2020
```

This can be verified by looking at the file.

```
Re-order - Notepad
File Edit Format View Help
Order ID=1
Product ID=P2
Total Amount=92000.0
Supplier Name=Dev
Supplier Address=AKOta
Delivery Date=02/02/2020
x-x-x-x-x-x-x-x-x-x-x-x-x-x-x-x
Order ID=2
Product ID=P3
Total Amount=92000.0
Supplier Name=Dev
Supplier Address=Akota
Delivery Date=02/02/2020
x-x-x-x-x-x-x-x-x-x-x-x-x-x-x-x
```

USE CASE 5: View Basket

We added two products from the product display tab, and we can see them in the basket here:



The screenshot displays the 'Basket' tab of a web application. The interface includes a navigation bar with tabs: Login, Display Products, Edit Products, Order, **Basket**, Checkout, and WISHLIST. Below the navigation bar is a table with the following columns: Product ID, Product Name, Price, VAT Price, Stock, Minimum, Maximum, Status, Category, and Quantity. The table contains two rows of data:

| Product ID | Product Name | Price | VAT Price | Stock | Minimum | Maximum | Status | Category | Quantity |
|------------|--------------|-------|-----------|-------|---------|---------|--------------|----------|----------|
| 1 | Fruity | 2.0 | 2.0 | 50 | 10 | 30 | Enough Stock | Eatery | |
| 2 | Doritos | 10.0 | 5.0 | 50 | 30 | 100 | Enough Stock | Eatery | |


Below the table are three buttons: REMOVE, ADD TO WISHLIST, and CONFIRM. Underneath these buttons is a section titled 'PREVIOUS BASKET:' which contains a scrollable list of previous items:

- 1 Fruity 2.0 2.0 50 10 30 Enough Stock Eatery null
- 2 Doritos 10.0 5.0 50 30 100 Enough Stock Eatery null
- 3 Camera 50.0 600.0 500 20 900 Enough Stock Electronics null
- 1 Fruity 2.0 2.0 50 10 30 Enough Stock Eatery null
- 2 Doritos 10.0 5.0 50 30 100 Enough Stock Eatery null
- 3 Camera 50.0 600.0 500 20 900 Enough Stock Electronics null

Along with this, we also see the basket that was previously present.

We can remove the products from the list, add them to the wish list and can continue to the checkout by clicking on confirm.

Lets remove one items:


— □ ×

Login
Display Products
Edit Products
Order
Basket
Checkout
WISHLIST

| Product ID | Product Name | Price | VAT Price | Stock | Minimum | Maximum | Status | Category | Quantity |
|------------|--------------|-------|-----------|-------|---------|---------|--------------|----------|----------|
| 2 | Doritos | 10.0 | 5.0 | 50 | 30 | 100 | Enough Stock | Eatery | |

REMOVE

ADD TO WISHLIST

CONFIRM

PREVIOUS BASKET:

1 Fruity 2.0 2.0 50 10 30 Enough Stock Eatery null
2 Doritos 10.0 5.0 50 30 100 Enough Stock Eatery null
3 Camera 50.0 600.0 500 20 900 Enough Stock Electronics null
1 Fruity 2.0 2.0 50 10 30 Enough Stock Eatery null
2 Doritos 10.0 5.0 50 30 100 Enough Stock Eatery null
3 Camera 50.0 600.0 500 20 900 Enough Stock Electronics null

USE CASE 8: Adding items in Wishlist:

We can add items from basket to the wish list and then back to wish list as well.

— □ ×

Login
Display Products
Edit Products
Order
Basket
Checkout
WISHLIST

| Product ID | Product Name | Price | VAT Price | Stock | Minimum | Maximum | Status | Category | Quantity |
|------------|--------------|-------|-----------|-------|---------|---------|--------------|-------------|----------|
| 2 | Doritos | 10.0 | 5.0 | 50 | 30 | 100 | Enough Stock | Eatery | |
| 1 | Fruity | 2.0 | 4.3 | 50 | 10 | 30 | Enough Stock | Eatery | |
| 2 | Doritos | 10.0 | 5.0 | 50 | 30 | 100 | Enough Stock | Eatery | |
| 3 | Camera | 50.0 | 600.0 | 500 | 20 | 900 | Enough Stock | Electronics | |
| 4 | Pen | 1.0 | 10.0 | 60 | 100 | 1000 | Re-Order | Stationary | |

ADD TO BASKET

Again, back to the Basket:

— □ ×

Login
Display Products
Edit Products
Order
Basket
Checkout
WISHLIST

| Product ID | Product Name | Price | VAT Price | Stock | Minimum | Maximum | Status | Category | Quantity |
|------------|--------------|-------|-----------|-------|---------|---------|--------------|-------------|----------|
| 2 | Doritos | 10.0 | 5.0 | 50 | 30 | 100 | Enough Stock | Eatery | |
| 1 | Fruity | 2.0 | 4.3 | 50 | 10 | 30 | Enough Stock | Eatery | |
| 2 | Doritos | 10.0 | 5.0 | 50 | 30 | 100 | Enough Stock | Eatery | |
| 3 | Camera | 50.0 | 600.0 | 500 | 20 | 900 | Enough Stock | Electronics | |
| 4 | Pen | 1.0 | 10.0 | 60 | 100 | 1000 | Re-Order | Stationary | |
| 2 | Doritos | 10.0 | 5.0 | 50 | 30 | 100 | Enough Stock | Eatery | |
| 1 | Fruity | 2.0 | 4.3 | 50 | 10 | 30 | Enough Stock | Eatery | |
| 2 | Doritos | 10.0 | 5.0 | 50 | 30 | 100 | Enough Stock | Eatery | |

REMOVE

ADD TO WISHLIST

CONFIRM

PREVIOUS BASKET:

1 Fruity 2.0 2.0 50 10 30 Enough Stock Eatery null
2 Doritos 10.0 5.0 50 30 100 Enough Stock Eatery null
3 Camera 50.0 600.0 500 20 900 Enough Stock Electronics null
1 Fruity 2.0 2.0 50 10 30 Enough Stock Eatery null
2 Doritos 10.0 5.0 50 30 100 Enough Stock Eatery null
3 Camera 50.0 600.0 500 20 900 Enough Stock Electronics null

We see that the wish list works as expected therefore this use case works correctly.

USE-CASE 7: CHECKOUT

After deciding the products from the basket, we then confirm these items and then look at the Checkout tab.

The screenshot shows a web application interface for the checkout process. At the top, there is a navigation bar with tabs: Login, Display Products, Edit Products, Order, Basket, Checkout (selected), and WISHLIST. Below the navigation bar, a list of selected products is displayed in a table-like format. The products are:

| | | | | | | | | | |
|---|---------|------|-------|-----|-----|------|--------------|-------------|------|
| 2 | Doritos | 10.0 | 5.0 | 50 | 30 | 100 | Enough Stock | Eatery | null |
| 1 | Fruity | 2.0 | 4.3 | 50 | 10 | 30 | Enough Stock | Eatery | null |
| 2 | Doritos | 10.0 | 5.0 | 50 | 30 | 100 | Enough Stock | Eatery | null |
| 3 | Camera | 50.0 | 600.0 | 500 | 20 | 900 | Enough Stock | Electronics | null |
| 4 | Pen | 1.0 | 10.0 | 60 | 100 | 1000 | Re-Order | Stationary | null |

Below the product list, the total price is shown as "TOTAL:" followed by a text box containing "51.0". Underneath, there are two input fields: "Account Number:" and "Sort Code:". At the bottom center, there is a blue button labeled "CONFIRM ORDER".

We can see that the list of selected products is listen along with other details. We also see the total price to pay along with option to see the Account number and sort code input options.

INTEGRATION:

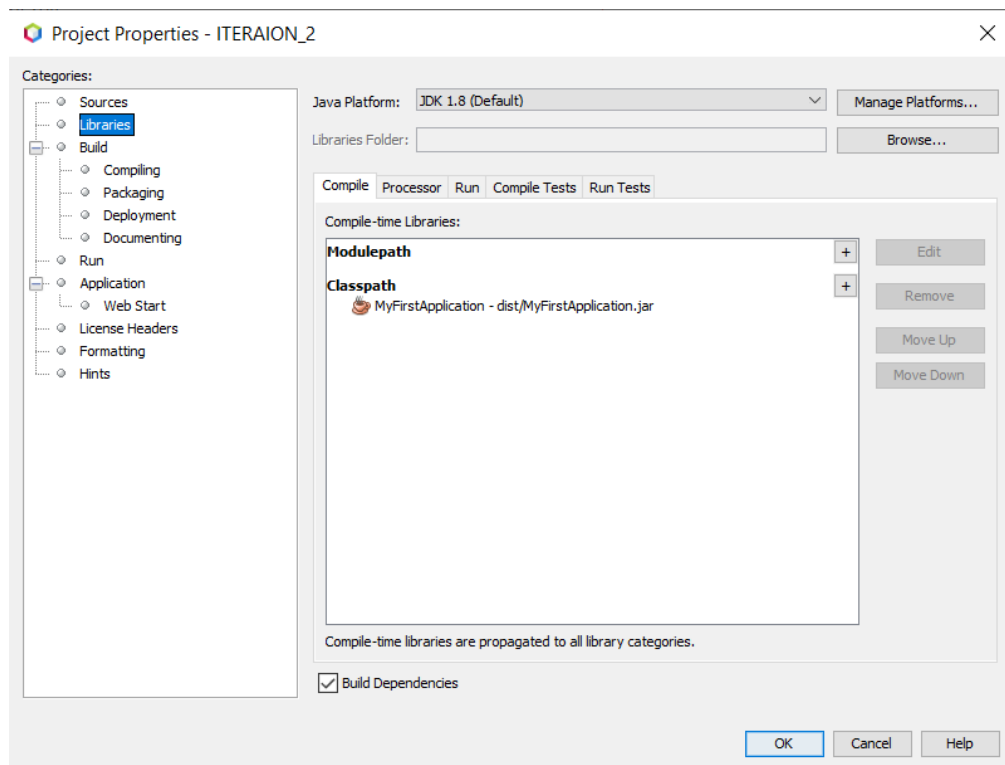
Implementing integration between two java projects in different packages was a bit challenging. But after hours of trying, I found a solution. I tried my best not to go for the solution involving copying classes into the same packages and finally found the appropriate solution.

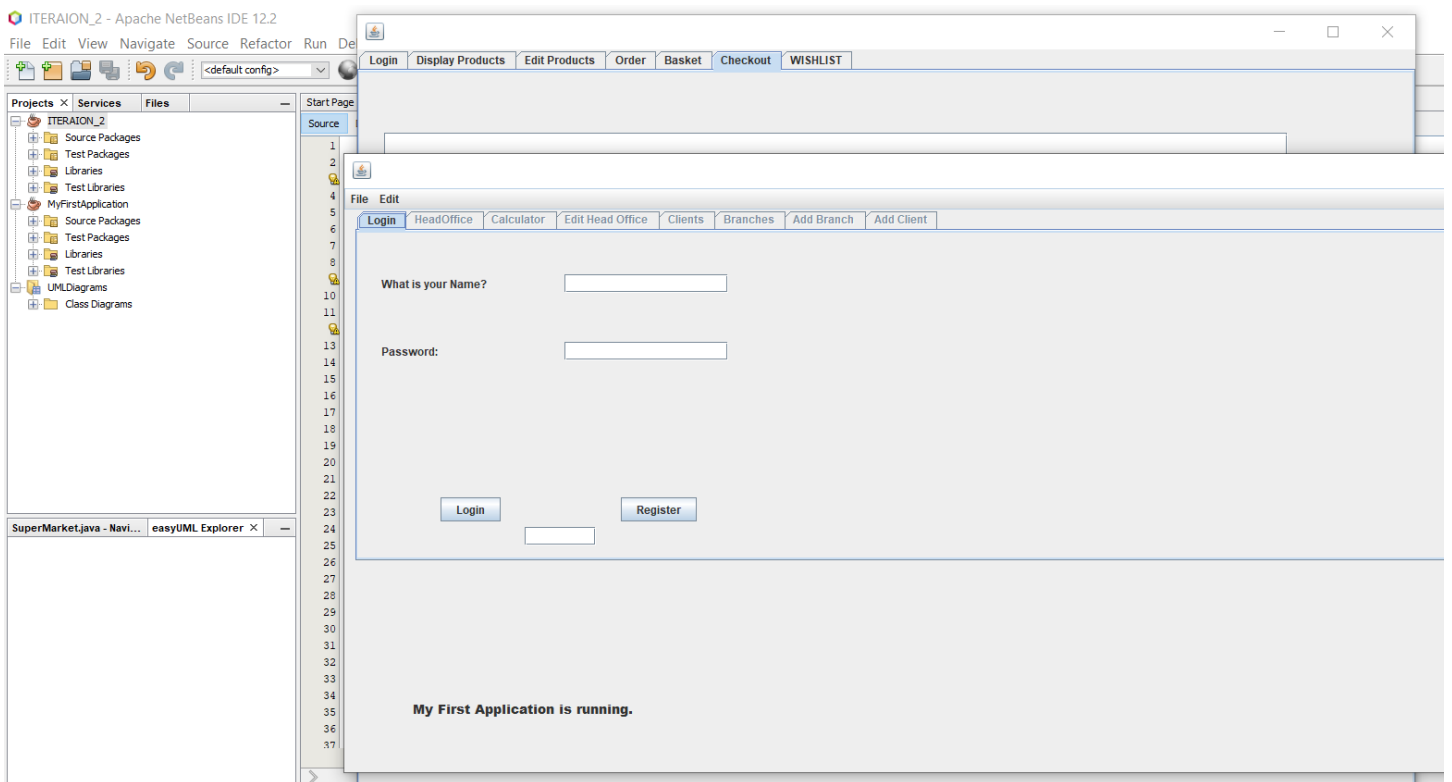
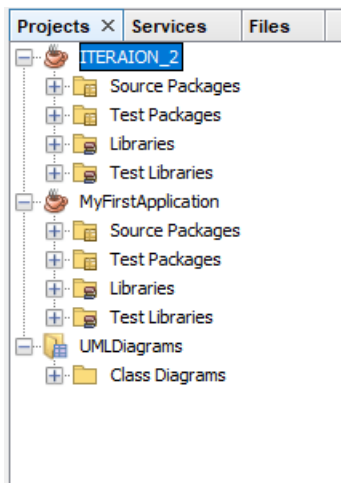
In order to integrate both packages together, we right click on our project and then select Libraries. And then add the other project that we made (Banking Software). This process set up the class path. After this, we simply import package into our Supermarket main JFrame by typing:

`import myfirstapplication.*;` (where myfirstapplication is the package of Banking software).

After this, our task was easy, we simply code for the confirm button inside our super market software which on clicking would redirect us to the login page of our Banking software.

Here is what it looks like after importing:





Here we obtain the second frame without opening it manually. Thus, integration is done correctly.