Criterion B

Class Design

Having a number of classes that specialize in performing their own separate tasks is helpful to organize the entire program in a meaningful manner and can help the owner not get confused about the complexity of the program. Below is a table highlighting the classes used in the program and a brief description of their function and components:-

Class Name	Description	Class Type	Object Calling
LoginPage	This class verifies the user's login credentials to enter the program	Display	Reprographicsdata
Reprographicsdata	This class is the main menu that uses buttons to other classes	Main	• Inventory • Printing and Photocopy • Issuing History • User Details
Inventory	Provides information about what items are available	Display	Reprographicsdata
Printing and Photocopy	Shows records of the printing and photocopy jobs in Reprographics	Display	Reprographicsdata
Issuing History	Shows the items that have been issued to people	Display	Reprographicsdata
User Details	Stores all the details about the users that can gain access to the program	Display	Reprographicsdata

Database Structure

I used one big database with multiple tables in order to execute different functions for each. In the table below I am providing details about each table and the Columns that each table has.

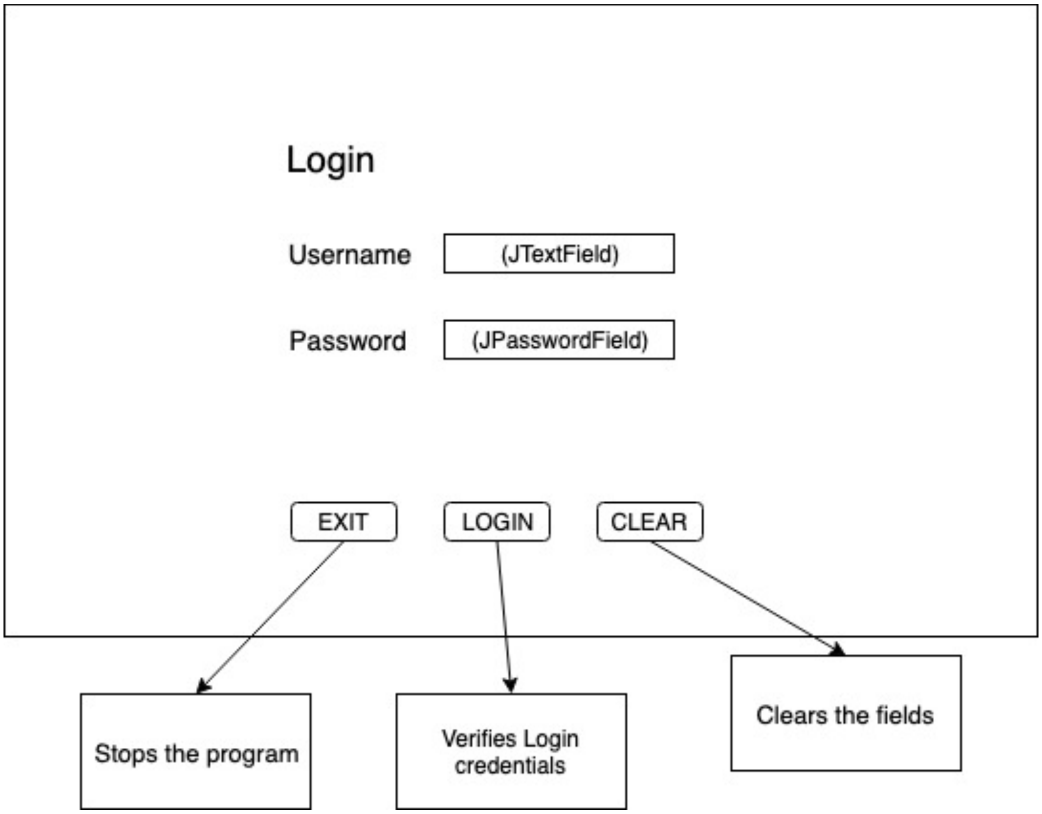
Table Name	Table Function	Columns
UserDetails (Username and Password)	The table will contain all the usernames and passwords belonging to the users of the software.	**'Id '**(INTEGER NOT NULL) **'Username '**(TEXT NOT NULL) **'Password '**(TEXT NOT NULL) **'Admin or not '**(BOOLEAN)
Printing and Photocopy	Contains details about all the printing and photocopying work that has been done at the Reprographics department.	**\`ID\`** (INTEGER NOT NULL UNIQUE) **\`Name\`** (TEXT NOT NULL) **\`Work Type\`** (TEXT NOT NULL) **\`Quantity\`** (INTEGER NOT NULL) **\`Choice of Paper\`** (TEXT NOT NULL)
Inventory	Contains details about the number of items available, required for order, item number and etc	**'ID '**(INTEGER) **'Item Name '**(TEXT NOT NULL) **'Type of Item '**(TEXT NOT NULL) **'Units in Inventory '**(INTEGER)

Issuing History	Contains details about each issued item	**`ID`** (INTEGER NOT NULL UNIQUE) **`Name`**(TEXT NOT NULL) **‘Issued Item ’**(INTEGER) **‘Quantity Issued ’**(INTEGER) **‘Email ’**(TEXT NOT NULL)
Courier	Contains details about packages delivered to campus for student	‘**Recipient Name’** (TEXT NOT NULL) **‘Origin’** (TEXT NOT NULL) **‘Email’** (VARCHAR NOT NULL) **‘Date of Arrival’** (TEXT NOT NULL)

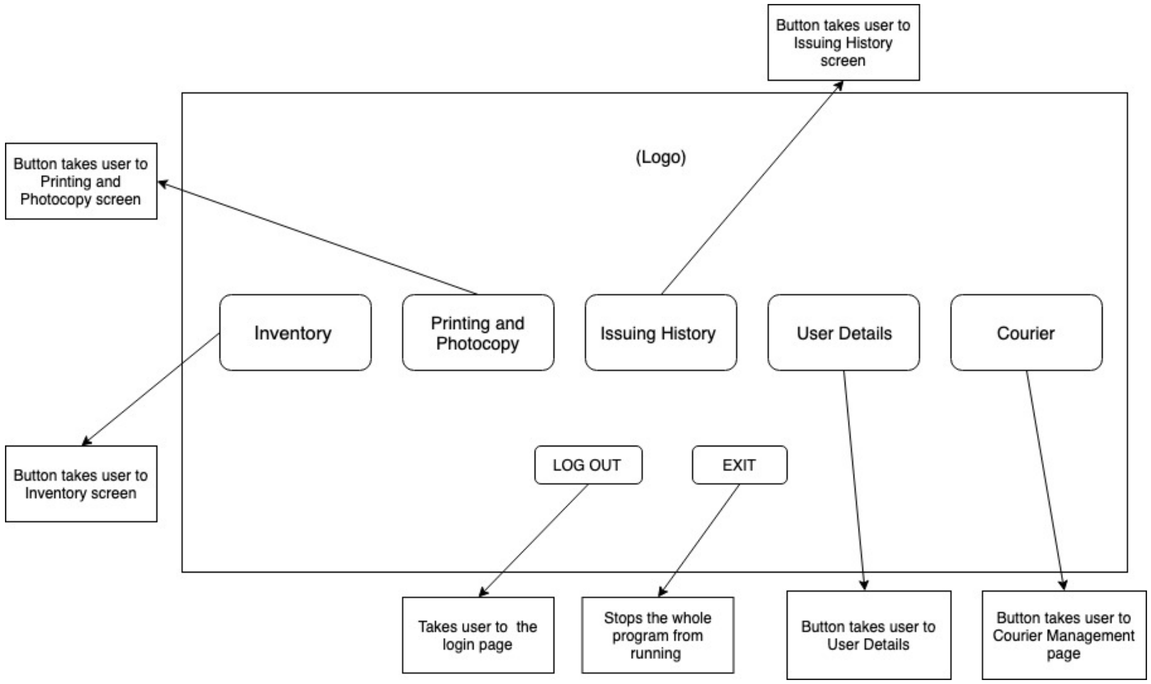
Design Layouts

The layouts were made using Draw.io to get a raw yet realistic idea of how the GUI would look in a fast and efficient manner.

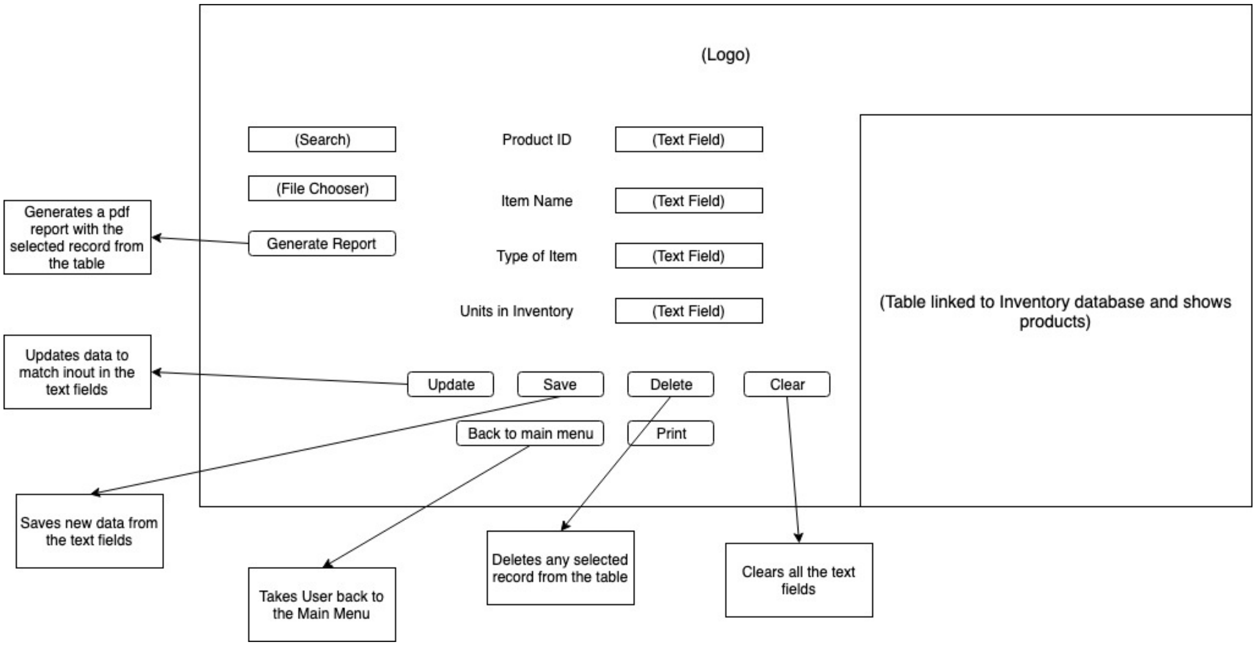
Login Screen



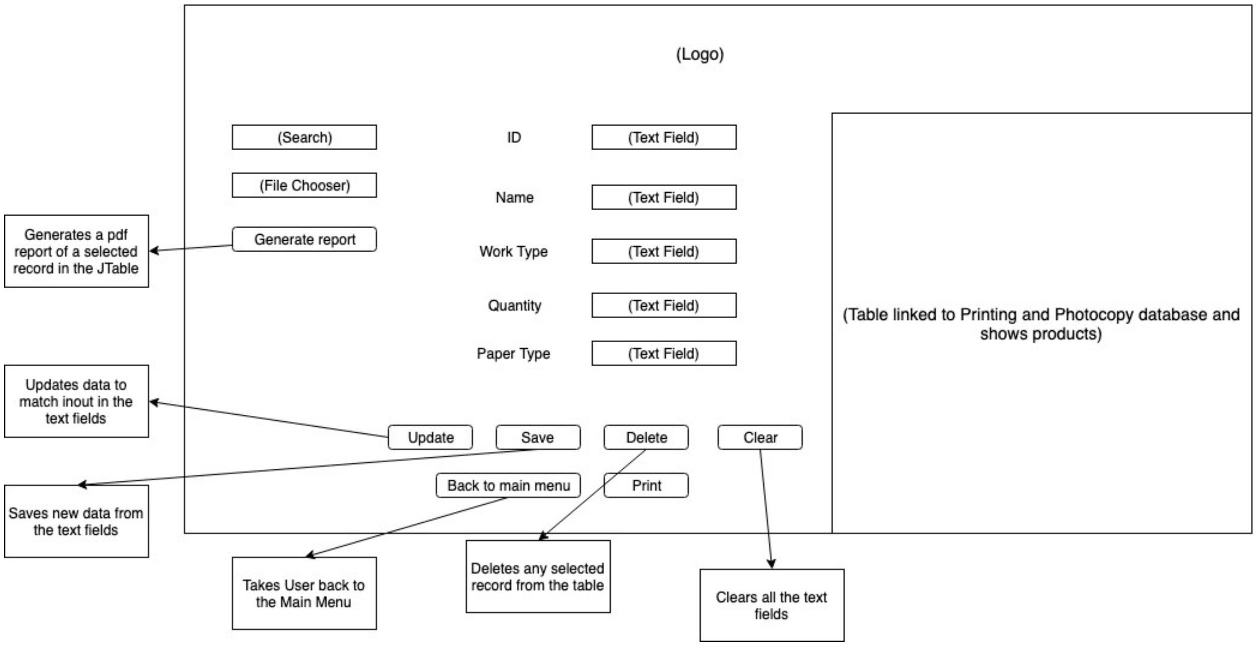
Menu Screen


Button takes user to Issuing History screen
(Logo)
Button takes user to Printing and Photocopy screen
Inventory
Printing and Photocopy
Issuing History
User Details
Courier
Button takes user to Inventory screen
LOG OUT
EXIT
Takes user to the login page
Stops the whole program from running
Button takes user to User Details
Button takes user to Courier Management page

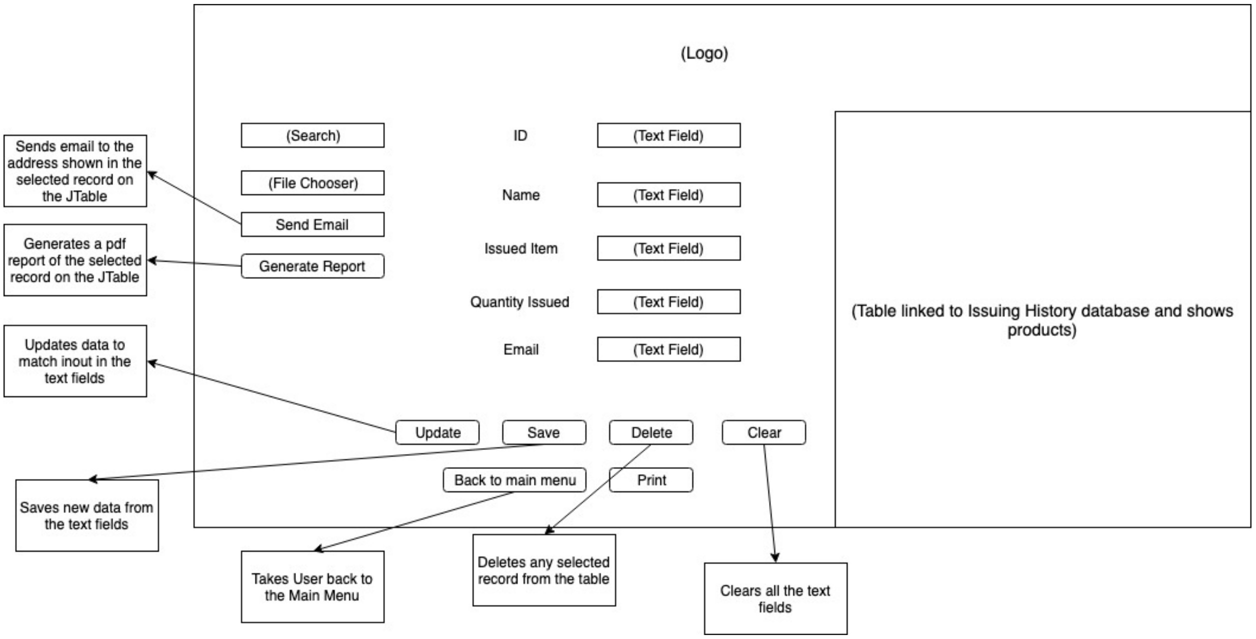
Manage Inventory Screen


(Logo)
(Search)
Product ID
(Text Field)
(File Chooser)
Item Name
(Text Field)
Generates a pdf report with the selected record from the table
Generate Report
Type of Item
(Text Field)
Units in Inventory
(Text Field)
(Table linked to Inventory database and shows products)
Updates data to match inout in the text fields
Update
Save
Delete
Clear
Back to main menu
Print
Saves new data from the text fields
Takes User back to the Main Menu
Deletes any selected record from the table
Clears all the text fields

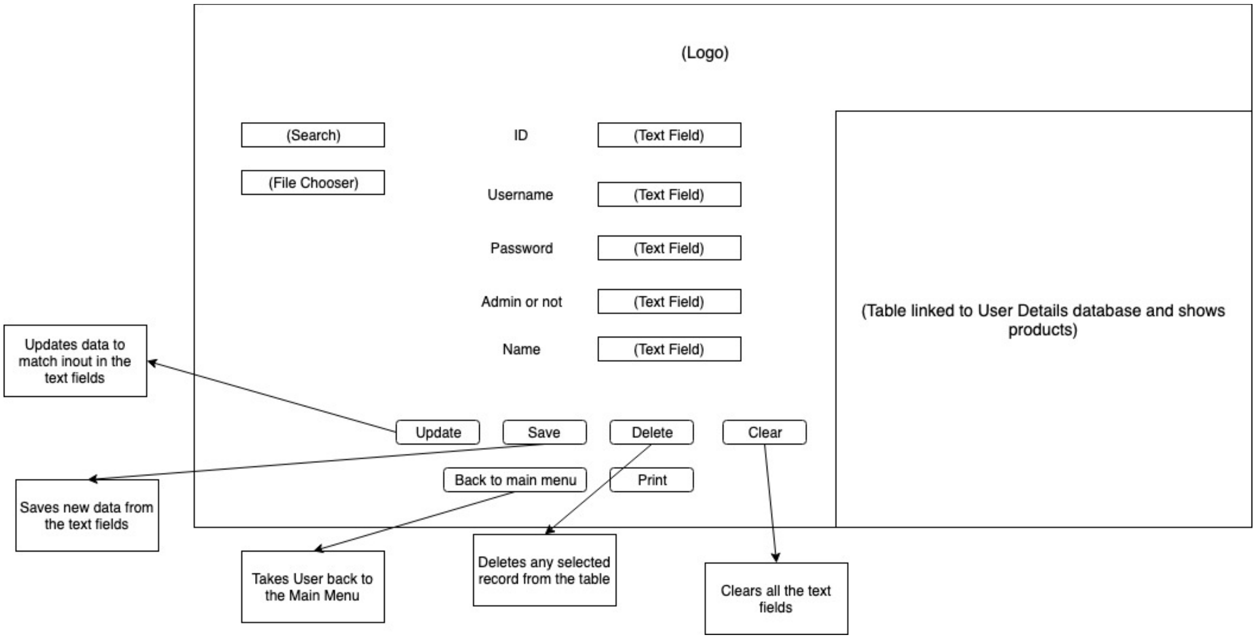
Printing and Photocopy Screen


(Logo)
(Search)
(File Chooser)
Generate report
Generates a pdf report of a selected record in the JTable
ID
(Text Field)
Name
(Text Field)
Work Type
(Text Field)
Quantity
(Text Field)
Paper Type
(Text Field)
(Table linked to Printing and Photocopy database and shows products)
Updates data to match inout in the text fields
Update
Save
Delete
Clear
Back to main menu
Print
Saves new data from the text fields
Takes User back to the Main Menu
Deletes any selected record from the table
Clears all the text fields

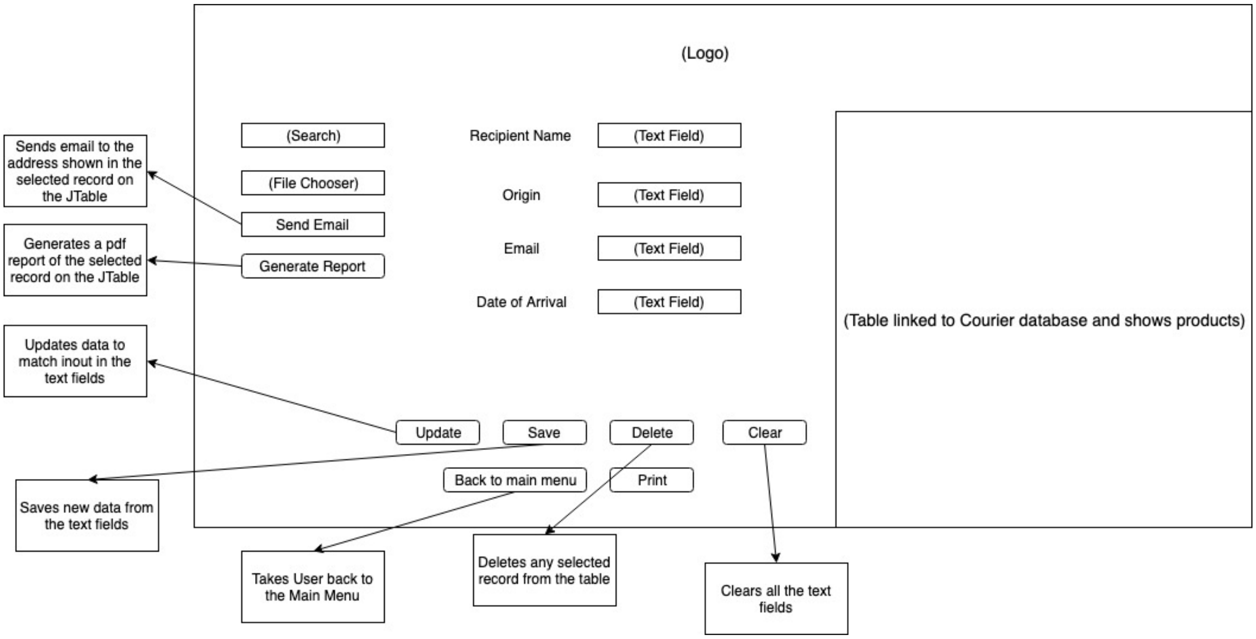
Issuing History Screen


(Logo)
(Search)
(File Chooser)
Send Email
Sends email to the address shown in the selected record on the JTable
Generate Report
Generates a pdf report of the selected record on the JTable
ID
(Text Field)
Name
(Text Field)
Issued Item
(Text Field)
Quantity Issued
(Text Field)
Email
(Text Field)
(Table linked to Issuing History database and shows products)
Updates data to match inout in the text fields
Update
Save
Delete
Clear
Back to main menu
Print
Saves new data from the text fields
Takes User back to the Main Menu
Deletes any selected record from the table
Clears all the text fields

User Details Screen


(Logo)
(Search)
ID
(Text Field)
(File Chooser)
Username
(Text Field)
Password
(Text Field)
Admin or not
(Text Field)
Name
(Text Field)
(Table linked to User Details database and shows products)
Updates data to match inout in the text fields
Update
Save
Delete
Clear
Back to main menu
Print
Saves new data from the text fields
Takes User back to the Main Menu
Deletes any selected record from the table
Clears all the text fields

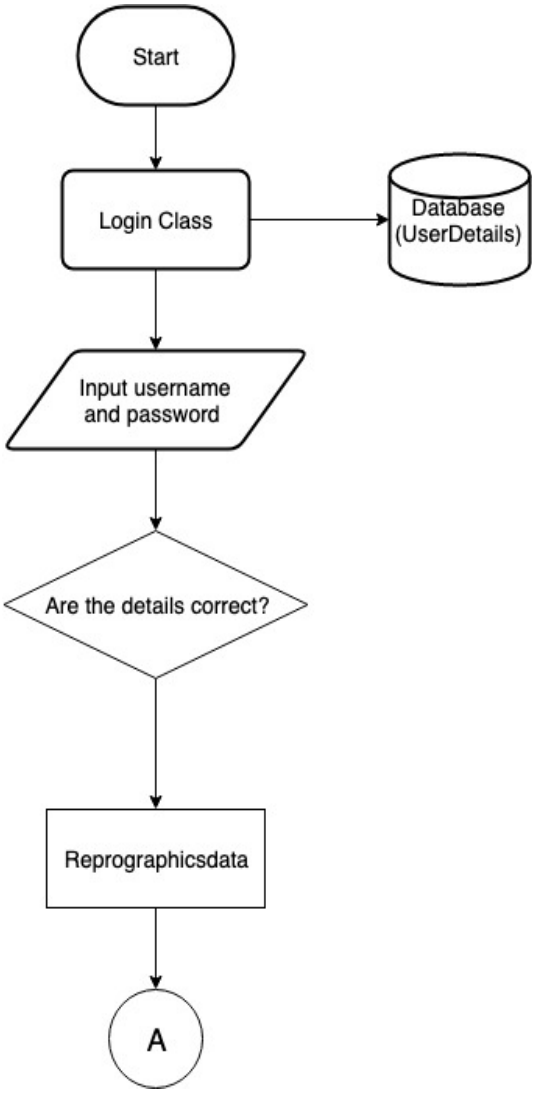
Courier Management Screen


(Logo)
Sends email to the address shown in the selected record on the JTable
(Search)
Recipient Name
(Text Field)
(File Chooser)
Origin
(Text Field)
Send Email
Email
(Text Field)
Generates a pdf report of the selected record on the JTable
Generate Report
Date of Arrival
(Text Field)
(Table linked to Courier database and shows products)
Updates data to match inout in the text fields
Update
Save
Delete
Clear
Back to main menu
Print
Saves new data from the text fields
Takes User back to the Main Menu
Deletes any selected record from the table
Clears all the text fields

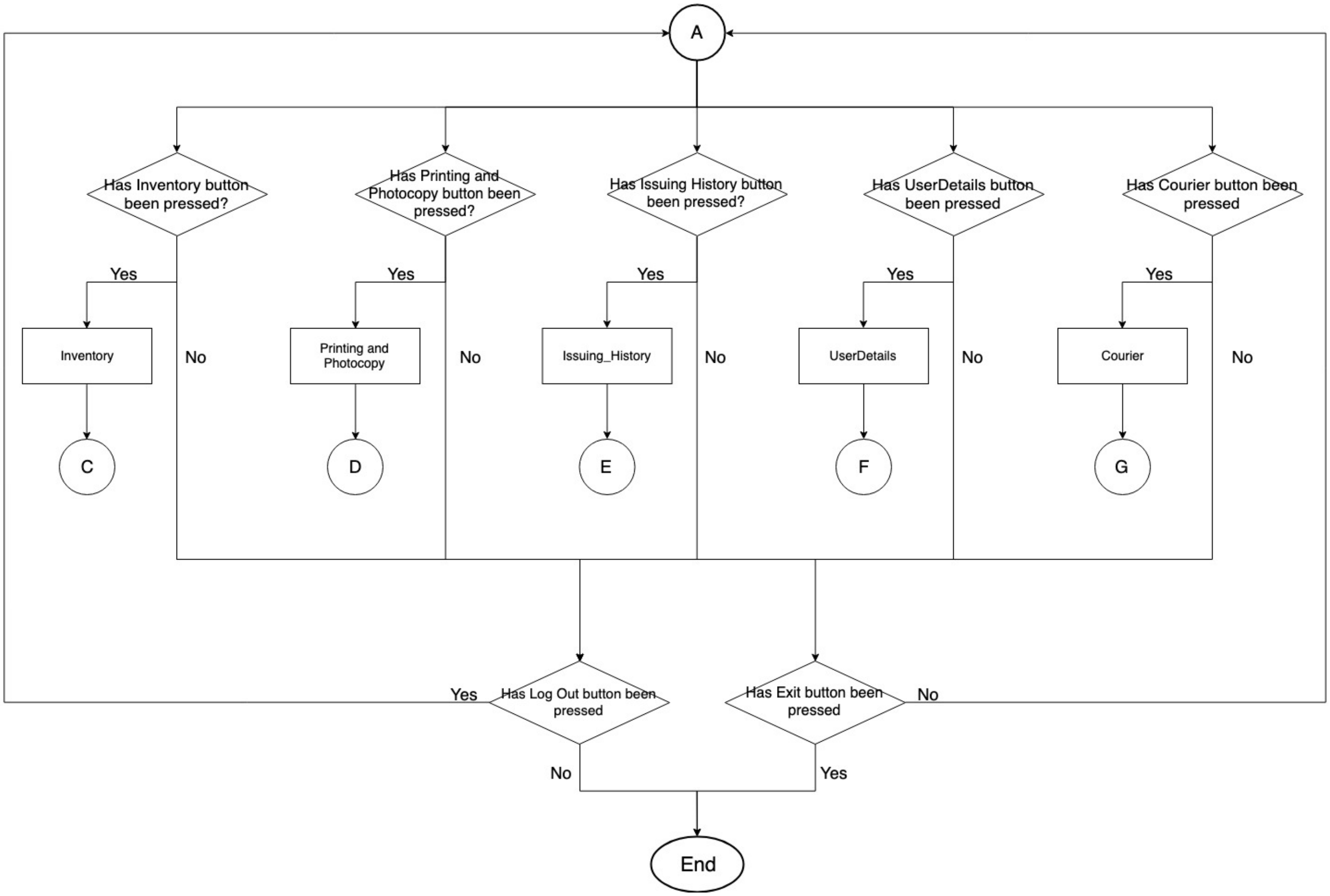
System Flowchart

The flowchart of the program allows us to view how the program is structured in a broader manner and look at how the different objects and methods are linked with each other in chronological order. Below is the system flowchart represented by a few diagrams segmented according to the various portions of the program made in Draw.io:-

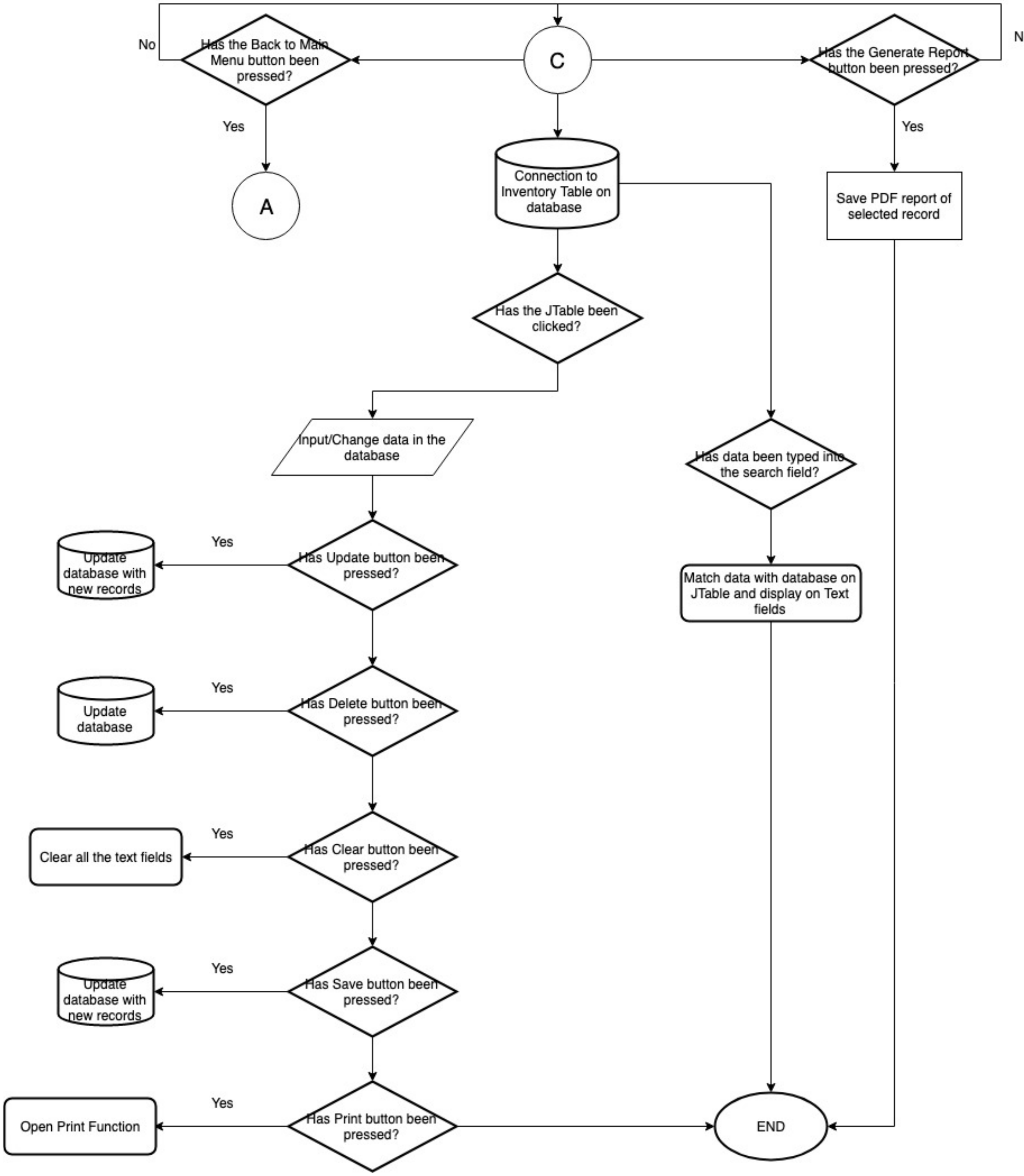
Beginning



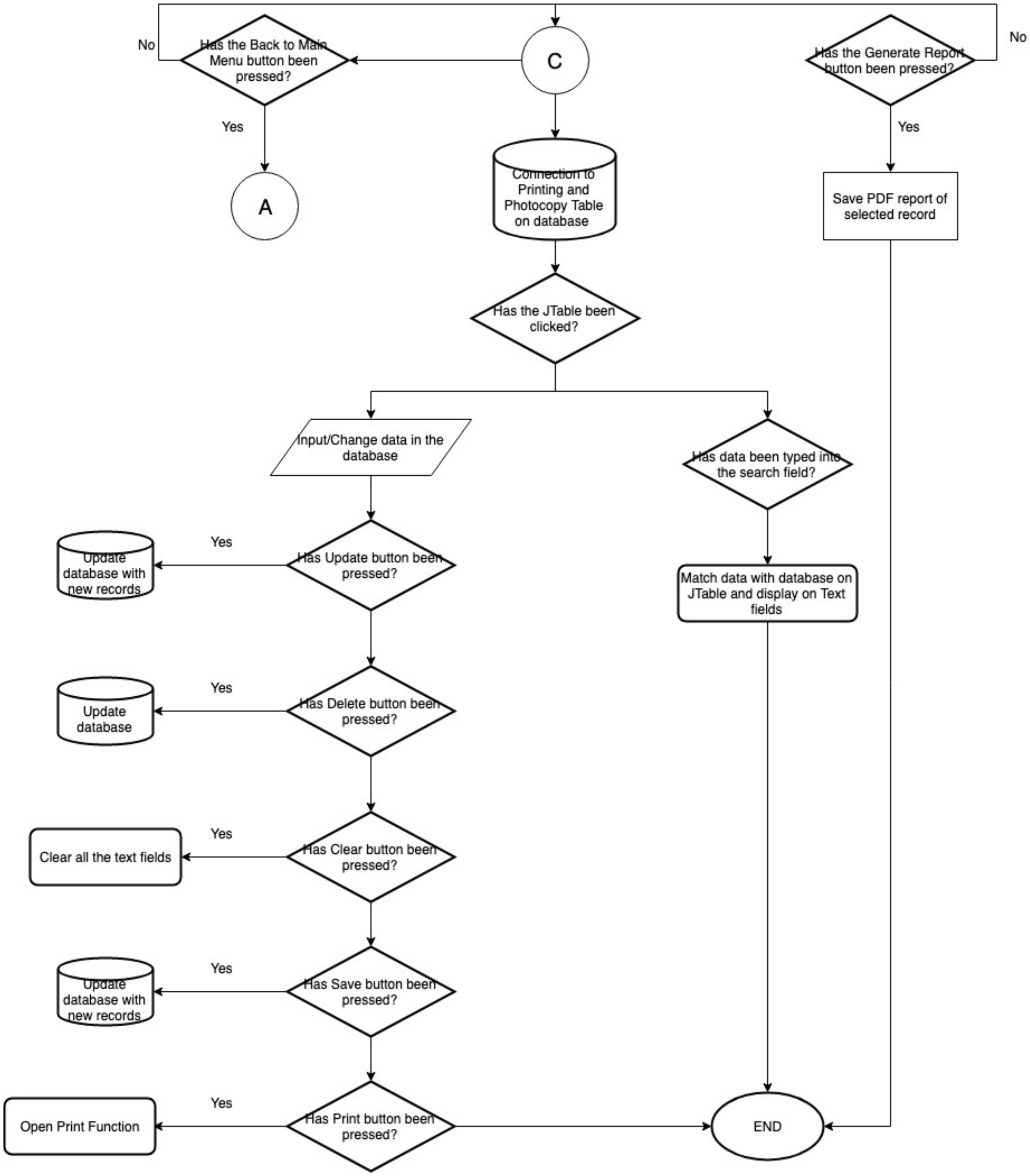
Main Menu


A
Has Inventory button been pressed?
Has Printing and Photocopy button been pressed?
Has Issuing History button been pressed?
Has UserDetails button been pressed
Has Courier button been pressed
Yes
Yes
Yes
Yes
Yes
Inventory
No
Printing and Photocopy
No
Issuing_History
No
UserDetails
No
Courier
No
C
D
E
F
G
Yes
Has Log Out button been pressed
Has Exit button been pressed
No
No
Yes
End

Inventory Management

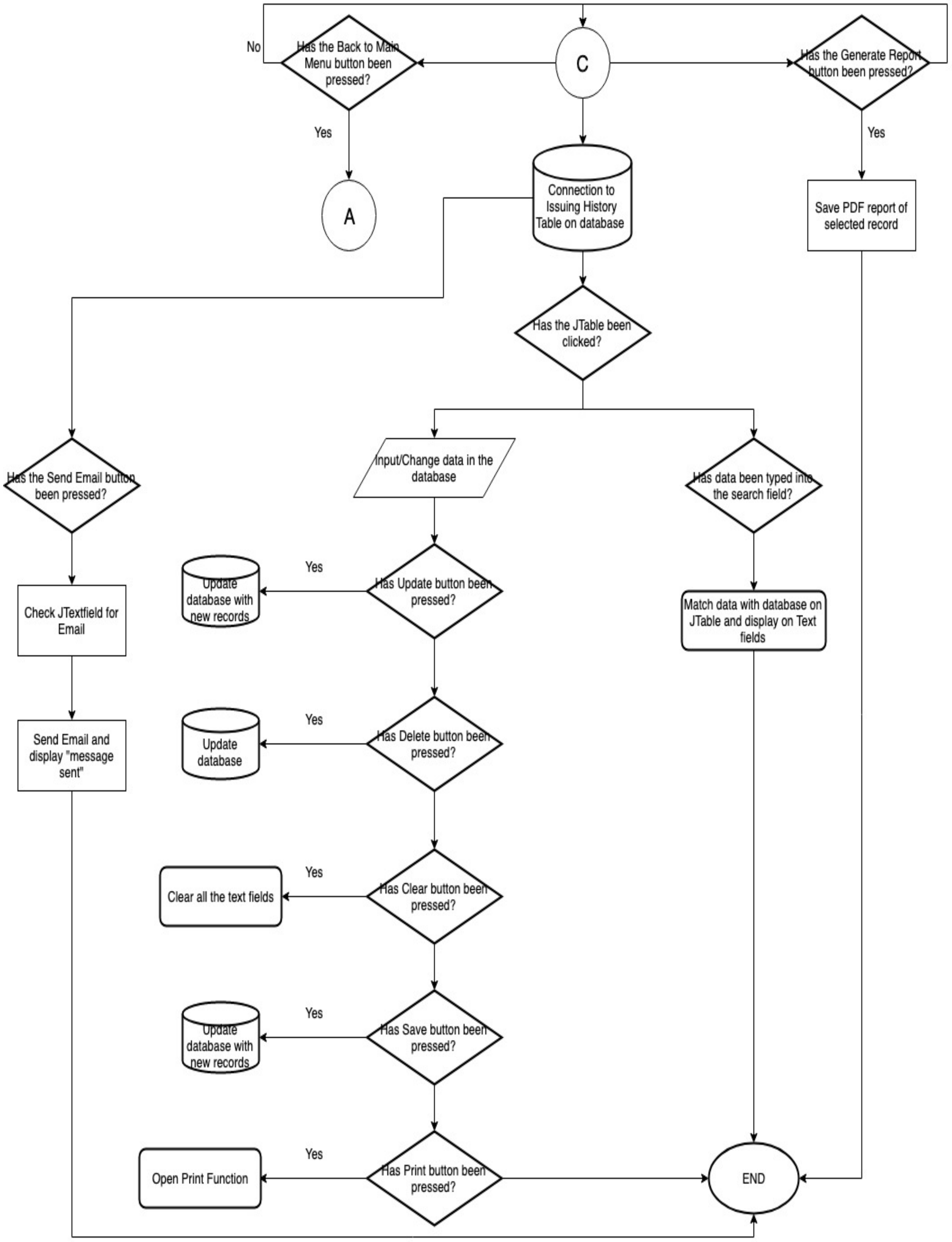

C
No
Has the Back to Main Menu button been pressed?
Yes
A
Has the Generate Report button been pressed?
No
Yes
Save PDF report of selected record
Connection to Inventory Table on database
Has the JTable been clicked?
Input/Change data in the database
Has data been typed into the search field?
Match data with database on JTable and display on Text fields
Yes
Update database with new records
Has Update button been pressed?
Yes
Update database
Has Delete button been pressed?
Yes
Clear all the text fields
Has Clear button been pressed?
Yes
Update database with new records
Has Save button been pressed?
Yes
Open Print Function
Has Print button been pressed?
END

Printing and Photocopy



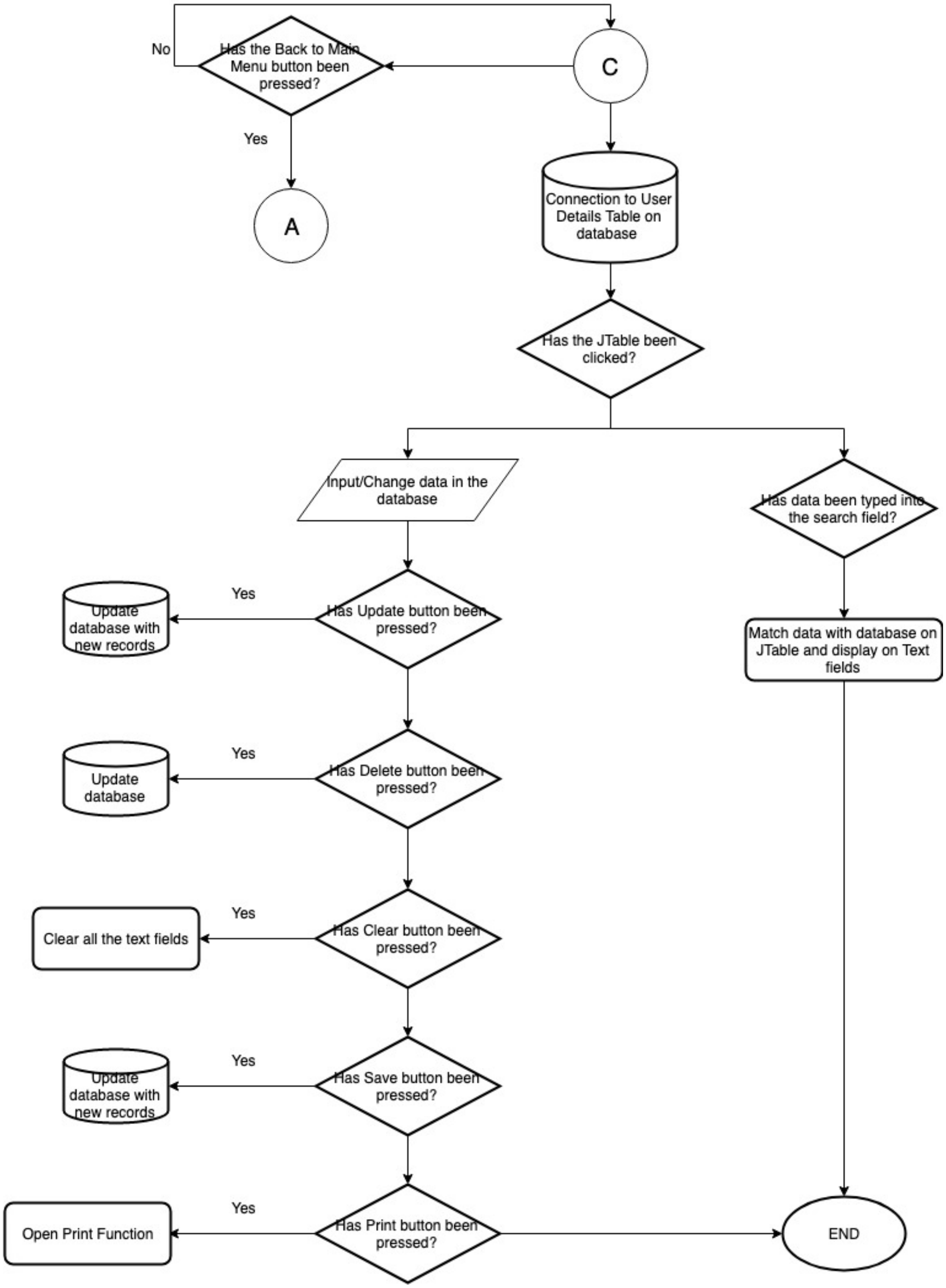
No
Has the Back to Main Menu button been pressed?
C
Has the Generate Report button been pressed?
No
Yes
A
Connection to Printing and Photocopy Table on database
Yes
Save PDF report of selected record
Has the JTable been clicked?
Input/Change data in the database
Has data been typed into the search field?
Yes
Update database with new records
Has Update button been pressed?
Match data with database on JTable and display on Text fields
Yes
Update database
Has Delete button been pressed?
Yes
Clear all the text fields
Has Clear button been pressed?
Yes
Update database with new records
Has Save button been pressed?
Yes
Open Print Function
Has Print button been pressed?
END

Issuing History

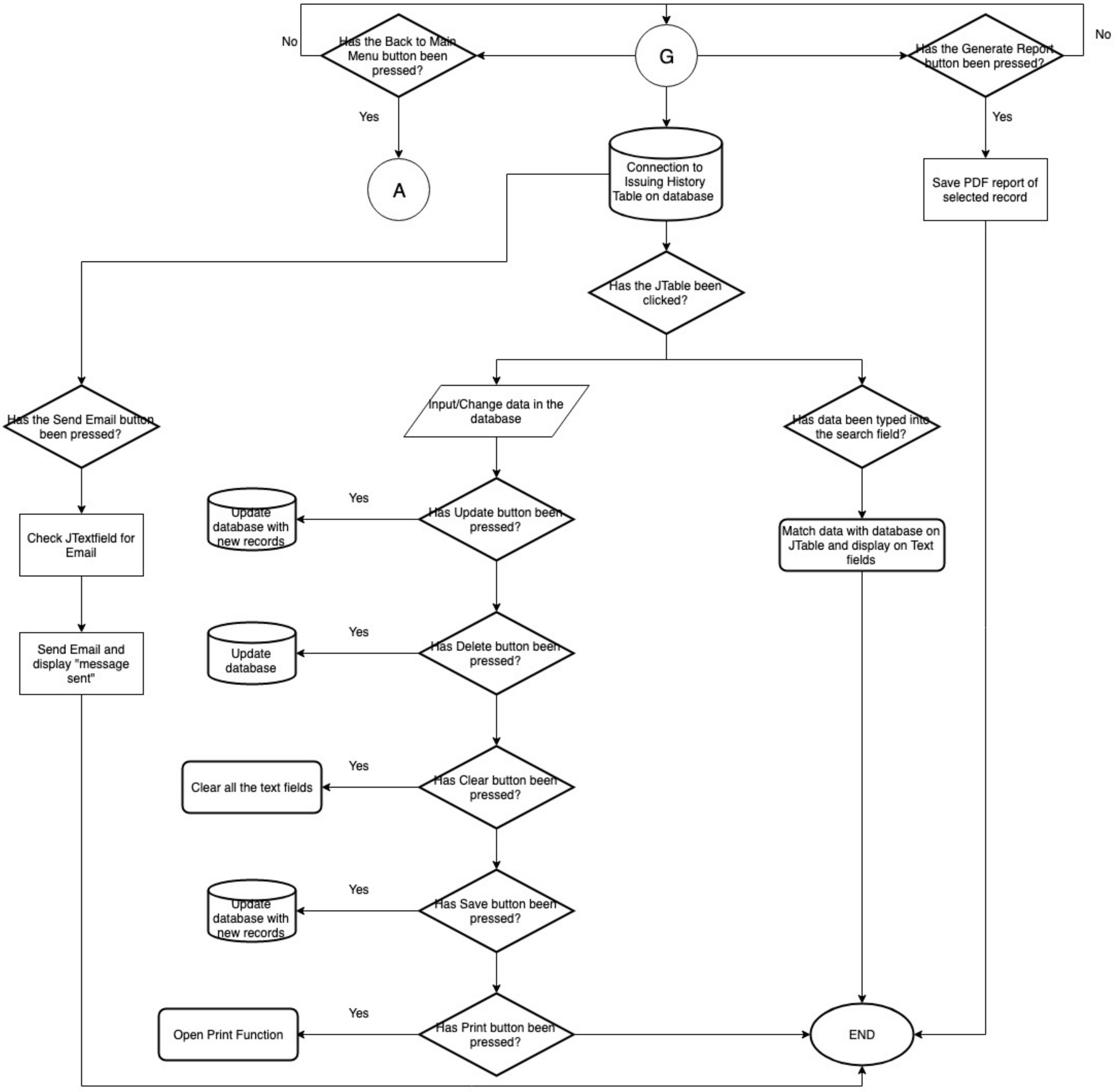


C
Has the Back to Main Menu button been pressed?
No
Yes
A
Has the Generate Report button been pressed?
No
Yes
Save PDF report of selected record
Connection to Issuing History Table on database
Has the JTable been clicked?
Has the Send Email button been pressed?
Check JTextfield for Email
Send Email and display "message sent"
Input/Change data in the database
Has Update button been pressed?
Yes
Update database with new records
Has Delete button been pressed?
Yes
Update database
Has Clear button been pressed?
Yes
Clear all the text fields
Has Save button been pressed?
Yes
Update database with new records
Has Print button been pressed?
Yes
Open Print Function
Has data been typed into the search field?
Match data with database on JTable and display on Text fields
END

User Details

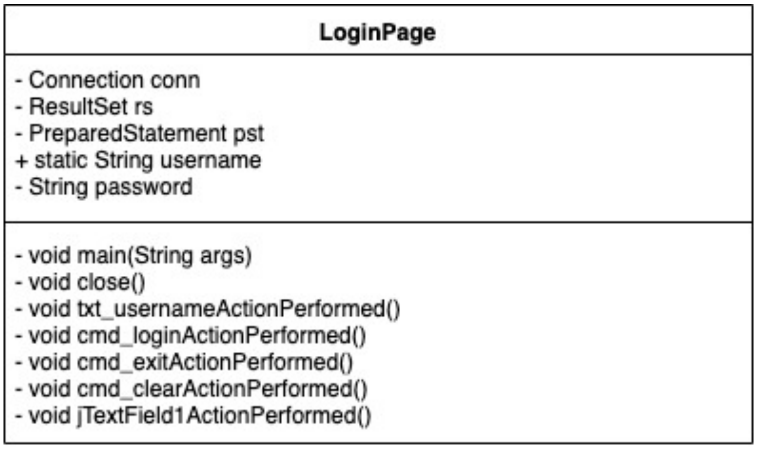

C
Has the Back to Main Menu button been pressed?
No
Yes
A
Connection to User Details Table on database
Has the JTable been clicked?
Input/Change data in the database
Has data been typed into the search field?
Yes
Has Update button been pressed?
Update database with new records
Match data with database on JTable and display on Text fields
Yes
Has Delete button been pressed?
Update database
Yes
Has Clear button been pressed?
Clear all the text fields
Yes
Has Save button been pressed?
Update database with new records
Yes
Has Print button been pressed?
Open Print Function
END

Courier Management


G
Has the Back to Main Menu button been pressed?
No
Yes
A
Has the Generate Report button been pressed?
No
Yes
Save PDF report of selected record
Connection to Issuing History Table on database
Has the JTable been clicked?
Has the Send Email button been pressed?
Check JTextfield for Email
Send Email and display "message sent"
Input/Change data in the database
Has Update button been pressed?
Yes
Update database with new records
Has Delete button been pressed?
Yes
Update database
Has Clear button been pressed?
Yes
Clear all the text fields
Has Save button been pressed?
Yes
Update database with new records
Has Print button been pressed?
Yes
Open Print Function
Has data been typed into the search field?
Match data with database on JTable and display on Text fields
END

UML Diagrams

UML Diagrams help us get a sense of roughly what each class contains, what the variables in it are and what its main methods and functions are. Below are UML diagrams for all important classes:-

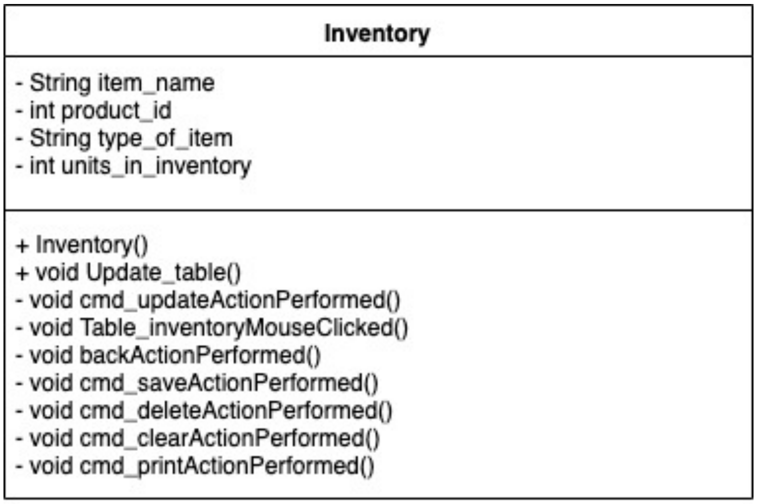


LoginPage

- Connection conn
- ResultSet rs
- PreparedStatement pst
+ static String username
- String password

```
- Connection conn
- ResultSet rs
- PreparedStatement pst
+ static String username
- String password
```

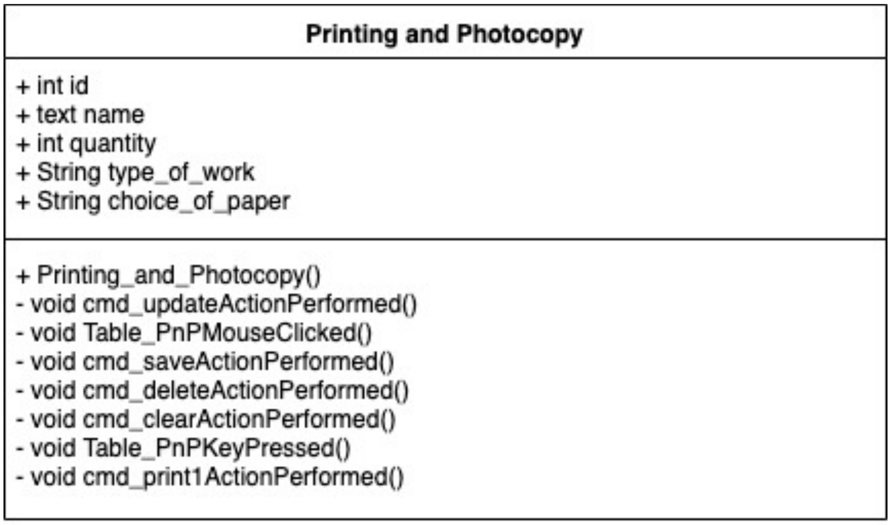
```
- void main(String args)
- void close()
- void txt_usernameActionPerformed()
- void cmd_loginActionPerformed()
- void cmd_exitActionPerformed()
- void cmd_clearActionPerformed()
- void jTextField1ActionPerformed()
```



Inventory

```
- String item_name
- int product_id
- String type_of_item
- int units_in_inventory
```

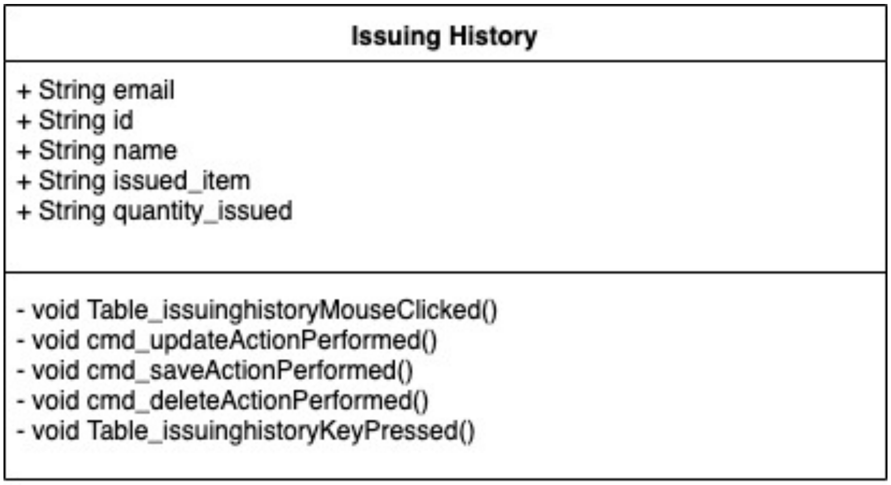
```
+ Inventory()
+ void Update_table()
- void cmd_updateActionPerformed()
- void Table_inventoryMouseClicked()
- void backActionPerformed()
- void cmd_saveActionPerformed()
- void cmd_deleteActionPerformed()
- void cmd_clearActionPerformed()
- void cmd_printActionPerformed()
```



Printing and Photocopy

```
+ int id
+ text name
+ int quantity
+ String type_of_work
+ String choice_of_paper
```

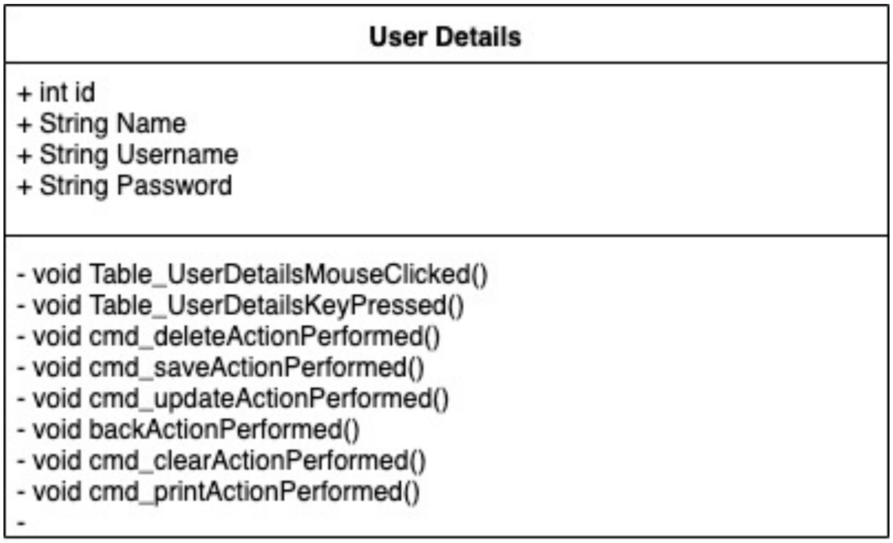
```
+ Printing_and_Photocopy()
- void cmd_updateActionPerformed()
- void Table_PnPMouseClicked()
- void cmd_saveActionPerformed()
- void cmd_deleteActionPerformed()
- void cmd_clearActionPerformed()
- void Table_PnPKeyPressed()
- void cmd_print1ActionPerformed()
```



Issuing History

```
+ String email
+ String id
+ String name
+ String issued_item
+ String quantity_issued
```

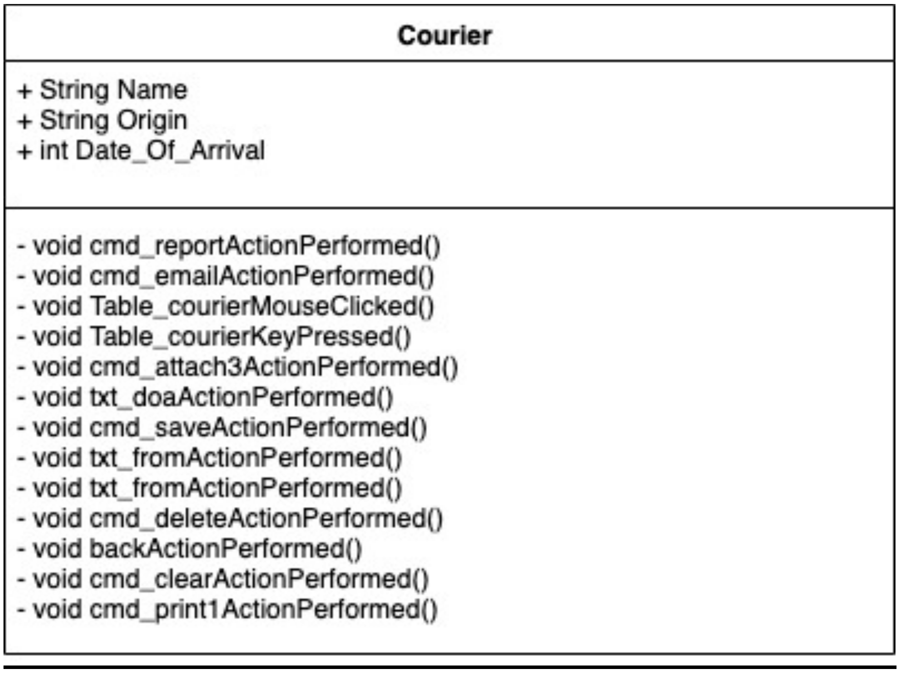
```
- void Table_issuinghistoryMouseClicked()
- void cmd_updateActionPerformed()
- void cmd_saveActionPerformed()
- void cmd_deleteActionPerformed()
- void Table_issuinghistoryKeyPressed()
```



User Details

```
+ int id
+ String Name
+ String Username
+ String Password
```

```
- void Table_UserDetailsMouseClicked()
- void Table_UserDetailsKeyPressed()
- void cmd_deleteActionPerformed()
- void cmd_saveActionPerformed()
- void cmd_updateActionPerformed()
- void backActionPerformed()
- void cmd_clearActionPerformed()
- void cmd_printActionPerformed()
-
```



Courier

+ String Name
+ String Origin
+ int Date_Of_Arrival

- void cmd_reportActionPerformed()
- void cmd_emailActionPerformed()
- void Table_courierMouseClicked()
- void Table_courierKeyPressed()
- void cmd_attach3ActionPerformed()
- void txt_doaActionPerformed()
- void cmd_saveActionPerformed()
- void txt_fromActionPerformed()
- void txt_fromActionPerformed()
- void cmd_deleteActionPerformed()
- void backActionPerformed()
- void cmd_clearActionPerformed()
- void cmd_print1ActionPerformed()

<u>Test Plan</u>

The final product will be tested against the success criteria outlined by me to ensure that the product meets the expectations of the client. The tests will evaluate comprehensively the functional aspects of the program. A tentative plan for this testing is as follows:-

Action Test	**Method of Testing**	**Expected Outcome**
Test if the program runs according to planned and the login page appears.	Click on the run option in Netbeans	The program should execute and a login page should appear.

Test if program can authenticate and grant access to authorized usernames and passwords only	User must input correct login credentials to check if the authentication works. The user must also input wrong credentials to check if the program can block access if the credentials are wrong.	The program must accept correct credentials and must reject wrong credentials.
Test if the program allows users to access different databases on the main menu page	Click on the buttons to go to the respective databases	The correct buttons should execute by closing down the menu page and opening up the database management page linked to the button.
Test if the program allows user to input, edit, save, update and delete record on all the databases	Use the JTextFields and the JButtons to input, edit, save, update and delete records on the different screens on the database.	The buttons should execute the desired actions and make changes accordingly to the database.
Test if the search button is functioning	The user must input data of any column of a record in the JTextField	The JTextFields designated to show the records must pull up the correct record with the data that has been input into the search bar.
Test if program allows user to send email onto the desired email address	Select a particular record and the email must show on the JTextField then press the send email button	Pressing the button should send an email to the designated email address, check the users inbox for email.
Test if the program generates a pdf report	Select a certain record on the JTable displaying the database and press Generate Report	A pdf report should be saved with the information of the selected record from the database.
Test if the program is able to print the databases.	Go to a database screen and press the print button	The print manager should open and the user should be able to print the database on the desired printer.