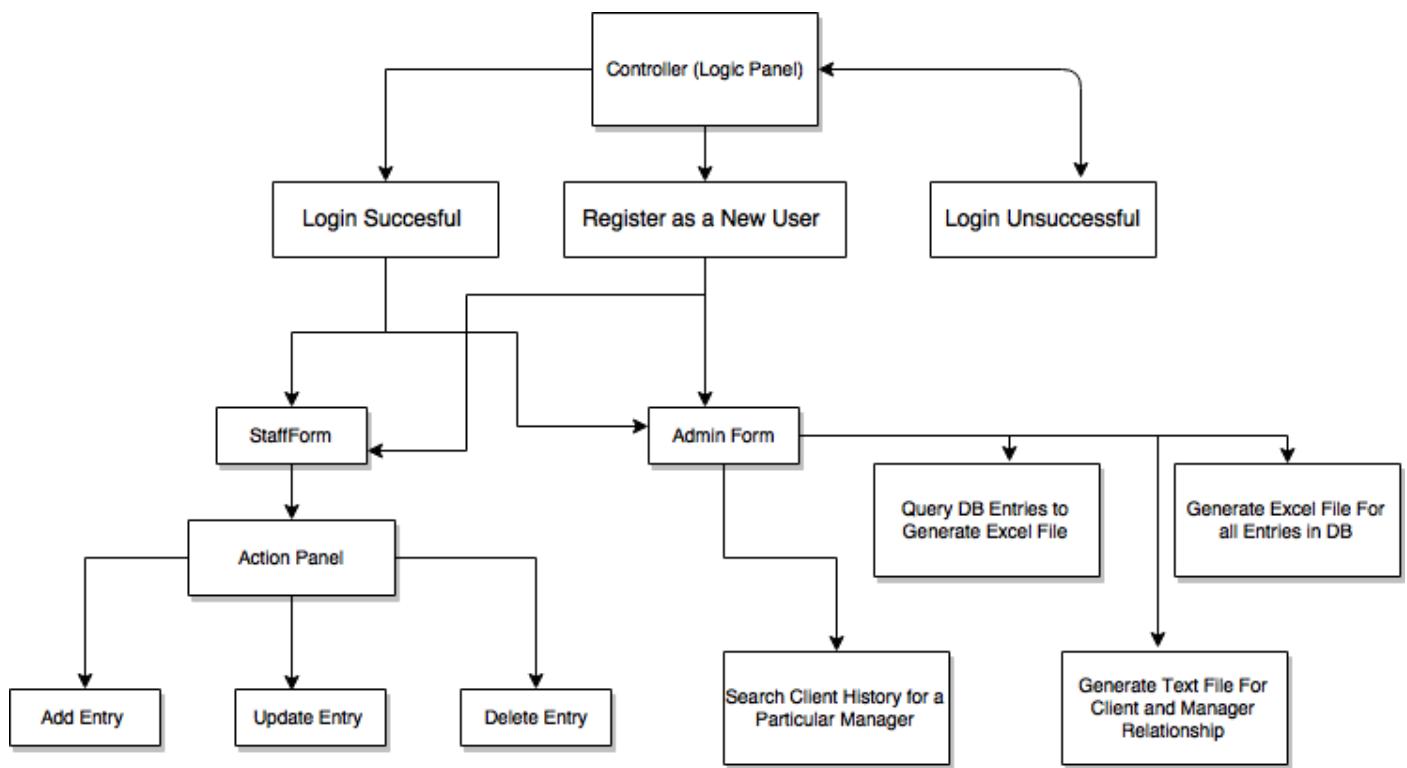


Overview of Complete Software

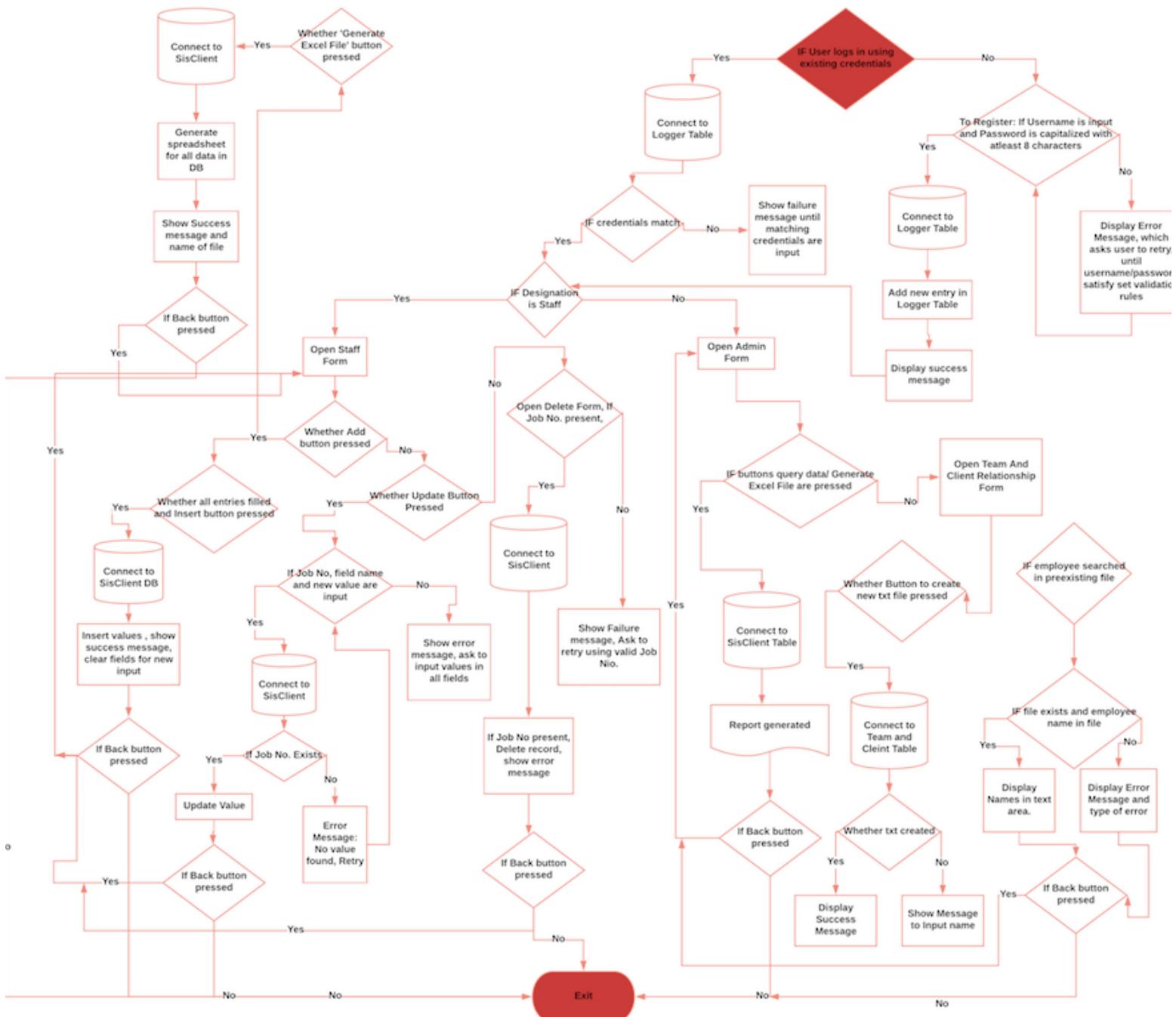


Description

The above diagram shows the procedure of control as a user enters the software. If the login is successful, then the user is directed to the staff form or the Admin form based on their preference. The Staff form allows the user to add, delete or update the existing database and create xml file to present the data in a tabular format. The Admin form allows the user to query the database, generate xml files and post comments for other staff members.

System Flowchart

The following flowchart shows how the program will function



Tables showing Fields and Datatypes

Table Name: SisClient

Field Name	Datatype	Description	Other Information
Job_no	INTEGER	Unique Job Number	Primary Key
Job_date	DATE	Current Date	
Client_name	VARCHAR (45)	Name of the Cleint	
Job_name	VARCHAR(45)	Article Name	
No_of_forms	VARCHAR(45)	Number of Forms per Machine	
Print_style	VARCHAR(45)	Single Side/ Double Side	
Paper_used	VARCHAR(45)	Dimensions and Type of Paper Used	
No_of_col	VARCHAR(45)	Number of colors used per Machine	
Paper_qty	INTEGER(11)	Quantity of Paper Used	
Plate	VARCHAR(45)	Provider and Quantity of Plates	
Paper_print	VARCHAR(45)	Amount of Paper to be printed	
Book_size	VARCHAR(45)	Dimensions of the Article to be printed	
No_of_pages	VARCHAR(45)	Number of Pages to be printed	
Plate_new/old	VARCHAR(45)	Plate Provider (Cleint or Factory)	
Paper_Required	VARCHAR(45)	Amount of paper required to print	
Paper_supp	VARCHAR(45)	Paper Supplier Name	
Paper_in_sheets	VARCHAR(45)	Amount of paper required in sheets	
Finish_quantity	VARCHAR(45)	Amount of Quantity to be printed	
Delivery_date	DATE	Deliver Date of the article	
Status	VARCHAR(45)	Whether ready OR In print	
Machine	VARCHAR(45)	Name of the Machine in use	
Space	VARCHAR(45)	Space where Paper is stored	
Team_Leader	VARCHAR(45)	Name of the Manager the Order is given to	

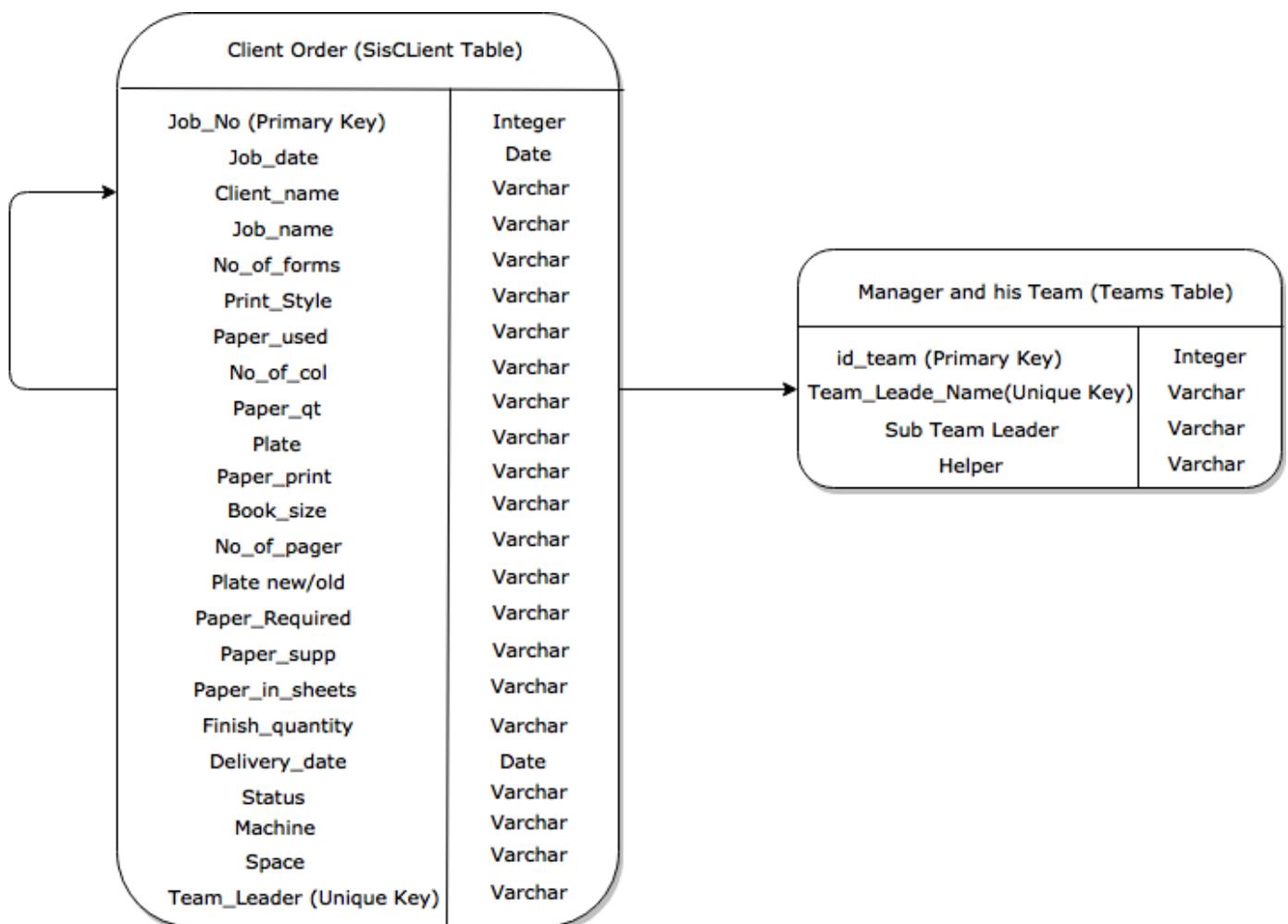
Table Name: Logger

Field Name	Datatype	Description	Other Information
Username	VARCHAR(45)		Primary Key
Password	VARCHAR(45)		
UserType	VARCHAR(45)	Whether Staff/Admin	

Table Name: Teams

Field Name	Datatype	Description	Other Information
Id_team	INTEGER(11)	Unique identification key	Primary Key
Team_Leader_Name	VARCHAR(45)		Unique Key
Sub Team Leader	VARCHAR(45)		
Helper	VARCHAR(45)		

Entity Relationship Diagram



Controller (Logic Panel)

The controller will be responsible to identify an authorized user. If the login details entered by the user match the details stored in the database, then the user is provided access. However, if the login details do not match, then the user is asked to re-enter his username, password and choose his designation in the office. The controller also allows the user to register as a new member, in which the user must type of unique username and a password the matches the security criteria of 8 alphanumeric characters.

Layout for Java Design

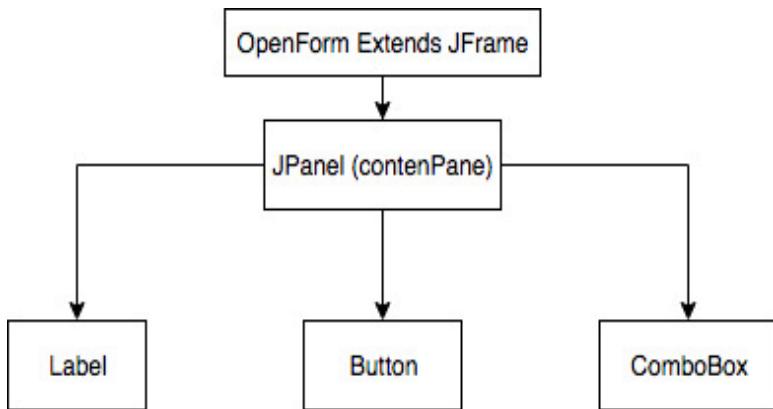
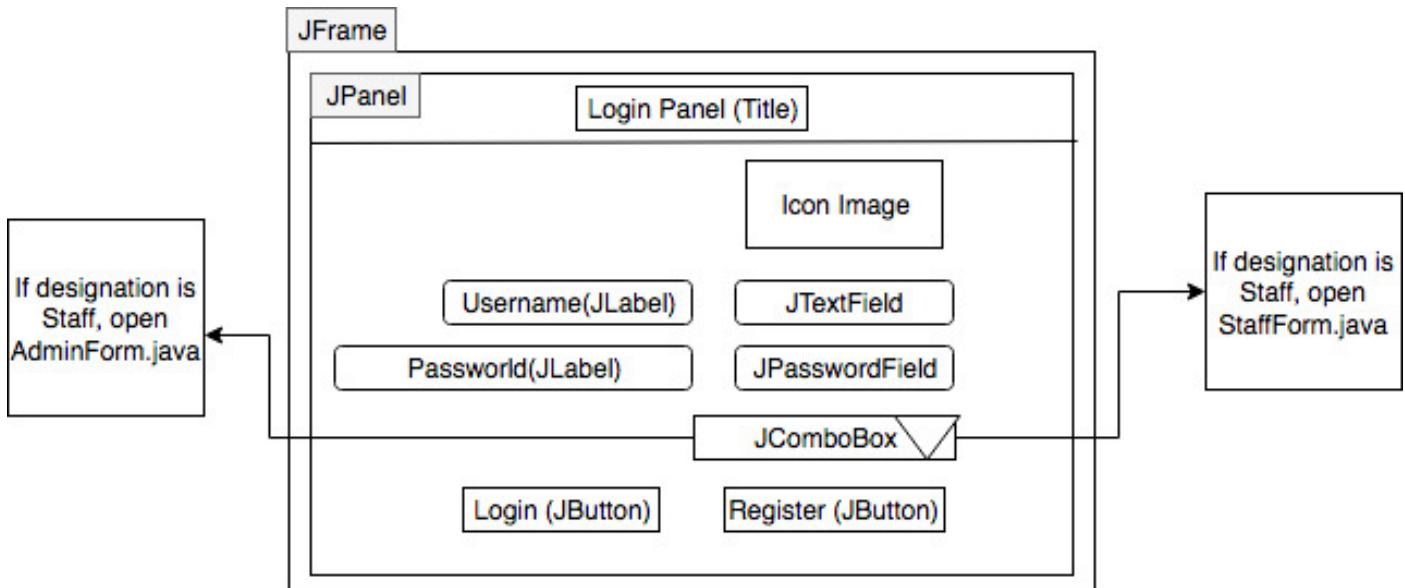


Table Design for MySQL

Field Name	Datatype
Username – (Key Field)	Text
Password	Text
Staff / Username	Text

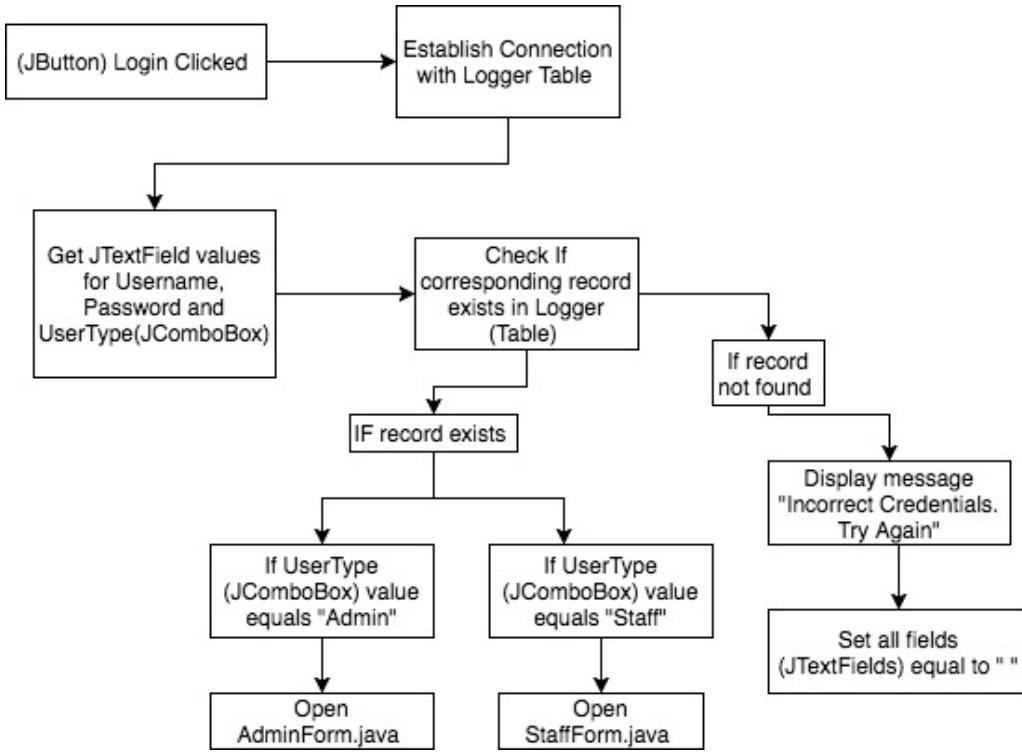
Design Layout



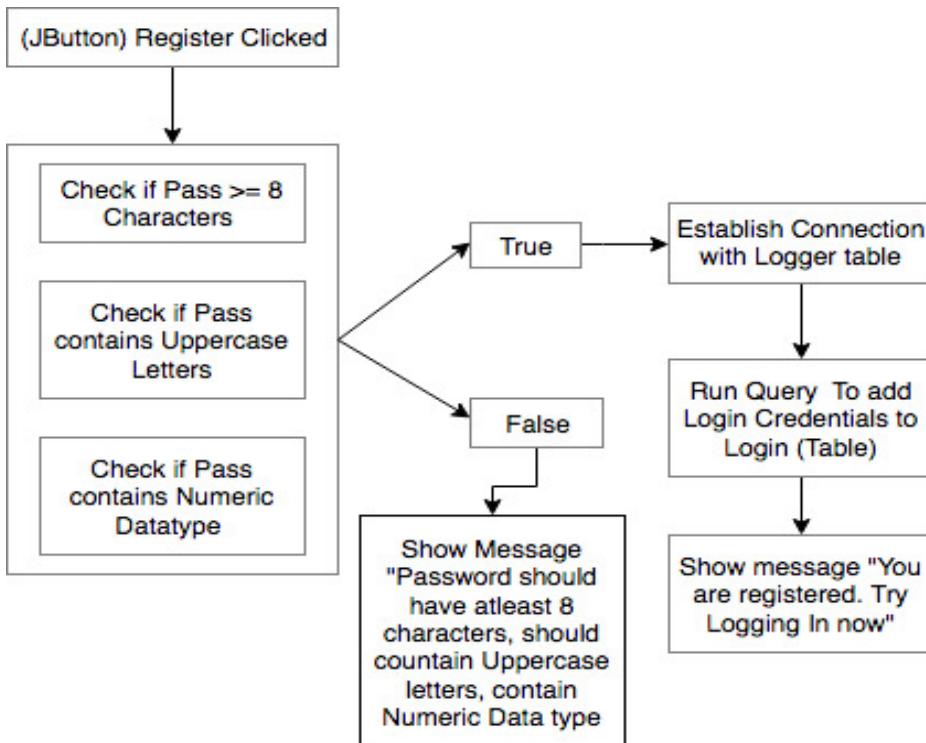
Styling Format

Container	Style
JPanel	White Background, Black Line Border
JLabel	Times, 14, Bold Italic
JButton	Times, 13, Bold
JComboBox	Helvetica, 13, Italic

Algorithm for Login Button



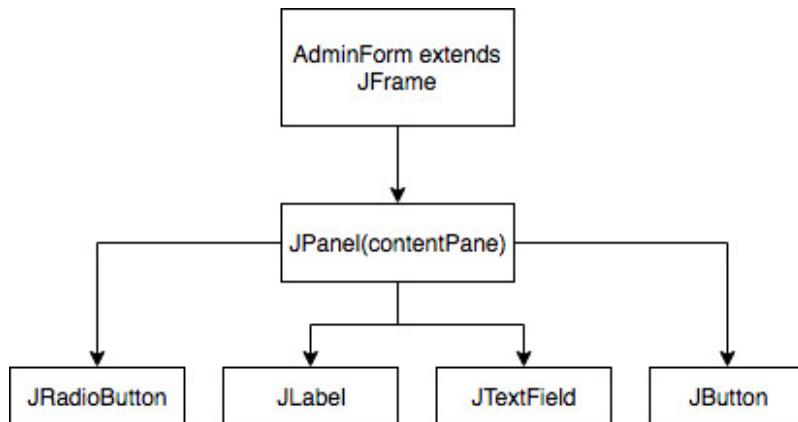
Algorithm for Register Button



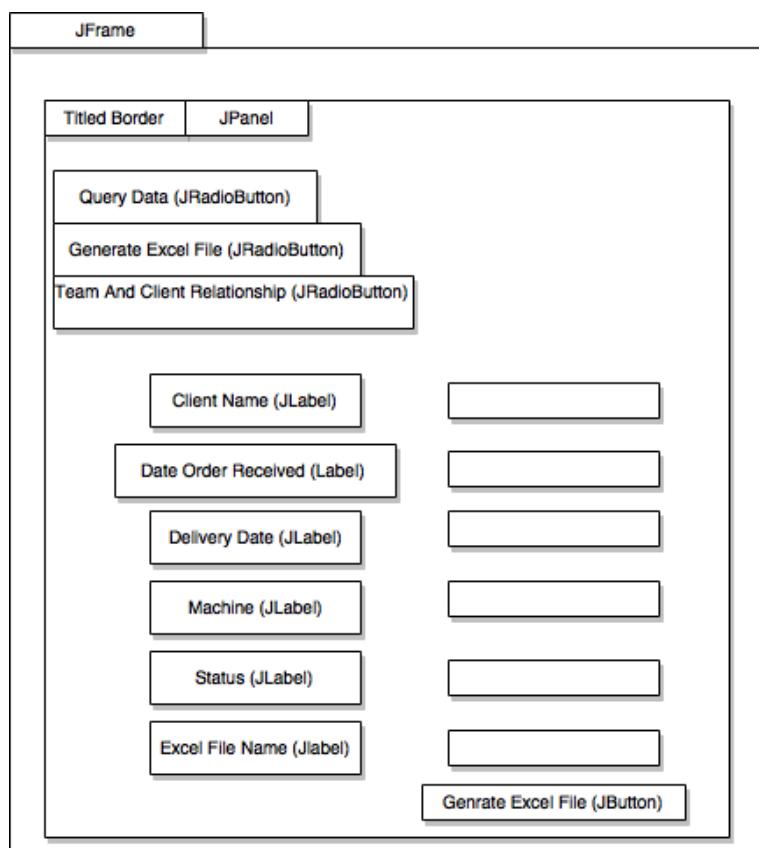
AdminForm

The following form is displayed when the user chooses the Admin as his/her designation in the Controller. The following form allows the user to run queries on the existing values of their database and then generate a tabular report in the form of an xml file. This form also allows the user to generate representation of all the values in their database.

Layout for Java Design



Design Layout



Styling Format

Container	Style
JPanel	White Background, Titled Border
JRadioButton	Times, 14, Bold Italic
JLabel	Times New Roman, 13, Bold
JComboBox	Helvetica, 13, Italic
JButton	Times, 14, Bold Italic

Queries

Table Name = SisClient

Query: Find all Records Matching the Client Name specified by the user

Field Name For Query	Query Used
Client Name = 'XNAME'	Select (*) from SisClient where Client Name = 'XNAME'

Query: Find all Records Matching the Date Order Received specified by the user

Field Name For Query	Query Used
Job_Date = 'YYYY/MM/DD'	Select (*) from SisClient where Job Date= 'YYYY/MM/DD'

Query: Find all Records Matching the Delivery Date specified by the user

Field Name For Query	Query Used
Delivery_date = 'YYYY/MM/DD'	Select (*) from SisClient where Delivery_date= 'YYYY/MM/DD'

Query: Find all Records Matching the Machine specified by the user

Field Name For Query	Query Used
Machine = 'XMACHINE'	Select (*) from SisClient where Machine= 'XMACHINE'

Query: Find all Records Matching the Status specified by the user

Field Name For Query	Query Used
Status = 'XSTATUS'	Select (*) from SisClient where Status= 'XSTATUS'

Team and Client Form

The diagram illustrates the layout of the Team and Client Form. It features a main window labeled 'JFrame' at the top. Inside, there is a panel labeled 'JPanel'. Within this panel, several components are arranged: a 'JTextField' labeled 'Enter File Name (JTextField)', a 'JButton' labeled 'Create txt File for Team and Client Relationship (JButton)', a 'JTextField' labeled 'Employee Name (JTextField)', a 'JTextField' labeled 'File Name (JTextField)', a 'JButton' labeled 'Search (JButton)', a 'JButton' labeled 'Back (JButton)', and a large 'JTextField' labeled 'RESULTS (JTextField)' at the bottom.

Styling Format

Container	Style
JPanel	White Background, Titled Border
JLabel	Times New Roman, 13, Bold
JButton	Times, 14, Bold Italic

PSEUDOCODE FOR BUTTON 'CREATE TEXT FILE FOR TEAM AND CLIENT RELATIONSHIP'

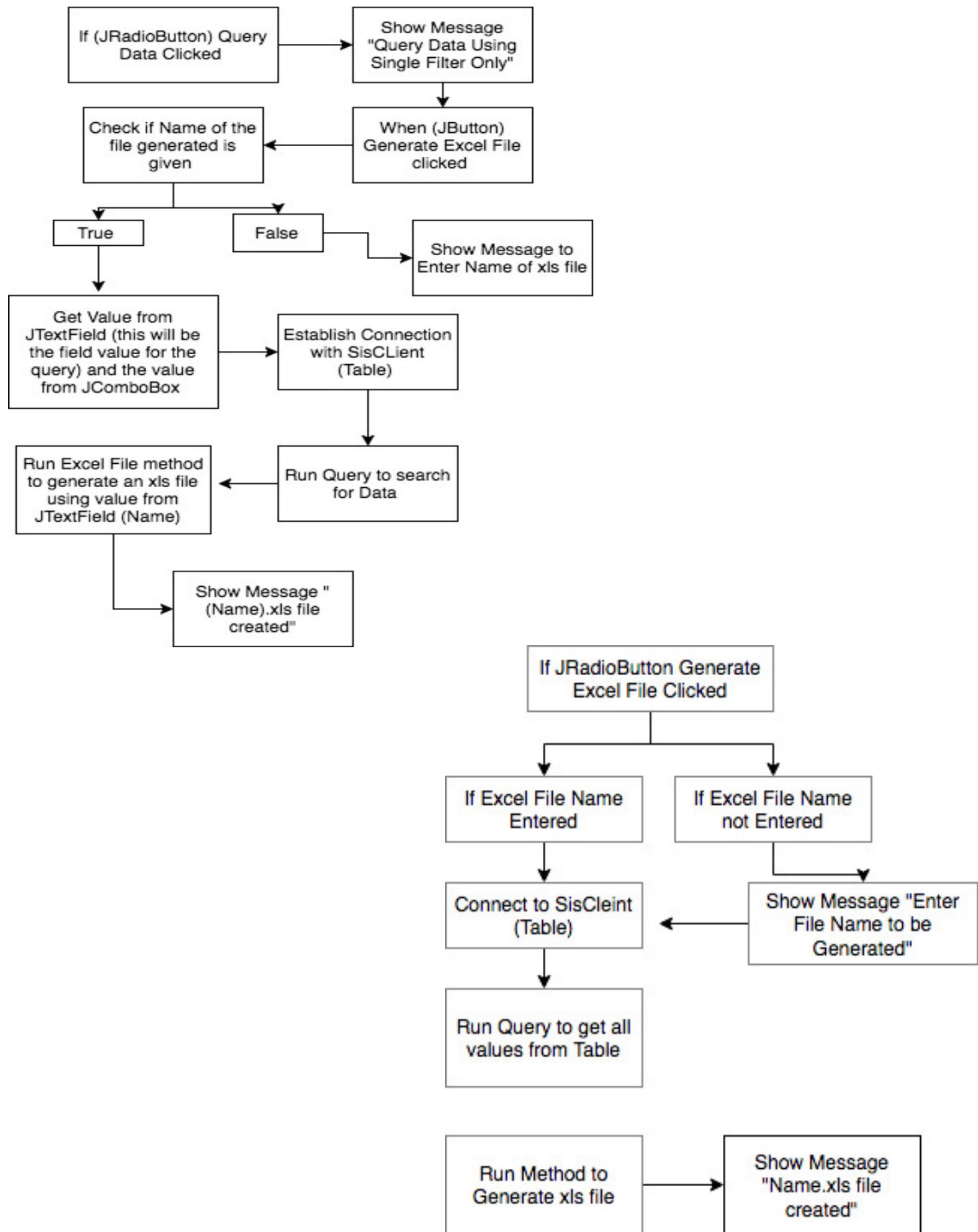
```
if (file name entered)
{
    connect to database
    left join sisclient with teams where
        sisclient.clientname=teams.team_leader_name
    create text file using fileoutputstream
    jpanel.showmessage("text file created")
}
else
{
    jpanel.showmessagedialog("enter file name")
}
```

PSEUDOCODE FOR SEARCH BUTTON

```
string method_parse file
{
    string ret
    string client[ ]
    list client_name
    string line
    string arr[ ]
    string file=get file name from jTextField
    if(file.exists and file.isDirectory)
    {
        line = scan next line from text file
        line=line.split(;)
        arr[ ]=line;
    }
    string empl_name = get text from employee name
    loop count from 0 to arr.length()
    {
        if(arr[ ].equals(empl_name))
        {
            client[ ]= scan client name from text file
        }
        else
        {
            ret = 'enter valid employee name'
        }
    }
    else
    {
        ret= 'enter valid file name'
    }
}
return ret
}
```

The user has two options either to select “Query Data” or “Generate Excel File” IN ADMINFORM

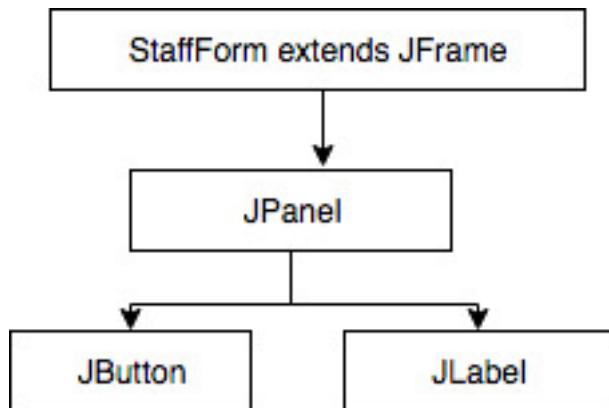
Flowchart Design for Button Functions



StaffForm

The following allows a staff member to Add a new set of records or Update/Delete an existing set of records from the database. The staff member clicks a button dependent on their preferred aim. Each button leads to a new Form.

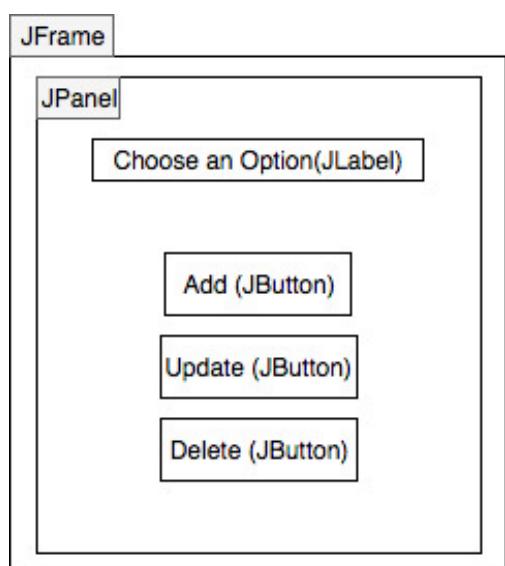
Layout for Java Design



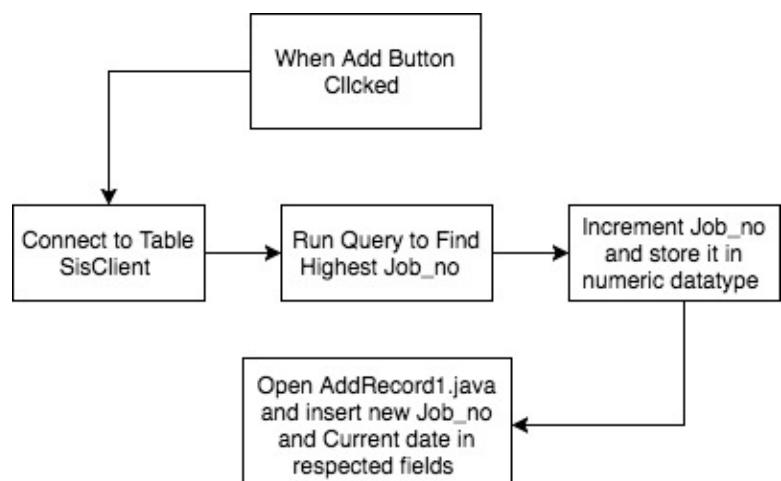
Styling Format

Container	Style
JPanel	Background White
JLabel	Times, 14, Bold
JButton	Times, !4, Bold Italic

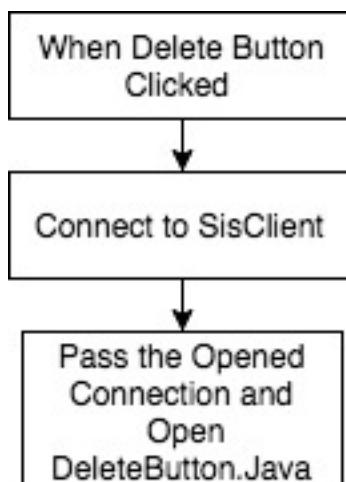
Design Layout



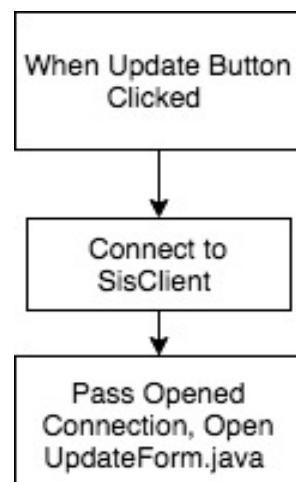
Algorithm for AddButton



Algorithm for Delete Button



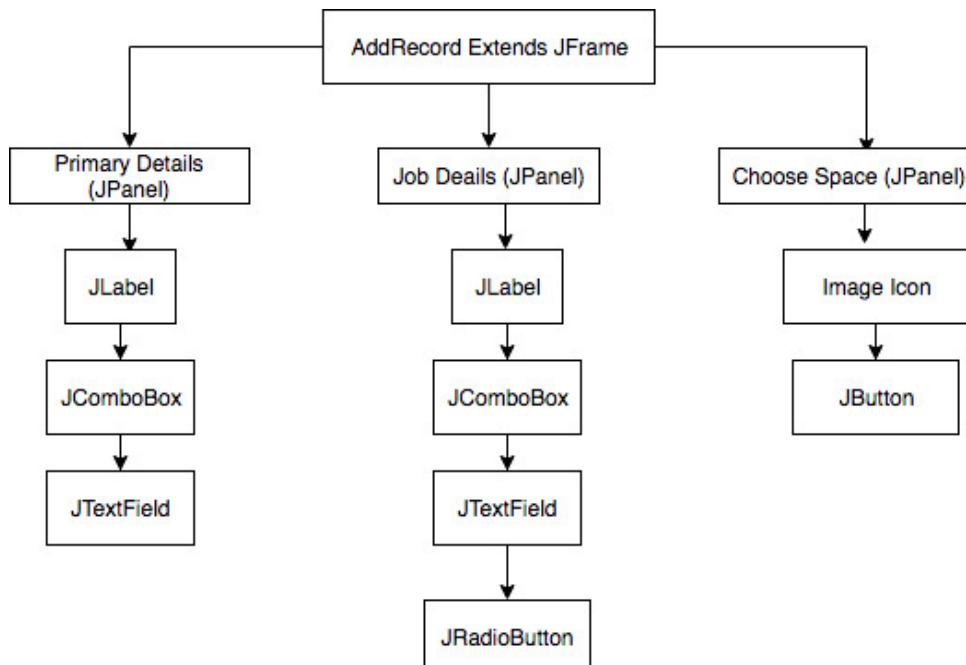
Algorithm for Update Button



ADDRecordForm

The following form allows the staff member to Add a record to the existing table. To do this, the data entered in the field must be of correct format and must not be left empty. The following form also allows the staff member to generate an xml file for the table in the database. For the sake of east navigation, a button which allows the staff member to go back to the previous form will be added.

Layout for Java Design



Design Layout

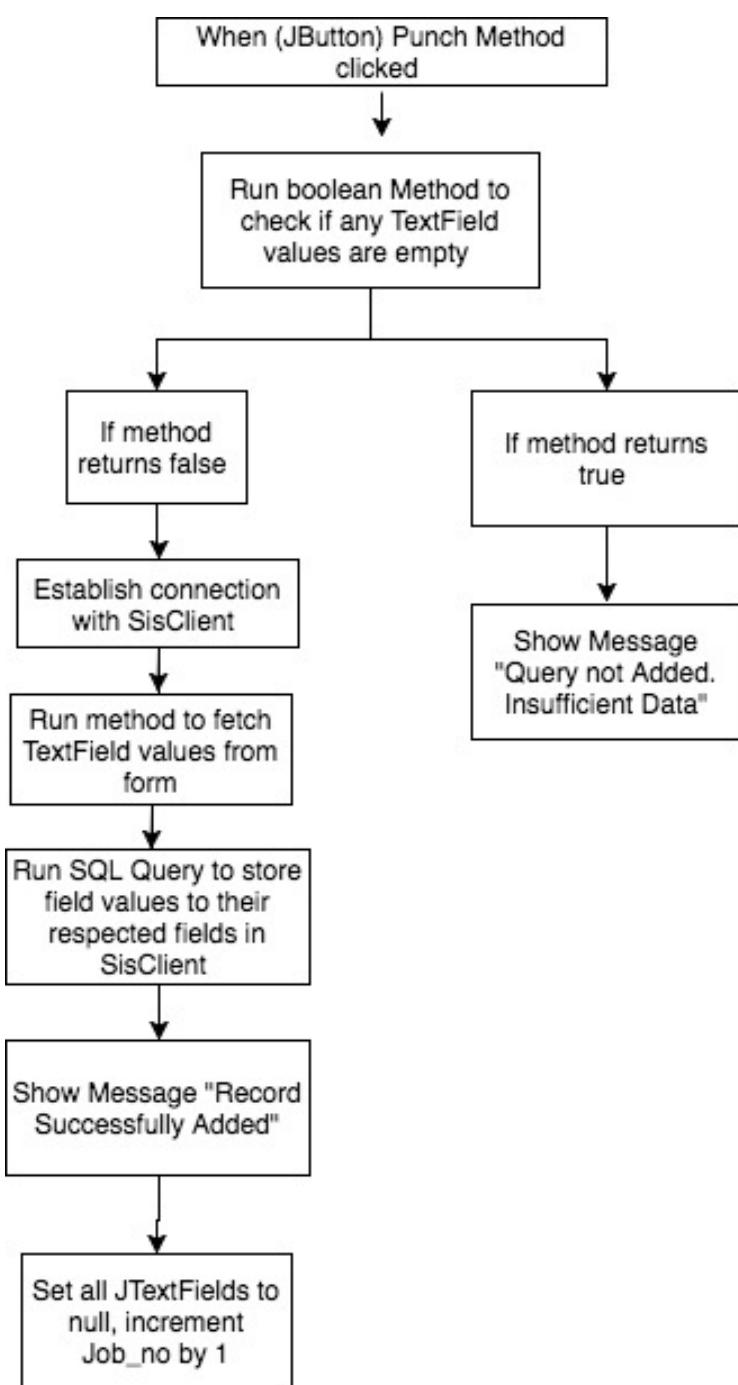
The screenshot shows the visual representation of the ADDRecordForm. It consists of three vertically stacked panels, each with a titled border and a JPanel header:

- Primary Details (Top Panel):** Contains fields for 'Job No (Jlabel)', 'Date(JLabel)', 'Client Name (JLabel)', 'JComboBox', and 'Job Name (JLabel)'.
- Job Deals (Middle Panel):** Contains fields for 'No of Forms (Jlabel)', 'Print Style (JLabel)', 'JComboBox', 'No. of Colors (Jlabel)', 'Paper Quantity(JLabel)', 'Plate (JLabel)', 'JComboBox', 'Paper Print (Jlabel)', 'Book Size(JLabel)', 'No of Pages(JLabel)', 'Old Plate(JRadio)', 'New Plate(JRadio)', 'Paper Required (Jlabel)', 'Paper Supp(JLabel)', 'Paper in Sheets (Jlabel)', 'Finish Quantity (JLabel)', 'Delivery Date (JLabel)', 'Status (JLabel)', 'JComboBox', and 'Machine (JLabel)', 'JComboBox'.
- Choose Space (Bottom Panel):** Contains an 'Icon Image' placeholder, and buttons for 'Punch Record (JButton)', 'Back (JButton)', and 'Generate Excel File (JButton)'. Below the icon image are three rows of three 'JLabel' components each.

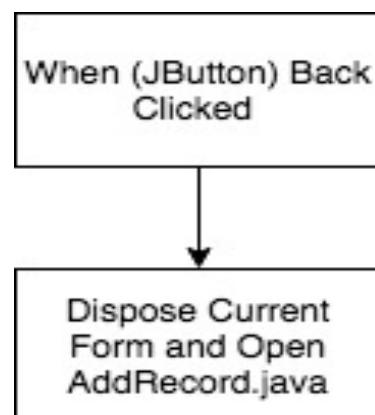
Styling Format

Container	Style
JPanel	Titled Border
JLabel	Times New Roman, 14, Bold
JRadioButton	Times, 14, Bold, Italic
JComboBox	Helvetica, 14, Italic
JButton	Lucida Grande, 13, Plain

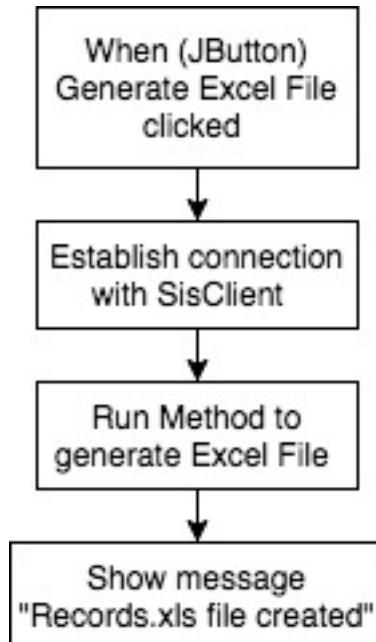
Algorithm for Punch Method Button



Algorithm for Back Button

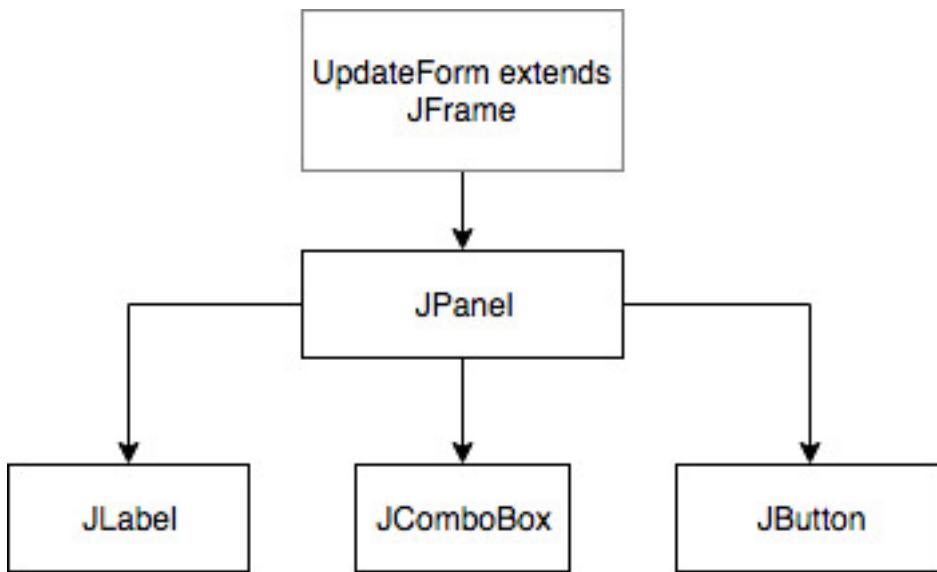


Algorithm for Generate Excel File

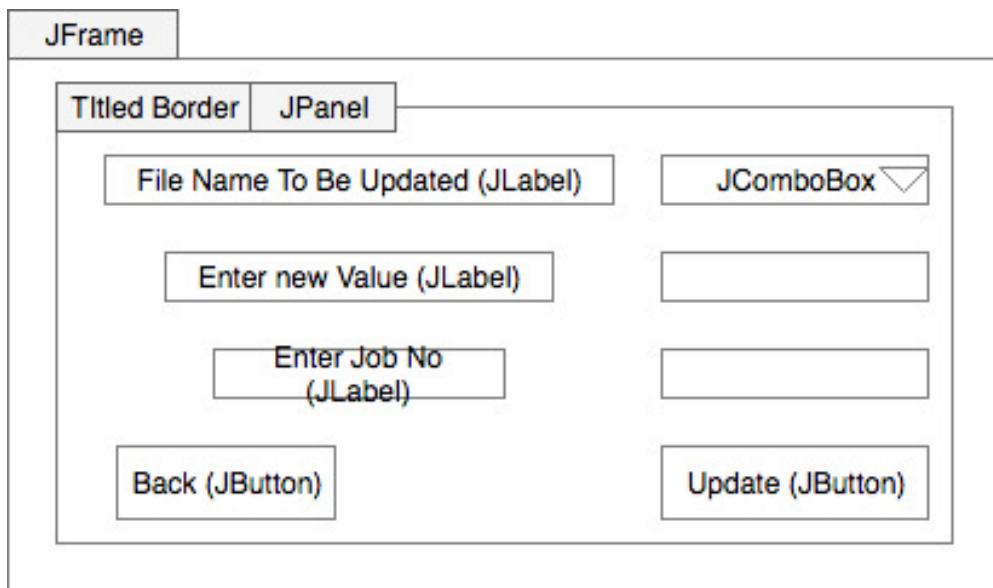


Update Record Form

The following form allows the user to update a particular value of a specified field from the database. For the sake of easy navigation, a back button has also been provided. The value that needs to be updated is first checked for its presence and then updated if in correct format. The Job_no for the record must be provided as it will be used to find the unique set of values that need to be changed.



Design Layout



Styling Format

Container	Style
JPanel	Title Border, Background: White
JLabel	Times, 14, Bold
JButton	Times New Roman, 14, Bold, Italic
JComboBox	Helvetica, 13, Italic

Query Explanation

Query: Update Field A, change value from X to Y, for Job_no Z

Query Used

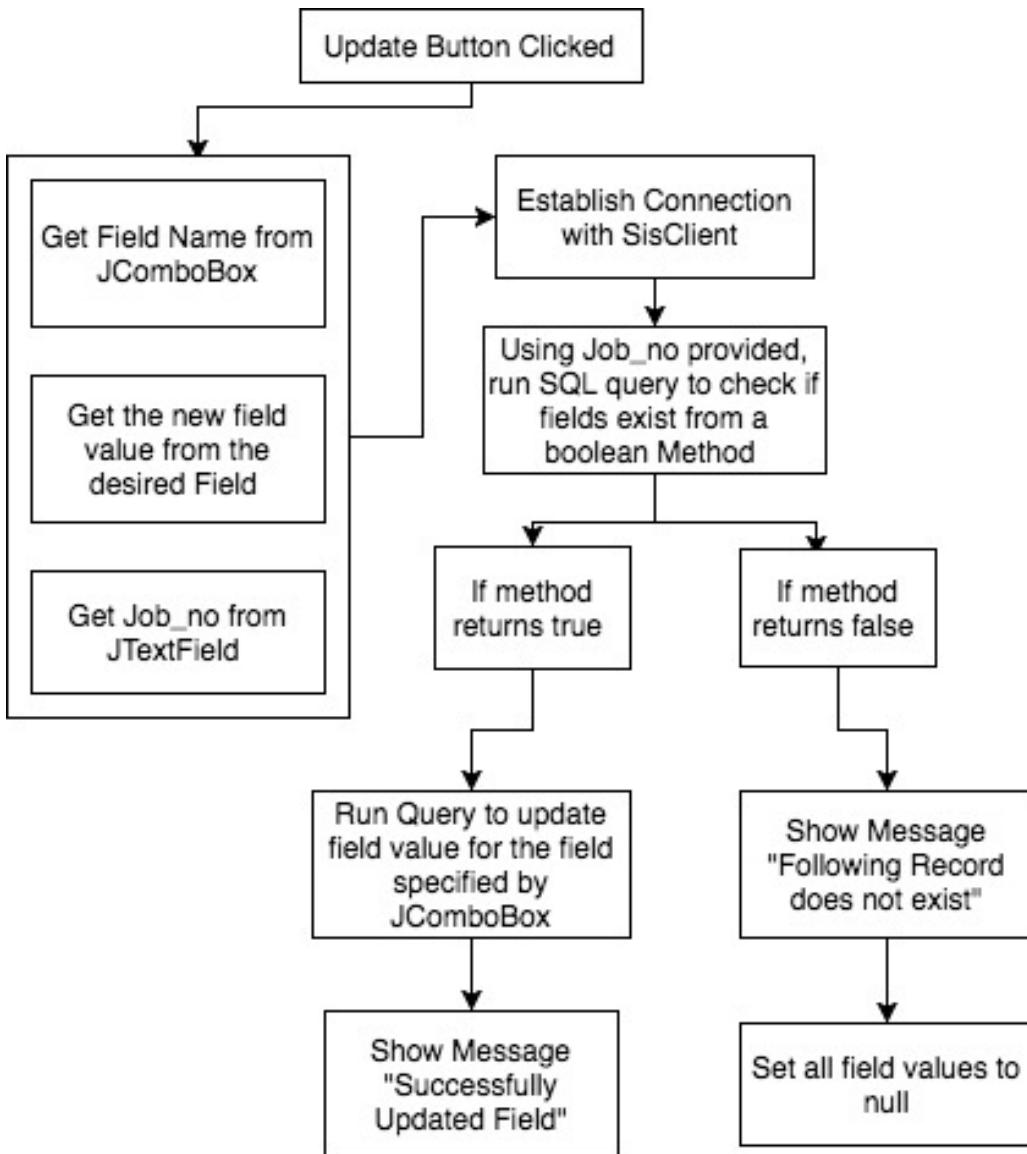
Update SisClient set A=Y where Job_no = Z

- Suppose the Status for Job_no 4 needs to be changed from 'Pending' to 'Ready'. The Query used for this would be as follows

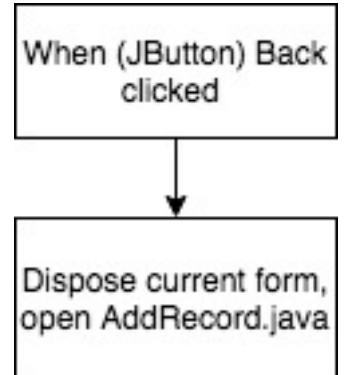
Query Used

Update SisClient set Status='Ready' where Job_no = 4

Algorithm for Update Button

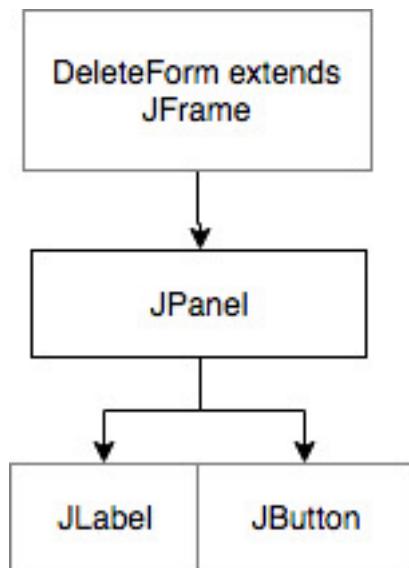


Algorithm for Back Button

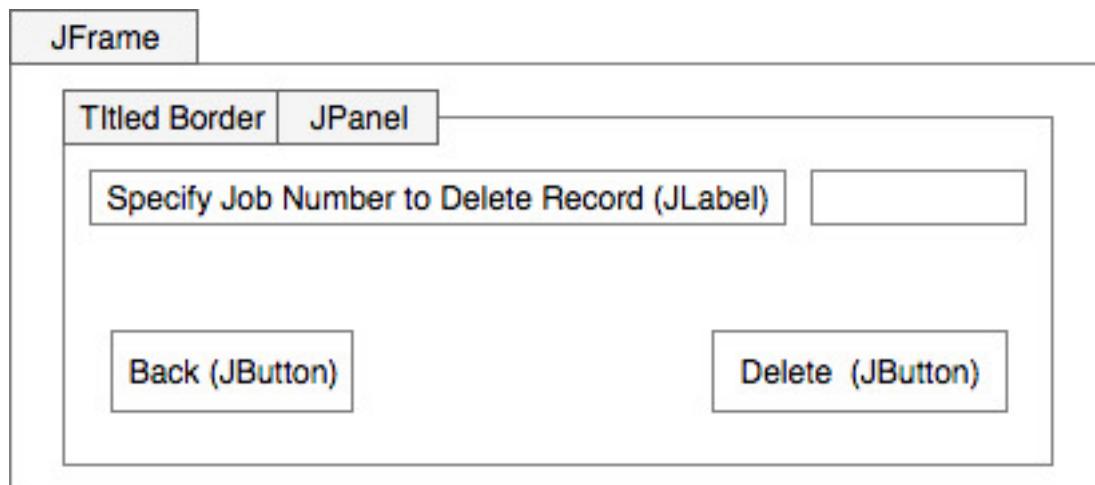


Delete Record Form

The following form allows the staff member to delete a particular record from the database. The staff member must enter the Job_no so that the required unique record is identified and deleted.



Design Layout



Styling Format

Container	Style
JPanel	Titled Border, White Background
JLabel	Times, 14, Bold
JButton	Times New Roman, 13, Bold Italic

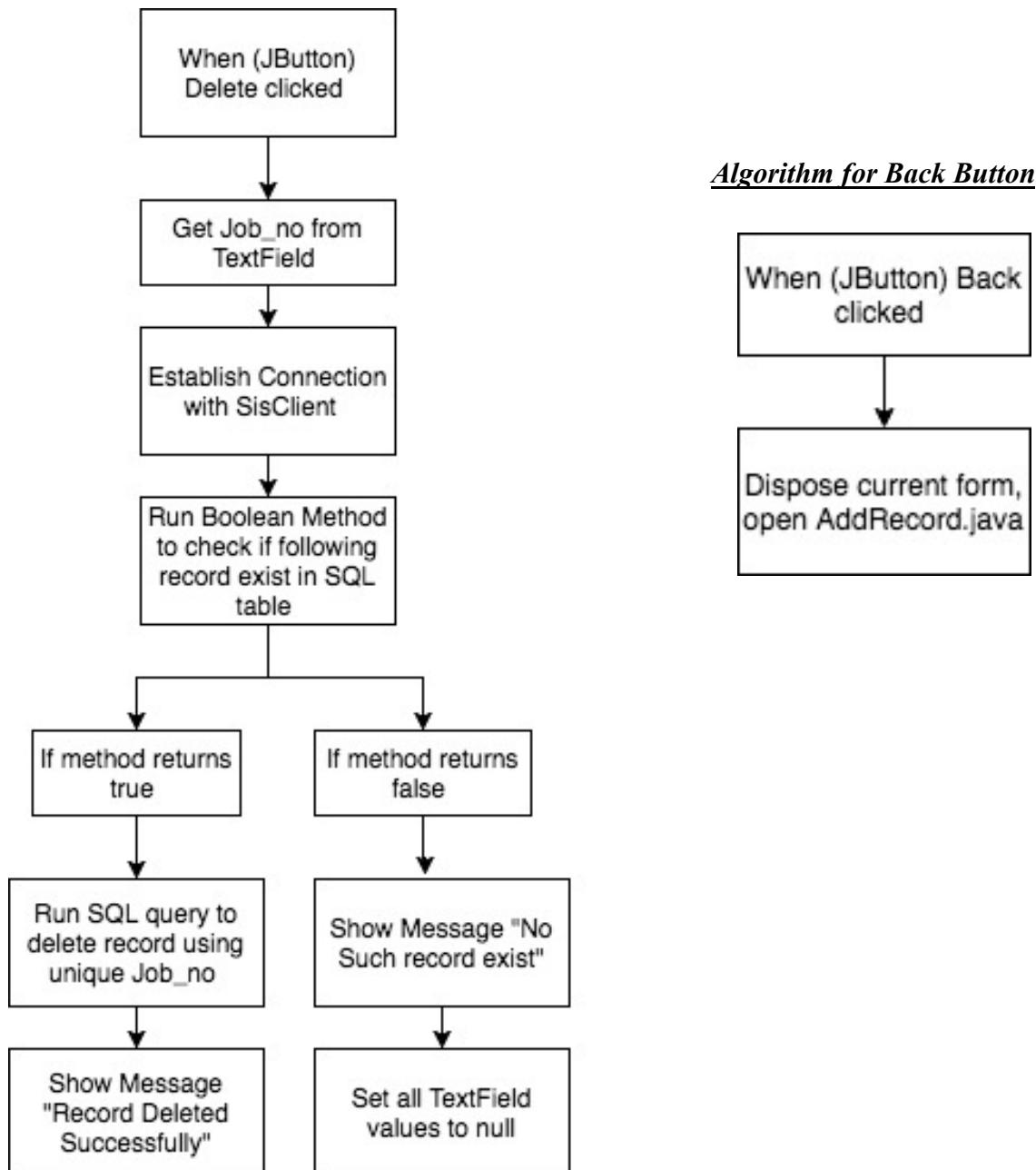
Query Explanation

Query Used
Delete from SisClient where Job_no = 'specified value'

- **Suppose the staff member wishes to delete the record with Job_no =5**

Query Used
Delete from SisClient where Job_no = 5

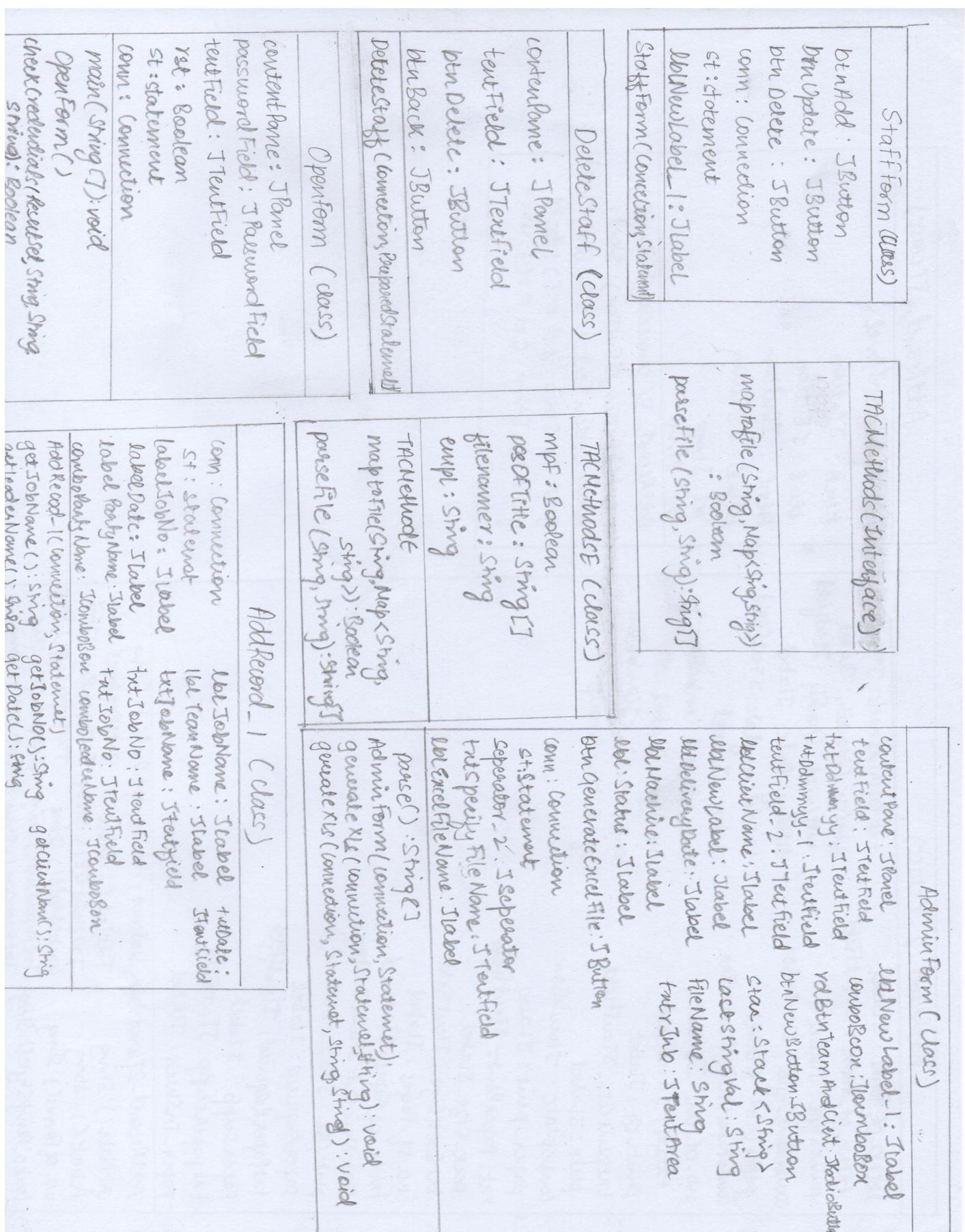
Algorithm for Delete Button



UML Diagram

Col 1- variables (private and public)

Col 2- methods and constructors



AddRecord - 2 (class)

```

var NO_OF_FORMS : JLabel
txtNO_OF_FORMS : JTextField
printStyle : JLabel
cmbboPrintStyle : JComboBox
paperUsed : JLabel
cmbboPaperUsed : JComboBox
NO_OF_VOL : JLabel
txtNO_OF_VOL : JTextField
print - Qt : JLabel
txtprintQt : JTextField
plate : JLabel
cmbboPlate : JComboBox
paper - print : JLabel
txt - paperPrint : JTextField
book - size : JLabel
txtBookSize : JTextField
no_of - Pages : JLabel
txtNo_of_Pages : JTextField
txtNo_of_Pages : JTextField
plateNewOld : JRadioButton
plateOld : JRadioButton
paperRequired : JLabel
txtPaperRequired : JTextField
paperSupp : JLabel
txtPaperSupp : JTextField
paper_inStock : JLabel

AddRecord - 2 (connection, statement) - cmbboPaperUsed : String
getPlate() : String
TxtLabel() : String
TxtprintQt() : String
TxtBookSize() : String
TxtNoOfPages() : String
TxtPaperSupp() : String
TxtInchQtr() : String
TxtPaperPrint() : String
PlateOld() : String

```

AddRecord - 3 (class)

```

contentPane : JPanel
btnF : JButton
btnB : JButton
btnC : JButton
btnD : JButton
al : String
a2 : String

AddRecord - 3 (connection, statement)
resetJButton, JButton) : void,
return JButton reset - 1 (JButton) : void
getA1() : String set A1() : void
getA2() : String set A2() : void

```

Overall Test plan

Test Type	Nature of test	Example
Software inserts, updates and deletes entries graphically from and to the database <i>(success criteria 1)</i>	After entering all the required fields in the AddRecord form the staff is able to make changes to the database including adding, editing fields in the tables. - Visual check for success message or error message when fields are missing.	After clicking button to update database table, the user should get a message of the following action has been completed or not.
Software should authenticates the user before allowing access to make changes to existing data <i>(success criteria 2)</i>	The username and password entered by the user should match with that stored in the database. If match is found access is granted. If not, then access is denied (error message should be displayed). - Visual check	
Software should contain validation checks on password when registering a new user. <i>(success criteria 6)</i>	When a new user is registered, the software should make sure that the password entered contains uppercase letters and the length is at least 8 characters. - Visual check for error message	
The software should take all the details of the orders that are given by the clients. The fields used in the form to take orders, must be checked for simple, extreme and inappropriate data <i>(success criteria 3)</i>	The form developed to insert order details should be detailed enough so that when excel file is created from the database, it ensures accurate information transfer to the person viewing the file. - Visual check - Excel file created	Field that are used to fill in the delivery date of the order, should ensure that the date entered by the user is in correct format, i.e. it should be checked against simple and inappropriate data.
The software should allow the company staff to allocate storage for paper rims and bundles arriving at the premises <i>(success criteria 4)</i>	Interactive buttons, when pressed by the staff should refer to a storage space available for storing paper - Visual check	
The user, whether supervisor or the staff should be able to navigate between forms of the software <i>(success criteria 5)</i>	Buttons should be used so to provide easy navigation through forms and the software is more user friendly. - Visual check	
The staff should not be given access to the admin level of the software, therefore the software should be developed and tested in a way so that the staff can only make changes to the DB <i>(success criteria 7)</i>	The admin should be able to create text files and generate excel files by querying data, while the staff should be only be able to create excel files for all the entries in the DB. - Check for Text files and Excel files	