

Criterion B: Design

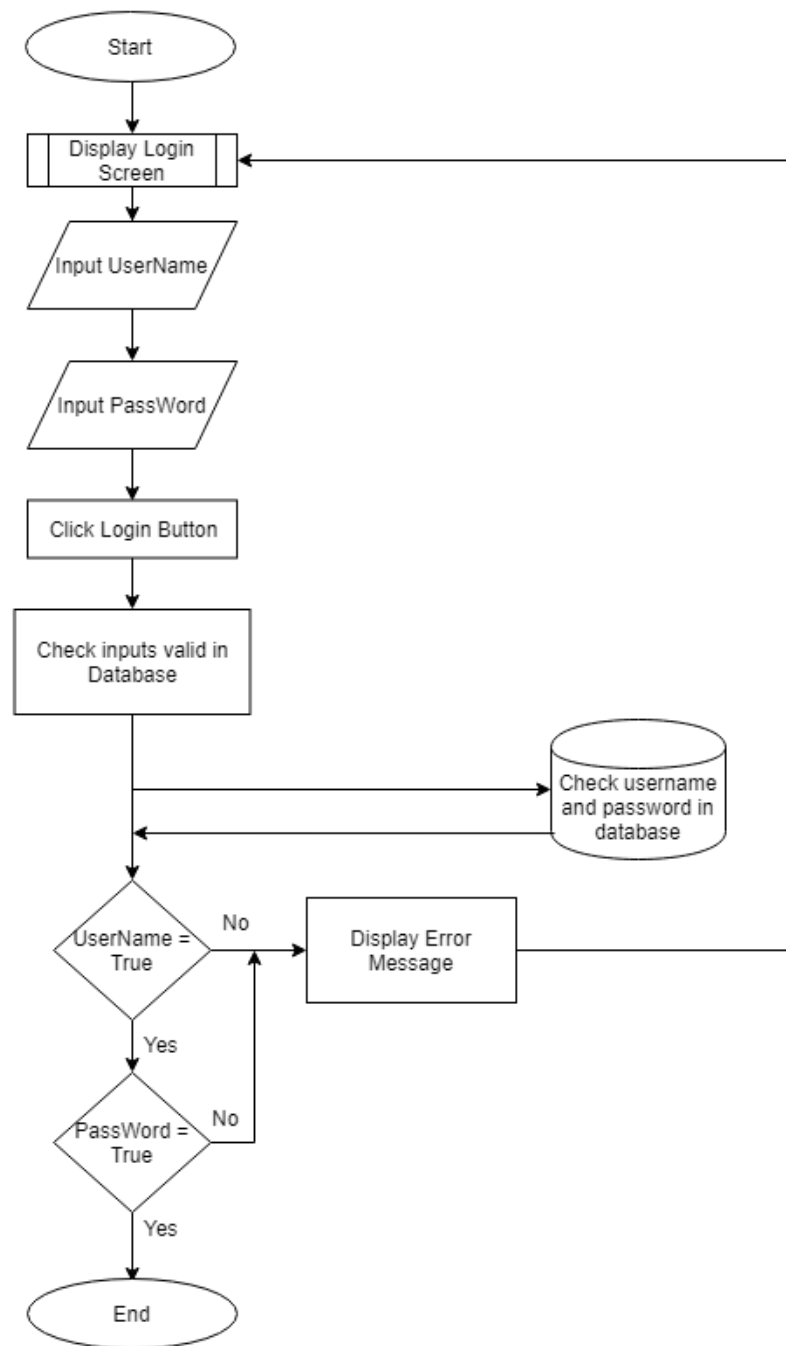


Figure 1.1 - Flowchart for login system

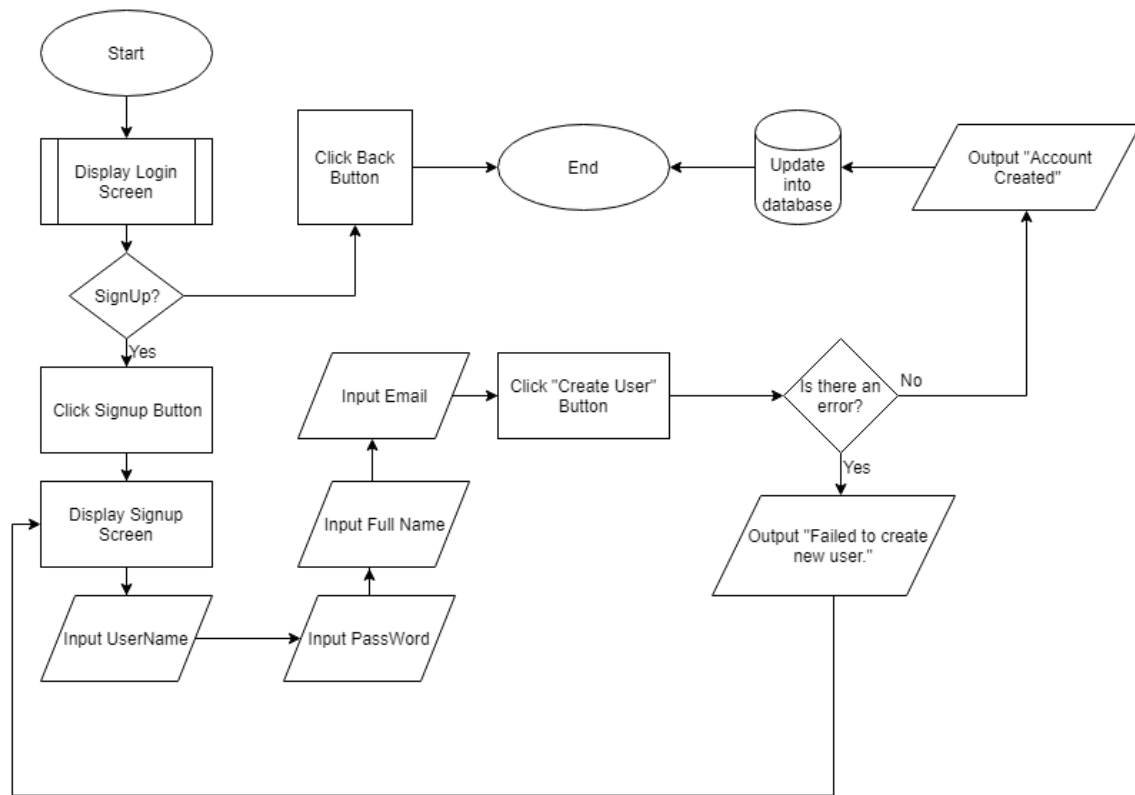


Figure 1.2 - Flowchart for the signup process

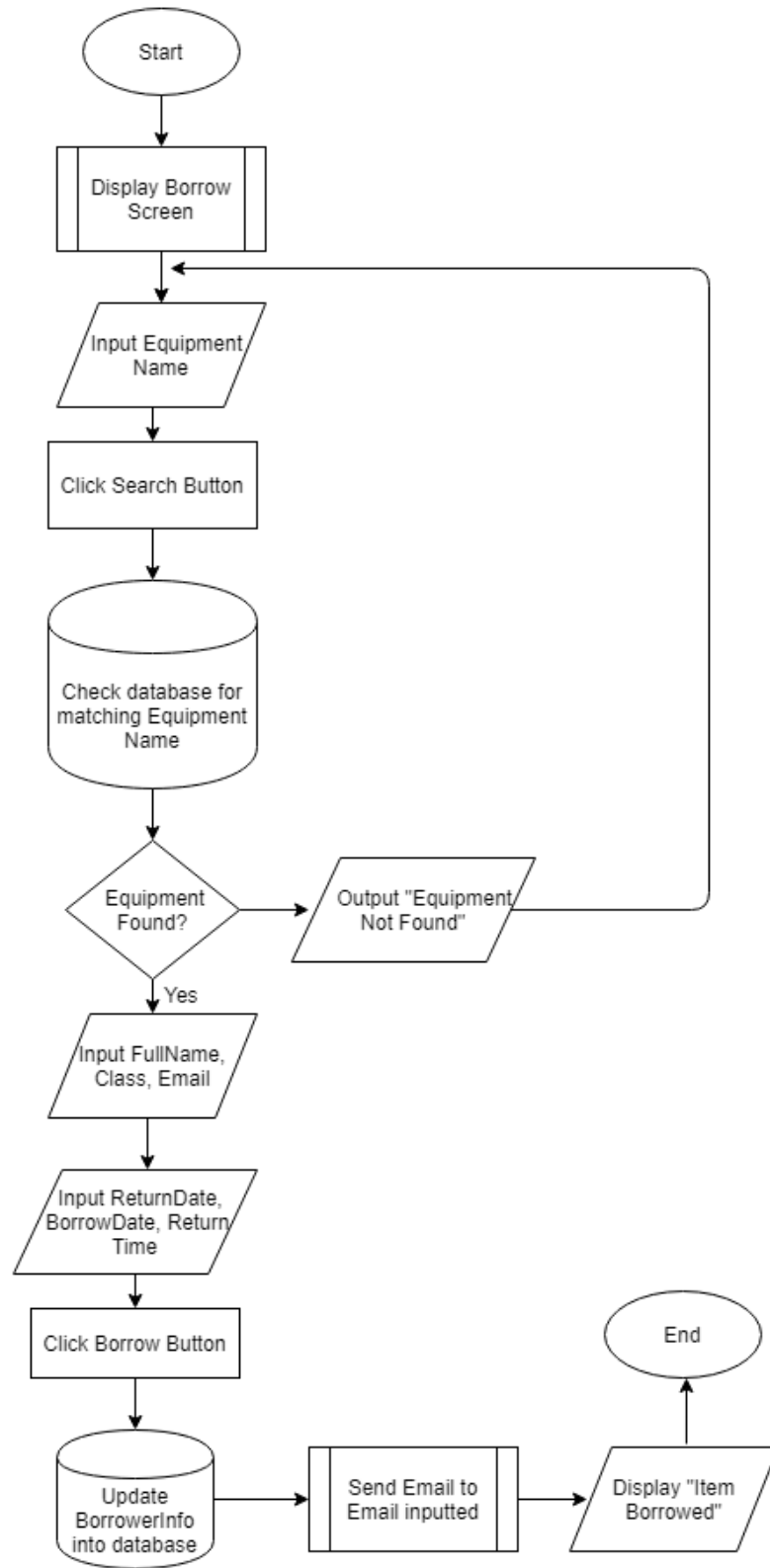


Figure 1.3 - Flowchart for Borrowing Process

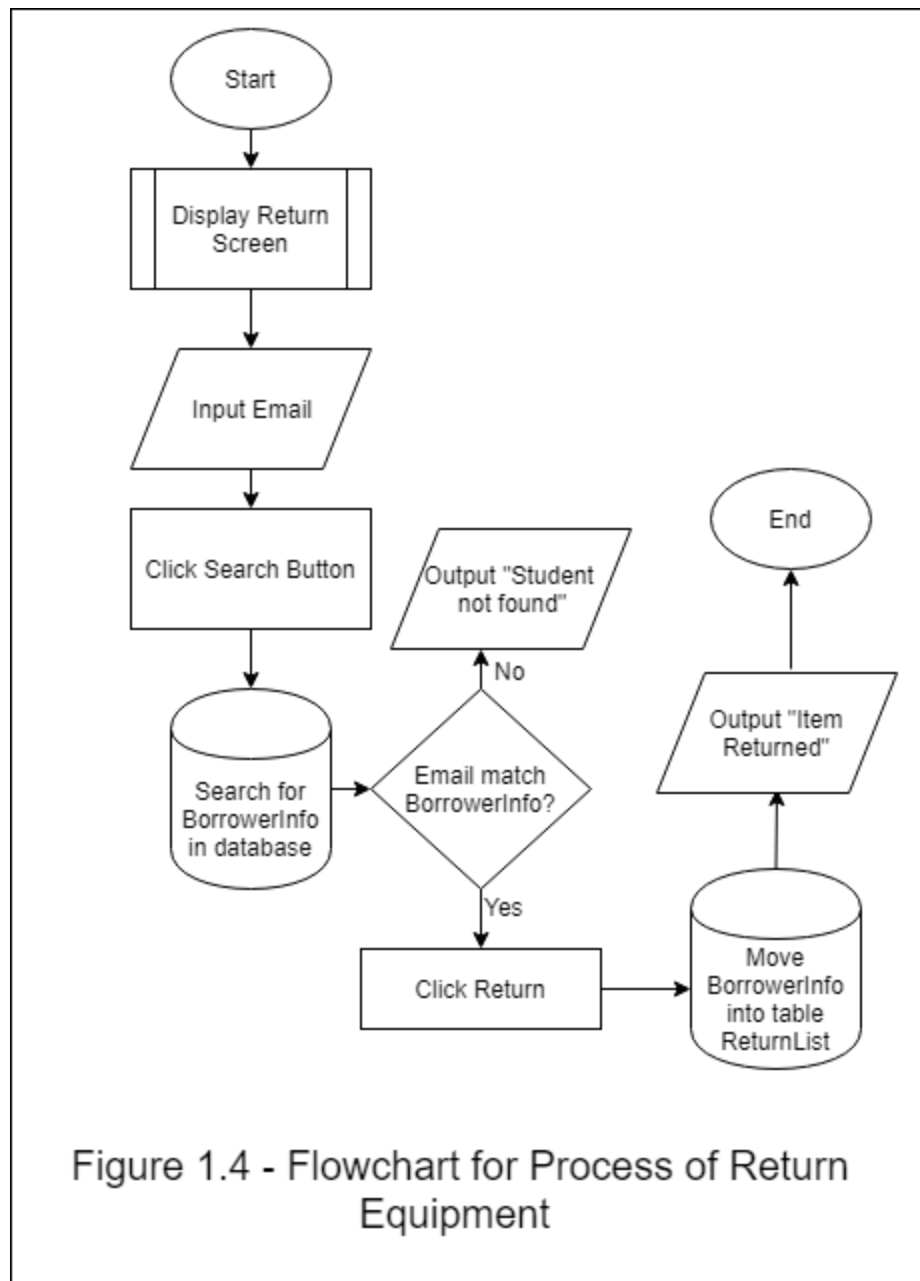


Figure 1.4 - Flowchart for Process of Return Equipment

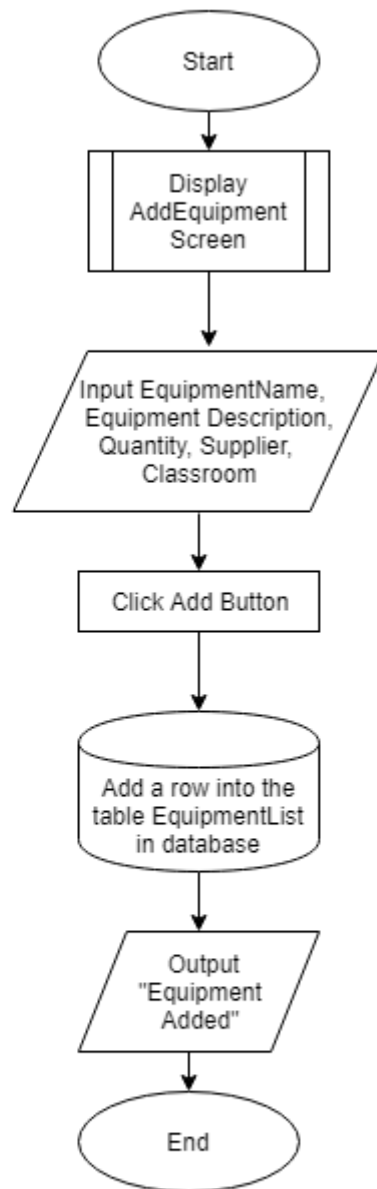



Figure 1.5 - Flowchart for adding new equipment

Equipment Management System

EQUIPMENT MANAGEMENT SYSTEM v1.0

By: Nguyen Bao Son Y13I

 Username:

Password:

Figure 2.1 – Login Screen

Equipment Management System

Create A New Account

Username:

Password:

Full Name:

Email:

Figure 2.2 – Signup Screen

Equipment Management System

EQUIPMENT MANAGEMENT SYSTEM v1.0

Borrow Equipment

Return Equipment

Borrowed List

Equipment List

Alert Student

Figure 2.3 – Main Dashboard

Equipment Management System

Borrowing an Equipment

Pencil
Pen
Colouring
Paintbrush
Eraser

Search

Date Borrowed: /
Return Time: Break
Return Date: /

Borrow

Back

Student Info

Full Name:

Email:

Class:

Search

Equipment Name:

Search

Click this button will allow this small window to popup

Figure 2.4 – Borrow Equipment Screen

Equipment Management System

Returning an Equipment

Email:

Equipment Info

Equipment Name:

Time Returned:

Borrowed Date:





Returned Date: / / 

Figure 2.5 – Return Equipment Screen

Equipment Management System   

Add Equipment

Equipment Name:

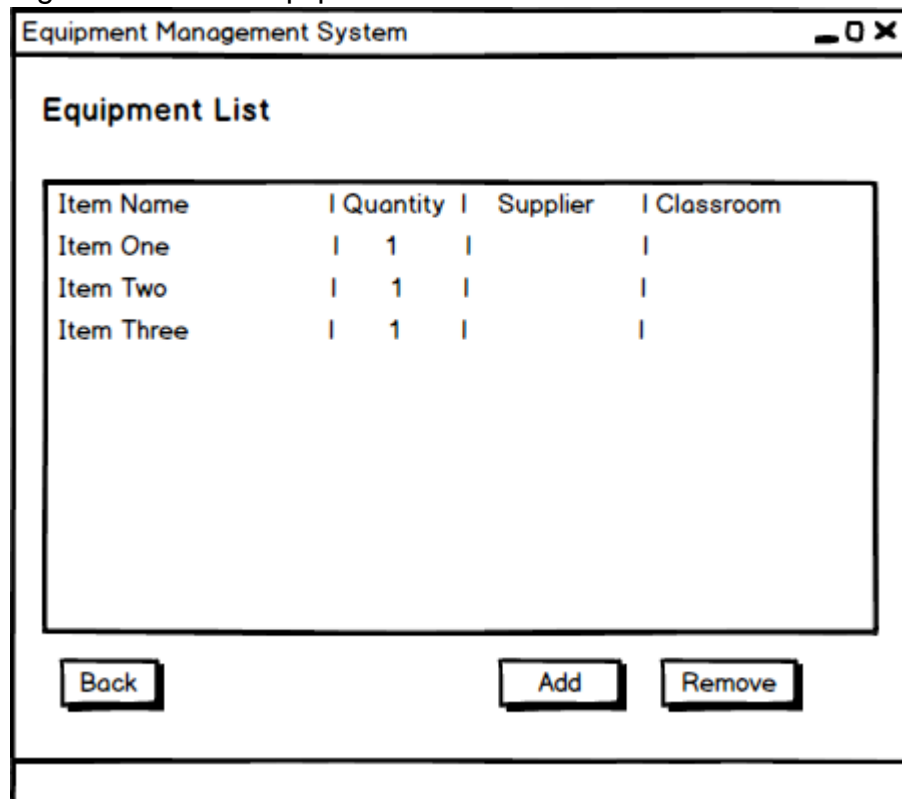
Equipment Description:

Quantity:

Supplier:

Classroom:

Figure 2.6 – Add Equipment Screen

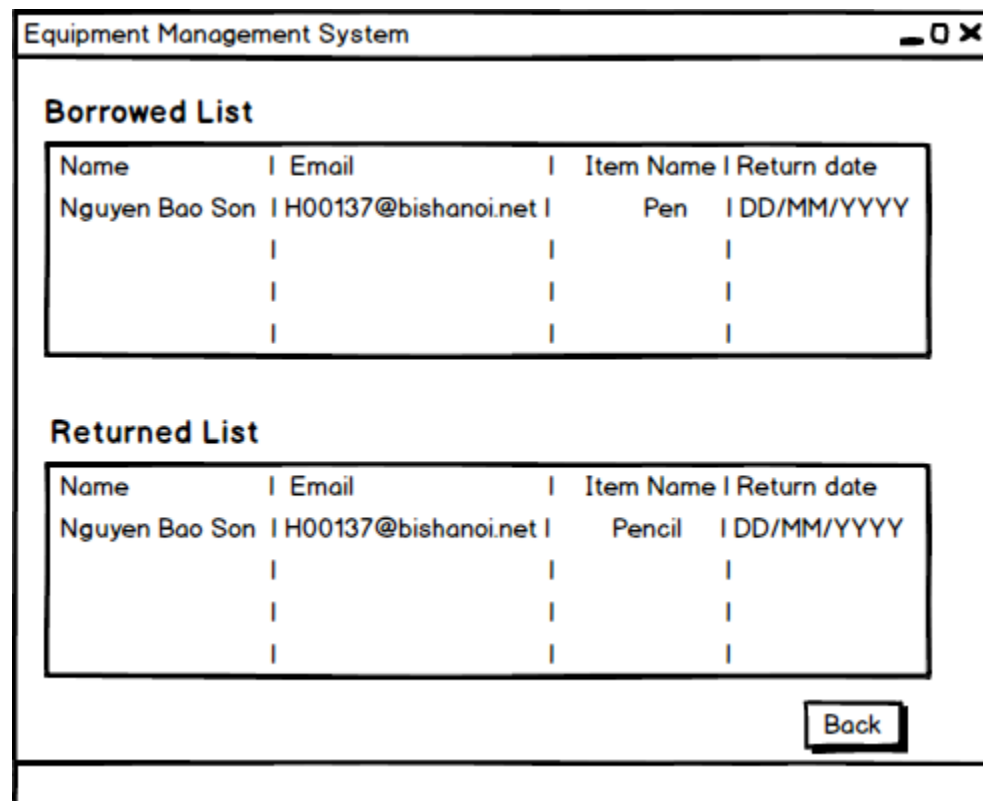


The screenshot shows a window titled "Equipment Management System" with a standard Windows-style title bar (minimize, maximize, close buttons). The main content area is titled "Equipment List". Below the title is a table with four columns: "Item Name", "Quantity", "Supplier", and "Classroom". The table contains three rows of data: "Item One" with quantity 1, "Item Two" with quantity 1, and "Item Three" with quantity 1. The "Supplier" and "Classroom" columns are empty. Below the table, there are three buttons: "Back", "Add", and "Remove".

Item Name	Quantity	Supplier	Classroom
Item One	1		
Item Two	1		
Item Three	1		

Back Add Remove

Figure 2.7 – Equipment List Screen



The screenshot shows a window titled "Equipment Management System" with a standard Windows-style title bar. The main content area is titled "Borrowed List". Below the title is a table with four columns: "Name", "Email", "Item Name", and "Return date". The table contains three rows of data: "Nguyen Bao Son" with email "H00137@bishanoi.net" and item "Pen", and two empty rows. Below the "Borrowed List" table is another section titled "Returned List". Below this title is another table with the same four columns: "Name", "Email", "Item Name", and "Return date". The table contains three rows of data: "Nguyen Bao Son" with email "H00137@bishanoi.net" and item "Pencil", and two empty rows. At the bottom right of the window, there is a "Back" button.

Name	Email	Item Name	Return date
Nguyen Bao Son	H00137@bishanoi.net	Pen	DD/MM/YYYY

Returned List

Name	Email	Item Name	Return date
Nguyen Bao Son	H00137@bishanoi.net	Pencil	DD/MM/YYYY

Back

Figure 2.8 – Borrowed List Screen

Equipment Management System

Alert Student

Student Email:

Figure 2.9 – Alert Student Screen

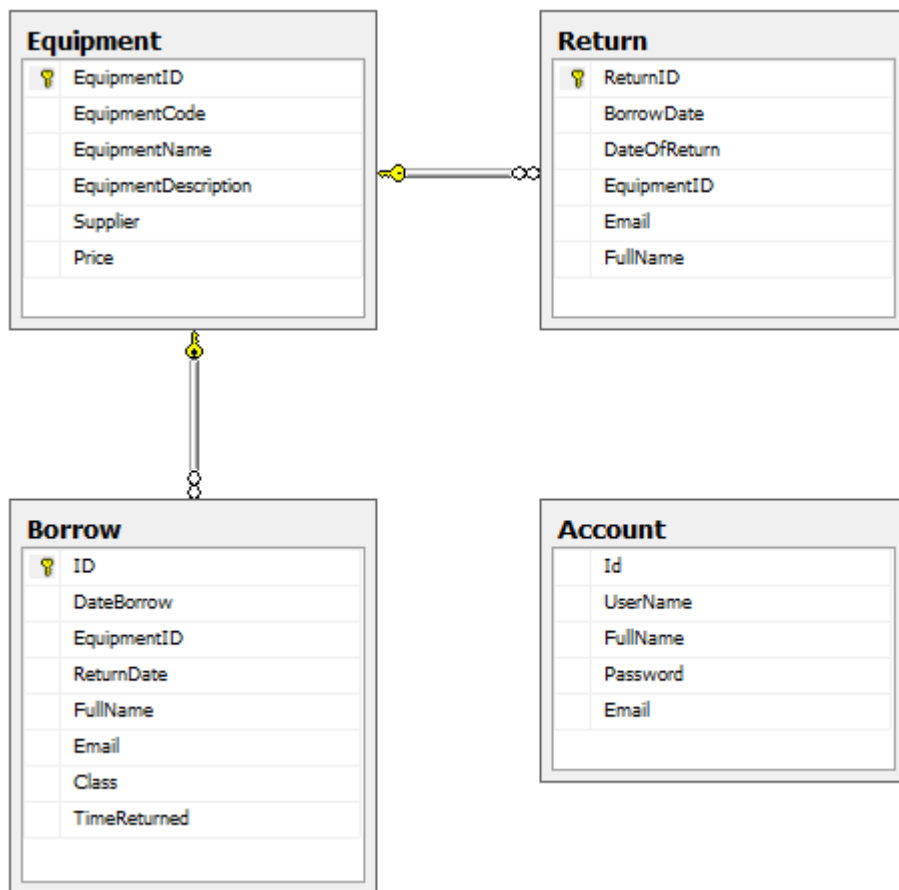


Figure 3.1 – Entity Relationship Diagram for database

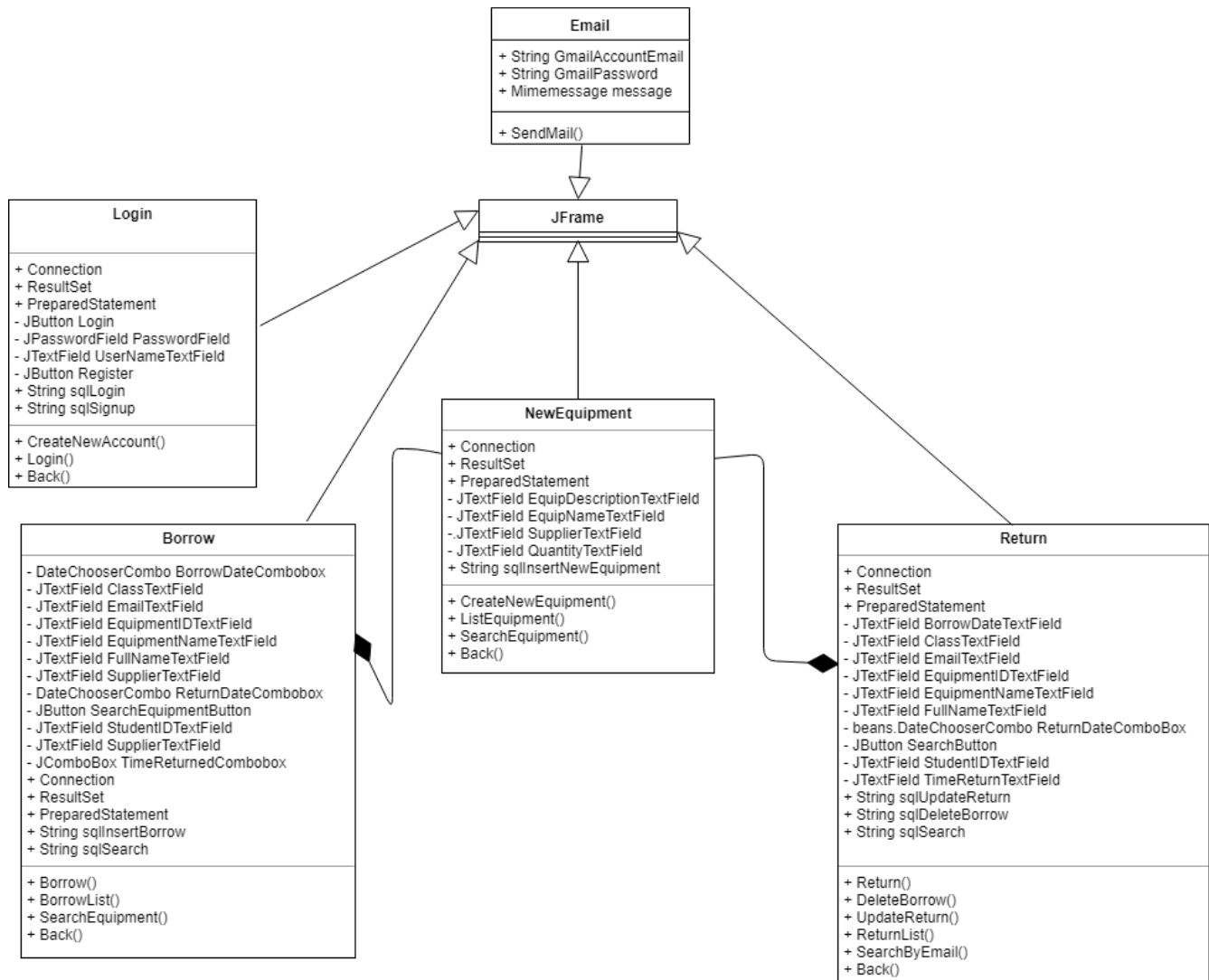


Figure 3.2 – UML Class Diagram

```
EquipmentList = new ArrayList()
EquipmentList = SELECT information from EquipmentTable in Database
output EquipmentList
```

Figure 4.1 – Pseudocode for the process of displaying Equipment List.

```
sum = 0
list = new ArrayList()
list = (SELECT Equipment Quantity From Database)
loop n from 0 to list.size()           //loop through the whole array list
    sum = sum + n
end loop

if sum > 0 then
    output "Sum of all borrowed equipment is " + sum
else
    output "There are no items borrowed"
endif
```

Figure 4.2 – Pseudocode for the process of counting total borrowed equipment.

```
string GmailAccount = xxx@xxx.com
string GmailAccountPassword = xxx

Configuration properties for prop

    mail.smtp.host = smtp.gmail.com

    mail.smtp.port = 465

    mail.smtp.auth = true

    mail.smtp.socketFactory.port = 465

    mail.smtp.socketFactory.class = javax.net.ssl.SSLSocketFactory

Session = sessionObject

sessionObject = (get instant session from getInstance method of Session class)

MimeMessage = messageObject

(SET message object Properties)

message.setFrom = SystemEmail

message.addRecipient <- userInput

message.setSubject = "Your Borrowed Item is Due In TODAY"

message.setText = "Dear Student \n The item you have borrowed is expected by Ms.
Goppert to be return today! Please return it on time"

(Sending Message by using Transport class)

if (message send successfully) then

    output "Email Sent Successful"

else

    output "Email Sent Failed"

endif
```

*Figure 4.3 – Pseudocode for the process of sending an email to alert the student.
Configuration of email server, properties learned from Myokong.com (Mkyong.com, 2010)*

Databases & Tables

	Id	UserName	FullName	Password	Email
1	2002	nbson	Nguyen Bao Son	1	ssbaoson@gmail.com

Figure 5.1 – Account Table in Database

	EquipmentID	EquipmentCode	EquipmentName	EquipmentDescription	Supplier	Quantity	Classroom
1	1	1	laptop	a cool laptop	Mr Toner	1	C123
2	2	2	Mac	a cool Mac	Mr Toner	1	C123
3	1006	test	Equipment Test	A testing equipment for development uses	Bao Son	9999	null

Figure 5.2 – Equipment Table in Database

ID	DateBorrow	EquipmentID	ReturnDate	FullName	Email	Class	TimeReturned
----	------------	-------------	------------	----------	-------	-------	--------------

Figure 5.3 – Borrow Table in Database

ReturnID	BorrowDate	DateOfReturn	EquipmentID	Email	FullName
----------	------------	--------------	-------------	-------	----------

Figure 5.4 – Return Table in Database

Test Plan

Action Test	Way of testing
Check if username and password entered matched an existing account	<ul style="list-style-type: none"> - Input username into the first text box on the login screen - Input password into the text box under the username text box on the login screen - Click button “Login” - If it matches an existing account, it transition to the dashboard. - If it does not match, then an error message “Incorrect username or password” will appear. The dashboard will not appear.
Check if new account can be created	<ul style="list-style-type: none"> - Click on the button “Register” on the login screen - The register screen will appear, then fill in all of the text box with information corresponding to the new account. - Click on the button “Create User”. - If account is created successful, a message will popup. - If the username and password text box is left empty, then an error message will be displayed.

Check if username and password textfield in Register screen can be empty.	<ul style="list-style-type: none"> - Click on the register button. - The register screen will appear, leave the username and password textboxes empty. - An error message will be display.
Check for abnormal data for Email text field.	<ul style="list-style-type: none"> - Enter data without a "@" in any text box next to the word Email. - There should be an error message displayed.
Check if the "Back to Login" button work	<ul style="list-style-type: none"> - Click button "Back to login" on the register screen - It then will transition back to the login screen
Check if Add Equipment works	<ul style="list-style-type: none"> - Click on the button "New Equipment" on the main dashboard - The new equipment screen will appear, fill in all of the text box with information about the new equipment. - Click on the button Add equipment - A message will appear indicating that a new equipment has been added.
Check if the search button works.	<ul style="list-style-type: none"> - Borrow screen: <ul style="list-style-type: none"> - Enter the information of the equipment in the borrow screen and click search. - The rest of the information about it will be displayed. - Return screen: <ul style="list-style-type: none"> - Enter the email of the borrower in the textbox and click search. - The information about the borrowed item will be displayed.
Check if the borrowed list update after someone borrow an item.	<ul style="list-style-type: none"> - Click on Borrow Equipment in the main dashboard. - Search for the item and fill in all of the textbox with relevant information. - Click borrow and go back to the main dashboard. - Go to the screen that displayed data on borrowed list. - If it is successful, the information about the borrow session will be displayed in a table.
Check if the returned list and borrowed list update after someone return an item.	<ul style="list-style-type: none"> - Click on Return Equipment in the main dashboard. - Search for the email of a borrower and the information about the borrow session will be displayed. - Click return and go back to the main dashboard. - Go to the screen that displayed data on returned list. - If it is successful, the information about the borrow session will be remove from borrowed list and will be update into returned list.
Check if an email that does not borrow anything can't return an item	<ul style="list-style-type: none"> - Go to the screen for returning equipment and enter an email that have not borrow anything. - It should return a message saying that this email has not borrow anything.

Check if an email is sent after borrowing an item	<ul style="list-style-type: none"> - Borrow an item and go back to the main dashboard. - Check mailbox for a message.
Check if an email is sent after the process of alerting the student.	<ul style="list-style-type: none"> - Click on the Alert Student button. - Enter the wanted email. - Check the mailbox for the alert.

Bibliography

Mkyong.com. (2010). JavaMail API – Sending email via Gmail SMTP example – Mkyong.com. [online] Available at: <https://www.mkyong.com/java/javamail-api-sending-email-via-gmail-smtp-example/> [Accessed 8 Jul. 2019].