Criterion B: Design

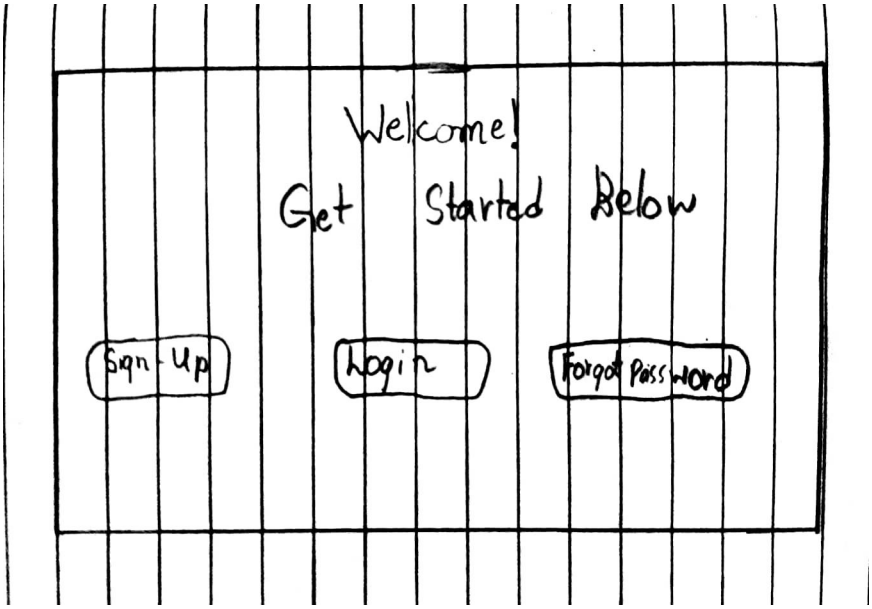
Table of Contents

1. Rough Sketches of the User Interface

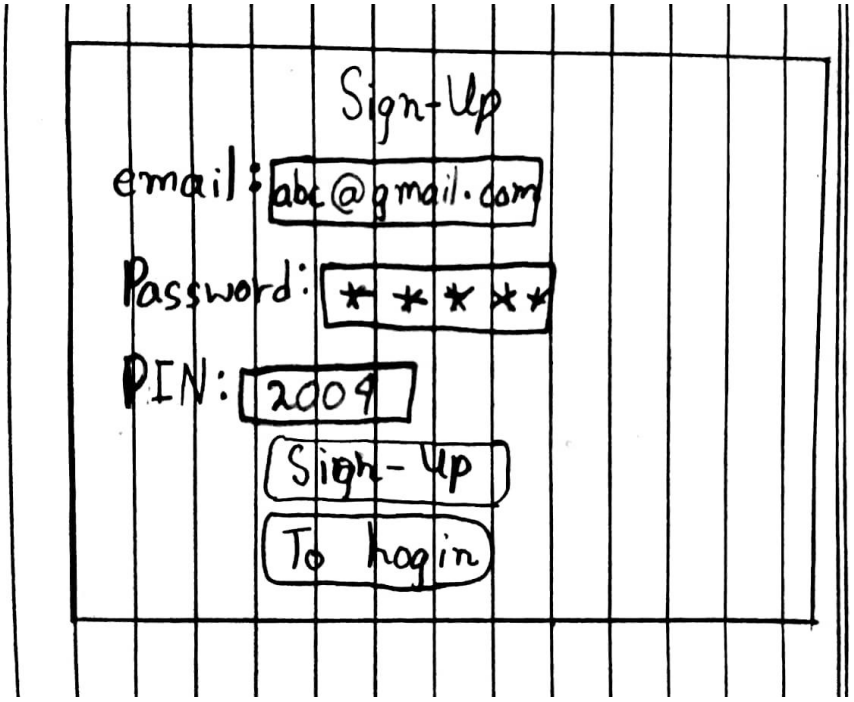
Each sketch is a rough representation of each class/frame that will be used in the product. The label for each sketch is as follows:

a. Sketch 1: Welcome
b. Sketch 2: signUp
c. Sketch 3: Forgot
d. Sketch 4: Login
e. Sketch 5: Homepage
f. Sketch 6: addAssignment
g. Sketch 7: deleteAssignment
h. Sketch 8: Assignment_Progress
i. Sketch 9: Deadline_Progress

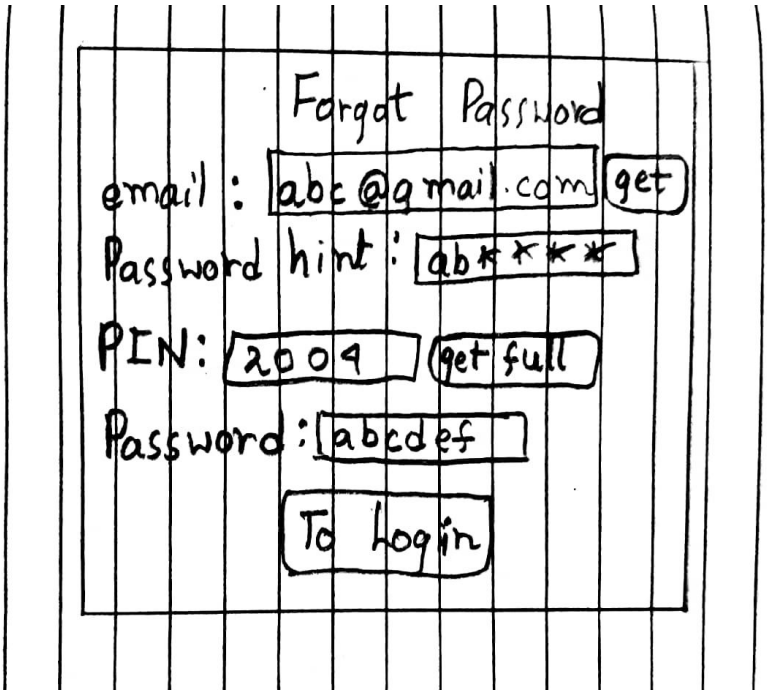
Sketch 1:



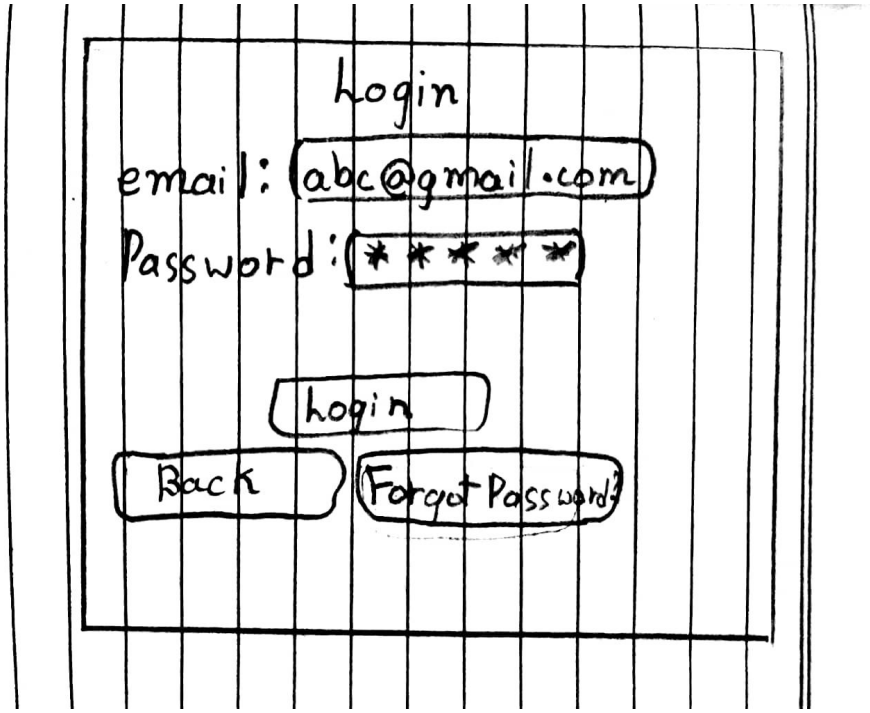
Sketch 2:



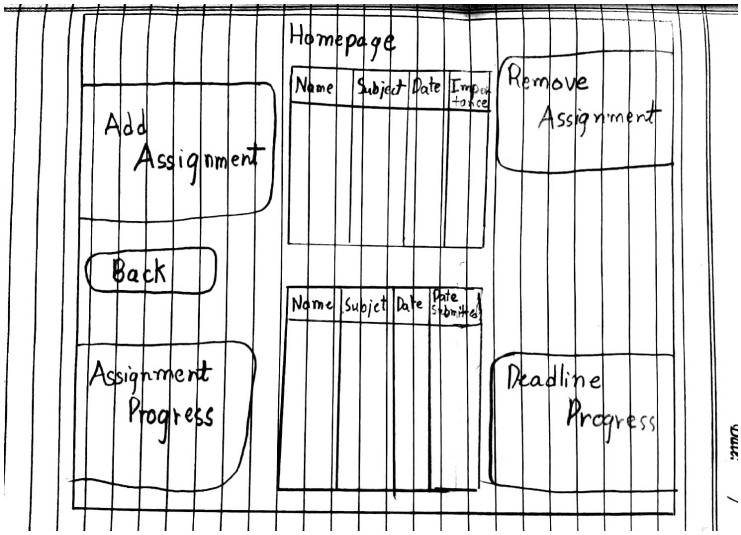
Sketch 3:



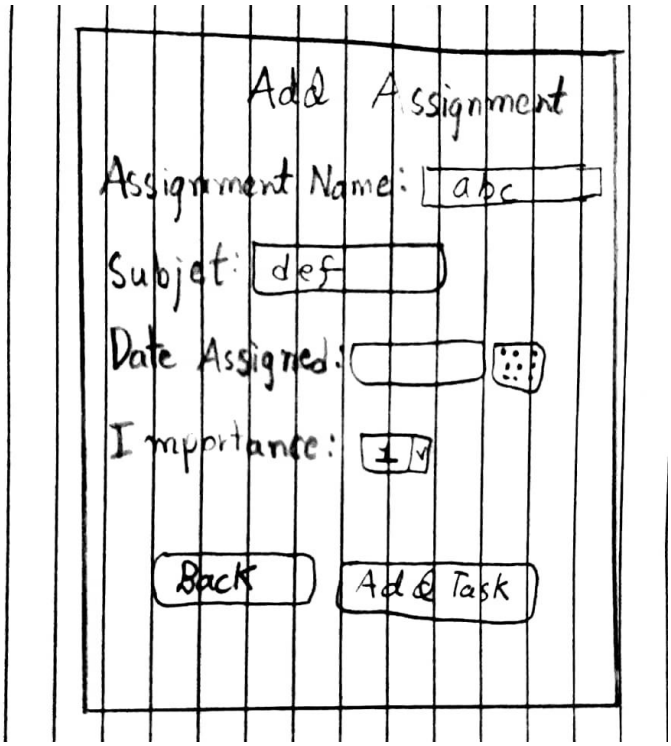
Sketch 4:



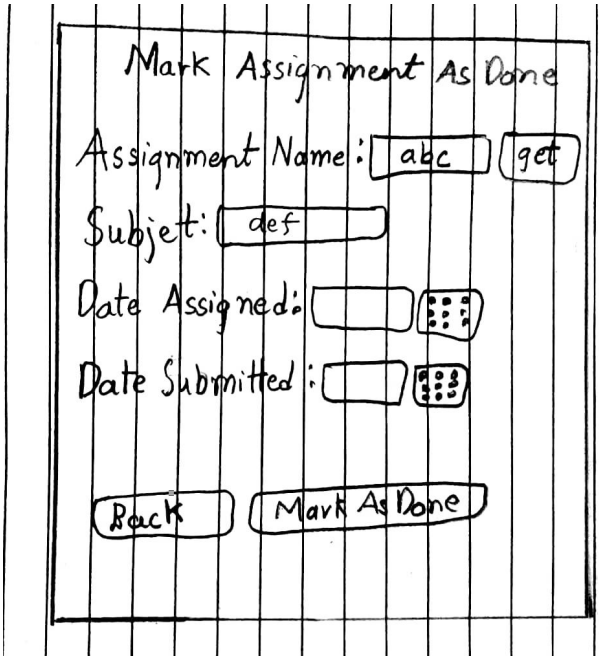
Sketch 5:



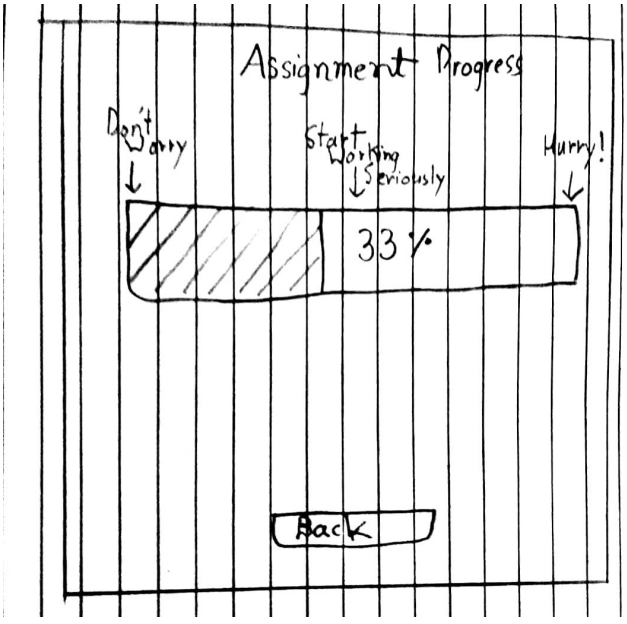
Sketch 6:



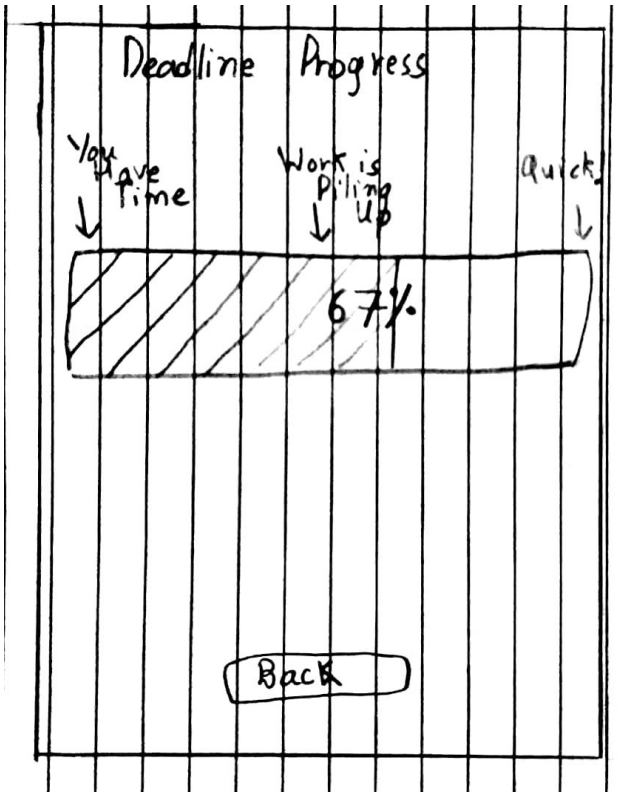
Sketch 7:


Mark Assignment As Done
Assignment Name:
abc
get
Subjet:
def
Date Assigned:
Date Submitted:
Back
Mark As Done

Sketch 8:


Assignment Progress
Don't worry
Start working seriously
Hurry!
33 %
Back

Sketch 9:


Deadline Progress
You have time
Work is piling up
Quick!
67%
Back

2. Classes

a. Welcome: This class will serve as the first screen the client will see. Here, the client will be prompted to log in or create an account. A password retrieving option will also be present if the password has been forgotten.

b. signUp: This class will give the client the option to create a new account, with the credentials used being the email, the password, and a PIN.

c. Forgot: This class will be used to retrieve the forgotten password. It will first give the client the hint of the password after the email has been entered and after the PIN has been entered, the full password will be revealed.

d. Login: This class will redirect the user to the homepage. It will require the user to input the email and the password to obtain access.

e. Homepage: This class will let the user access all the functions of the app. The user will be able to view the pending and completed assignments immediately. They will also have the option to be redirected to sections where they can add assignments, mark them as done, and view total progress in regards to total progress and deadline proximity.

f. addAssignment: This class will give the user to option to add new assignments based on assignment name, the subject through which it was assigned, and its difficulty level, ranging from one to three.

g. deleteAssignment: This class will give the user the option to mark assignments as done, which will remove them from the pending assignments list and add it to the completed lists one. It will use the assignment name, subject, and the dates in which it was assigned and submitted as bases for the insertion of data.

h. Assignment_Progress: This class will show the user the total progress made in terms of how many assignments are left to be done. It will highlight the urgency of doing tasks once enough assignments with higher levels of difficulty have been assigned.

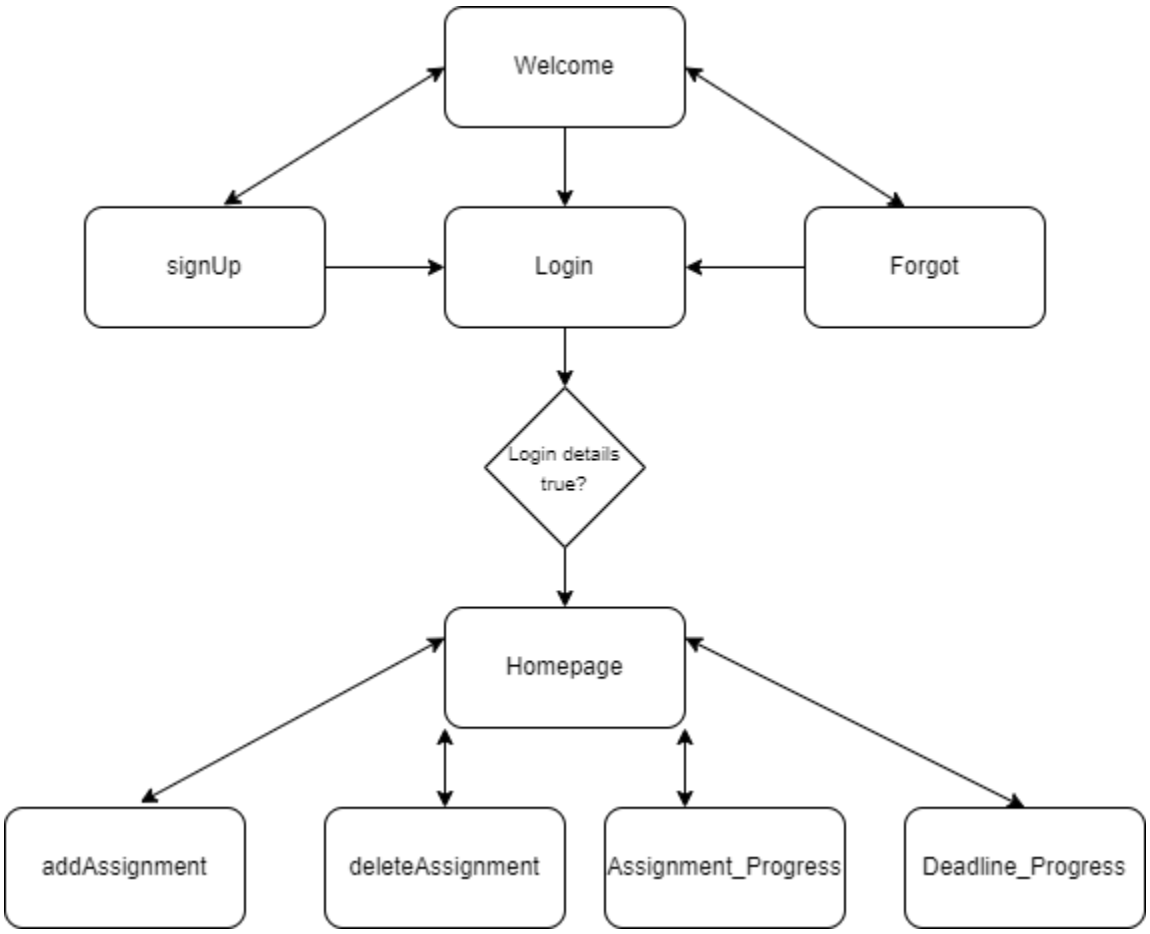
i. Deadline_Progress: This class will show the user the total progress made in terms of how behind the user is relative to their deadlines. It will emphasize the urgency of doing tasks based on how close due dates are to the current date.

j. Connector: This class will connect the project to the database. The methods of this class will be utilized in the other classes to offer them an easy mode of connection.

3. Flowcharts

3.1 Movement Between Windows

Chart 1.1: This flowchart movement outlines the basic movement from one class/window to another in the program.



3.2 Flowcharts for Classes

The flowcharts outlining the program flow of the more complex classes are depicted here. They are as follows.

Chart 1.2: logIn

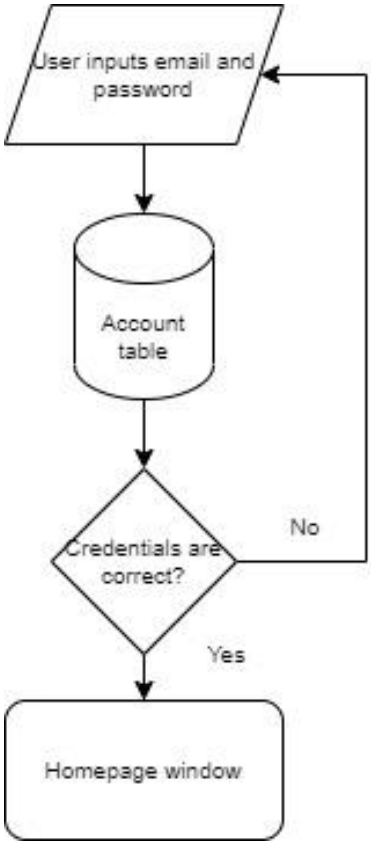


Chart 1.3: Forgot

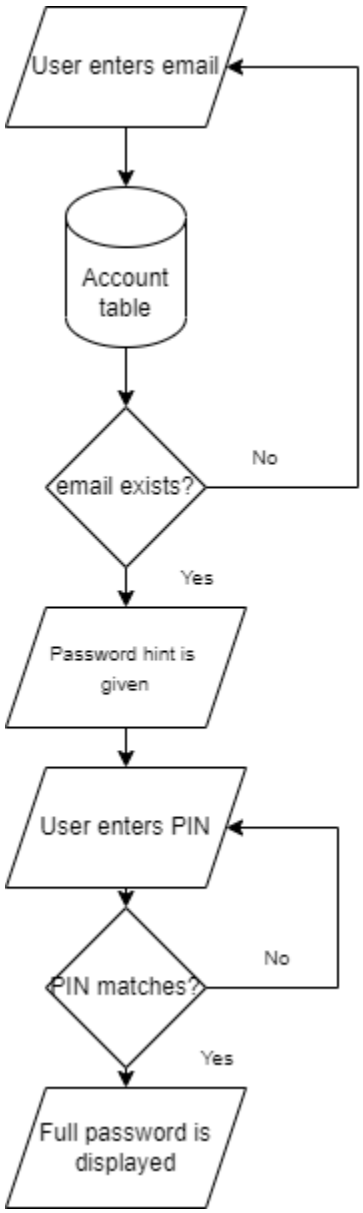
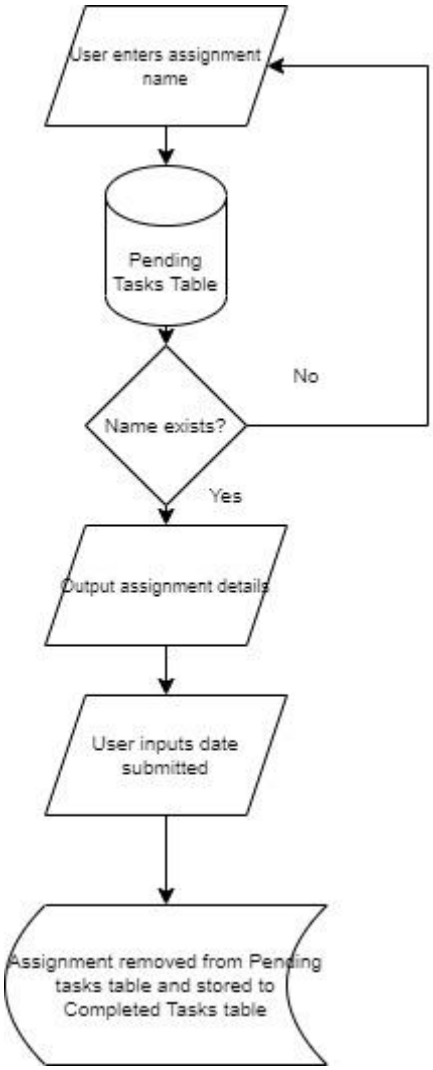


Chart 1.4: deleteAssignment



4. SQLite Implementation

The following SQLite tools will be used in the product to store and utilize data.

4.1 SQLite Tables

Table 1: Account

Field Name	**Data Type**	**Description**
Name	TEXT	Used to store the name
Email	TEXT	Used to store the email; also used as the primary key
Password	TEXT	Used to store the password hint
PIN	TEXT	Used to store the PIN
Decrypted_Password	TEXT	Used to store the full password

Table 2: Pending

Field Name	**Data Type**	**Description**
Name	TEXT	Used to store the name of the assignment; also used as the primary key
Subject	TEXT	Used to store the subject of the assignment
Date_Due	TEXT	Used to store the deadline of the assignment
Importance	TEXT	Used to store the level of importance of the task

Table 3: Completed

Field Name	Data Type	Description
Name	TEXT	Used to store the name of the assignment; also used as the primary key
Subject	TEXT	Used to store the subject of

		the assignment
Date_Due	TEXT	Used to store when the assignment was due
Date_Submitted	TEXT	Used to store when the assignment was submitted

4.2 SQLite Queries

a. **INSERT:** This query will be used to store both the credentials and assignment details into the databases. It will be used to enter records into the tables “Account”, “Pending Tasks”, and “Completed Tasks”.
b. **SELECT:** This query will be used to retrieve credentials and assignment details from the databases. It will be used to retrieve records from the “Account” table for the “Login” and “Forgot” classes. It will also be used to retrieve assignment details from “Pending Tasks” tables for the “deleteAssignment” class.
c. **DELETE:** This query will be used to delete records from the database. It will be used to remove the assignments from the “Pending Tasks” table after it has been marked as completed and moved to the “Completed Tasks” table.

5. Pseudocode for Advanced Algorithms

5.1 Password Hint Creator

```
INPUT Password
Hint = Password.substring(0, 2)
Asterisk = “”
Loop for i = 0 to Password.length() - 2
        Asterisk = “*” + Asterisk
End loop
Hint_Password = Hint + Asterisk
```

5.2 Assignment Progress

```
i = 0
n = 0
Sum = 0
        while (Pending.hasNext())
                i = fetched value from “Importance” column
                n = i * 10
                Sum = Sum + n
                ProgressBar.value(Sum)
        end loop
```

5.3 Deadline Progress

```
N = 0
date = “”
dateSubstring = “”
dateInt = 0
now = date.today()
nowSubstring = now.substring(0,4) + now.substring(5,7) + now.substring(8)
nowInt = nowSubstring.toInteger()
while (Pending.hasNext())
        date = fetched value from “Date_Due” column
         dateSub = date.substring(0,4) + date.substring(5,7) + date.substring(8)
         dateInt = dateSub.toInteger
        if (dateInt - nowInt >= 14)
                N = N + 10
        else if (dateInt = nowInt >= 7 AND dateInt  - nowInt < 14)
```

```
            N = N + 20
        else if (dateInt - nowInt >= 2 AND dateInt - nowInt < 7)
            N = N + 80
        else
            N = 100
        end if

        if (N > 100)
            N = 100
        end if
        ProgressBar.value(N)
end loop
```

6. Test Plans

Action to Test	Success Criteria Tested	Method
How user-friendly the GUI is	1, 6	Check the legibility of the labels and buttons and how coherent the flow from one window to the other is
Signing up and logging in	2, 5	See if access to the homepage is ensured through, and only, after the correct login details are entered
Inputting the wrong information	11	Display the appropriate message when wrong information is inputted.
Password Hints	3	Check if only certain parts of the password can be seen; assisting the user to remember while still maintaining privacy
Two-step verification for full step password retrieval	3, 4	Check if the full password is revealed only after both the email and PIN are entered
Show all pending and completed tasks easily	9	Verify if both the completed and pending tasks can be seen upon immediate access into the homepage
Easily add new tasks and mark them as done	6	Check how simple and easy it is to add and remove assignments from the pending list.
Highlight the urgency of doing certain tasks	10, 12	Show immediate progress using the progress bars as the basis.