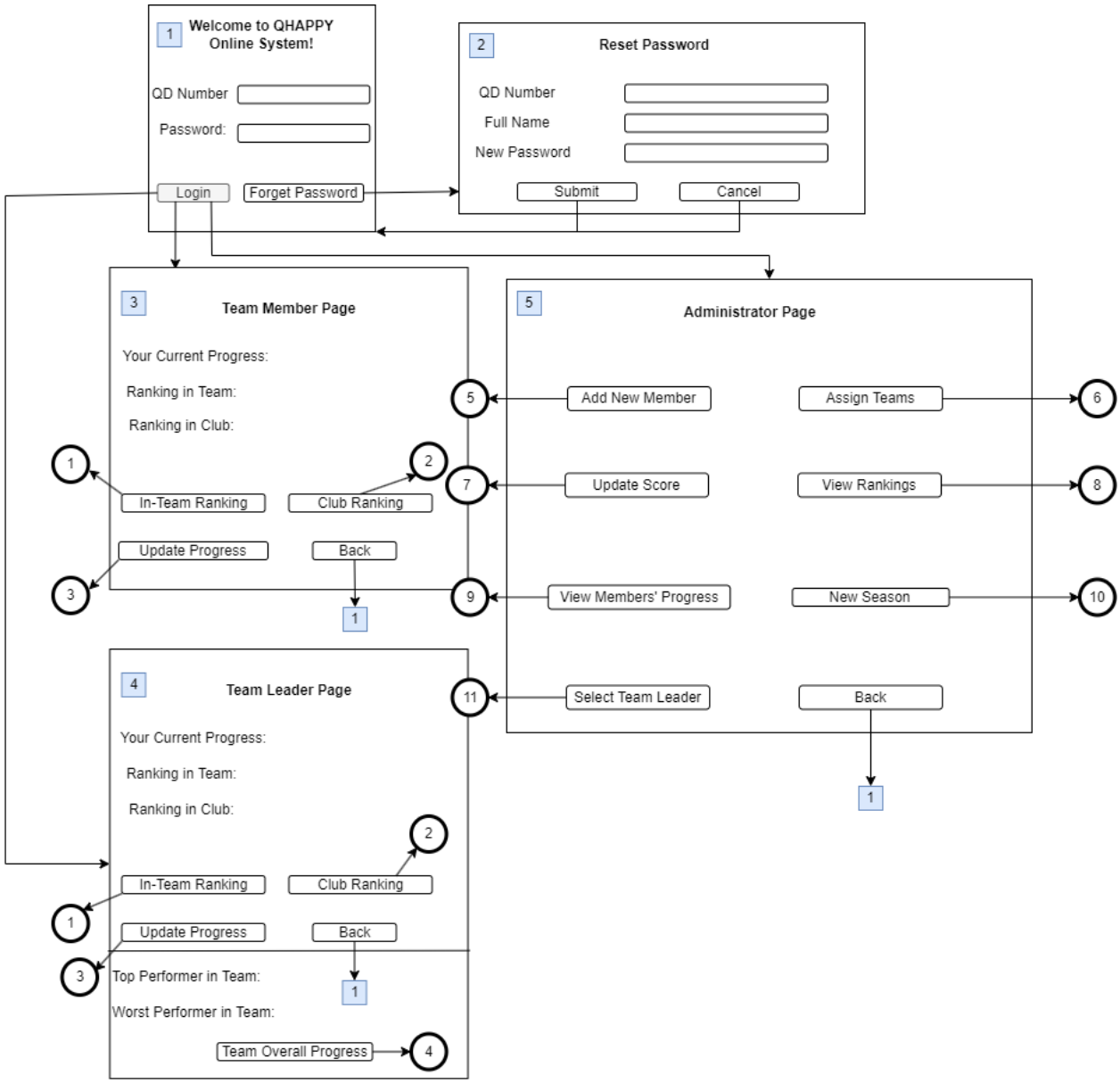
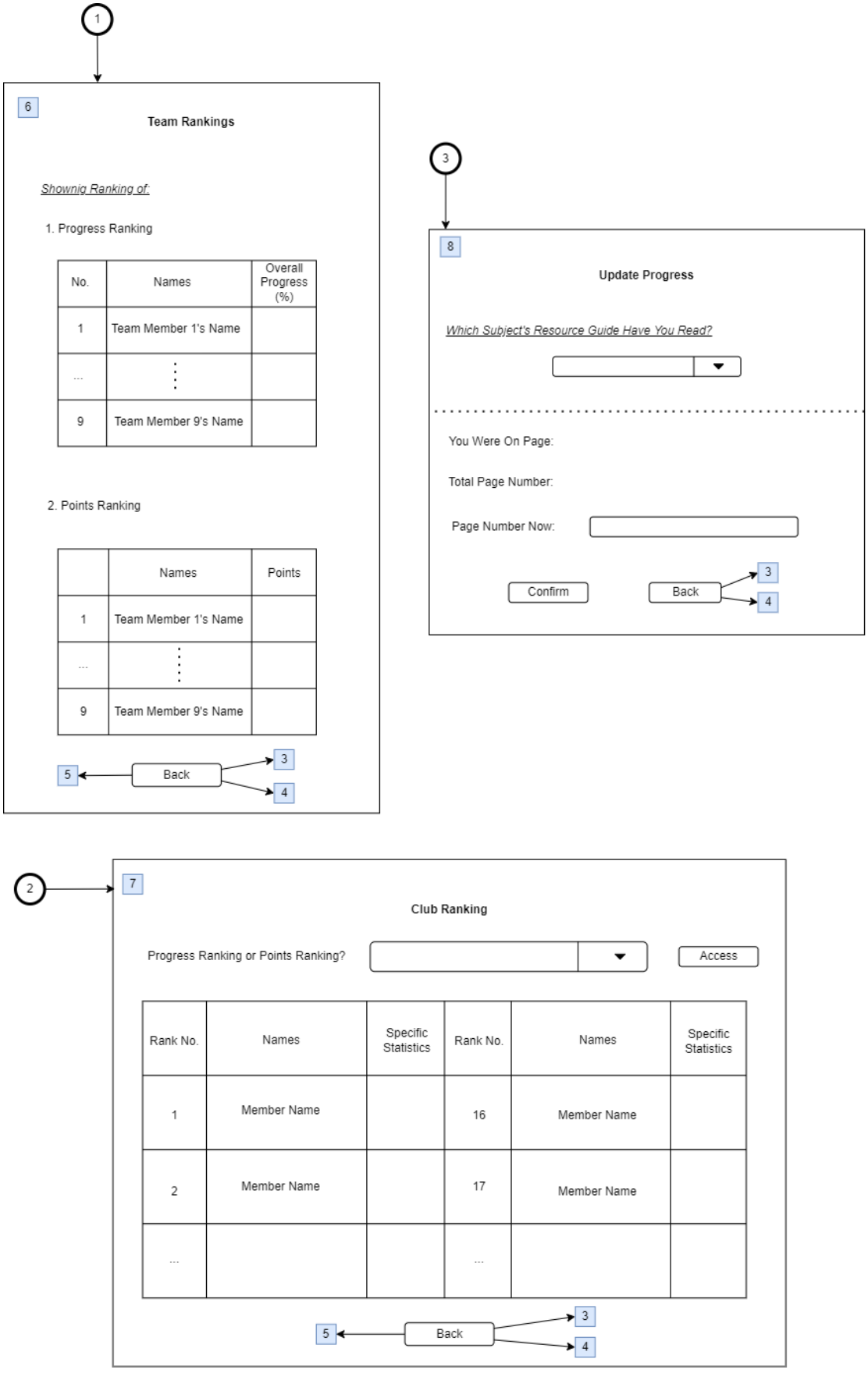
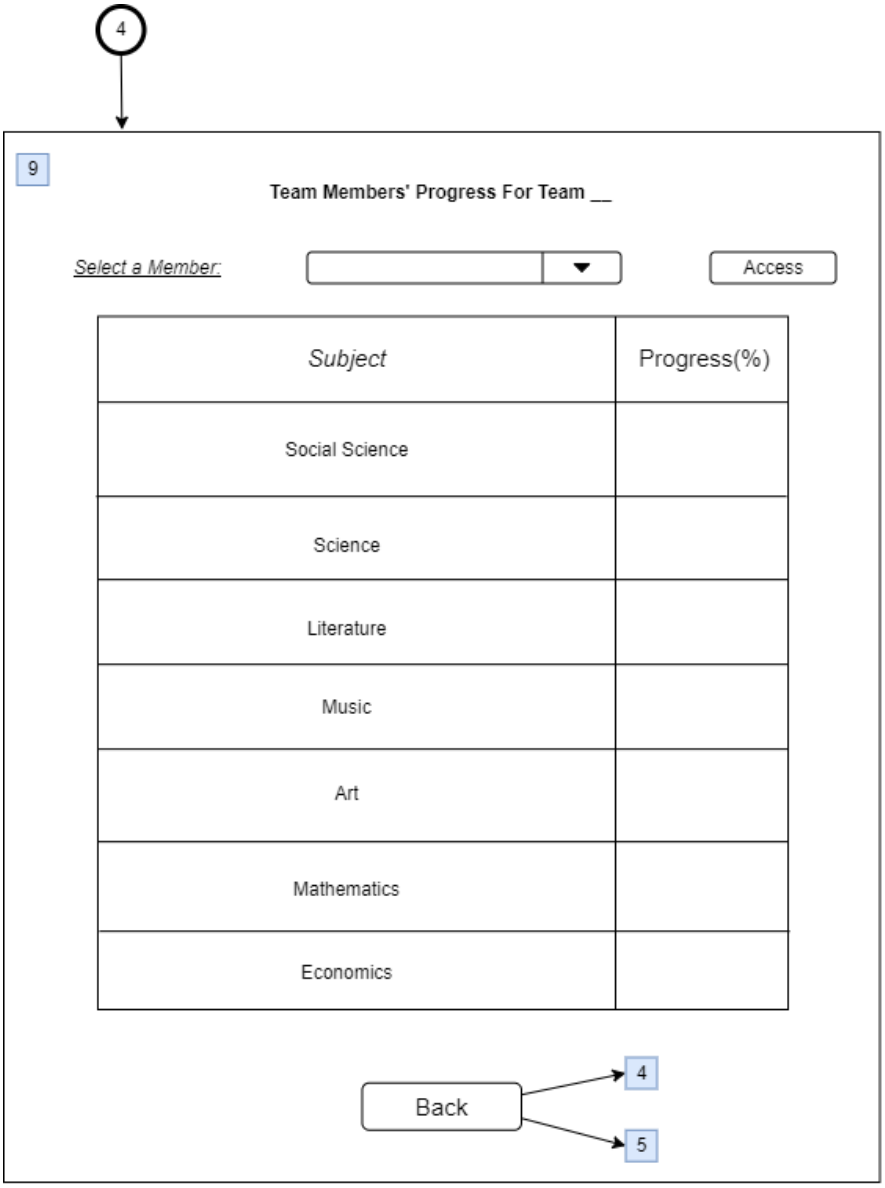
Criteria B: Design

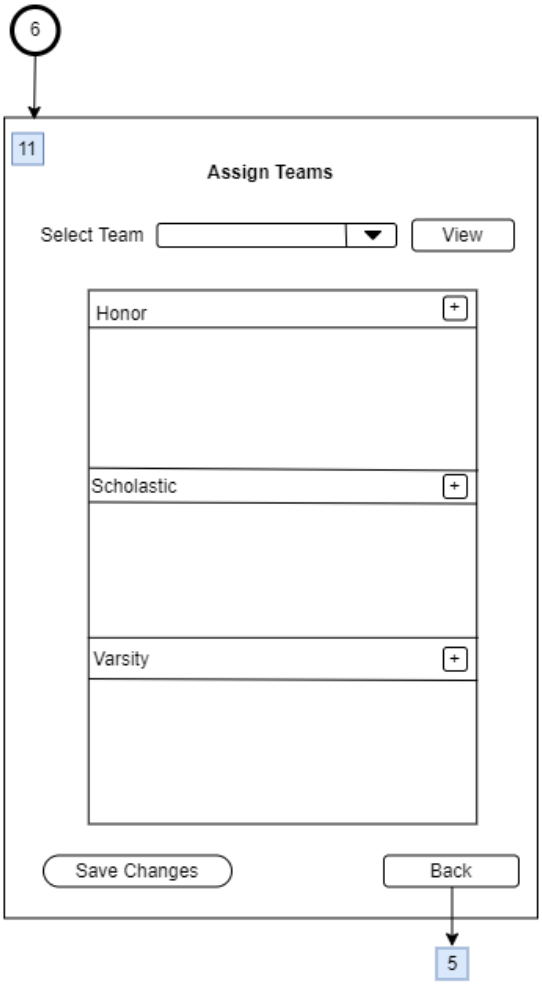
Table of Contents

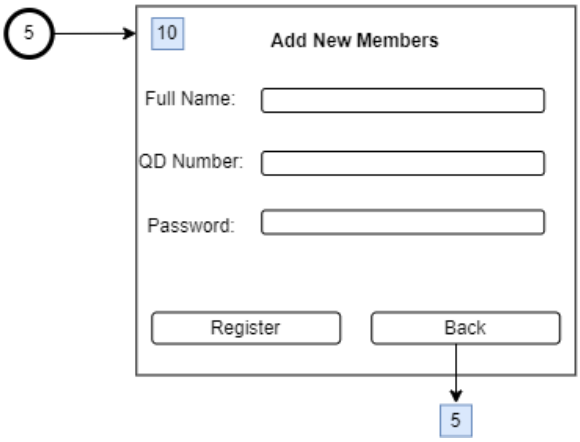
1. ***Overall Product Setup***

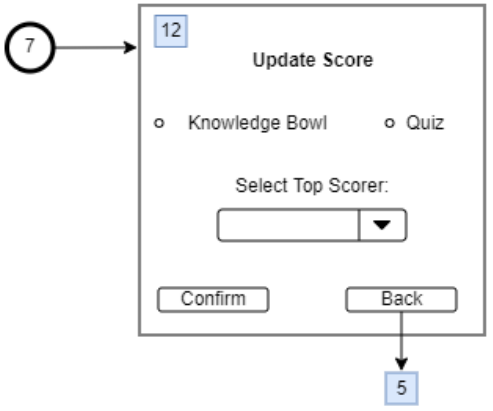


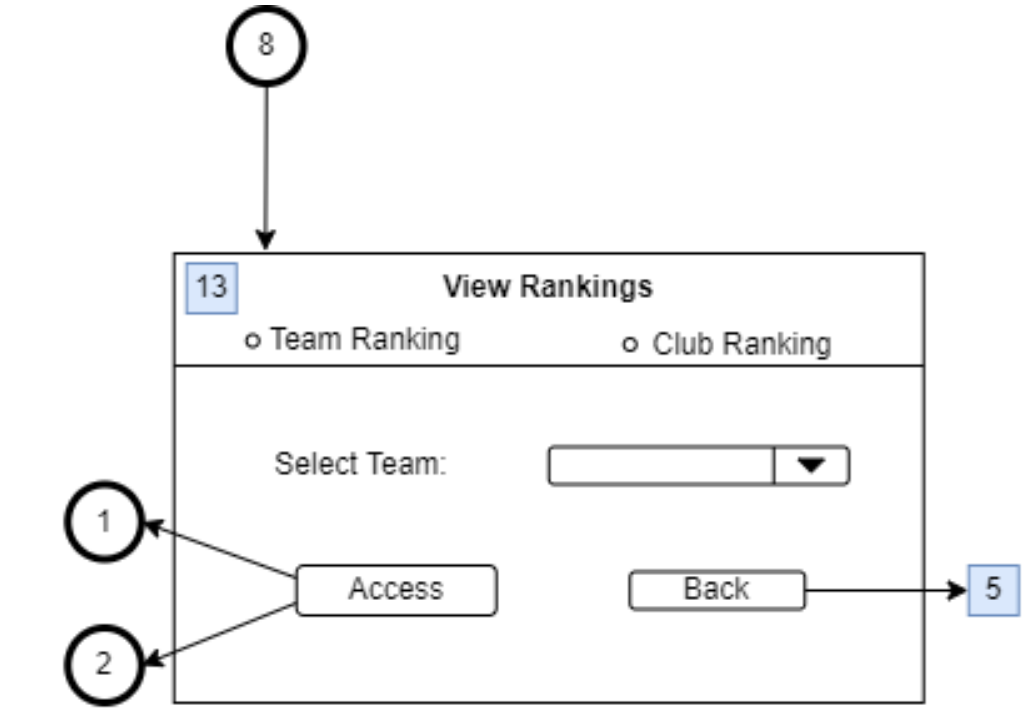

1
6
Team Rankings
Shownig Ranking of:
1. Progress Ranking
No.
Names
Overall Progress (%)
1
Team Member 1's Name
...
9
Team Member 9's Name
2. Points Ranking
Names
Points
1
Team Member 1's Name
...
9
Team Member 9's Name
5
Back
3
4
3
8
Update Progress
Which Subject's Resource Guide Have You Read?
You Were On Page:
Total Page Number:
Page Number Now:
Confirm
Back
3
4
2
7
Club Ranking
Progress Ranking or Points Ranking?
Access
Rank No.
Names
Specific Statistics
Rank No.
Names
Specific Statistics
1
Member Name
16
Member Name
2
Member Name
17
Member Name
...
...
5
Back
3
4

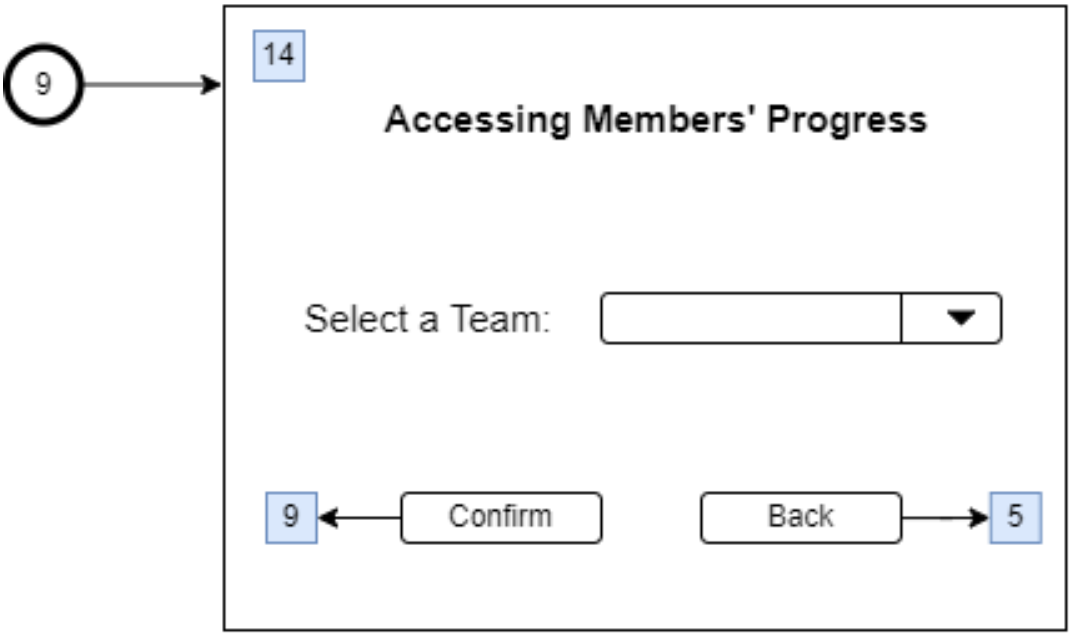

4
9
Team Members' Progress For Team __
Select a Member:
Access
Subject
Progress(%)
Social Science
Science
Literature
Music
Art
Mathematics
Economics
Back
4
5

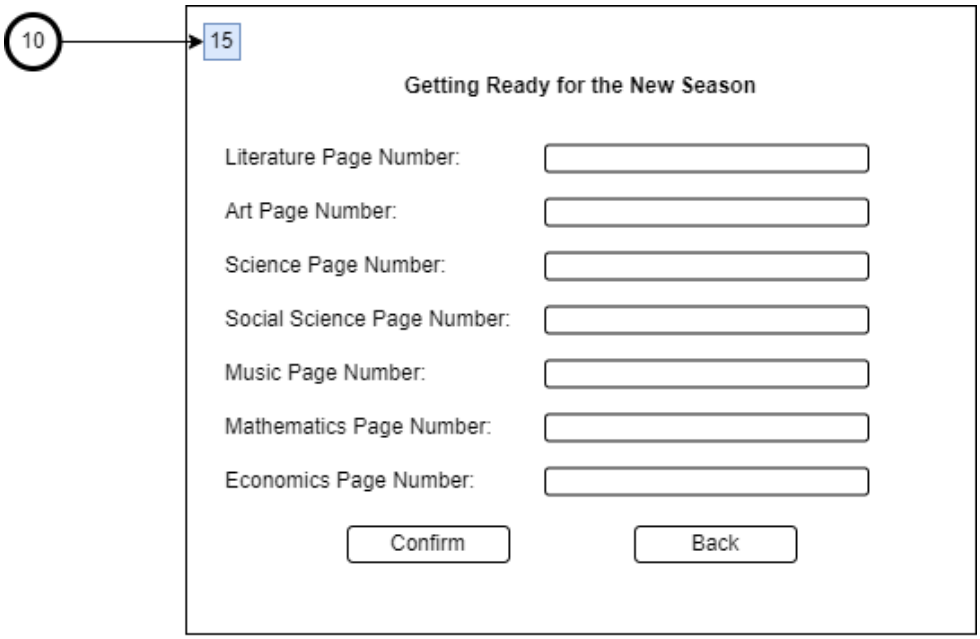

6
11
Assign Teams
Select Team
View
Honor
Scholastic
Varsity
Save Changes
Back
5

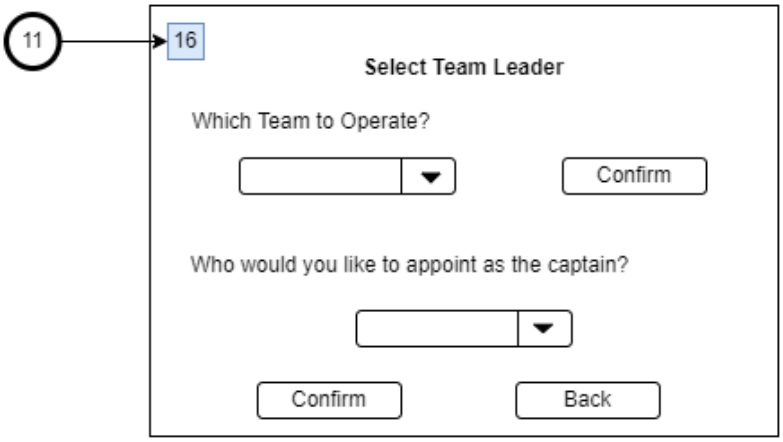

5
10
Add New Members
Full Name:
QD Number:
Password:
Register
Back
5


7
12
Update Score
Knowledge Bowl
Quiz
Select Top Scorer:
Confirm
Back
5

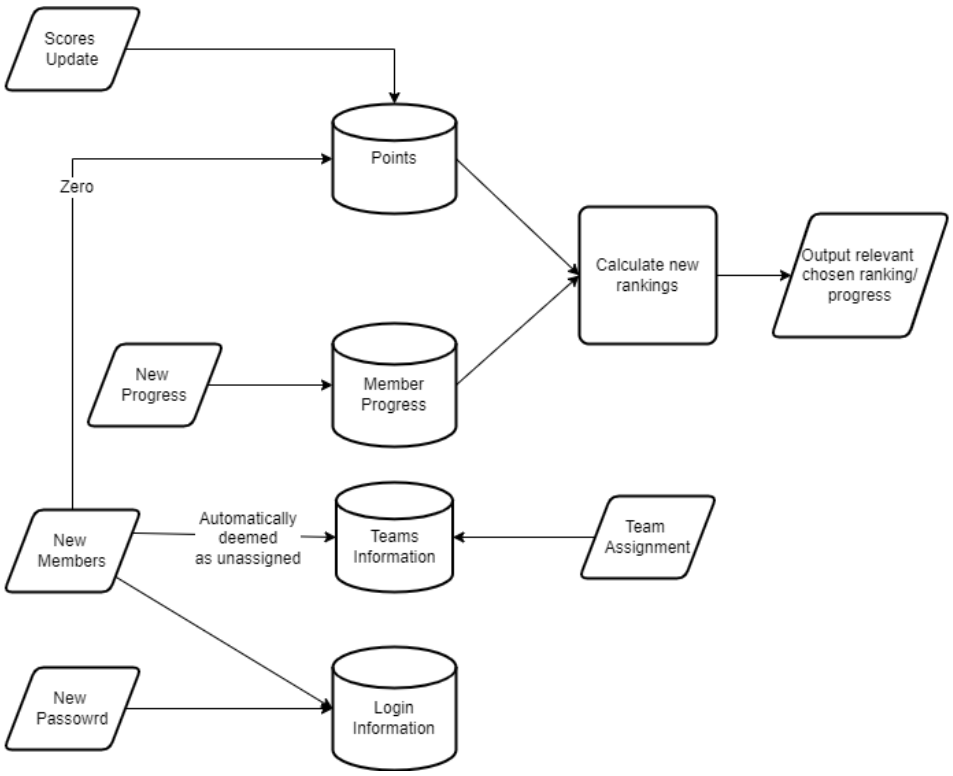

8
13
View Rankings
Team Ranking
Club Ranking
Select Team:
1
Access
Back
5
2


9
14
Accessing Members' Progress
Select a Team:
9
Confirm
Back
5


10
15
Getting Ready for the New Season
Literature Page Number:
Art Page Number:
Science Page Number:
Social Science Page Number:
Music Page Number:
Mathematics Page Number:
Economics Page Number:
Confirm
Back

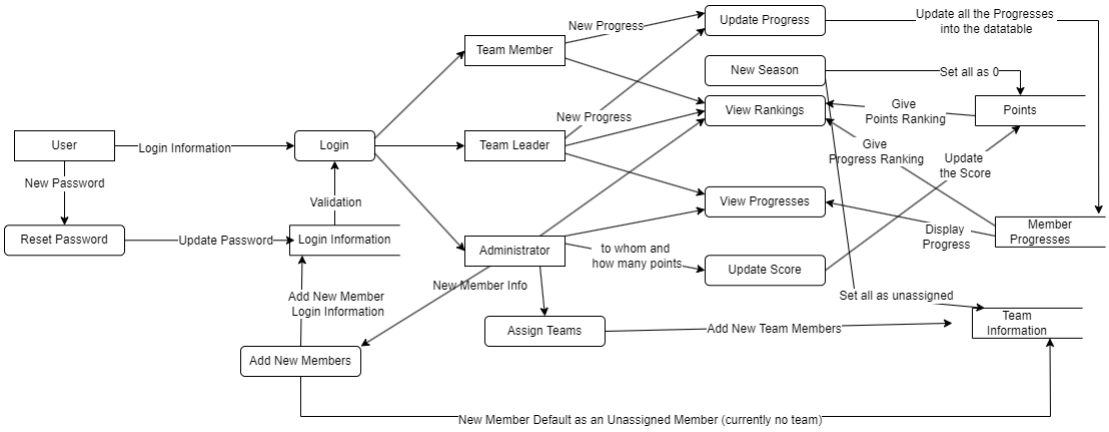

11
16
Select Team Leader
Which Team to Operate?
Confirm
Who would you like to appoint as the captain?
Confirm
Back

2. ***System Flowchart***



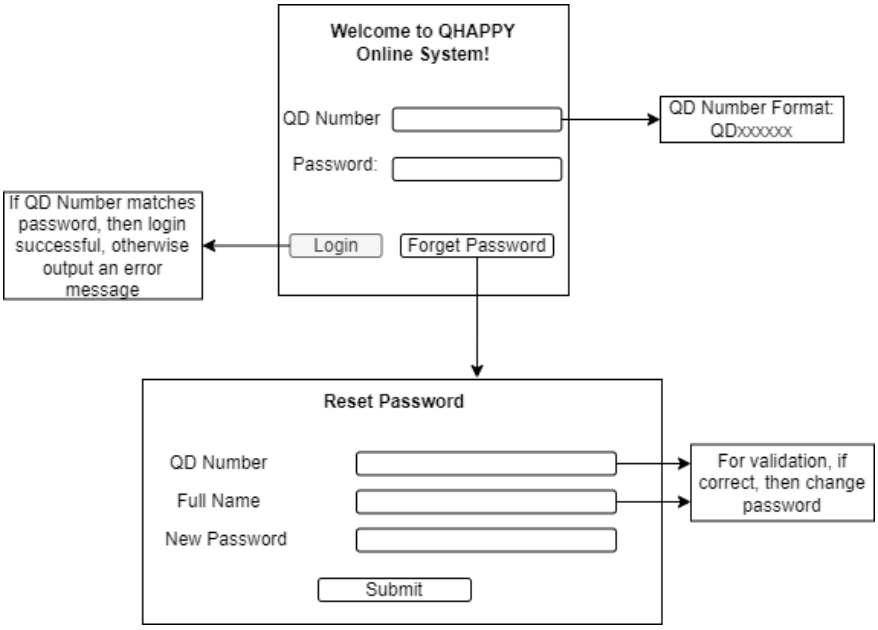
Then, the system flowchart above is a demonstration of how the system would work out in simple terms.

3. ***Data Flow Diagram***

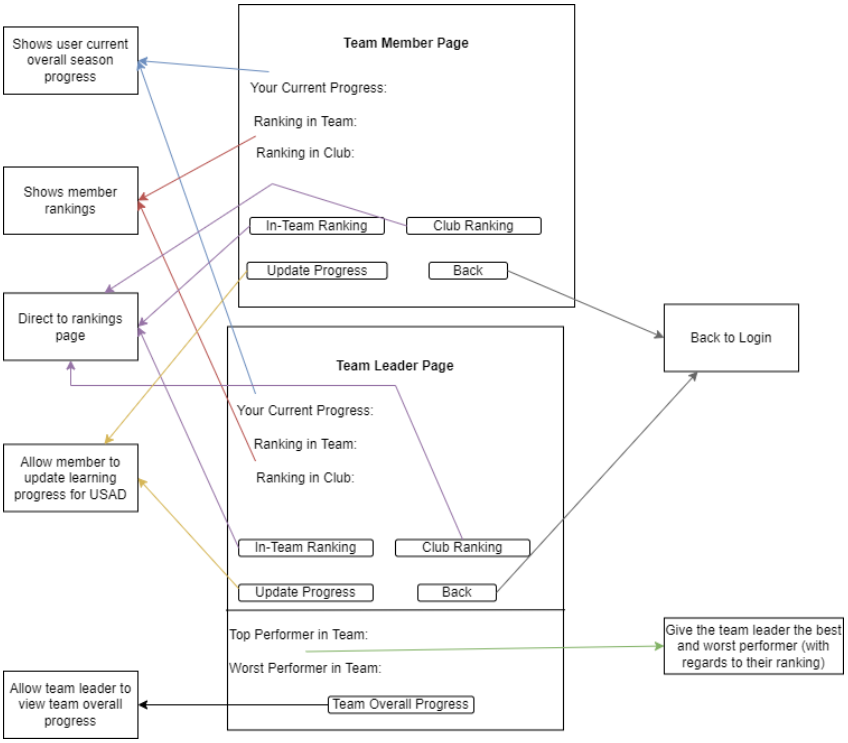


The above Data Flow Diagram gives a brief overview of how data would be transferred and updated within this close system.

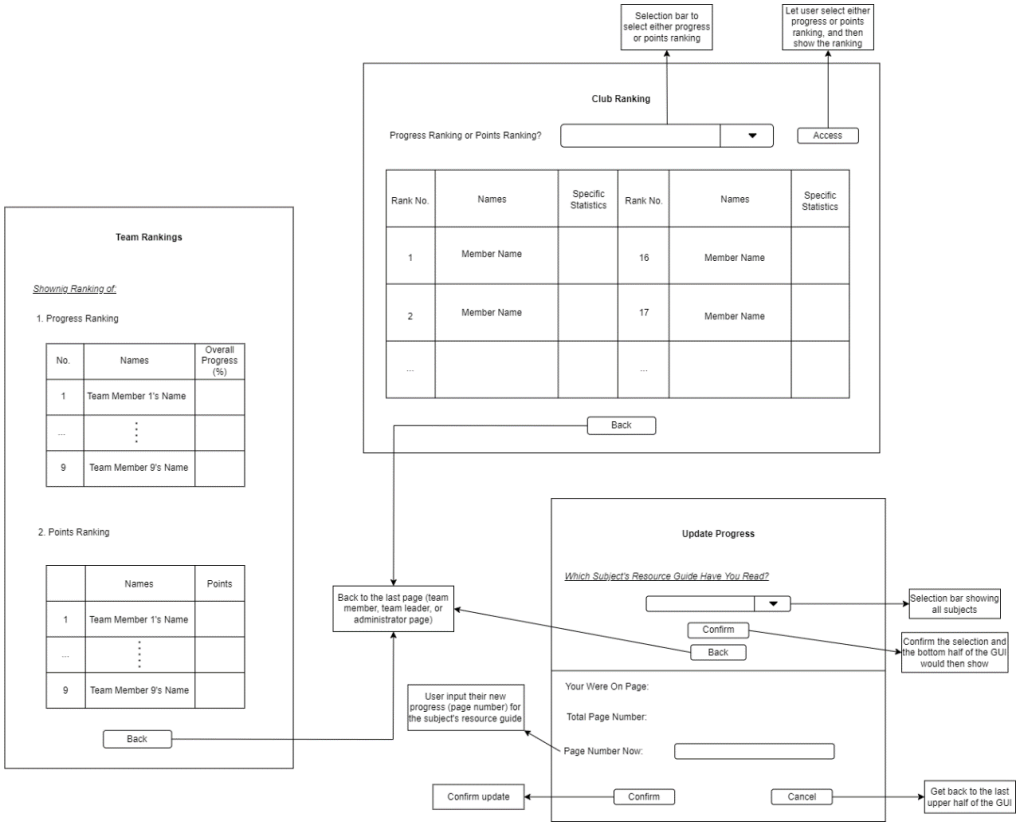
4. ***Annotated GUIs***



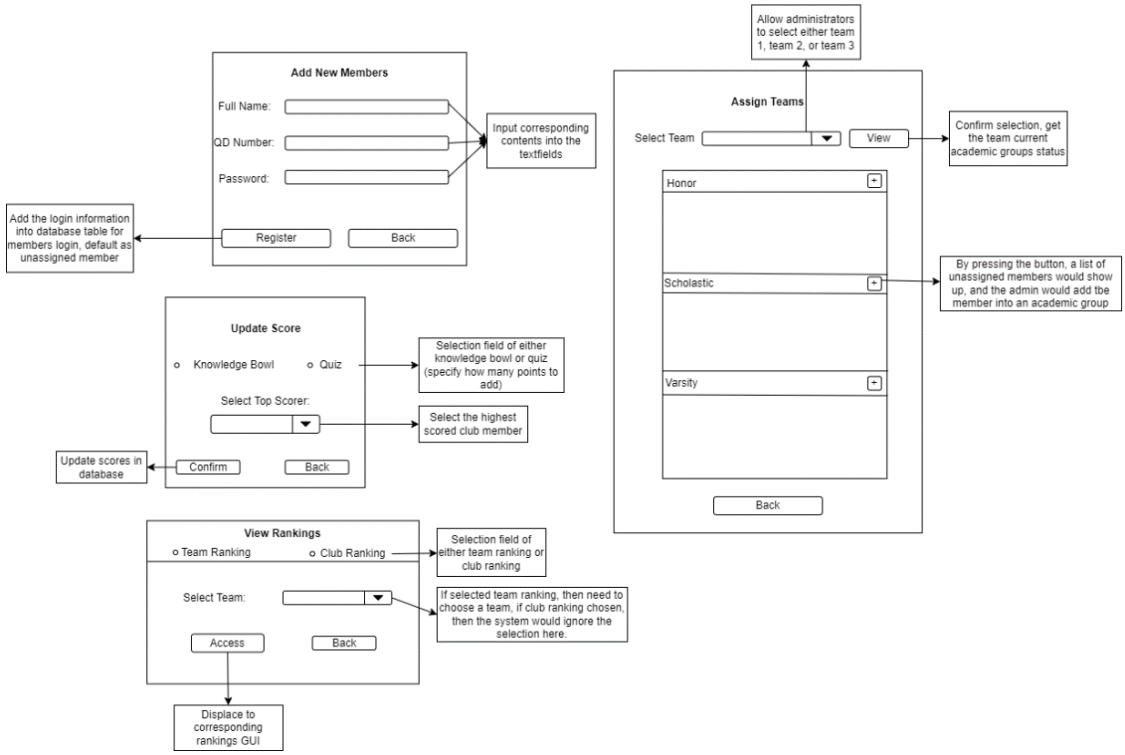
When logged in, users would be directed to different GUI windows that correspond to their statuses – either member, leader, or administrator



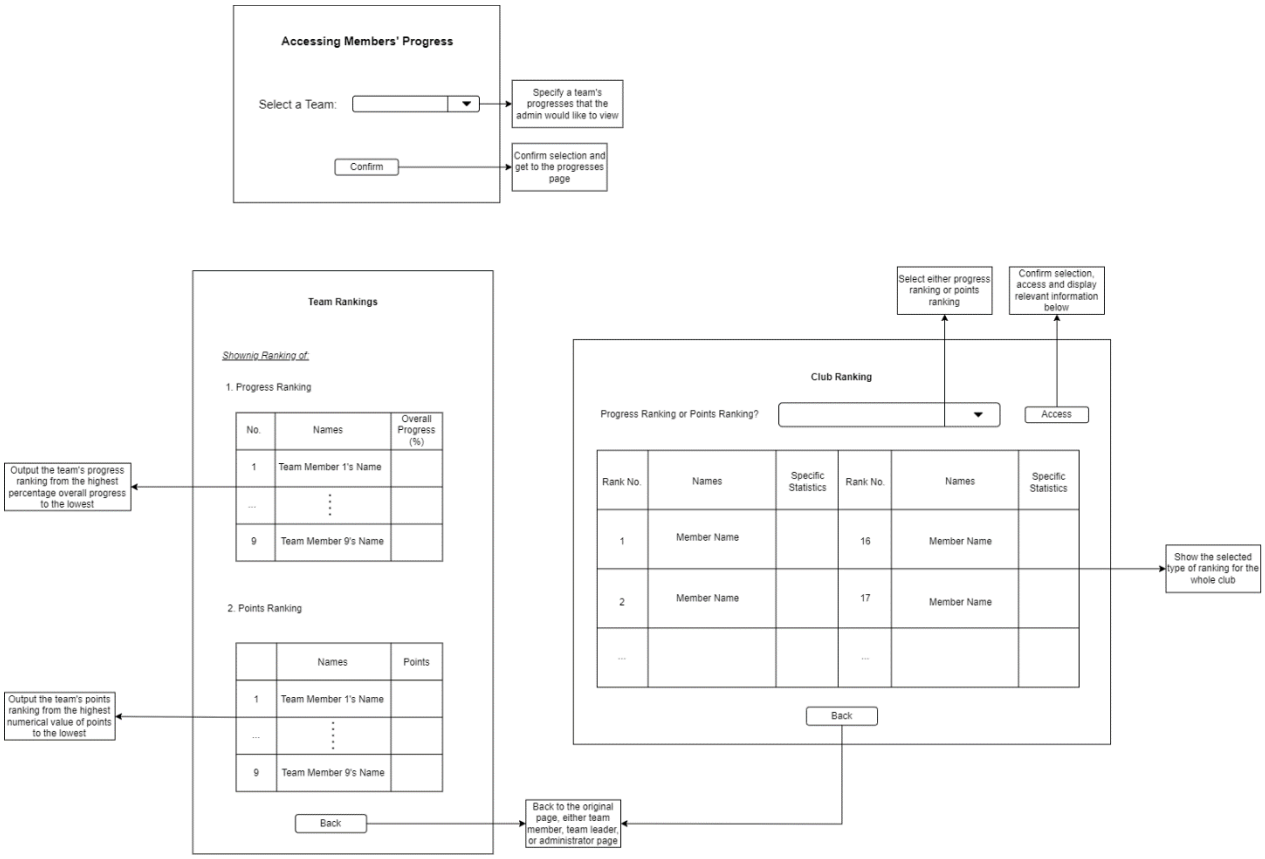
Team leader and team member have majorly same page, except that team leader could have more accesses to his/her members' information



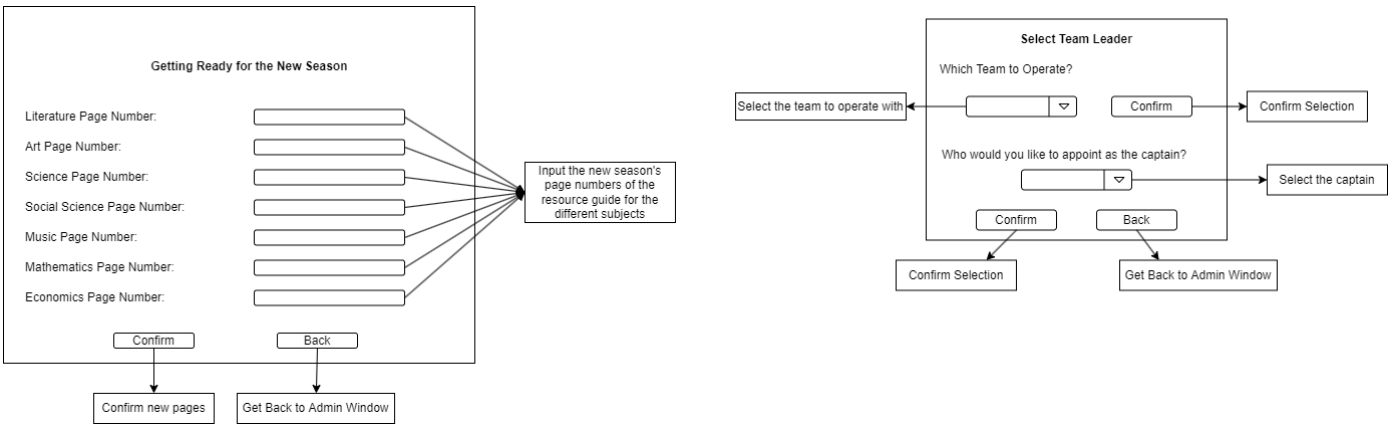
The rankings mainly have two different ways: progress ranking and points ranking. For the worst and best performers that would be displayed in the team leader's page, it only takes in the points ranking.



Some functions within an administrators' team management rights are shown above. In the plus sign, there would be an option where the admin could choose whether or not to assign the member as a team leader. The team leader could not be changed afterwards.



There would be two kinds of rankings: progress ranking and points ranking, for all system's users to access. At last, the annotated GUIs that are related to starting a new season.

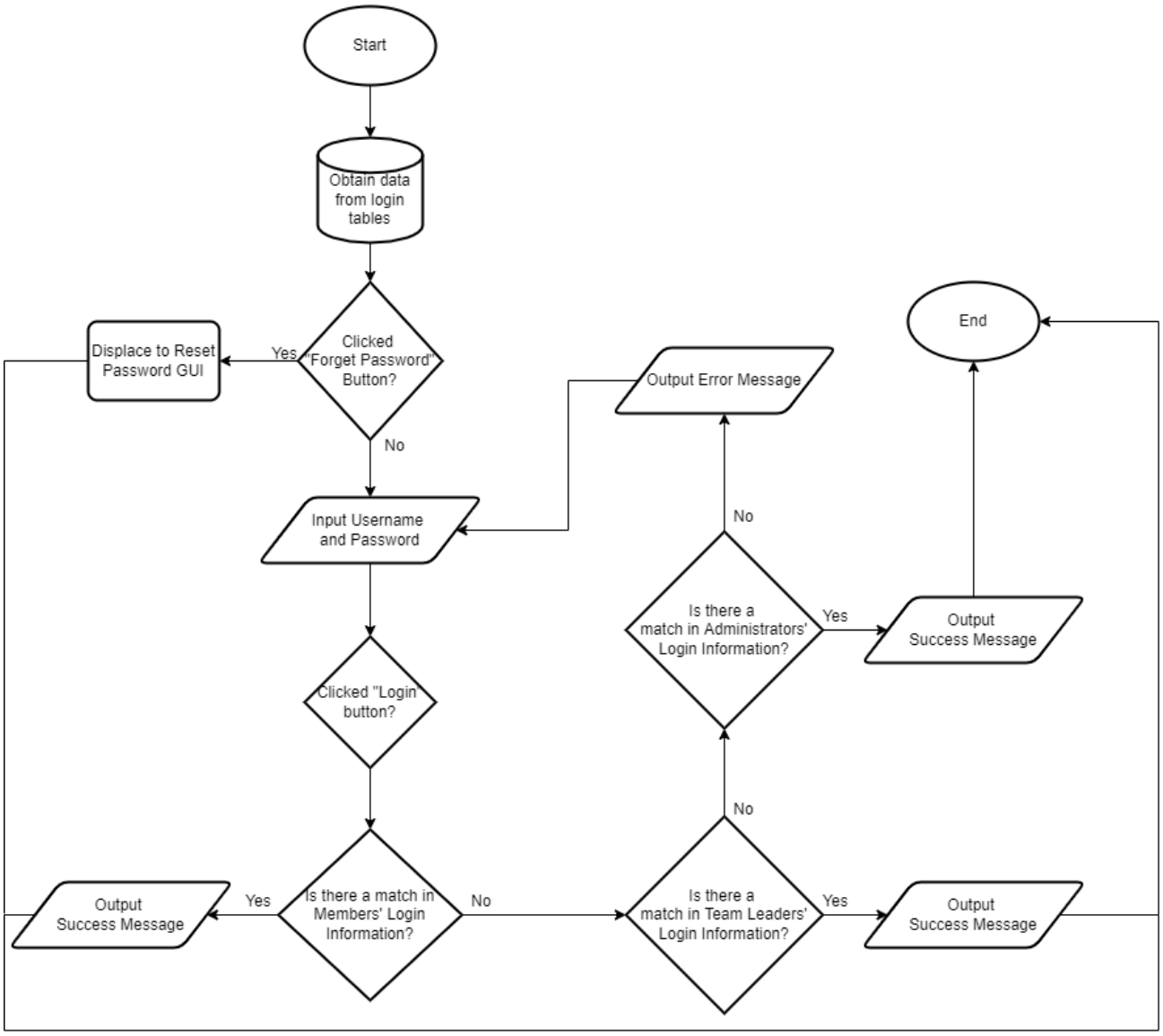


The New Season UI allows the administrator to input the new page numbers of the new season's resource guides, while the Select Team Leader UI gives the admin the right to select a leader for a specific team.

5. ***Flowcharts***

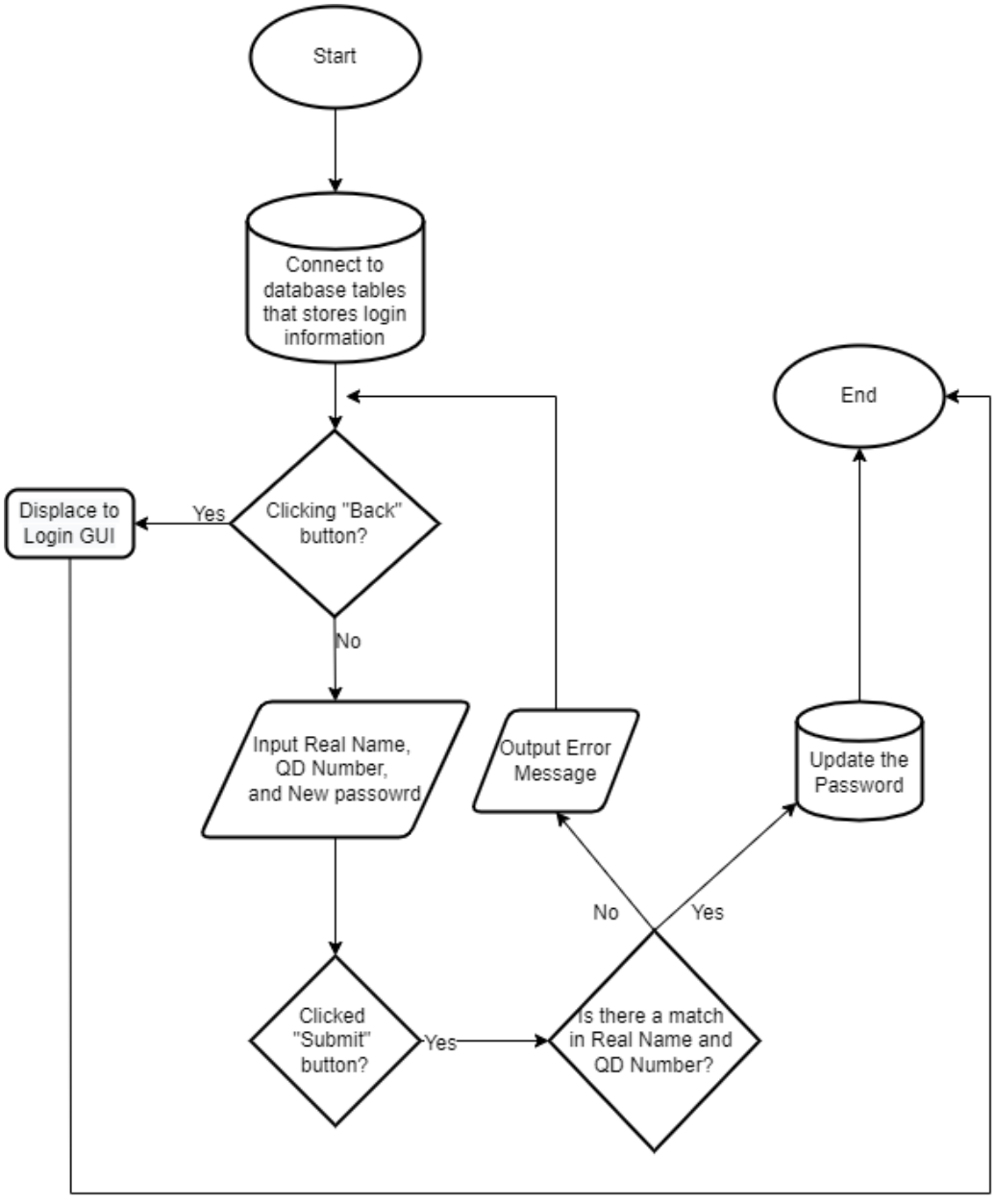
Login GUI:

This flowchart presents the different conditions in which the users may get into the entire QHAPPY Online System. The entered data in text fields where passwords and QD numbers are required would be compared to the three different databases and eventually see if there is a match. If either the QD Number or the password is incorrect, then an output message would show and the user would be required to input new username (QD number) and password.



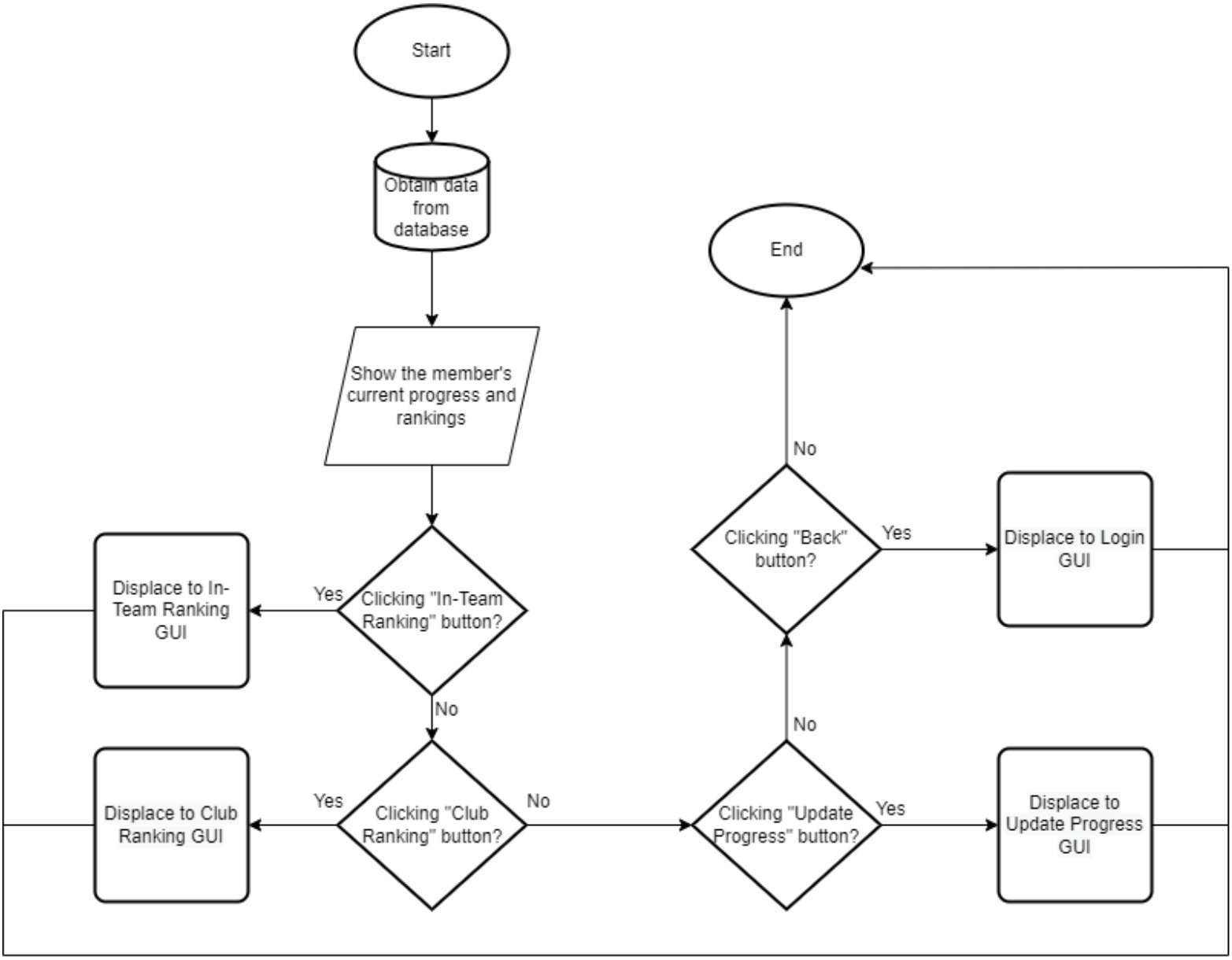
Reset Password GUI:

The flowchart below shows that in this GUI, the users need to put in their real name and QD number for validation (this information are rather private) and then be able to enter a new password. The input would, again, be compared to data already existing in the login database tables and if there is a match in name and QD number, then the new password can be set. Pressing the “cancel” button would lead the user back to the Login GUI.



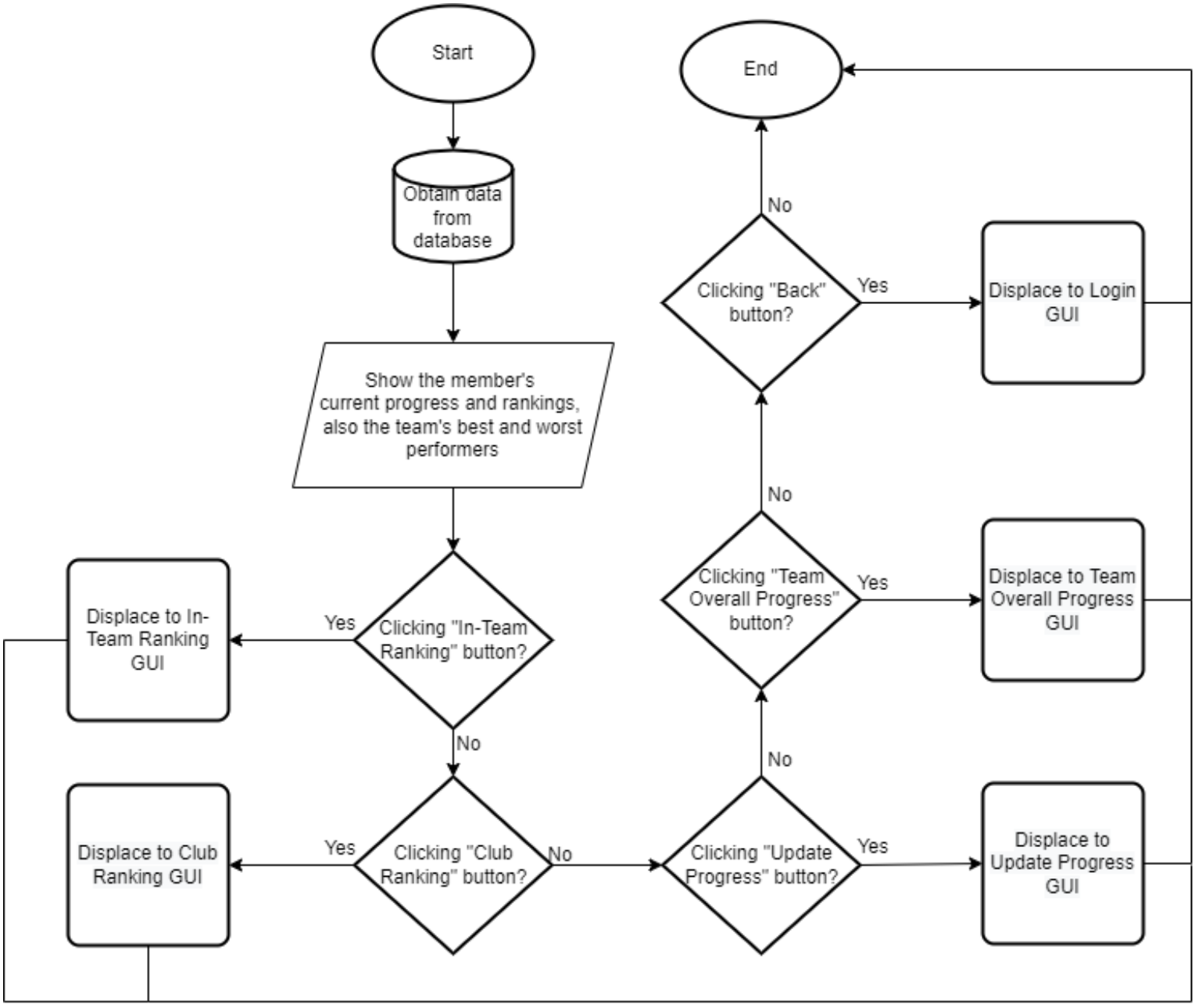
Team Member Page GUI:

For this GUI, the main function is navigation and showcase of important indexes. For the navigation function, when clicking on the different buttons, the GUI shall then lead the users to the relevant windows; also, the GUI should have a specified presentation of the specific member's progress and ranking.



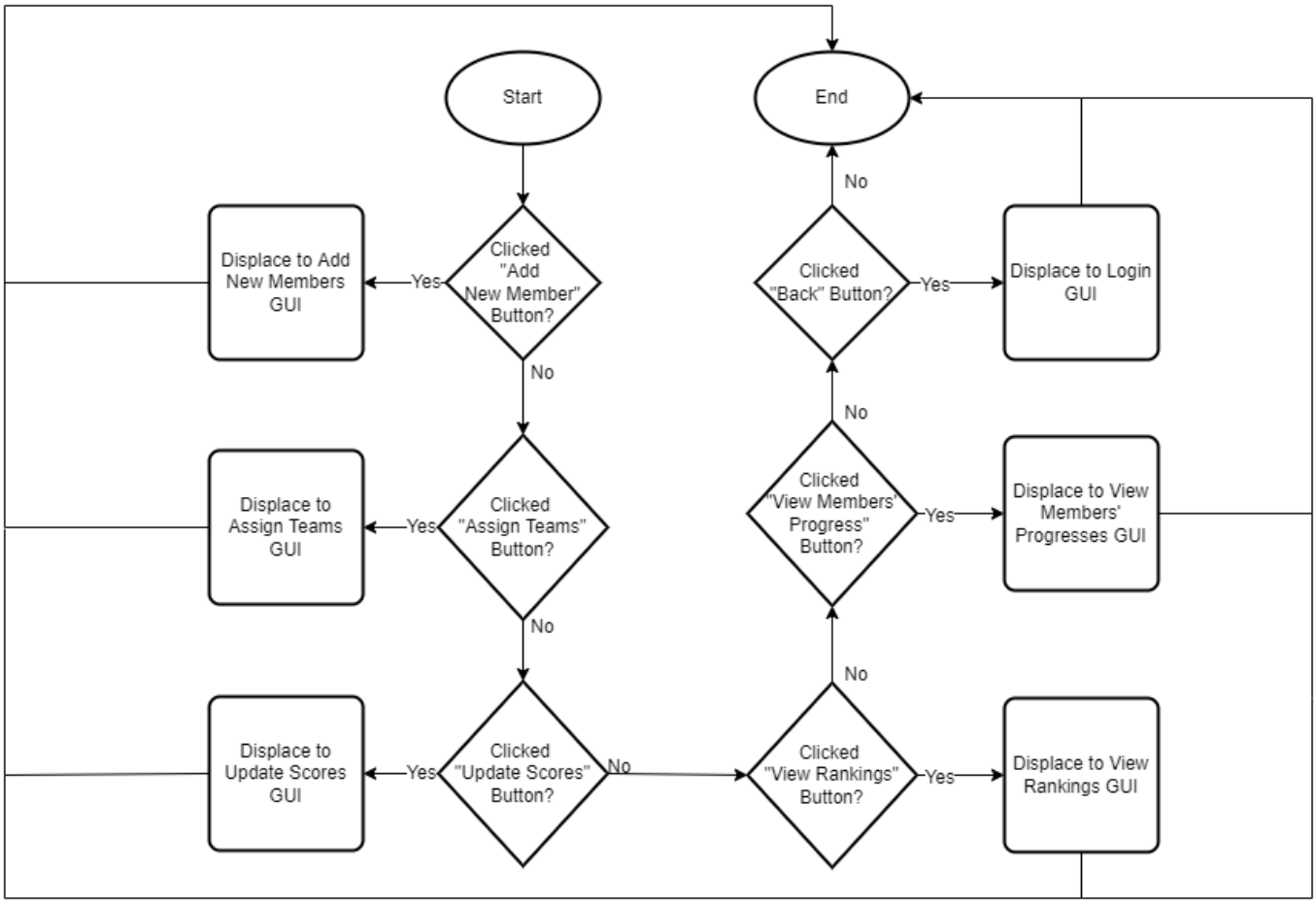
Team Leader Page GUI:

This GUI is an extension of the Team Member Page GUI. It would have a separate space showing the team leaders on the best and worst performer in team with regards to their rankings, and have an additional navigating button leading the team leaders to Team Overall Progress GUI.



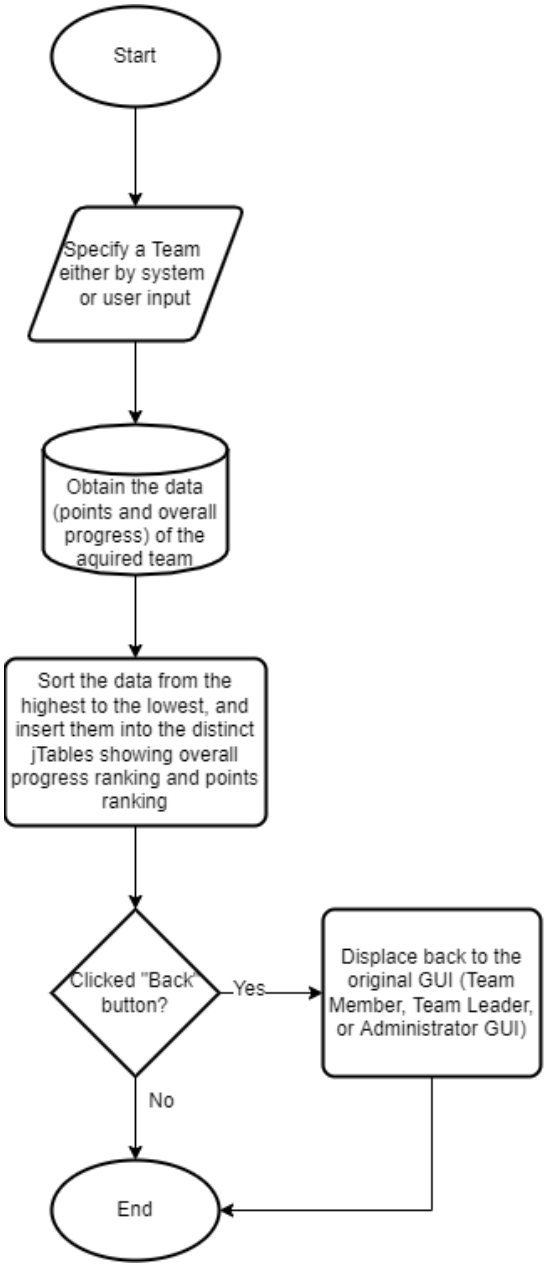
Administrator Page GUI:

This page is purely navigational. The buttons would lead to different additional GUI windows. The only exception is the “Retire” button. If the administrator indicates to retire, the system would check if there are other administrators and if there is one or more except from the current user, it would allow the club official to retire and then this specific user’s username and password would be deleted from the database table.



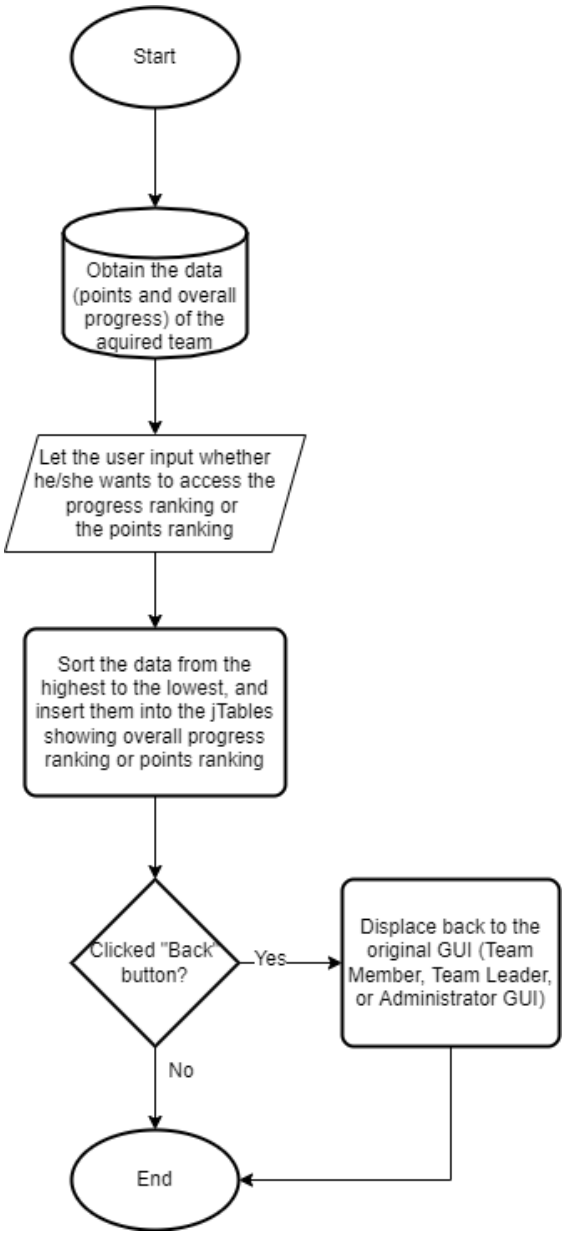
Team Ranking GUI:

The Team Ranking GUI would firstly specify a team that the team ranking is showing. Then, it would access the specific team after receiving the search requirements and present both the progress and the points ranking within the team. The ranking would be made by extracting the overall progress and points from database table and list them in a descending order from top to bottom. The “back” button should lead the users back to their relevant navigation window.



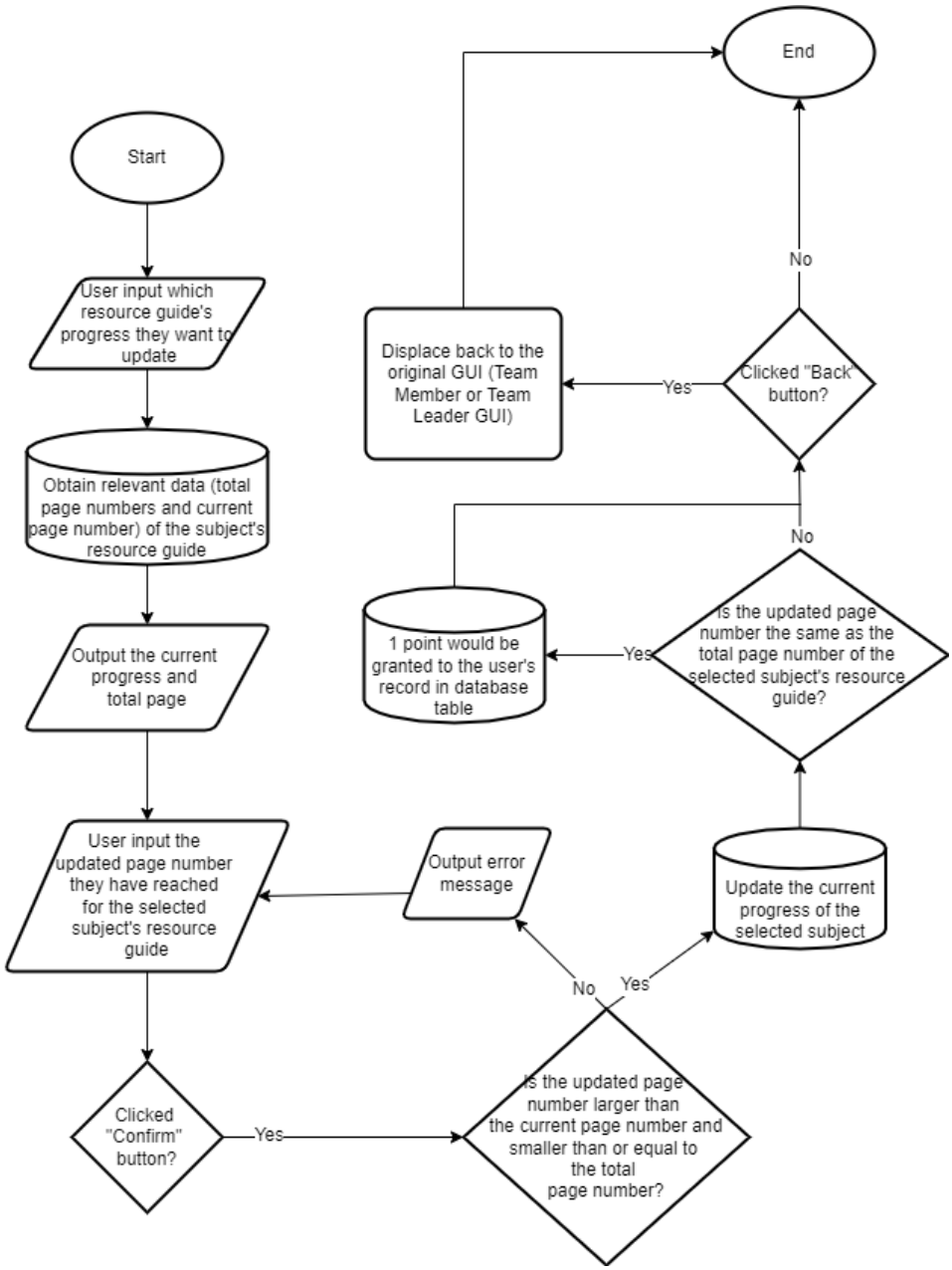
Club Ranking GUI:

The Club Ranking GUI basically does the same as Team Ranking GUI, except that it would showcase rankings of the whole club. Users may select from the selection bar whether they want to access the progress ranking, the points ranking, or the overall ranking. The overall ranking is calculated by adding up the rank number of a member's progress and points ranking and list them in an ascending order.



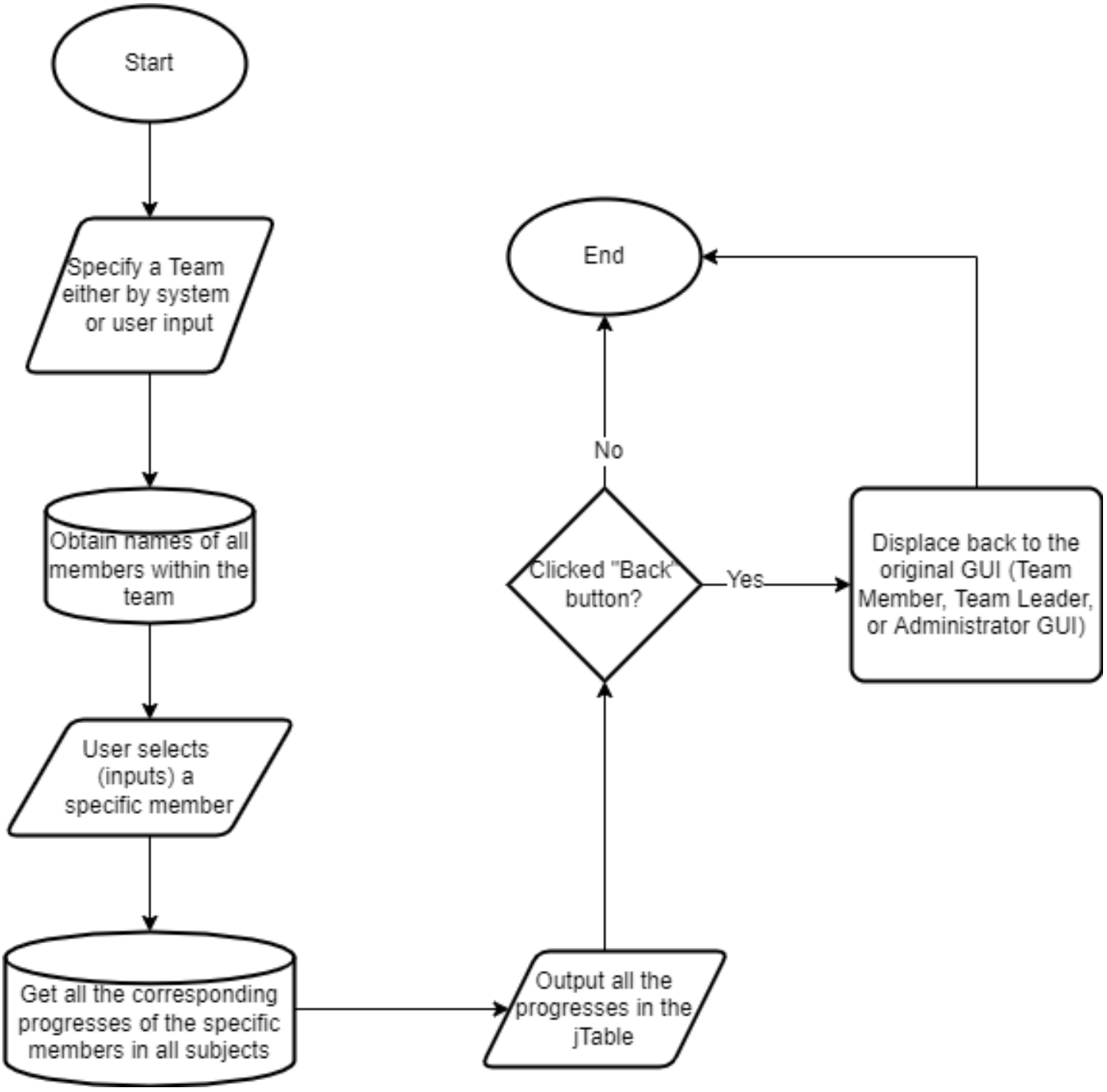
Update Progress GUI:

The users should select which book they have read, and the GUI shall then show which page number the user was on, and the total number that the resource guide of this subject contains. It is expected that the updated page number lies within the range bound by the two numbers mentioned in the previous sentence. With a success message, data in the database tables should then be updated. If they have finished the book, then 1 point should be added in their database table.



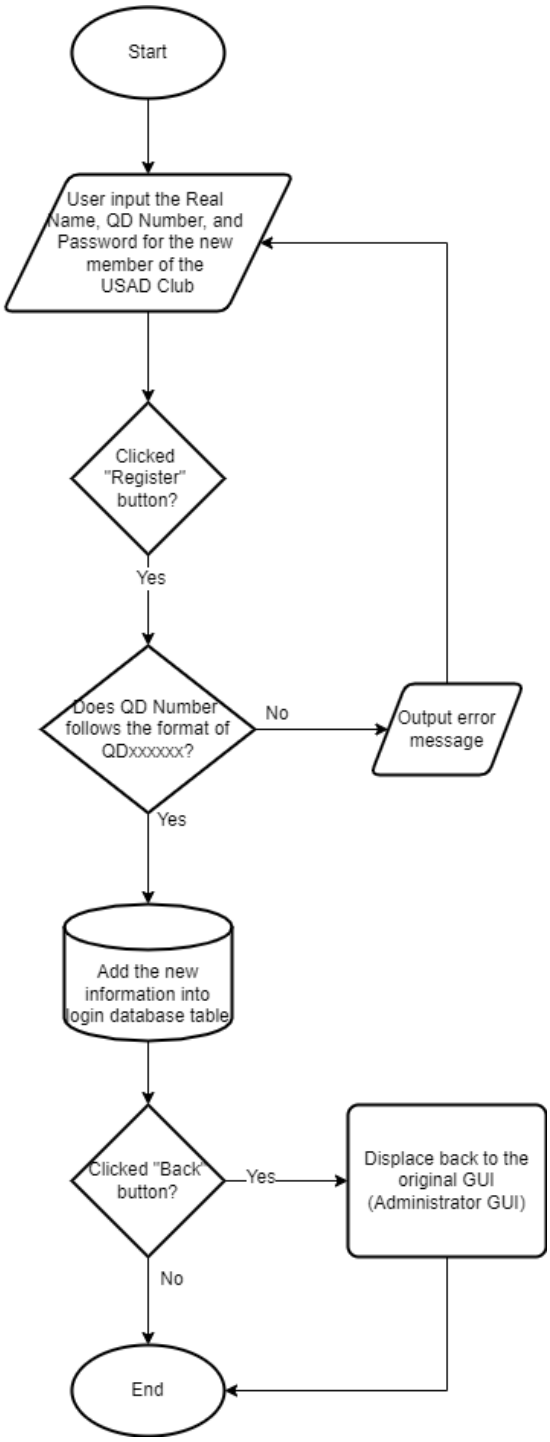
Team Members' Progresses GUI:

Again, this GUI would require the user to have a specific indication of which team that he/she wishes to access. Then, the selection bar would contain all the members of the team and the user need to select one of them to be able to view the holistic progress (progresses on each subject's resource guide).



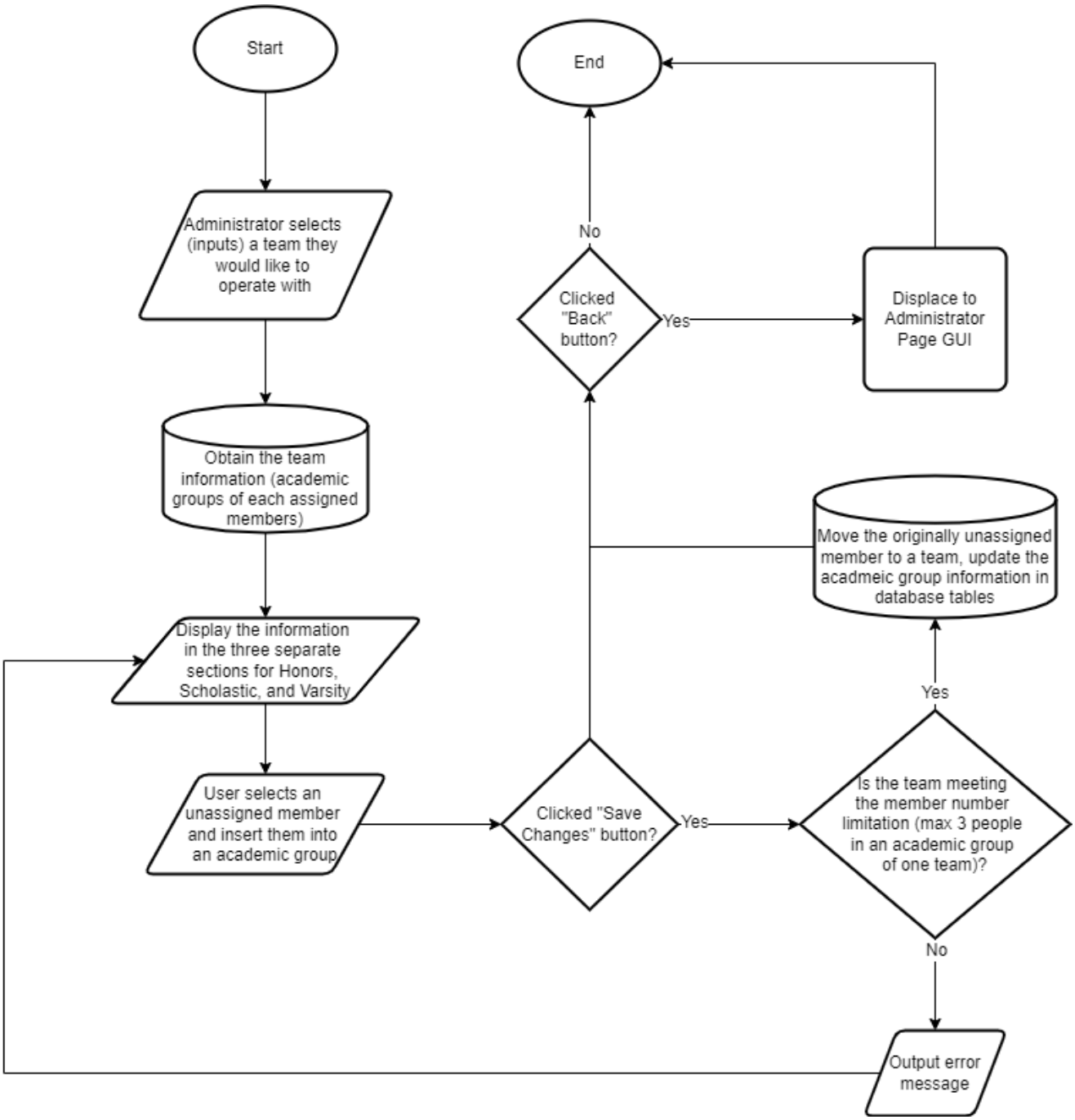
Add New Members GUI:

This GUI would ask the users to input the full name, the QD number, and a designated initial password from the administrator. The QD number should align with the format QDxxxxxx. After the data is entered and “register” is clicked, then the data would be stored into login database tables.



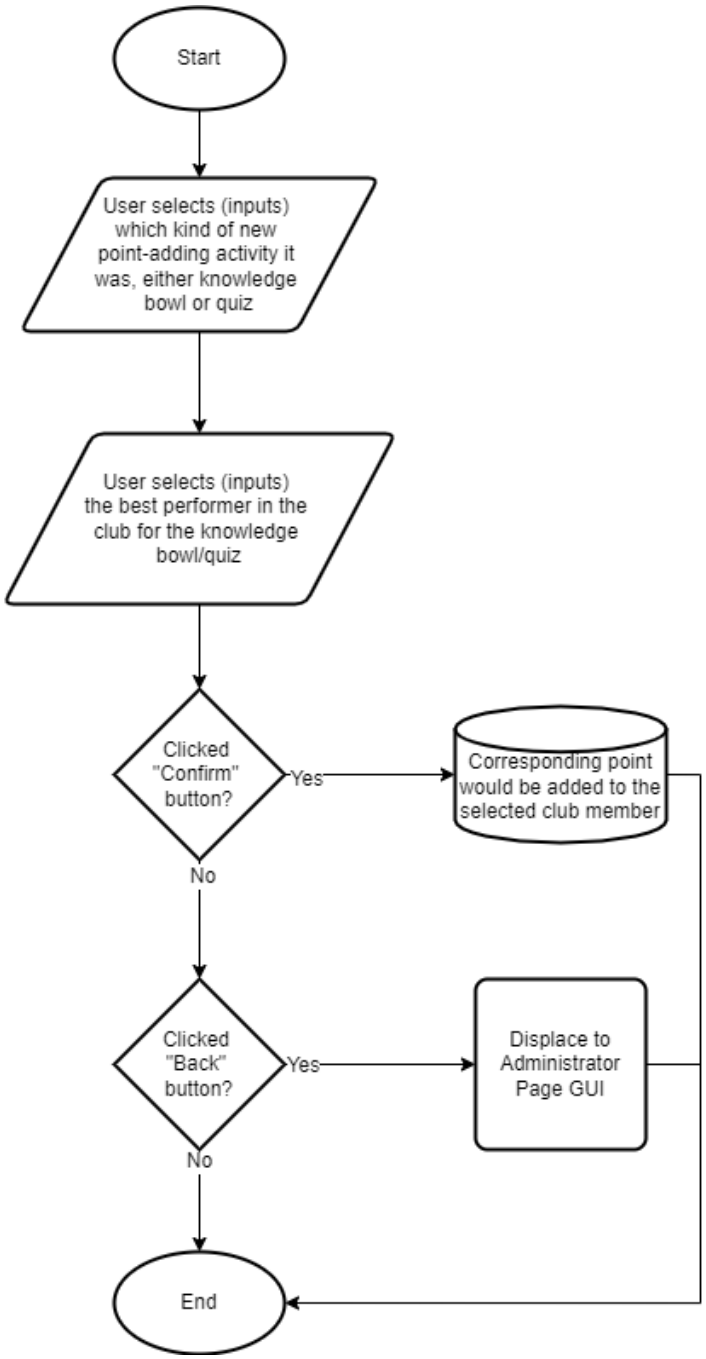
Assign Teams GUI:

The administrator, on this GUI, would firstly need to select a team, and the composition of the whole team would be shown, and the administrator can then select a specific academic group by clicking the + icon. Then, the list of unassigned members would be shown and the administrator could select a member from the list until the academic group is full (has three members).



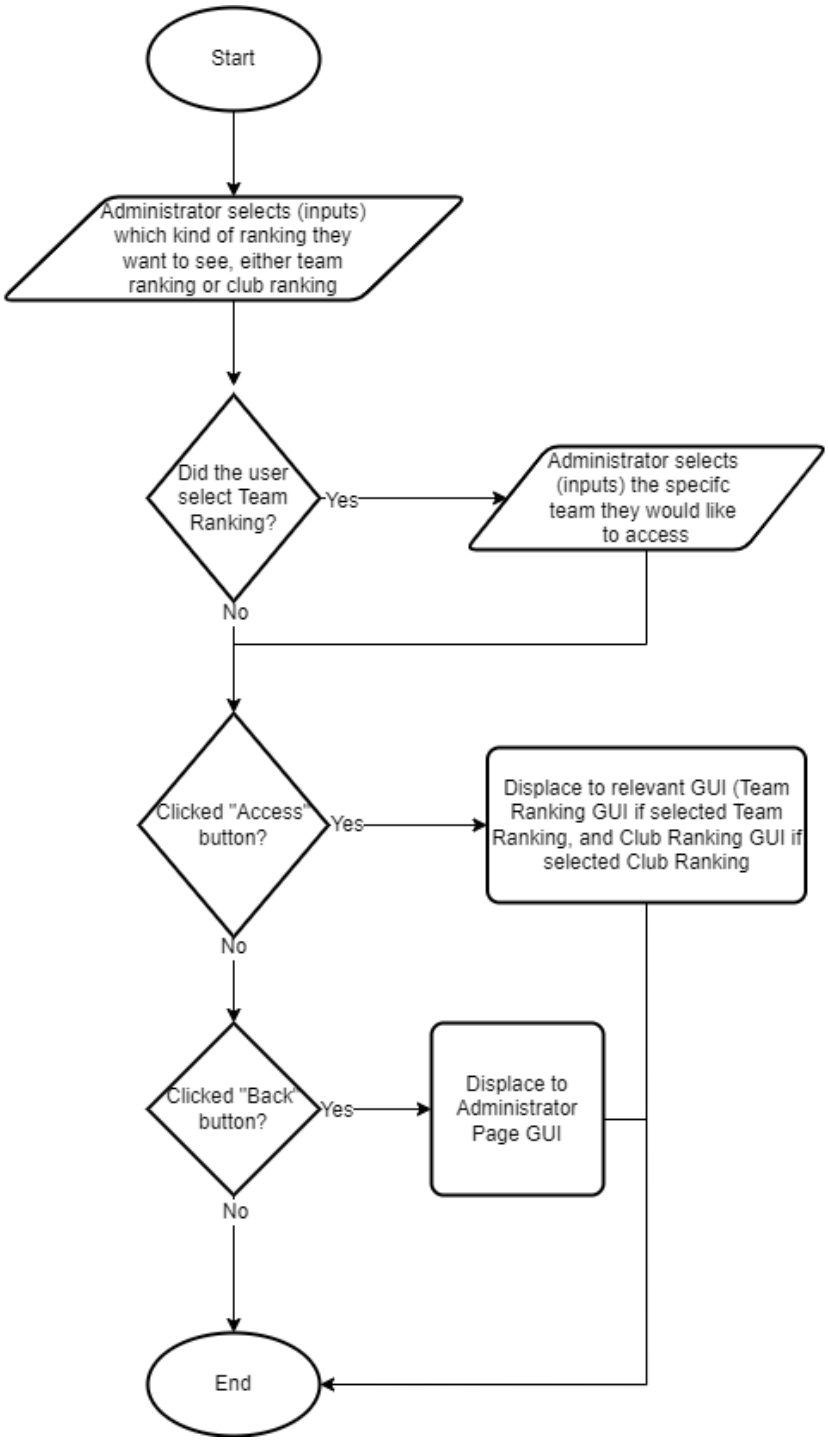
Update Score GUI:

In this GUI, the administrator would be required to select if the new score-updating form, either quiz or knowledge bowl, to determine the number of points that would be added. Then, the administrator would need to select a member in the club as the score earner. After clicking the “confirm” button, the point would be added to the selected member in the database table.



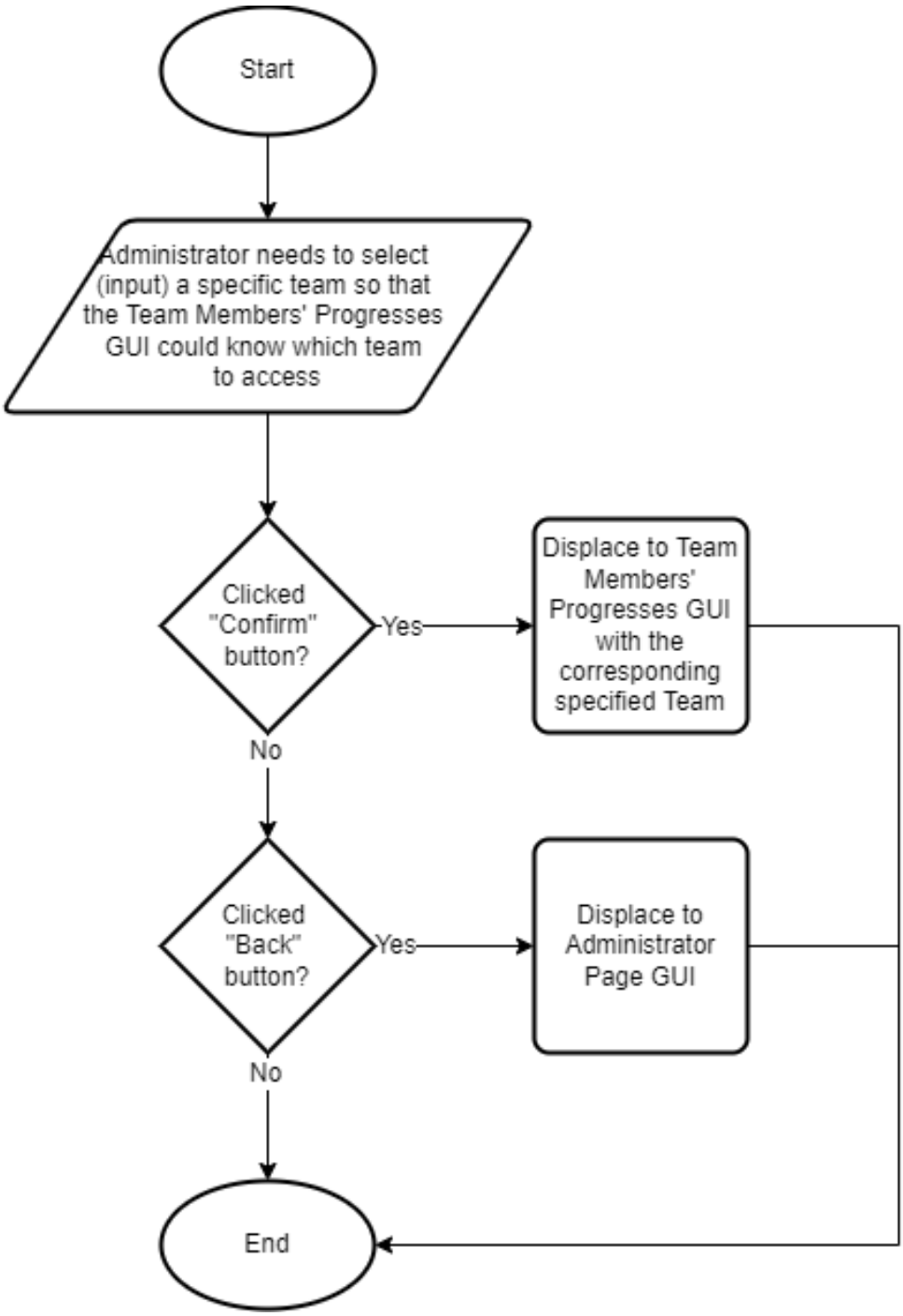
View Rankings GUI:

This window is to allow administrators to select which kind of ranking they would like to access, and if they select team ranking, then they should specify which team. Then, the system would lead them to relevant window that shows the ranking.



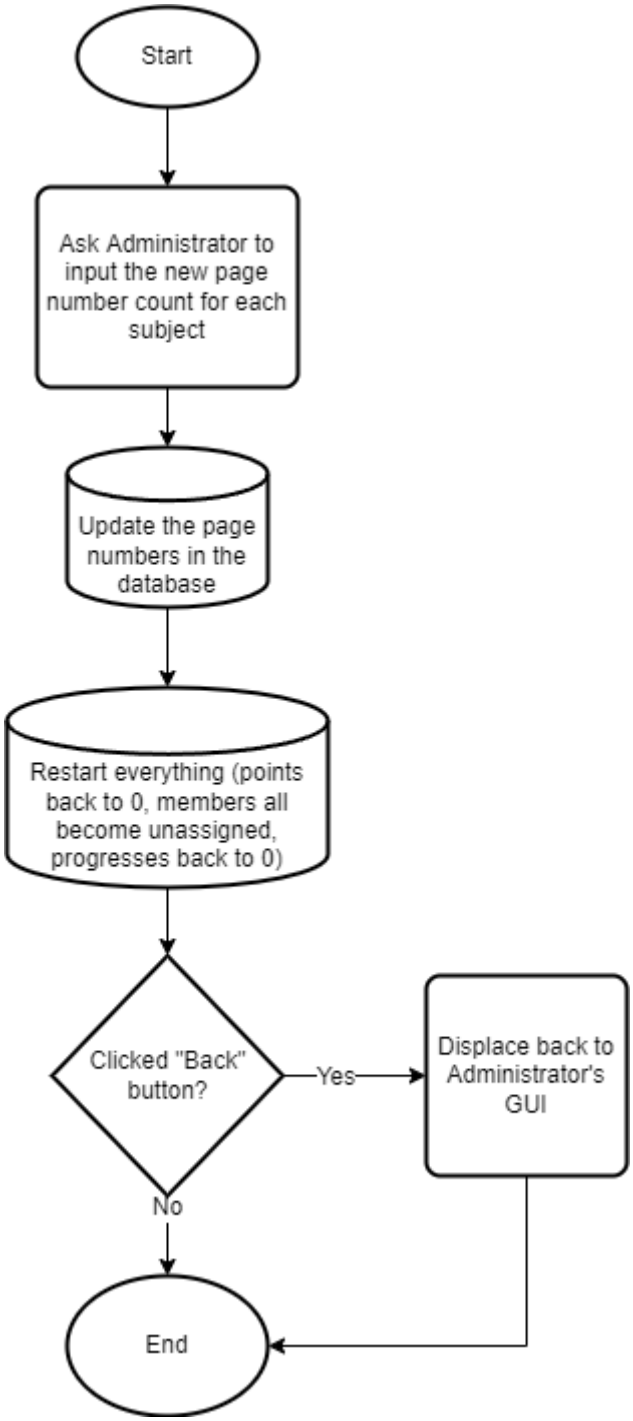
Accessing Members' Progress GUI:

If the club administrator wants to access progresses of a team of members, again, they would need to select which specific team, and then the system would pass on to show the corresponding window with the correct team, showing the members' progresses (Teams Members' Progresses GUI).



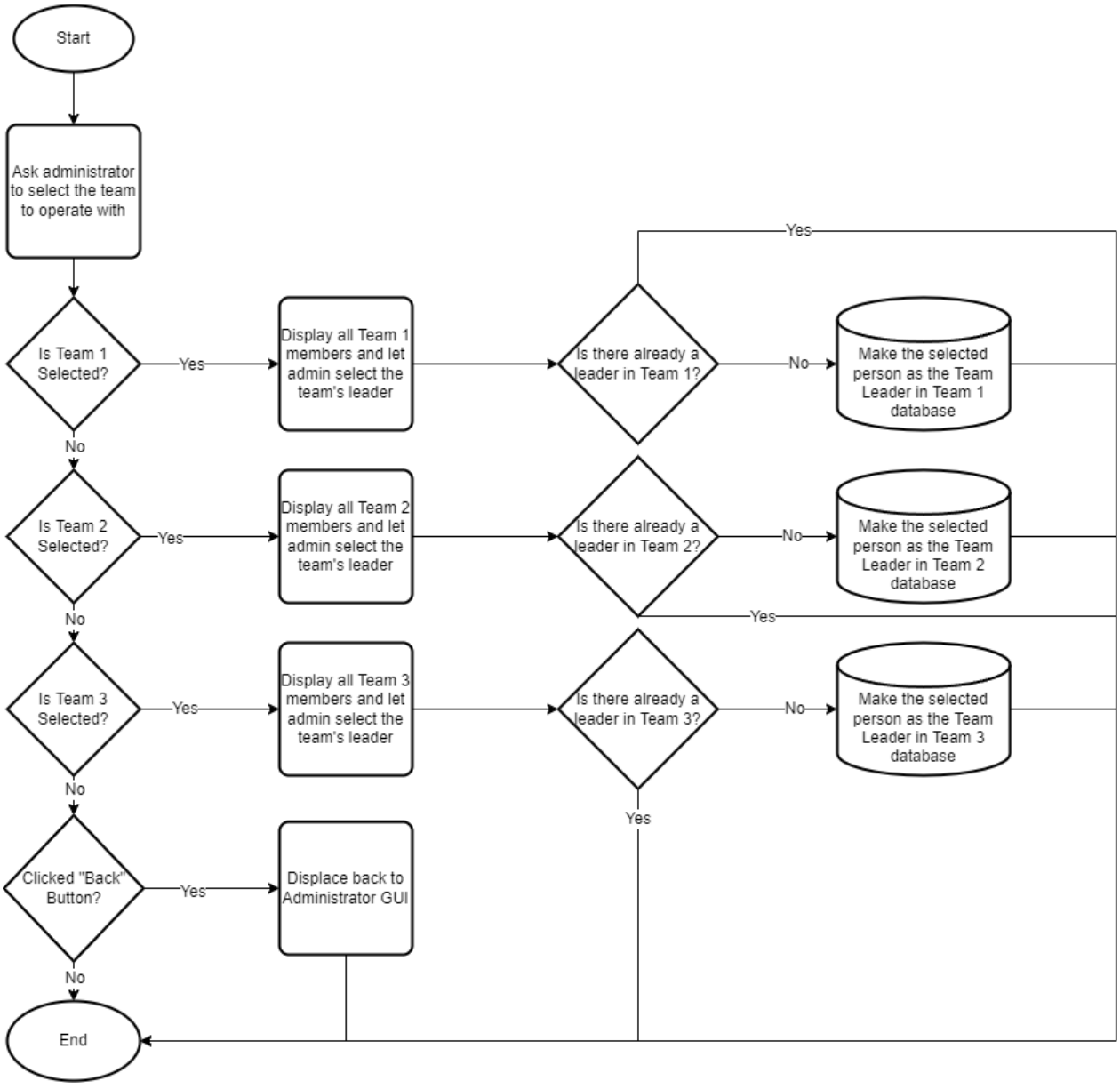
New Season GUI:

With a new season started, new study materials would be released. Then, for each subject, the total page number for their corresponding resource guides would change. So, the administrator needs to update the new page numbers.

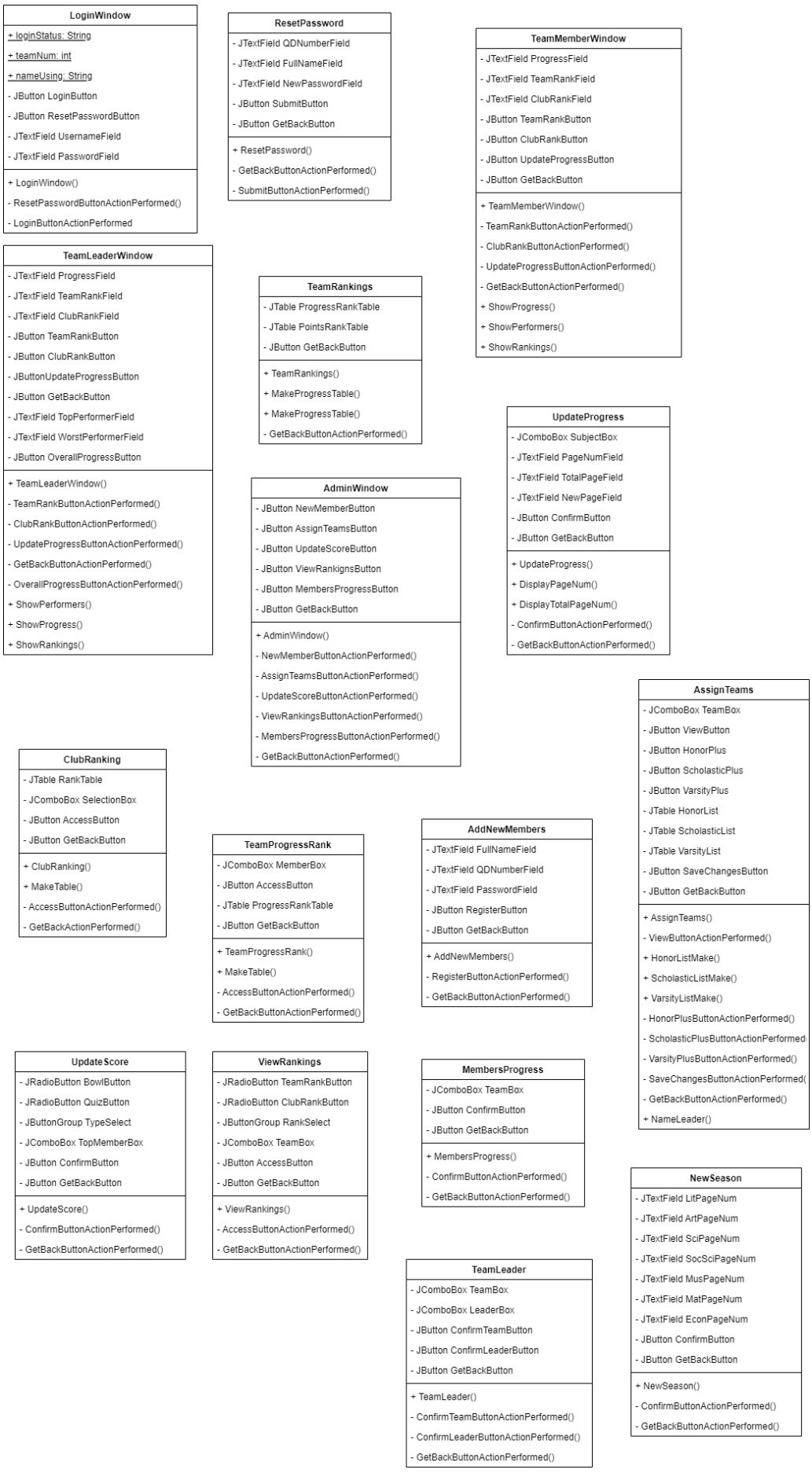


Team Member Select GUI:

The administrator could use this UI to select a specific leader for a specific team, and the database would make the corresponding changes. But this update can only be done when there are currently no leaders in the chosen team.



6. ***UML Diagrams***



The above image gives an overview on the different classes that would be included in the system, and their basic structures.

7. ***Databases and Tables***

DATABASE TABLES

- **MEMBER_LOGIN**

Field Name	Data Type	Description
NAME	Varchar	Stores the real names of the users
QDNUMBER	Varchar	User needs to input he correct username (QD Number because it is more personal, increases safety)
PASSWORD	Varchar	User needs to type in a corresponding password to login

- **TEAM_LEADER_LOGIN**

Field Name	Data Type	Description
NAME	Varchar	Stores the real names of the users
QDNUMBER	Varchar	User needs to input he correct username (QD Number because it is more personal, increases safety)
PASSWORD	Varchar	User needs to type in a corresponding password to login

- **ADMIN_LOGIN**

Field Name	Data Type	Description
NAME	Varchar	Stores the real names of the users
QDNUMBER	Varchar	User needs to input he correct username (QD Number because it is more personal, increases safety)
PASSWORD	Varchar	User needs to type in a corresponding password to login

- **TEAM1_MEMBERS**

Field Name	Data Type	Description
Name	Varchar	Including the full name of the corresponding user with a specified login
Lit_Progress	Integer	Recording the progress of the members in their readings of the literature resource guide
Art_Progress	Integer	Recording the progress of the members in their readings of the art resource guide
Sci_Progress	Integer	Recording the progress of the members in their readings of the science resource guide
SocSci_Progress	Integer	Recording the progress of the members in their readings of the social science resource guide
Mus_Progress	Integer	Recording the progress of the members in their readings of the music resource guide
Mat_Progress	Integer	Recording the progress of the members in their readings of the mathematics resource guide
Econ_Progress	Integer	Recording the progress of the members in their readings of the economics

		resource guide
Overall_Progress	Integer	Recording the progress of the members in their readings of all of the resource guides
POINTS	Integer	Recording the points that members have earned throughout the season
Team_Leader	Boolean	See if the member is the team leader
GROUP	Varchar	Record the academic group that the member is in. Could only be one of the three academic groups.

- **TEAM2_MEMBERS**

Field Name	Data Type	Description
Name	Varchar	Including the full name of the corresponding user with a specified login
Lit_Progress	Integer	Recording the progress of the members in their readings of the literature resource guide
Art_Progress	Integer	Recording the progress of the members in their readings of the art resource guide
Sci_Progress	Integer	Recording the progress of the members in their readings of the science resource guide
SocSci_Progress	Integer	Recording the progress of the members in their readings of the social science resource guide
Mus_Progress	Integer	Recording the progress of the members in their readings of the music resource guide
Mat_Progress	Integer	Recording the progress of the members in their readings of the mathematics resource guide
Econ_Progress	Integer	Recording the progress of the members in their readings of the economics

		resource guide
Overall_Progress	Integer	Recording the progress of the members in their readings of all of the resource guides
POINTS	Integer	Recording the points that members have earned throughout the season
Team_Leader	Boolean	See if the member is the team leader
GROUP	Varchar	Record the academic group that the member is in. Could only be one of the three academic groups.

- **TEAM3_MEMBERS**

Field Name	Data Type	Description
Name	Varchar	Including the full name of the corresponding user with a specified login
Lit_Progress	Integer	Recording the progress of the members in their readings of the literature resource guide
Art_Progress	Integer	Recording the progress of the members in their readings of the art resource guide
Sci_Progress	Integer	Recording the progress of the members in their readings of the science resource guide
SocSci_Progress	Integer	Recording the progress of the members in their readings of the social science resource guide
Mus_Progress	Integer	Recording the progress of the members in their readings of the music resource guide
Mat_Progress	Integer	Recording the progress of the members in their readings of the mathematics resource guide
Econ_Progress	Integer	Recording the progress of the members in their readings of the economics

		resource guide
Overall_Progress	Integer	Recording the progress of the members in their readings of all of the resource guides
POINTS	Integer	Recording the points that members have earned throughout the season
Team_Leader	Boolean	See if the member is the team leader
GROUP	Varchar	Record the academic group that the member is in. Could only be one of the three academic groups.

- **MEMBERS_UNASSIGNED**

Field Name	Data Type	Description
Name	Varchar	Including the full name of the corresponding user with a specified login
Lit_Progress	Integer	Recording the progress of the members in their readings of the literature resource guide
Art_Progress	Integer	Recording the progress of the members in their readings of the art resource guide
Sci_Progress	Integer	Recording the progress of the members in their readings of the science resource guide
SocSci_Progress	Integer	Recording the progress of the members in their readings of the social science resource guide
Mus_Progress	Integer	Recording the progress of the members in their readings of the music resource guide
Mat_Progress	Integer	Recording the progress of the members in their readings of the mathematics resource guide
Econ_Progress	Integer	Recording the progress of the members in their readings of the economics

		resource guide
Overall_Progress	Integer	Recording the progress of the members in their readings of all of the resource guides
POINTS	Integer	Recording the points that members have earned throughout the season
Team_Leader	Boolean	See if the member is the team leader
GROUP	Varchar	Record the academic group that the member is in. Could only be one of the three academic groups.

- **PAGE_NUMBERS**

Field Name	Data Type	Description
Lit_Page	Integer	Storing the total page number of literature resource guide. To be updated when a new season starts as new resource guides are published
Art_Page	Integer	Storing the total page number of literature resource guide. To be updated when a new season starts as new resource guides are published
Sci_Page	Integer	Storing the total page number of literature resource guide. To be updated when a new season starts as new resource guides are published
SocSci_Page	Integer	Storing the total page number of literature resource guide. To be updated when a new season starts as new resource guides are published
Mus_Page	Integer	Storing the total page number of literature resource guide. To be updated when a new season starts as new resource guides are published
Mat_Page	Integer	Storing the total page number of literature

		resource guide. To be updated when a new season starts as new resource guides are published
Econ_Page	Integer	Storing the total page number of literature resource guide. To be updated when a new season starts as new resource guides are published

8. ***Test Plan***

Success Criteria	Nature of Test	Expected Result
The login window should be able to differentiate team leaders, team members, and administrators, and lead the users to their corresponding windows	1. Unit Test: Testing if the login window can operate and lead the users into their windows; 2. Validation Test: Try a QD Number not in the form of QDxxxxxx 3. Validation Test: Try a username and password not in either of the three databases	1. Successfully logging into corresponding windows; 2. Relevant error message would output if the QD Number is not in the form of QDxxxxxx; 3. Error message would pop up indicating wrong username or password.
The administrators could create new accounts for new team members, manage the teams, and assign team leaders. An error message should pop up if the team or one of its groups has too much students	1. Functional Test: See if the administrator could be able to operate these mechanisms; 2. Validation test: try to assign a new member into a team already with maximum number of members (9)	1. All of the functions can work deliberately and swiftly for an administrator of the club; 2. A relevant error message should indicate such an error, saying that no new members could be assigned.
Administrators could add a new test/knowledge bowl and input the result into the system, while the system	1. Functional Test: Check whether administrator could add a game; 2. Function Test: add a new quiz/knowledge bowl and check	1. Administrators should be able to add new quiz/knowledge bowl; 2. Real-time ranking should be

would grant corresponding points to the members with the highest score	the ranking	updated
On a team leader's window, the student may be able to view all of his/her team members' progresses	1. Functional Test: Check if a team leader user could have access to all of his/her team members' progresses	1. There should be a window showing all of the team members' progresses within a specific team for that the team leader could view
Both the students and the administrators should be able to view real-time rankings.	1. Functional Test: Go into all users' windows and click to access the rankings;	1. Rankings should be able to be viewed by all users of the system;
Students could update their current progresses on reading the resource guides. The administrator should be able to click into a tab that can show each student's progress	1. Unit Test: See if the update section of the whole system can work; 2. Type Check: try input a string of characters when the update window is asking for an integer page number output; 3. Range Check: the updated page number should be larger than the previous record, and within the total page number recorded in the database table.	1. The progresses should be able to update accordingly with the new data entries; 2. An error message should pop up indicating that an integer input is expected; 3. An error message would pop up indicating that the updated page number is out of range and the user should re-input a new integer that is within the range.

Club members' overall progress in reading the resource guides should be presented in their window	1. Functional Test: A progress should be shown on the club members' windows	1. The percentage should correspond to the overall_progress stored in the database
Proper error messages would pop up where necessary (e.g., the students are stating that they are reading onto a page number that exceeds the total page number, login failed, etc.)	1. Validation Test: try all places where potential errors, especially with regards to data type and range, could occur.	1. Relevant error messages indicating the specific problem should pop up and let the user to re-enter a new data.