Project Interim Demo

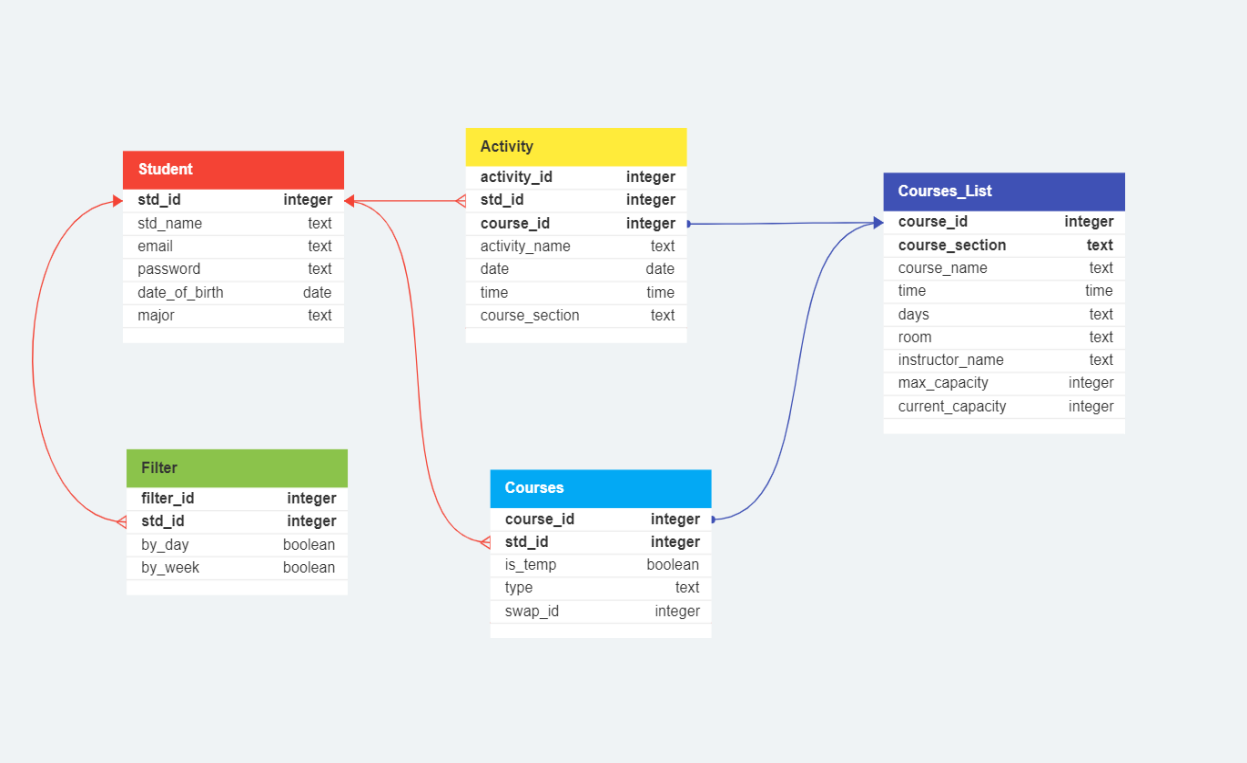
- Hussain Umer Farooqui – 08040
- Muhammad Shayan Wasif – 08071
- Salman Adnan - 07885

Enrollment Management System

Description:

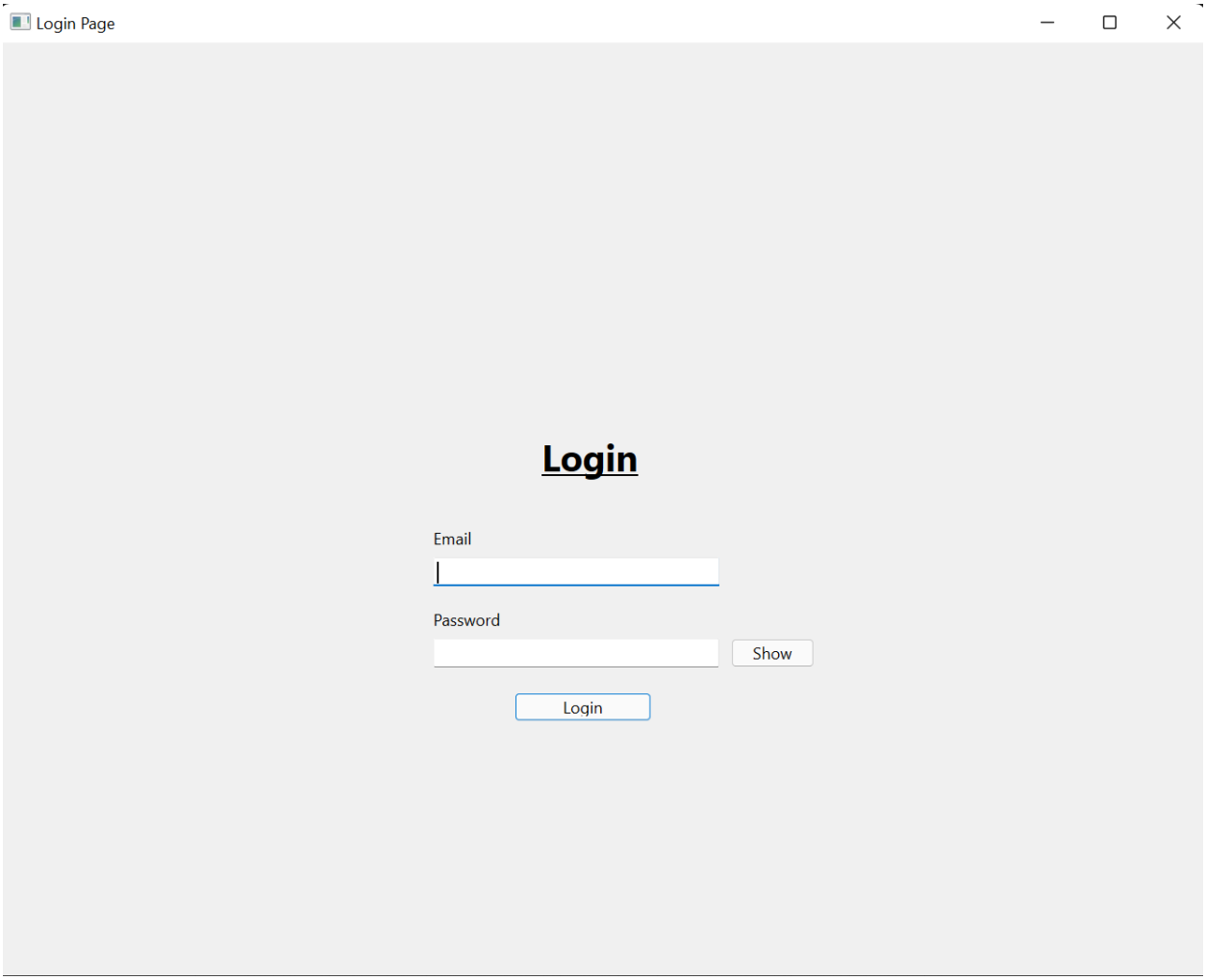
Our project centers on an enrollment system, enabling students to construct their timetables from a provided list of courses. Additionally, students have the flexibility to make adjustments, including adding, dropping, or swapping courses up-to a specific date. The system will incorporate safeguards to prevent any scheduling conflicts. Each student will be provided with unique credentials for secure access to the system.

Relational Schema:



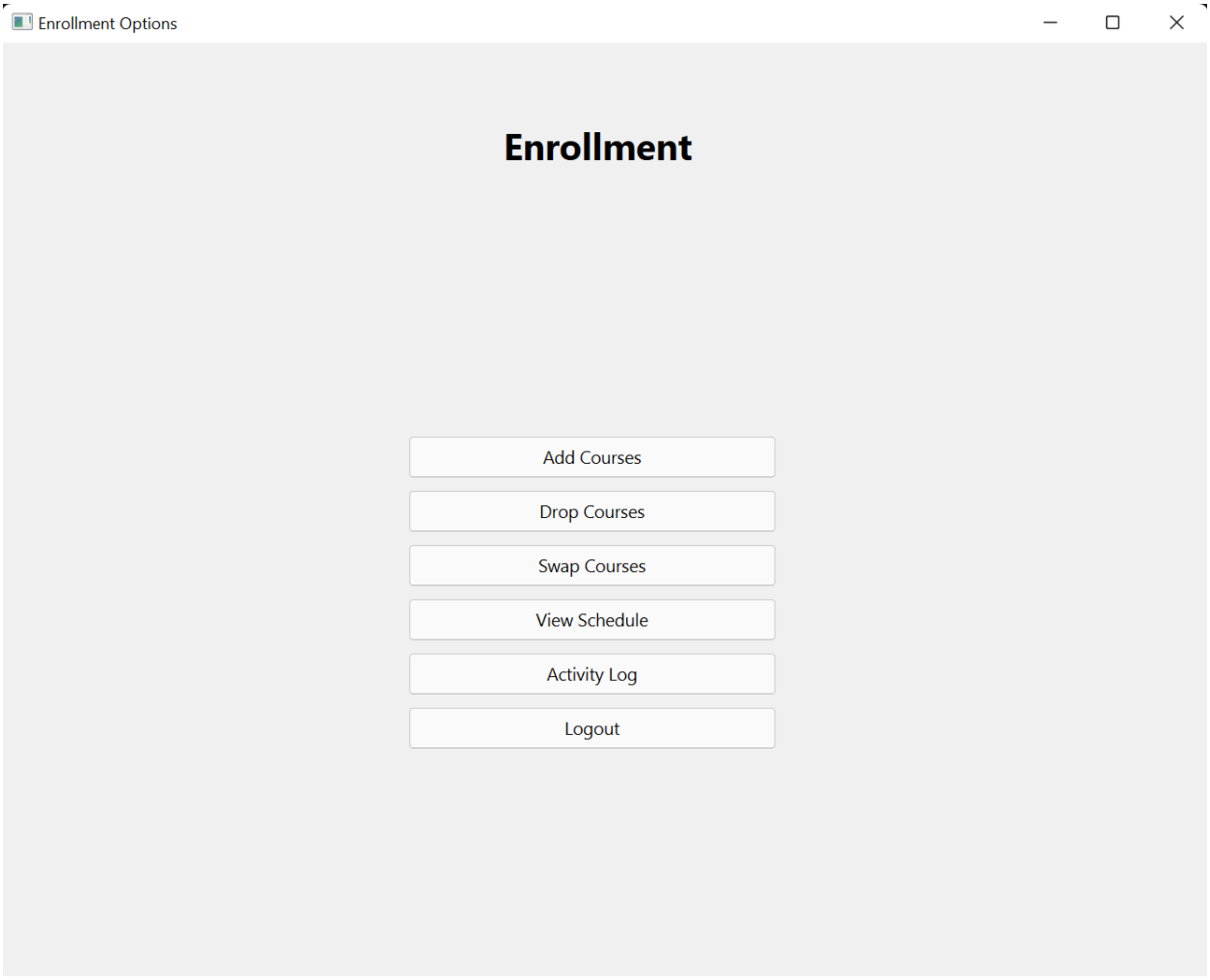
UI Screens:

1. Login Screen:



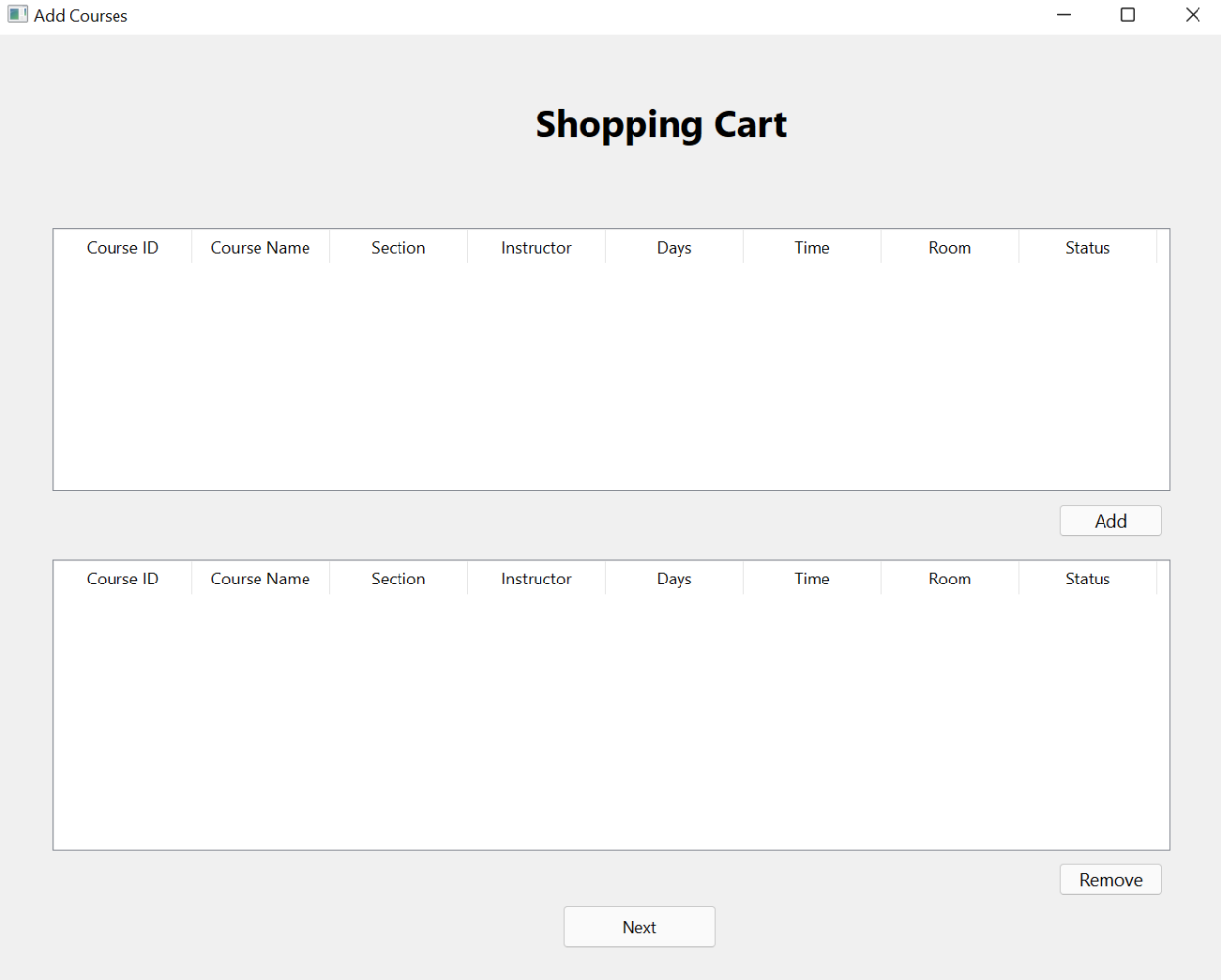
The first screen of our enrollment management system is the login page. The credentials of the students will be stored in the “Students” table. The user will enter his student email and password which will be checked from the Student database. If the credentials entered are correct the user will have proceed to the next screen else the dialog box will pop up showing ‘Invalid Email or Password’.

2. Enrollment Options:



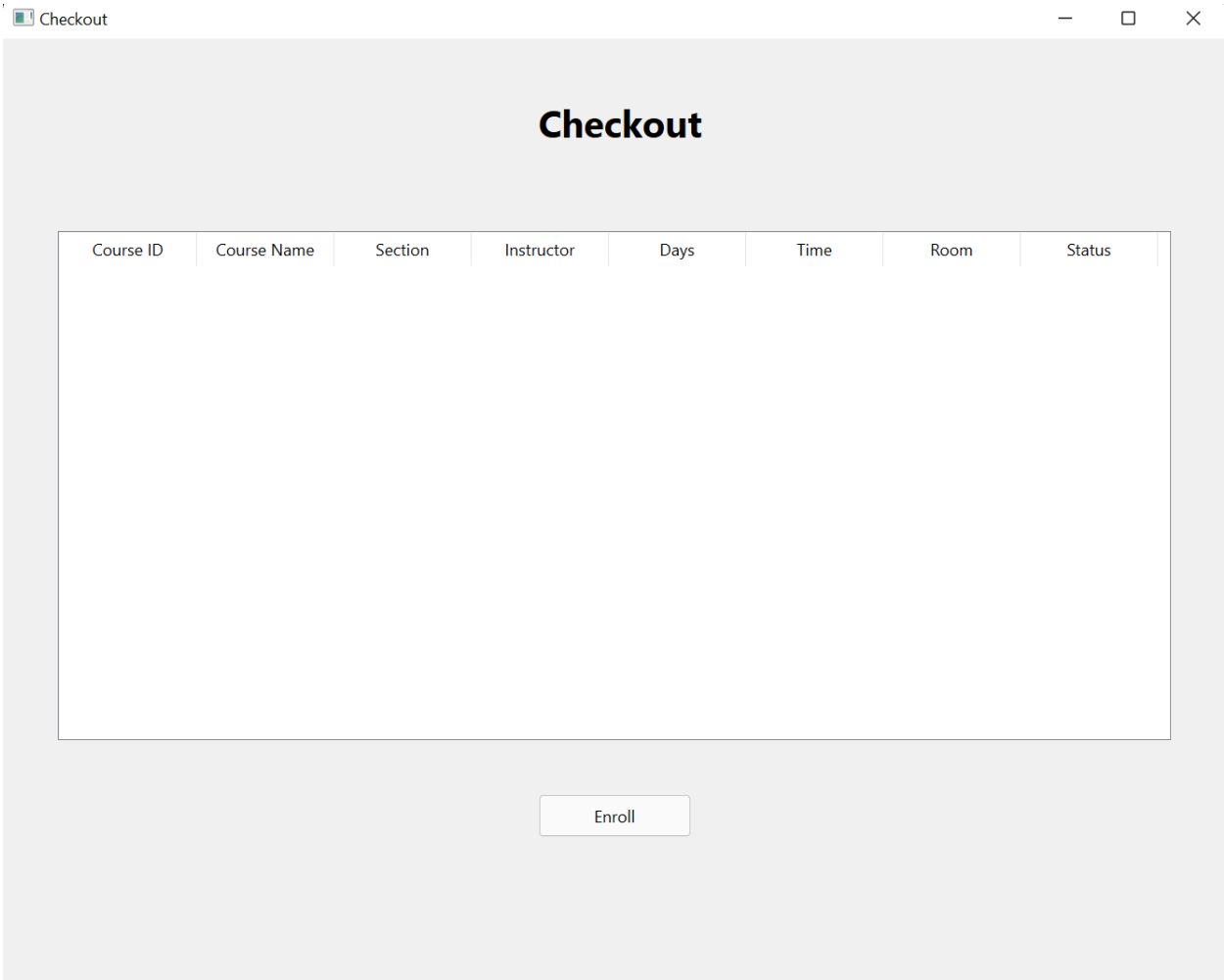
This screen shows all the options that this management system offers and clicking on one of the buttons will take them to the corresponding screen. Once the user proceeds on the relevant screen and perform their task then closes that window they will be brought back to this options menu.

3. Add Courses:



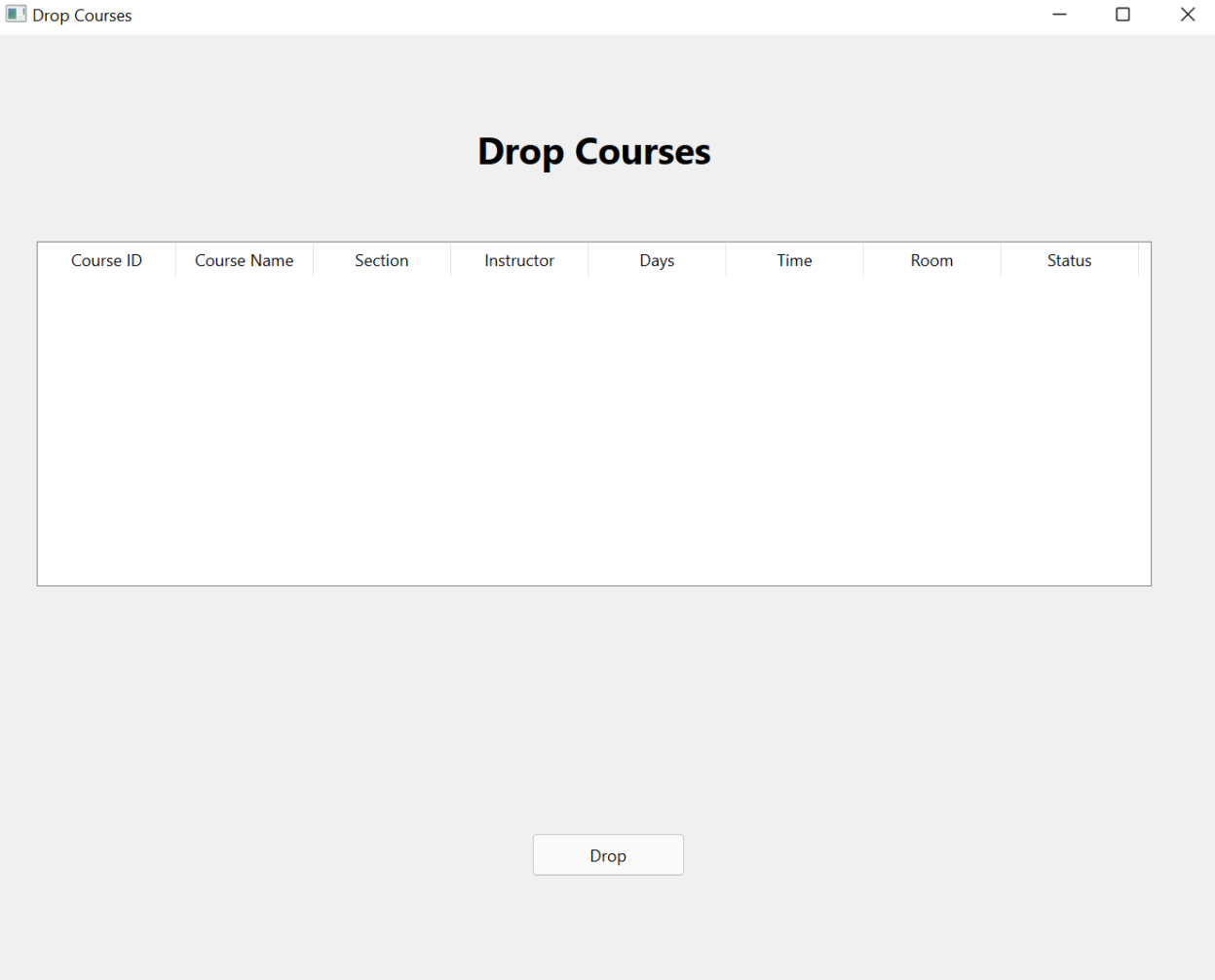
This screen features the shopping cart where the user adds his desired courses from the given list of courses stored in the “Courses_List” table. The first box contains all of the courses saved in the database, and as he selects a course and clicks 'add,' that course is added to the second box. The 'is_temp' value in the "Courses" table then becomes 'True', and the 'type' value in the "Courses" table becomes 'create'. Then he can go to the checkout page by clicking Next.

4. Checkout(Enroll) :



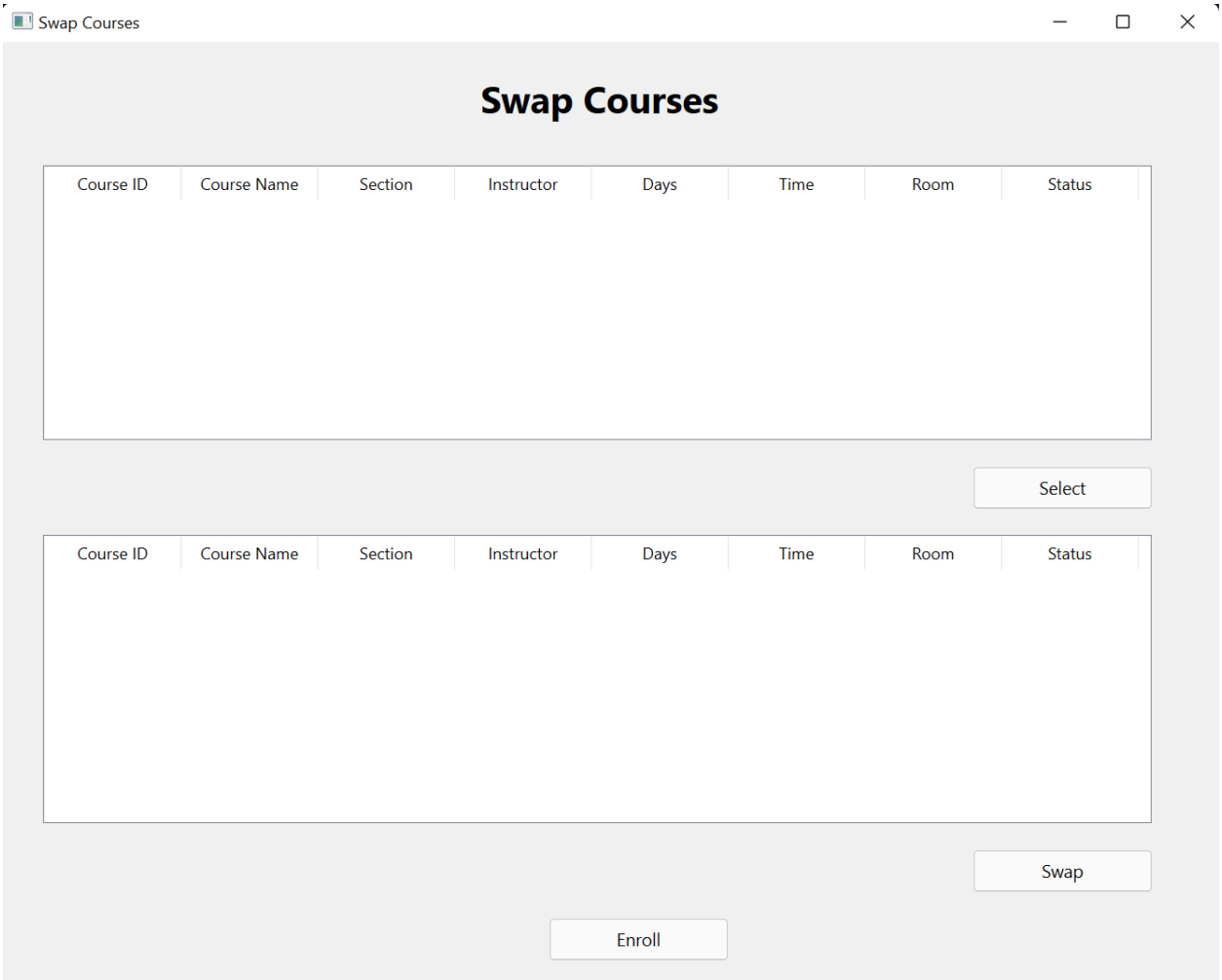
This screen shows all the courses added in the shopping cart and once the user clicks on the Enroll button two conditions are checked; for every course whether the current capacity does not equal to zero through the “Courses_List” table and time conflict is checked comparing the timings of the courses in shopping cart. If both requirements are met, enrollment in that specific course is successful, and a dialog box appears listing all of the enrolled courses. In the “Courses” table, the ‘is_temp’ value becomes ‘False’ and the ‘type’ value becomes 'null' once the enrollment is successful.

5. Drop Courses:



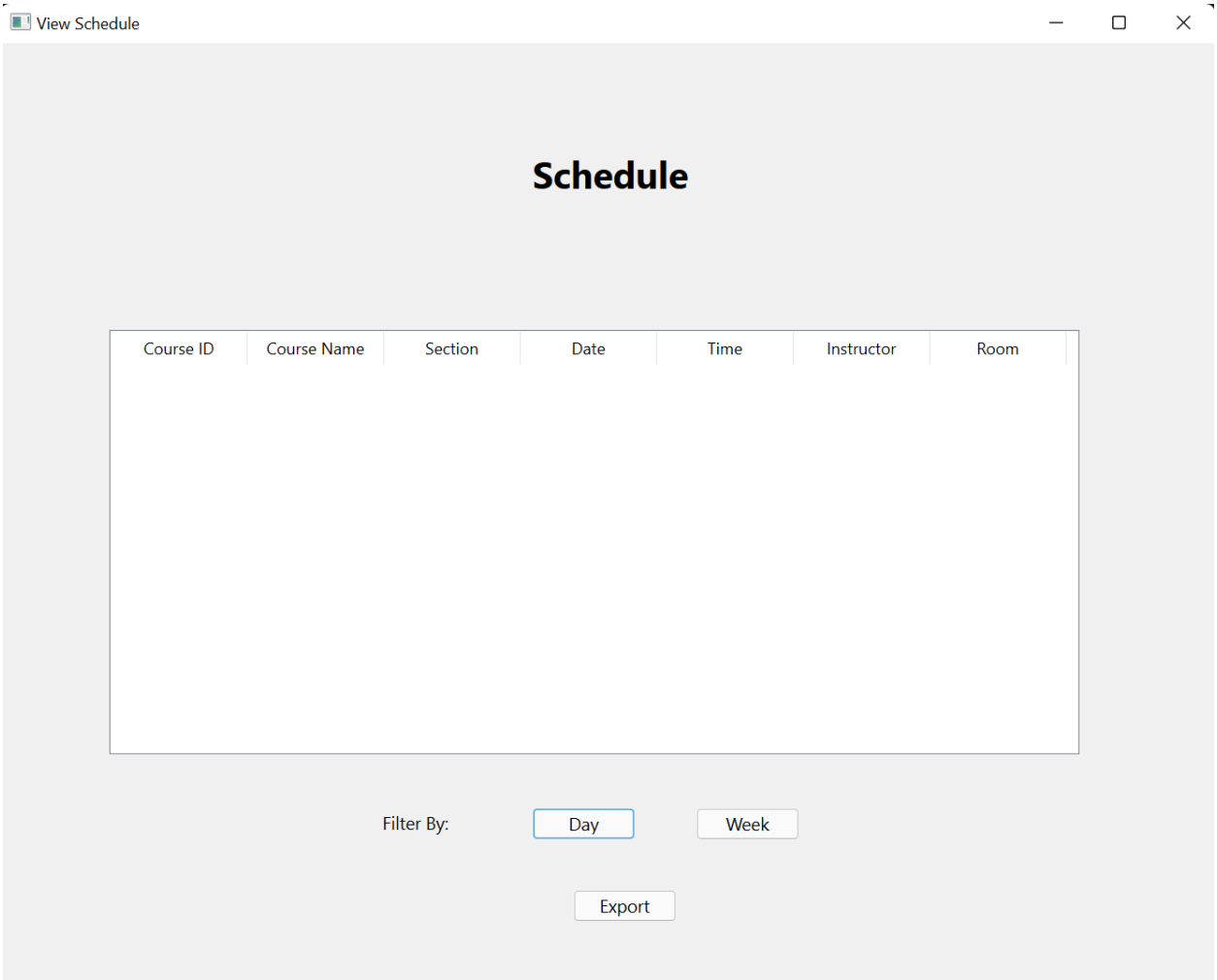
This screen covers the drop course feature of our enrollment management system. Once the user has enrolled themselves in the courses, those courses will be shown in this table. And those courses will be retrieved through the “Courses” table. As the user selects any course and clicks drop, a dialog box will ask for confirmation and then that course will be deleted from his courses in the “Courses” table.

6. Swap Courses:



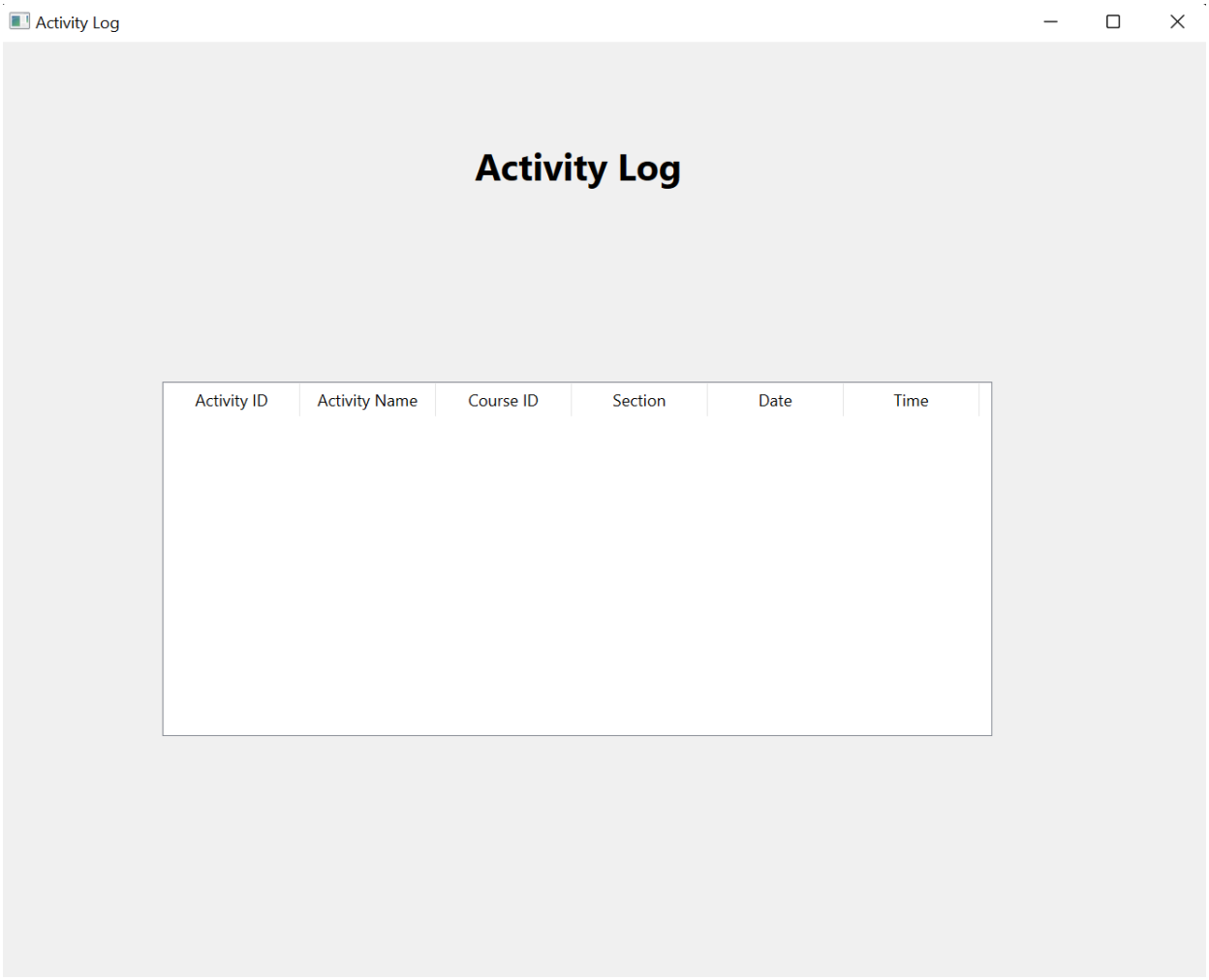
This screen features the swap courses option of the enrollment system. The first table in the screen would show all the courses in which user has not enrolled and their current capacity is greater than zero. These courses would be retrieved from the “Courses_List” table excluding the courses in the ‘courses’ table also considering the current capacity. Once the user chooses any course from the first table and click on the select button then the second table would show any possible swaps with the user’s current enrolled courses. The ‘is_temp’ value will be changed to ‘True’, the ‘type’ value will be changed to ‘swap’, and the ‘swap_id’ value will be changed to the ‘course_id’ of the course from the "Courses" table that we are swapping from. Then, clicking on swap will display a swap confirmation message, and clicking on Enroll will confirm the swap.

7. View Schedule:



This screen shows the user their weekly or daily schedule by choosing the filter option. The courses will be retrieved from the “Courses” table where ‘std_id’ is a foreign key to “Student” table and ‘std_id’ of “Filter” table is foreign key to the “Student” table. Through this relation we will retrieve all the courses of the user. If the user wants to download their schedule they can use the option of export.

8. Activity Log:



This screen shows all the activity performed by the user including swap, add, view, and filter. If the activity we are doing is create timetable, remove course, and swap course, then, the activity is stored in the "Activity" table where 'course_id' is the course we are doing the changes on, 'activity_name' is the name of the activity e.g. create timetable, date and time is current date and time, and course section is the section of the course we are adding, removing, swapping. If the activity is filter by day, filter by week, or export timetable from View Schedule Screen, then, 'course_id' and 'course_section' is null, and activity_name becomes the activity we are doing e.g. filter by week.