

Change Impact for Evol 3 TEAM 8 Final

In Evol 3, our team was requested to make the changes below:

- Add JavaFX GUI for Attendance Taking
- Add JavaFX GUI for Admin App
- Block student from registering in the multiple sections of the same course
- Allow admin app add new professor and new courses/sections

Overall, we didn't have to refactor our old code too much. Our code pretty much stayed the same except for minor changes that will be described below.

Attendance Taking:

To implement GUI for the student/professor user, we needed to solidify our communication protocol between client and server. Following are the protocol:

COMMUNICATION PROTOCOL

Message Types: LOGIN, LOGINSUCCESS, SELECT, SELECTSTUDENT, ENTER, TEXT, DATE, LOGOUT

* LOGIN: The server sends a prompt to the client and asks for a netID and password. The client displays text fields for the netID and password and send back the netID and password to the server.

* LOGINSUCCESS: The server sends a success message (includes netid) to the client to confirm successful login.

* SELECT: The server sends a list of **NUMBERED options*** to the client. The client displays these options as buttons. When the user clicks a button, the client sends the corresponding option NUMBER to the server.

* SELECTSTUDENT: The server sends a list of ***student NETIDs*** to the client. The client displays these netIDs as buttons. When the user clicks a button, the client sends the corresponding student netID to the server.

- Future improvement: Have sender send numbered options so it can be SELECT message type instead

* ENTER : The server sends a prompt to the client and asks for text input. The client displays a text field and a submit button. When the user enters text and clicks the submit button, the client sends the text to the server.

* TEXT: The server sends a message to the client. The client displays the message as text.

* DATE: The server sends a prompt to the client and asks for a date input. The client displays a date picker and a submit button. When the user selects a date and clicks the submit button, the client sends the date to the server.

* LOGOUT: The server sends a message to the client to confirm logout. The client displays a confirmation dialog. If the user confirms exit, GUI stops and resources are closed.

All message types except LOGIN, LOGINSUCCESS, LOGOUT have at least 3 lines.

- * Line 1: Message Type
- * Line 2: Message to display to user
- * Line 3: Either other messages or Go back to previous page message

Admin App:

We refactored our old admin app to have clear model, view, controller modules which helped writing JavaFX GUI.

Block student from registering in the multiple sections of the same course:

We updated our SQL statements in the database adapter module to block such operation

Allow admin app add new professor and new courses/sections:

We added a new option for the Admin user to add a professor, course, and section. This new feature is also reflected in the GUI.