Security Assurance

Our user interface is designed to take user input and generate SQL queries that will show the user specific information based on the information tab that they selected. One of the biggest problems with using SQL backed web applications is a user entering SQL language into the user input. This is SQL injection and is typically used to expose sensitive or private information from the database to unauthorized users. Our interface has some measures instilled to prevent this from occurring. Because of this concern we took away some user inputs and instead replaced them with buttons. For example, in the admin view, when they have the choice to filter instructors, instead of having the user type out salary, department, or name; we have the user select a button which will bring them to a sorted list based on the button they pressed. The same goes with viewing the salaries of all the departments. We made it so that the user is showed all the information at once rather than having them input that they want to see minimum, average or maximum at one time. In the student and instructor views we have the user input severely restricted so that the user has to put in the bare minimum information required for a successful SQL query. Also, if the user attempts SQL injection an error will occur as the combined SQL statement will not make sense and an error will come from the SQL server and send that message back to the interface. Overall, in order to limit the possible effects of SQL injection we decided to limit the chances for a user to enter input that could lead to this problem, and then in the spots where user input is absolutely essential, we limited the possibility of poor outcomes.