Consider the sqlbasic app that we have used in class.

In this assignment you will add some simple extensions to the app that will implement defense-in-depth measures for protecting against SQL injections and implementing proper authentication. Please extend the app as follows:

1. Add a "session" attribute to the appusers table.

2. When the user logs in, store the session ID in the "session" attribute of the user's record. Please do not make any other changes to the database.

3. When the user navigates to the site, the back-end (1) checks if the user is logged in and has an active session; and (2) if so, the back-end looks up the user record based on the session ID and shows a simple welcome page showing the user's name and the contents of the "info" column. Otherwise, the user is directed to a login/create account page.

4. When the user logs out, the session ID is deleted from the user's record (or is replaced with some place holder value such as "not logged in".

5. Add an option to allow users to register (i.e., add their user name and password)

5. Use the node.js's bcrypt package to securely store and verify passwords (in the SQL dabatabse). You can also find a very simple sample file here.

6. Use node.js's password strength checker package to check whether the user's password is strong according to OWASP 10 requirements covered in class.

7. Add a self-signed HTTPs certificate.

8. Configure the client-sessions package to have the session expire after 10 mins inactivity (which the program already uses).

9. Add CSP protection and make session cookies HTTPOnly to ensure some protection against XSS.

10. Make sure that the webapp has a privilege-restricted database account.

What to submit:

1. Working code [ZIP]

2. README file including (1) List of all group members; and (2) Instructions for creating a database and running the program.

3. Screenshots (videos are even better!) illustrating:

- **That multiple users can create an account.**
- **That multiple users can login and correct information is shown upon login/visiting the website with a valid session.**
- **That user's can logout and after logout the user is directed to login if they visit the site.**
- **An explanation and screenshots explaining how the code meets each requirement.**