COMP 307 – Principles of Web Development

Assignment #2

Due: November 4, 2015 at 23:30 on myCourses

Question 1: Project Preparation (1 point)

Given the project description, select your team and the project you would like to build.

For this question submit the following things in a single PDF document:

- Team member names
- Select a team leader
- Describe the website you would like to create
- Identify a minimum of 4 technologies (preferably more)
- You will need my approval
- All team members submit this question/assignment individually to myCourses for the TA to grade
- The team leader must email Question 1 to me for review and approval

Question XAMPP & The Slim framework and Bootstrap (19 points)

You must do this question individually. This question is not part of your group project.

Do this assignment in XAMPP (or compatible).

Your solution must include: HTML5, CSS, JavaScript bindings to DOM, JSON communication to a PHP script that will access a mySQL database and return a Reply to the client. Use the Slim framework and something (like a CSS) from Bootstrap.

Problem to solve – Create a login/log-out web page that uses encrypted JSON communication with the server, and a server Session ID to track the user's login status.

This is what you need to build:

- A mySQL database called Q2DB with two tables: members and sessions. The members table has the following fields: member ID, shared key, user name, and password. The session table has the following fields: member ID and session ID. All members belonging to this web page are recorded in the members table. When a member logs in successfully a record is created in the session table. When a member logs out successfully the preexisting record in the session table is deleted. In this way the session table tracks which members are currently logged in. Populate the members table with sample records. Leave the session table empty, indicating that no one is logged in.
- Create a login web page. This page uses Caesar cipher encryption and JSON to pass login credentials to the server. Dream up a way to show off your abilities to make a boring login page exciting. When the user presses the login button a JavaScript is invoked that converts the user name and password into a JSON message. Then using the shared key belonging to

this client, encrypt the password stored in the JSON string. Then send this JSON to the server asking for the login.php script through the Slim framework.

- The server has a login program written in PHP called login.php. The received packet's JSON payload is extracted by this PHP script. It uses the unencrypted user name and the members table to locate the user's password and shared key. Using the key the JSON password field is decrypted and compared with the password obtained from the table. If they match then the user has successfully logged in. An entry is appended to the session table storing the user's member ID and a unique session number (use aa variable that increments to create the unique session ID). If the login was not successful then no record is appended to the session table. The login script sends a reply web page to the client. If login was successful a web page is returned saying the user successfully logged in. It will also have a log-out button that calls a server logout.php script when pressed, through Slim. This reply web page contains a hidden field with the session ID assigned to the client by the database. When the user presses log-out this hidden session ID field is sent to the server. If the login was not successful then a web page is returned stating this fact. The web page does not have a hidden session ID. The web page has a hyper link to the login page.
- The logout.php script uses the session ID provided by the hidden field together with the session table to confirm if the user is actually logged in. If it finds the session ID in the session table then the user had previously logged in and we proceed to log the user out. To do this the member ID number is extracted from the session table record. The session record is deleted and a reply message is sent to the user stating that they are now logged out. The web page displays the user's user name, which is obtained from the members table using the member ID from the deleted session field. A hyper link is provided that leads to the login page. If the provided session ID was not found in the session table then it was not a valid log-out request. In this case the session table is left unmodified. A reply web page is returned to the user stating that they are not a valid user. A hyper link is provided leading to the login page.

That's it. Enjoy.

Submission Guidelines

- Submit your assignment to the submission box in myCourses identified as Assignment #2.
- Your written answers must be in: PDF, ODF, JPG, or RTF formats.
- Your code should be zipped.

Grading Guidelines

- This assignment is worth a total of 20 points.
- Each question has identified a point-weight beside the question title. This is the maximum score for that question.
- Each question is graded proportionally. In other words, if 75% of the question is correct then you receive 75% of the point-weight for that question.
- Your grades will be posted on myCourses under the Grades link from the icon bar.
- If you have any issues with the grade do not hesitate to communicate with me or the TA