THE UNIVERSITY OF WINNIPEG

Applied Computer Science
ACS-3909-002
Advanced Internet Programming

Winter 2024

Assignment 3

Due date: April 4, 2024, 11:59 pm (CST) **Total Marks: 34**

Objectives

This assignment represents a remarkable chance for you to employ your knowledge of cookies, sessions, authorization, and data persistence in a practical setting. It includes two specific tasks that are outlined below:

1) **(10 marks) Develop a web-based program that carries out the tasks listed:**
   a) The homepage is designed to generate and dispatch a signed cookie.
   b) Navigating to a second page, the signed cookie is then transmitted back to the server.
   c) The signed cookie received is displayed on this second page.

   You need to consider the following requirements:
   - Use the following cookieSecret:
     cookieSecret: 'This is the last assignment for 3909-002 course!@@#'
   - The signed cookie set is 'MySignedCookie': '<your_student_ID>SignedCookie'
   - Set the maxAge to 1204800500.
   - To navigate to a second page, implement 'get' method for '/cookieShow' endpoint to send the cookie back to the server to display on that site.

   8 marks are for the web application, and 2 marks are for an output verifying the cookie functionality.

2) **Part 1 (20 marks)** As discussed in "Week 8- Cookies and Sessions", using a memory store for session data is unsuitable in a production environment. Based on "L08E03_SesssionExample", it is easy to set up MongoDB to store session data, namely ***usernames*** and ***passwords***.

   The attached zip file includes the working directory for this task and demo videos. There are four items to complete:
   1) Check the ***package.json*** dependencies and install all relevant modules
   2) Add the ***connection string*** to your database (replace <password> with your DB password)
   3) Complete the implementation of ***checklogin*** function
   4) Implement the ***post*** endpoint for '***/processReg***'
   5) A screenshot of your database storing a sample of usernames and passwords

   1) and 2) are worth 2 marks each. 3) and 4) are worth 7 marks each for a working web application described above. Finally, 2 marks for the screenshot.

   Part 2 (4 marks) MD5 (Message Digest 5) is a widely used cryptographic hash algorithm that secures and verifies the integrity of documents and data. Use MD5 to secure the passwords of the users in the previous task. See the related video in the attached zip file.
   Note: Don't submit a separate file for this part. Just include your solution in part 1.

Submission Instructions

Zip all files and folders into a single archive named *StudentNumber_Assignment3.zip*. Submit the zip file through Nexus. Do **not** include your node_modules folders.