COMP 2537 – Web Development 2

Assignment 4

BCIT CST — Due: May 23rd, 11:59 PM

Late submissions result in zero

This is a project group submission (groups based upon your COMP 2800 course)

Introduction

You will create an app in Node.js that performs logging using the Morgan package as well as add a chat component to your app.

Details

Create an app (high level of fidelity) that uses the Morgan package for logging. You will also add a chat component that allows users on your system to chat once they are logged in.

Requirements for the logging are:

- Use the Morgan package with Express in Node.js
- Save all log information to a file called access.log that is in a folder called logs
- Logging should be set to rotate daily (so use the rotating-file-stream package)
- Information stored in each access should be:
  - Referrer (this is any other sites that linked to your site/app)
  - URL (the URL that was accessed – could be a wrong URL too)
  - The user-agent (e.g., Windows 10 with Chrome, Safari on iPhone 12, etc.)

Requirements for your chat component:

- Allow for users of your system to chat
- User names are displayed (like the example code)
- When users come and go, this is announced to all users in chat (again, like the example code)
- Users must be logged in in order to use the chat component (and therefore there aren't any 'anonymous' users/names – *unlike* the example)
- The chat component should be something that isn't automatically seen by the user and that they have to "connect" or "open" first
- Provide the ability for users to connect and disconnect from the chat gracefully (e.g., an iframe, another page, or something elegant)

Grading Scheme

Your assignment will be marked out of 20:

- 2 marks for users can chat
- 2 marks for users connect first before chatting
- 2 marks for users disconnect/leave chat
- 2 marks for messages displayed to all other users when users enter or leave chat
- 4 marks for logging working correctly
- 2 marks that users have to be logged in before being able to chat
- 2 marks for users having different colors than others in chat

Bonus marks (additional – not mandatory):

- 4 marks for saving all chat to the DB and allowing the user to see previous chat
- 2 marks for replacing :) with ☺

You will lose marks for:

- **Not using strict**
- Invalid code of any type
- Runtime errors
- Not following the requirements of the assignment
- Using variable names or file names that are very clearly example code or example files (e.g., lab7-starter.js)
- Content that is boilerplate (e.g., dummyimage images, lorem ipsum)

If you have questions pertaining to any design choices, please see your lab instructor or lecture instructor.

Submission Requirements

Zip up your submission and upload to the COMP 2537 learning hub “Assignment 4” folder, one submission from each group.

The uploaded app must contain all of your names, full names and student IDs that show up in the main page. Failure to do so will result in not getting marked for work done. If I have to do a marks change after the course is over because of this, the entire group will get docked marks.

Submit naming format:

> Your zip file containing your source code files in a zip named COMP2537-assignment4-<lastname>-<firstname>.zip (only one zip file for all source code) and only one submission per project group