National University of Singapore
School of Computing

IS3106 Enterprise Systems Interface Design and Development
Assignment 1

Learning Objectives

At the end of this assignment, you should be able to:

- Perform backend development using ExpressJS.
- Working with the templating engine in ExpressJS.
- Integrate ExpressJS with MongoDB.

System Description

In this assignment, you will be building a **Q&A Forum System** (like Stack Overflow[1]). The system allows users to ask technical or general questions, answer others' questions, and vote on helpful responses. This assignment is only worth **15%** of the course grade and is mainly used to assess your ability to perform both backend development using ExpressJS and the templating engine in ExpressJS with proper integration with MongoDB and an HTML template that was covered in Labs 1-3.

To keep things simple, the system will support 2 main types of content:

1. Questions (posted by users)
2. Answers (responses to questions)

There are no admin roles, and all users have the same privileges. Users can ask and answer questions freely.

General Requirements

- This is an **individual assignment**. You are supposed to work on it on your own. Please refrain from discussing codes with your friends. **Your submission will be assessed carefully to ensure that there is no plagiarism (both with your classmate's submissions and sample codes online).**
- You **must** develop the backend using **ExpressJS** and must adopt a **templating engine in ExpressJS**. The second assignment would be based on the same system except that you are supposed to use React for the development process. As such, for this assignment, you should adopt a more traditional approach to the development of rendering the page on the server side as much as possible and avoid using RESTful APIs and AJAX for most/all of the implementation.
- The use of other Javascript frameworks such as React, Vue, Angular, NextJS, etc. is **NOT allowed** for this assignment.
- You are free to use any images and templates[2] that you find online. Your assignment submission will not be published online so you do not need to be concerned with copyrights. However, if you choose to deploy the project for your use later, you must ensure that the media files do not infringe copyright. We will not be liable for any copyright infringement.

Function Requirements

Your system should support the following functionalities, but you are allowed to include additional features.

1. User Management

- **User Registration**
- **User Login & Logout**
- **View My Profile and Project Management**
    - Each user should have at least a username, email, profile picture, and bio.

see question/answer history

[1] https://stackoverflow.com/questions
[2] The use of HTML/CSS templates are fine, but you should refrain from copying large chunks of codes from online or sample projects that you might obtain from online.

2. Question Management

- **Post a new question**
  - Each question should have a title, body, tags, and timestamp
- **View a list of all questions**
  - The portal should allow the user to sort the questions using different filters (e.g. most recent, hot, tags, etc)
- **View a single question with its answers**
- **Edit a question** (only the author of the question can edit his/her own question)
- **Delete a question** (only the author of the question can delete his/her own question)

3. Answer Management

- **Post an answer to a question**
- **View all answers to a question**
- **Edit an answer** (only the author of the answer can edit his/her own answer)
- **Delete an answer** (only the author of the answer can delete his/her own answer)

4. Voting Management

- **Upvote a question**
- **Downvote a question**
- **Upvote an answer**
- **Downvote an answer**
- Users should only be able to vote **once per question/answer**, and votes should be toggleable (e.g. upvoting after downvoting removes the downvote).

The list of features shown above is not exhaustive. You can also consider adding additional features or enhancements to the above. Please note that the grading is also based on the quality of the feature implemented – i.e. implementing all listed features above does not by default lead to full credit for the feature (10%) grading component.

System Submission

1. Deliverable: completed ExpressJS project (without the "node_modules" folder)
2. Together with your ExpressJS project, you should also create a README.txt file with the following information:
   a. Your name
   b. Your student number
   c. Database details and instructions for deploying your project
   d. Indicate and describe any extra features you have incorporated into your system.
3. You should also prepare a PDF document with screenshots of every page / most pages for the grading of the UI/aesthetics.
4. All the files (ExpressJS project and the README.txt) should be zipped up into a single file with your name (e.g. JohnDoe.zip) and uploaded to the Canvas (**Assignment 1**)
5. Deadline: **Week 9 Friday, 23 Mar 2025 11:59 pm**
6. You will be penalized for late submission.
7. The system submission is worth **15%** of the entire course grade.

Evaluation Criteria

Weightage: 15% of the course grade.

- 10% Features
  - Grading is mainly based on correctness rather than code quality
  - Please note that you do not by default get 10% if you have implemented all the above listed features. We also grade based on the quality of the implementation (usability, implementation of validation, UX, etc)
- 5% - UI and aesthetic