DataByte

The Official Machine Learning and Data Science Club of NITT

FIRST-YEAR INDUCTION

TASK-2

WEB DEVELOPMENT

Problem Statement

The task is to build a group based social network where users can create or join groups based on their interests or activities to interact with each other, share content, and participate in group activities.

Basic Mode:

- Implement authentication that allows users to register and login on to the site.
- Allow users to create groups with a suitable form getting basic details of the group and join groups based on their interests.
- The dashboard should contain a list of group cards along with basic details of the group like purpose, number of members, date of creation, join-in button.
- If a user is already a member of that group, then navigate him to that group page showing chat history. If not, add the authorised user to that group as a new member.
- Add a search bar that allows the user to search for groups by name or keywords.

Hacker Mode:

- Validate the registration forms (check for duplicate usernames, determine password strength, etc).
- Implement a chat feature that allows users to communicate with each other in real-time.
- Only the admin (the creator of the group himself and the one he assigns the role) of the group can enable/disable messaging of the group.
- Allow users to filter by group tags in the search feature.
- Support for image and file upload.

Brownie Points:

- Generate group codes or link and share to other registered users within the website to join the group.
- Send notifications through email or in-app notifications.

GUIDELINES:

- It is mandatory to complete the basic mode of the task before it can be considered complete. While the hacker mode is optional, it is recommended that the mentee implements it if interested.
- Strictly build the website following the list of allowed frameworks.
- Plagiarism is strictly forbidden, and any instance of it will result in disqualification from any Databyte activity.

LIMITATIONS:

Build the code in JS / TS only.

List of allowed frameworks

- Frontend: React, Angular, Vue.js
- Backend: Express (NodeJS), Flask (Python), Echo(GO), Django (Python)

Not allowed:

Raw PHP, Laravel (PHP), Spring (Java), MVC framework, CSS libraries / frameworks (such as Bootstrap, Material-UI, Semantic UI)

EVALUATION METRICS:

The mentee's work will be evaluated based on the implementation of the basic features suggested and UI/UX, as well as their creativity and originality. If any extra features have been implemented, the mentee will receive additional brownie points. Plagiarism will not be tolerated and will result in disqualification from the task.

SUBMISSIONS:

Host the website in any platform and submit the repository link containing the complete code with a well structured README file explaining in detail your work and features implemented.

RESOURCES:

- Python based Frameworks:
- Django: Django Web Framework (Python) - Learn web development | MDN

  Getting started with Django

- Flask: Tutorial — Flask Documentation (2.3.x)

  https://opensource.com/article/18/4/flask

- JavaScript based Frameworks:
- NodeJS: https://nodejs.org/en/docs/

  Express 4.x - API Reference

  Node.js - MDN Web Docs Glossary: Definitions of Web-related terms

- ReactJS: React

  Getting started with React - Learn web development | MDN

- Angular: https://angular.io/docs

  Getting started with Angular - Learn web development | MDN

- Vue:Getting started with Vue - Learn web development | MDN

  Introduction | Vue.js

- Echo: https://echo.labstack.com/guide/
- MySQL: (Relational Database) MySQL Tutorial
- MongoDB:(Non-Relational Database) https://docs.mongodb.com/manual/tutorial/getting-started/ https://mongoosejs.com/docs/api.html