Assignment: Building an Interactive Website to Validate ChatGPT Efficiency

Objective

This assignment combines web development with an in-depth evaluation of ChatGPT's efficiency across various domains. You will create a website that documents your research, displays data visualizations, and provides insights on ChatGPT's performance. This assignment will develop your skills in web design, API interaction, data handling, and presenting technical findings.

Total Marks: 20 (15 points for the project, 5 points for the presentation)

Part 1: Setting Up and Hosting Your Website (5 points)

Objective: Develop a personal website using a provided template, where you'll showcase your ChatGPT efficiency evaluation results. Host this website on GitHub Pages for public access.

Steps

1. **Download and Customize the Website Template**
    - **Access the Template**

      Download the website project template from this link.
    - **Customize Content**

      Open index.html in a text editor of your choice (e.g., Visual Studio Code, Atom, or Sublime Text).
        - **Introduction**: Add an introductory section explaining the purpose of your project, combining both web development and ChatGPT efficiency evaluation.
        - **Profile Section**: Include your name, a brief bio, and an overview of your technical background relevant to this project.
        - **ChatGPT Efficiency Section**: Create sections dedicated to the ChatGPT evaluation project:
            - **Research Approach**: Briefly describe the setup, tools, and technologies used (MongoDB, Node.js, ChatGPT API).
            - **Dataset Overview**: Include a summary of the dataset used (questions across History, Social Science, and Computer Security) and the purpose of these evaluations.
            - **Results**: Provide a placeholder for the visualizations from Part 2, which will later display ChatGPT's performance in each domain.
        - **Styling and Design Adjustments**: Feel free to personalize the template, such as modifying colours, fonts, or layouts to enhance readability and aesthetic appeal.
2. **Preview and Test Locally**
    - Save your changes and open the index.html file in a web browser to view your website locally. Make any necessary adjustments to ensure a professional appearance.

3. **Hosting on GitHub Pages**
    - **Create a GitHub Account**: If you don't have a GitHub account, sign up at GitHub.com.
    - **Host Your Website**:
        - Create a new GitHub repository and upload all the files from the template, including your customized index.html.
        - Follow a tutorial (such as this YouTube video) to set up GitHub Pages and publish your site.
    - **Submission**: Once your website is live, submit the GitHub Pages URL through the assignment portal. Ensure your website is publicly accessible and fully functional before the deadline.

Points Breakdown for Part 1:

- Template Customization (3 points)
- Content Quality and Relevance (1 points)
- Successful GitHub Hosting (1 points)

Part 2: ChatGPT Efficiency Evaluation (10 points)

Objective: Assess the accuracy and response times of ChatGPT across multiple domains using a client-server setup. You'll interact with a MongoDB database and utilize Node.js and the ChatGPT API to record and analyse responses.

Steps

1. **Setting Up MongoDB**
    - **Install MongoDB**: Set up MongoDB on your machine or use a cloud-based MongoDB service.
    - **Create Database and Collections**:
        - Name the database ChatGPT_Evaluation.
        - Establish three collections within the database: History, Social_Science, and Computer_Security.
    - **Dataset Population**: Populate each collection with at least 50 questions from the dataset in the Google Drive repository.
        - Each document (entry) in the collection should include:
            - The **question** text.
            - An **anticipated answer** based on your own research.
            - A **field for ChatGPT's response**, which will be populated later.
2. **Client-Server Interaction with ChatGPT**
    - **Server Setup with Node.js and Express.js**:
        - Develop a server application in Node.js using Express.js to handle communication with MongoDB and the ChatGPT API.
        - Implement routes to:
            - Retrieve questions from MongoDB.

        - Send these questions to ChatGPT for responses.
        - Store ChatGPT's responses back in MongoDB within the respective document.
    - **API Integration**:
        - Use the ChatGPT API to obtain answers for each question in MongoDB. Ensure your server handles requests efficiently to avoid rate-limiting or other issues.
    - **Code Testing**: Run the server locally and test that data is being fetched from MongoDB, processed through ChatGPT, and saved correctly.
3. **Evaluation and Visualization**
    - **Accuracy and Performance Evaluation**:
        - Compare ChatGPT's responses to your anticipated answers and calculate the accuracy for each domain.
        - Measure and record the response time for each query.
    - **Data Visualization**:
        - Use a graphing library (e.g., Chart.js) to create visualizations that display:
            - **Accuracy Rates** across History, Social Science, and Computer Security.
            - **Average Response Time** for each domain.
            - Any other metric you find insightful.
        - Embed these visualizations on your website (from Part 1) under the "Results" section.

Points Breakdown for Part 2:

- Database and Data Population (4 points)
- API and Server Integration (4 points)
- Data Visualization and Analysis (2 points)

Part 3: Presentation (5 points)

Objective: Deliver a 10-minute presentation summarizing your project, covering both the website creation and ChatGPT efficiency evaluation.

Presentation Content

1. **Introduction**
    - Briefly introduce yourself and outline the objectives of your project.
2. **Website Overview**
    - Describe the website and highlight key sections, explaining how it serves as a platform for the ChatGPT evaluation results.
3. **Technical Approach**
    - Describe your approach to setting up MongoDB, integrating the ChatGPT API, and handling client-server communication with Node.js.

    - Provide a high-level overview of your code, with emphasis on the server functions and API interactions.
4. **Data Analysis and Key Findings**
    - Present your visualizations and discuss the results:
        - How did ChatGPT perform across the different domains?
        - Were there any trends in accuracy or response times?
        - Any limitations or challenges you encountered.
5. **Concluding Remarks**
    - Conclude with a reflection on what you learned from combining web development and AI evaluation. Mention any improvements you'd consider for future projects.

Points Breakdown for Part 3:

- Content Coverage (2 points)
- Clarity and Organization (2 points)
- Visual and Oral Presentation (1 point)

Submission Guidelines

- Submit the GitHub Pages link to your website through the assignment portal.
- Ensure all website content, including visualizations and analyses, is live and accessible.
- Prepare for your in-class presentation scheduled **for November 27, 2024**.

Deadline: **November 27, 2024**