Hackathon Task: Data-Driven Web Application

Overview

In this task, I have created a web application that allows users to upload CSV files, displays the uploaded data with pagination, and includes a subscription pricing calculator based on the uploaded data.

Instructions

1. Clone the Project from GitHub

- Open a terminal and navigate to the directory where you want to clone the project.
- Run the following command:
  git clone https://github.com/Dishu099/data-driven-web-app.git
  cd data-driven-web-app

2. Open the Project in Visual Studio Code

- Launch Visual Studio Code.
- Open the project directory in Visual Studio Code.

3. Run the Project

- Ensure you have a live server extension installed in Visual Studio Code.
- Open the `index.html` file.
- Right-click on the `index.html` file and select `Open with Live Server`.
- This will start a local server and open the application in your default web browser.

4. CSV Upload Service

- The application will have an interface for uploading CSV files.
- Users can select and upload a CSV file.
- The service should efficiently handle large datasets.
- Implement real-time progress indicators to keep users informed during uploads.

5. Data Display and Pagination

- After the CSV file is uploaded, the data will be displayed in a table.
- Implement pagination to allow users to navigate through large datasets smoothly.
- Ensure the UI remains responsive even when dealing with extensive data.

6. Subscription Pricing Calculator

- Utilize the uploaded CSV data to create a Subscription Pricing Calculator.
- This calculator will use the data to calculate and display subscription pricing based on the uploaded data.

Project Structure

data-driven-web-app/

- **index.html:** The main HTML file containing the structure of the web application.
- **style.css:** The CSS file for styling the web application.
- **script.js:** The JavaScript file containing the logic for CSV file upload, data display, pagination, and the subscription pricing calculator.

Key Features

1. CSV Upload Service

- **Functionality:** Allow users to upload CSV files.
- **Efficiency:** Handle large datasets efficiently.
- **Real-time Progress:** Show progress indicators during file upload.

2. Data Display and Pagination

- **Display Data**: Show uploaded data in a table.
- **Pagination**: Implement pagination for large datasets.
- **Responsive UI:** Ensure the UI remains responsive.

3. Subscription Pricing Calculator

- **Functionality:** Calculate and display subscription pricing based on uploaded data.