FSD Mini Project Report

TY Btech CSF

Batch A1

Student Hub

1. Project title: StudentHub: Smart Student Assistant App

2. Team Members

PA03 - Neeraj Varma

PA13 - Atharva Yadav

PA15 - Yashvardhan Tekavade

PA19 - Dhruv More

PA20 - Aayush Sandhane

3. Overview of various full stack technologies used in the mini project:

Our student assistant app, StudentHub, leverages a comprehensive set of technologies to deliver a sophisticated and user-friendly experience. The technologies include:

<u>Frontend:</u>

1. HTML/CSS/JS: HTML, CSS, and JavaScript are programming languages that allow developers to communicate with computers and instruct them on how to perform specific tasks.

HTML:

Hypertext Markup Language is the standard markup language for creating web pages. It describes the structure of a web page and tells the browser how to display the content.

CSS:

Cascading Style Sheets is used to style and layout web pages. It can alter the font, color, size, and spacing of content, split it into multiple columns, or add animations and other decorative features.

JavaScript:

A scripting or programming language that allows you to implement complex features on web pages. It can add dynamic interactions to web pages, applications, servers, and even games.

2. ReactJS: React is a library for building composable user interfaces. It encourages the creation of reusable UI components, which present data that changes over time. Lots of people use React as the V in MVC. React abstracts away the DOM from you, offering a simpler programming model and better performance. React can also render on the server using Node, and it can power native apps using React Native. React implements one-way reactive data flow, which reduces the boilerplate and is easier to reason about than traditional data binding.

React Features:

- JSX – JSX is JavaScript syntax extension. It isn't necessary to use JSX in React development, but it is recommended.
- Components – React is all about components. You need to think of everything as a component. This will help you maintain the code when working on larger scale projects.
- Unidirectional data flow and Flux – React implements one-way data flow which makes it easy to reason about your app. Flux is a pattern that helps keeping your data unidirectional.
- License – React is licensed under the Facebook Inc. Documentation is licensed under CC BY 4.0.

Backend:

1. Node.js with Express.js: For the server-side, we have chosen Node.js with Express.js. This powerful combination facilitates the handling of server operations and enables seamless communication between the frontend and backend, ensuring a smooth user experience.

2. Django: Django serves as the backbone of our backend infrastructure. It excels in managing data logic, handling server-side operations, and ensuring the overall stability of the application.

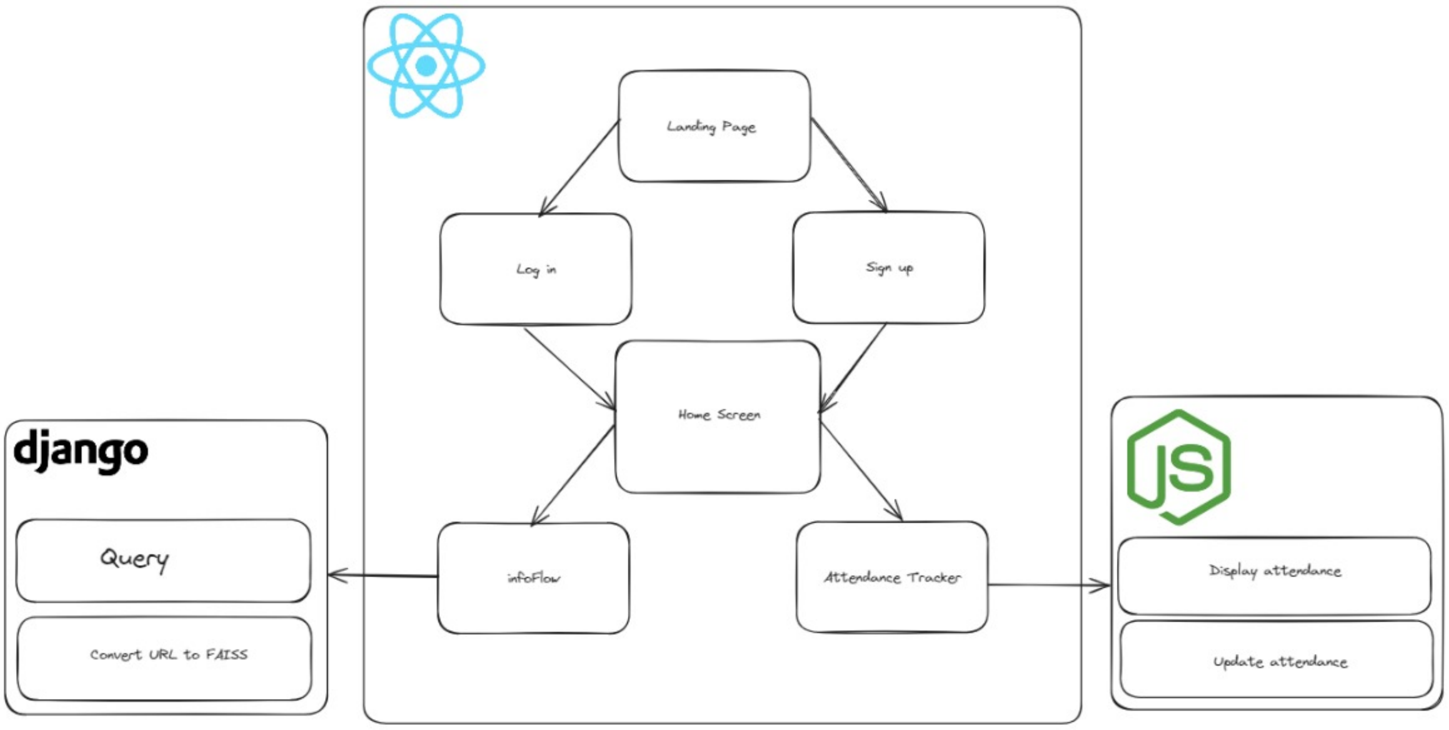
3. MongoDB: As a NoSQL database, MongoDB is employed to store user data and attendance records. Its flexibility allows us to adapt to evolving data requirements and ensures efficient data management.

4. SQLite: For lightweight local storage needs, SQLite is integrated into the backend, providing a reliable solution for specific data storage requirements.

5. Chatbot Integration:

- Python: Python powers our chatbot functionality.
- Langchain: LangChain is an open-source framework that makes it easier to create applications using large language models (LLMs). It provides a standard interface for chains, many integrations with other tools, and end-to-end chains for common applications.
- Hugging face: Hugging Face is a machine learning (ML) and data science platform that helps users build, deploy, and train ML models. It also provides the infrastructure to demo, run, and deploy artificial intelligence (AI) in live applications.
- Google palm API: Google's PaLM API (Pathways Language Model API) provides access to Google's large language models. These models can be used for a variety of applications, including: Chat, Content generation, Code generation, Reasoning, Writing.

4. Workflow/ architecture diagram with explanation



5. Future scope and conclusion

Enhanced Access Control:

- Implement role-based access control to differentiate between teacher and student logins.
- Teachers will have access to an administration panel for tasks such as editing attendance and performing other administrative duties.

- Students will have a personalized dashboard showcasing attendance records, upcoming lectures, and additional relevant information.

Expanded Student Features:

Personalized Dashboards: Students can access a comprehensive dashboard displaying their attendance history, upcoming lectures, and important announcements.

Assignment Assistance: Integrate advanced features into the chatbot to assist students in writing assignments. The chatbot can provide guidance, suggest resources, and offer assistance in structuring content.

Administrative Tools:

Attendance Management: Enable teachers to efficiently manage attendance records, making necessary edits and updates through the administration side of the app.

Data Analytics: Implement analytics tools to provide insights into attendance trends, helping teachers make informed decisions.

Chatbot Advancements:

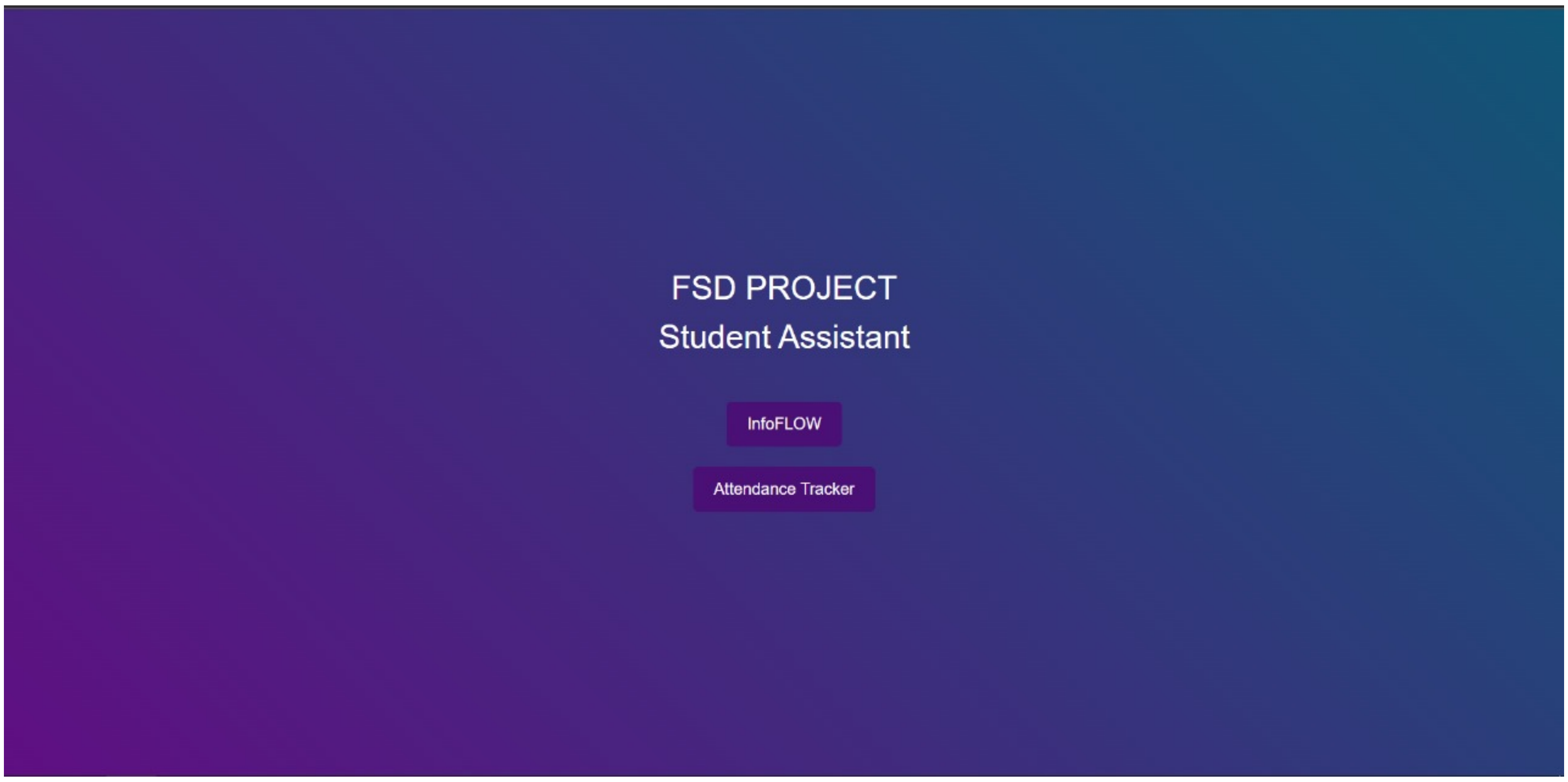
Multifunctional Chatbot: Expand the chatbot's capabilities to offer simplified answers based on YouTube videos for a broader range of subjects.

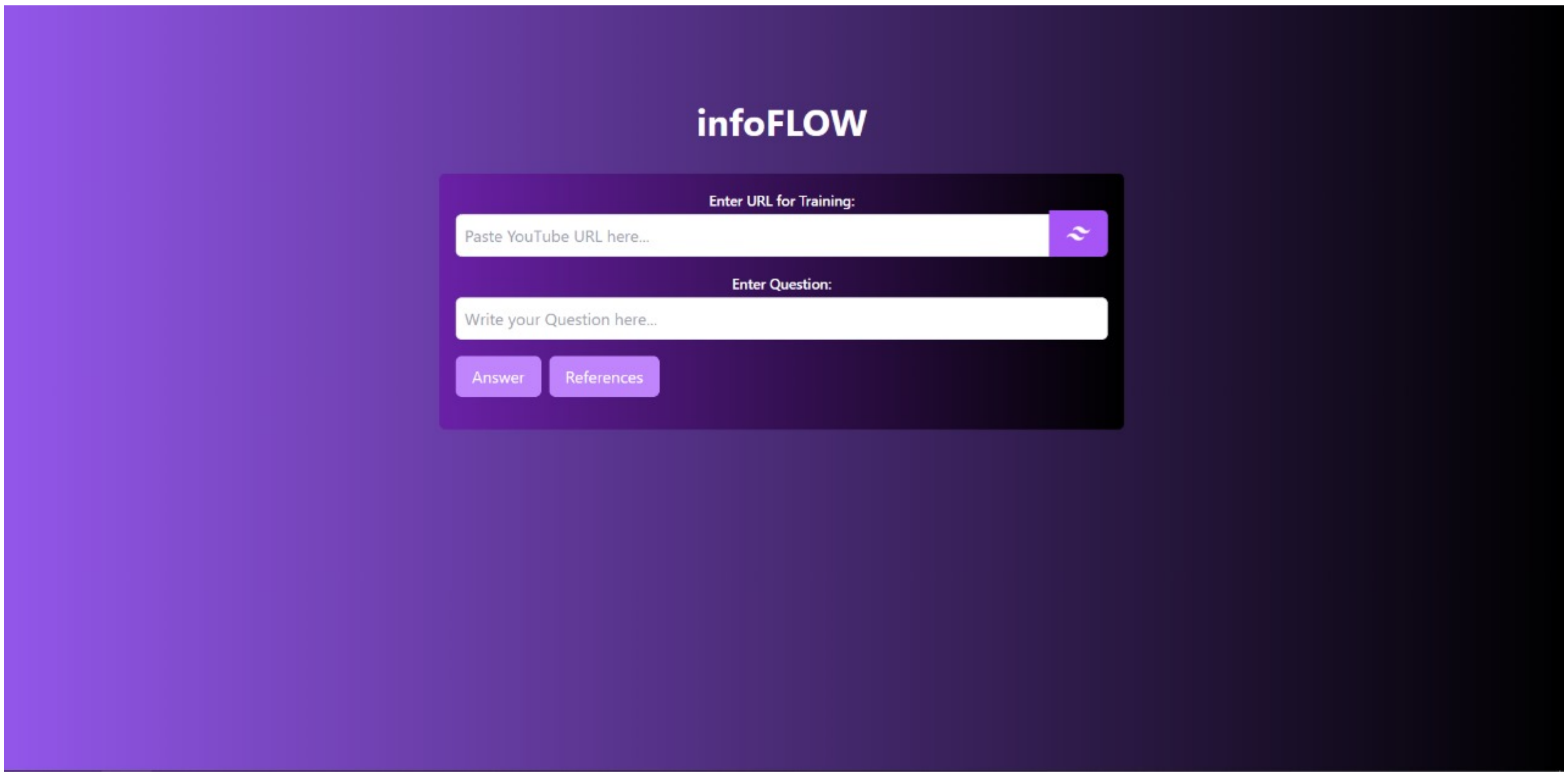
Assignment Support: Enhance the chatbot's ability to provide assistance in various academic areas, offering guidance on assignments and answering subject-related queries.

Conclusion:

As StudentHub evolves, our vision is to transform it into a fully integrated student app. The implementation of role-based access control will tailor the experience for both teachers and students. Students will benefit from personalized dashboards, assignment assistance through an advanced chatbot, and a consolidated view of their academic journey. Teachers, on the other hand, will have administrative tools at their disposal for efficient attendance management and data analysis. This holistic approach aims to create a comprehensive solution that addresses the diverse needs of students and educators alike.

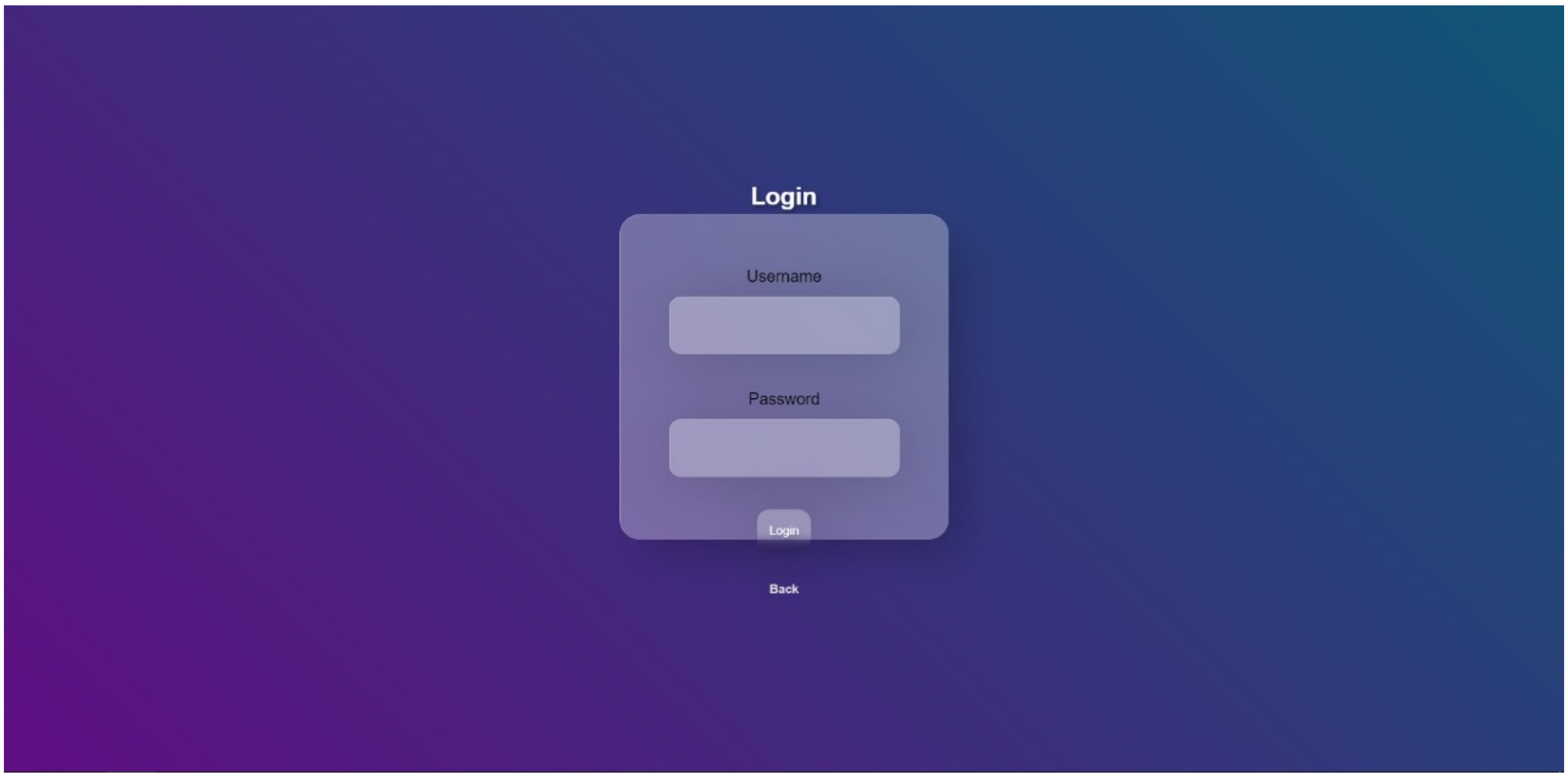
6. Project UI


FSD PROJECT
Student Assistant


infoFLOW

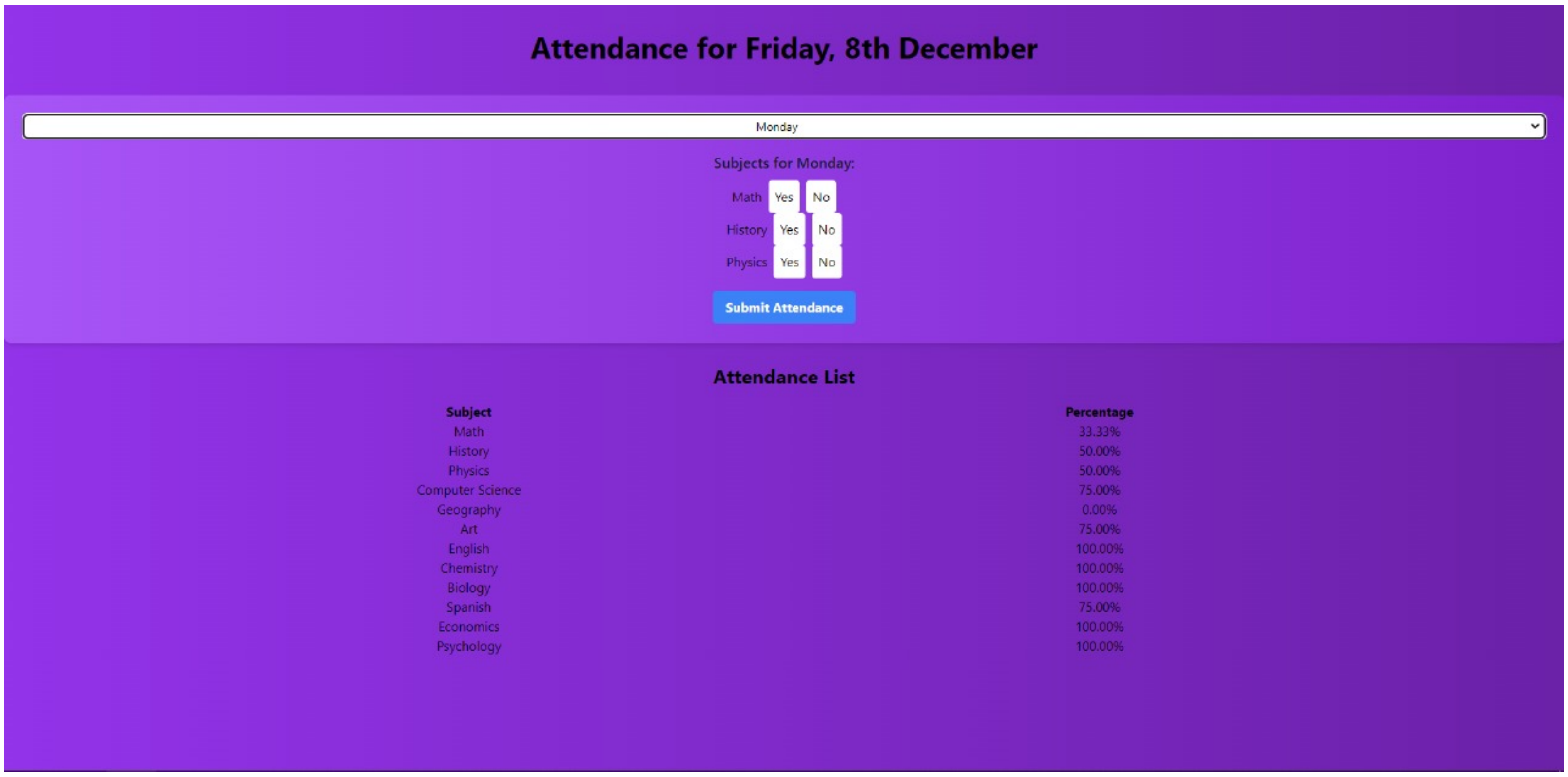
Enter URL for Training:
Paste YouTube URL here...

Enter Question:
Write your Question here...


Login

Username

Password

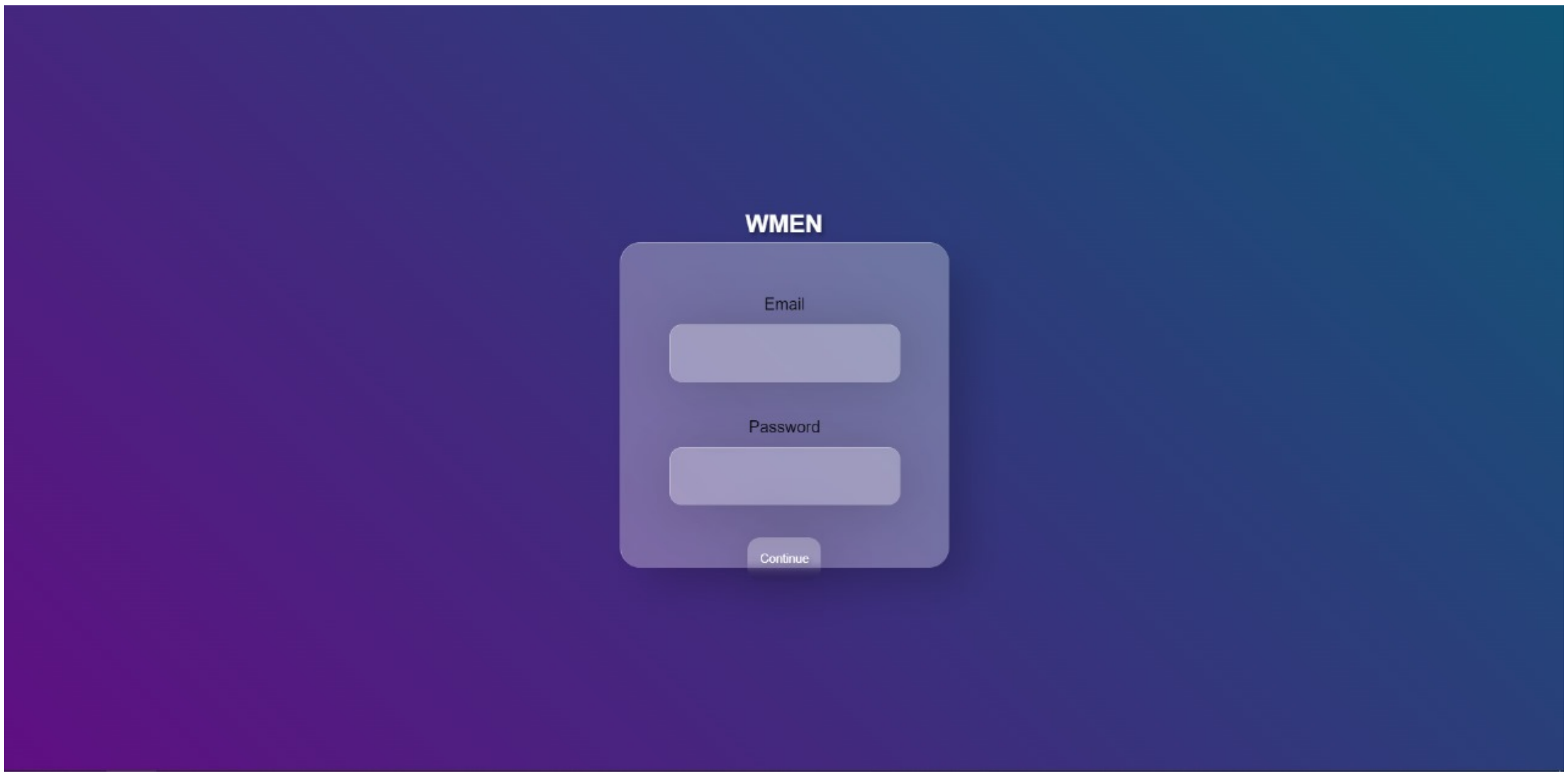

Attendance for Friday, 8th December

Monday

Subjects for Monday:

Attendance List

Subject	Percentage
Math	33.33%
History	50.00%
Physics	50.00%
Computer Science	75.00%
Geography	0.00%
Art	75.00%
English	100.00%
Chemistry	100.00%
Biology	100.00%
Spanish	75.00%
Economics	100.00%
Psychology	100.00%



7. References

HTML, CSS, and JavaScript:

- HTML (https://developer.mozilla.org/en-US/docs/Web/HTML)
- CSS (https://developer.mozilla.org/en-US/docs/Web/CSS)
- JavaScript (https://developer.mozilla.org/en-US/docs/Web/JavaScript)

React:

- React Documentation (https://reactjs.org/docs/getting-started.html)

- React - A JavaScript library for building user interfaces (https://reactjs.org/)

Express:

- https://expressjs.com/en/5x/api.html

MongoDB:

- MongoDB University (https://university.mongodb.com/)

MERN Stack Projects:

- Build and Deploy a MERN Stack App - Full Tutorial (https://www.youtube.com/watch?v=7CqJlxBYj-M)