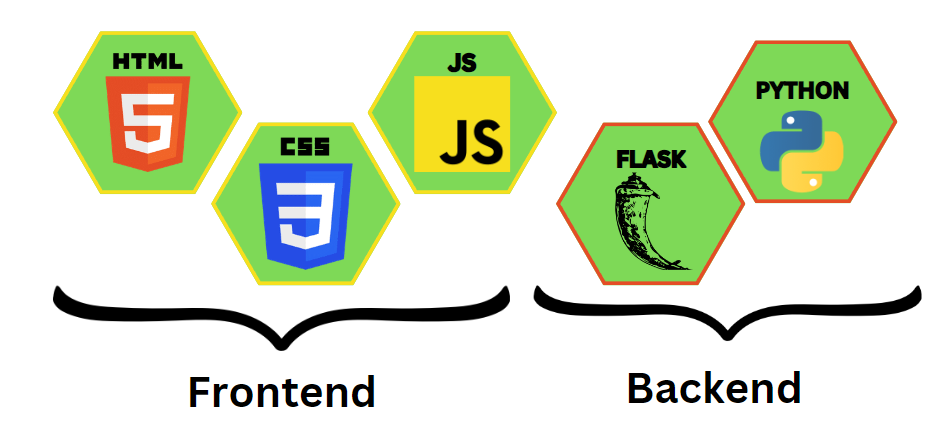
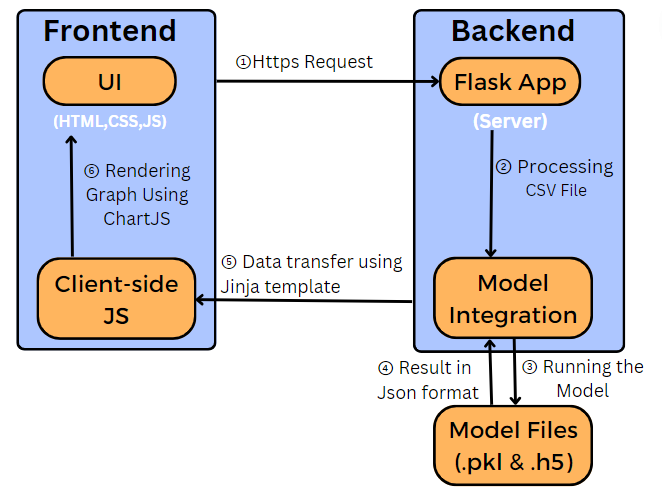
**Tech Stack and Development :**

1. **Tech Stack Used**

****

1. **Working Flow :**

The basic working model of the project is shown in figure below :

****

1. **Brief Explanation:**

**I. For Frontend:**

**HTML:** HTML is responsible for the overall layout and organization of the user interface, providing a semantic structure that is enabling the browsers to render and display the content properly.

**CSS:** CSS is working alongside HTML to control the visual appearance, presentation and styling of the web pages.

**Javascript:** JS helps to manipulate the HTML and CSS elements, respond to user actions, validate input, perform calculations, make https requests, and much more. With JavaScript, we have enhanced the user interface by adding interactive features like dynamic content updates, animations, and user-driven events.

**II. For Backend:**

**Flask :** Flask is a micro web framework written in Python that allows us to build web applications quickly and efficiently. It has provided various tools and libraries for handling HTTP requests, routing, rendering templates, and managing sessions. The main reason for choosing flask over other backend-frameworks like Django is Flask follows a minimalistic approach, providing just the essentials for web development, making it lightweight, flexible, simpler and faster for building the backend of our website.

**Python :** Flask and Python are used together to create the backend of our website, which primarily focuses on handling server-side operations and processing data. The backend serves as the backbone of the website, facilitating csv file storage, retrieval, manipulation and communication with the frontend. Flask and Python combined, results into a robust and efficient platform for building reliable and scalable backend system for our web application.

**III. Integration:**

**Python:** Python is used to integrate machine learning models into the Flask backend, allowing the web application to leverage the power of predictive analytics and data-driven decision-making. In our project, Flask is serving as the backend framework responsible for handling incoming requests from the frontend, passing the required data to the ML model, and returning the predictions back to the frontend.Flask and Python combined, provides a robust and scalable infrastructure for integrating the ML model seamlessly into the web application, ensuring efficient communication and smooth user experience.

1. **Screenshots:**

**Lokesh bhai maine jo tujhe screenshot ka email bheja hai😅 unme se 3 screenshot yaha laga dena**

1. **Code Snippet:**

**Aditya 4-5 code ke screenshot bhejega vo yha par laga dena**