Full Stack Development with MERN

Frontend Development Report

Date	13 July 2024
Team ID	PNT2022TMIDSWTID1721142115
Project Name	Video Conferencing Application
Maximum Marks	10 marks

Project Title: Video Conferencing Application [Smart Meet]

Date: 13. 07. 2024

Prepared by: Meetminds (SWTID1721142115)

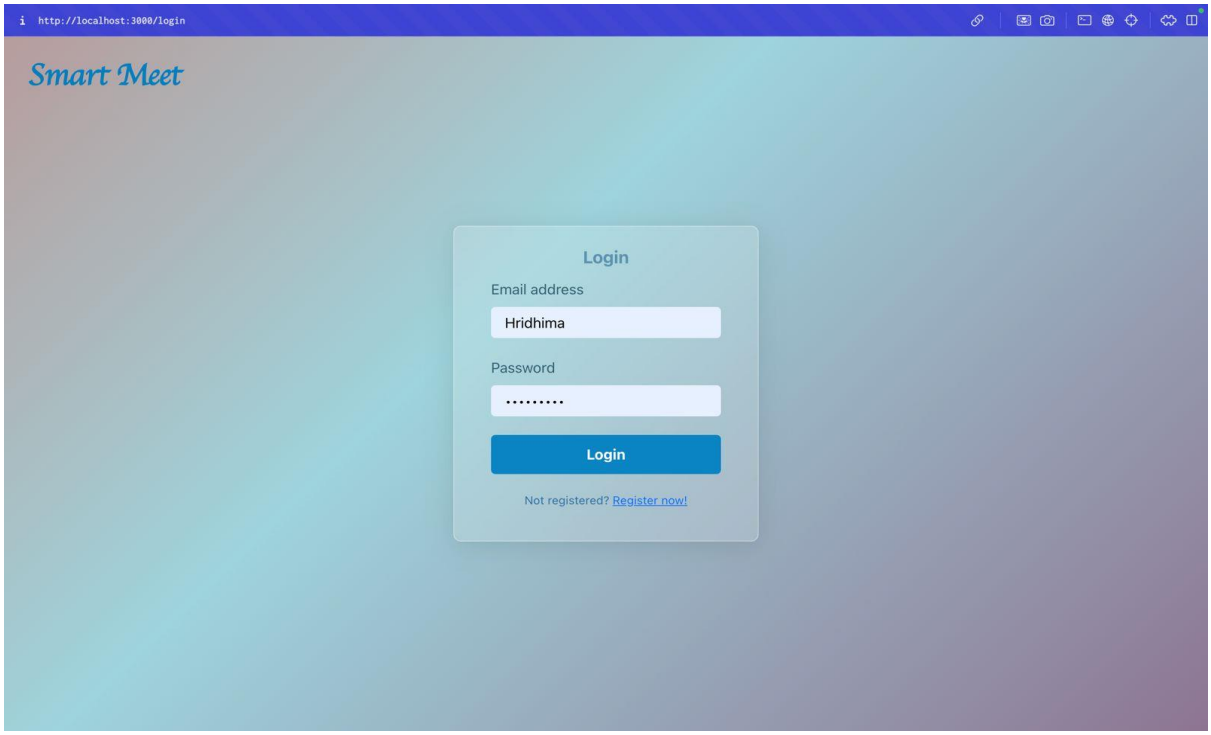
Objective

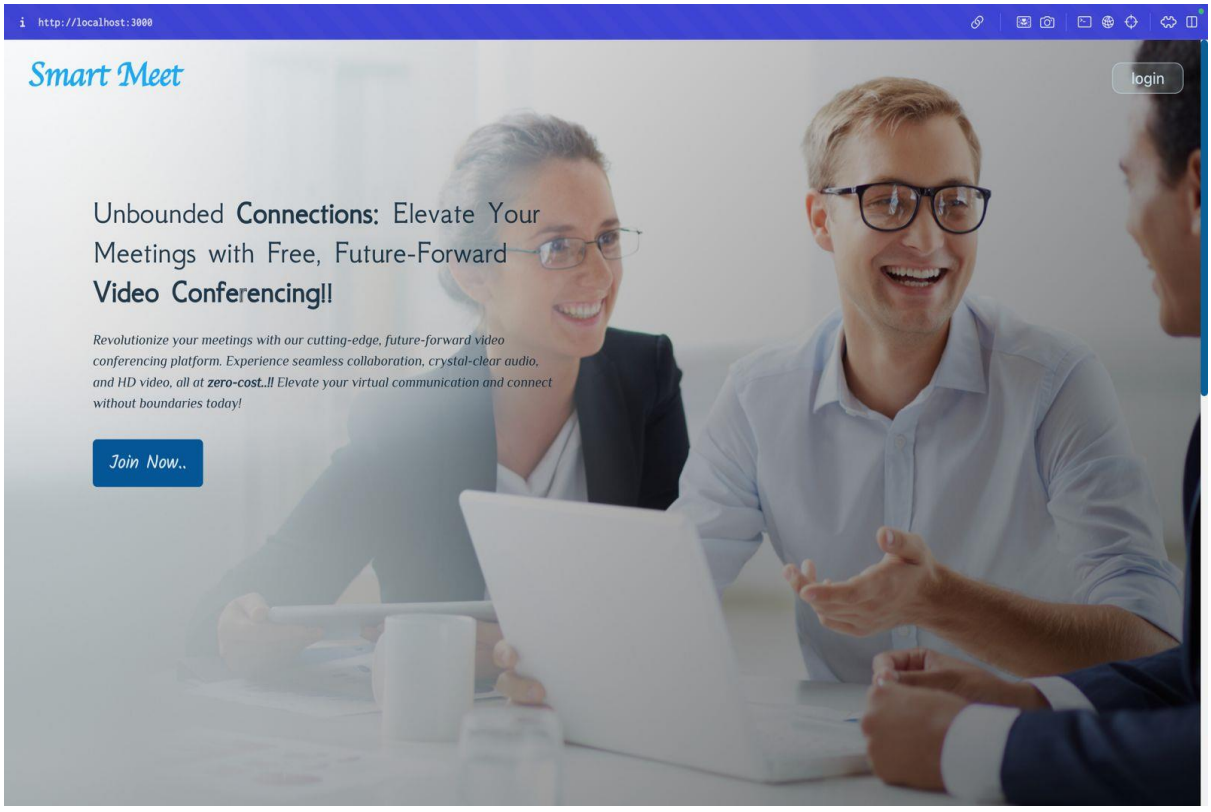
The objective of this report is to document the frontend development progress and key aspects of the user interface implementation for the Video Conferencing Application project.

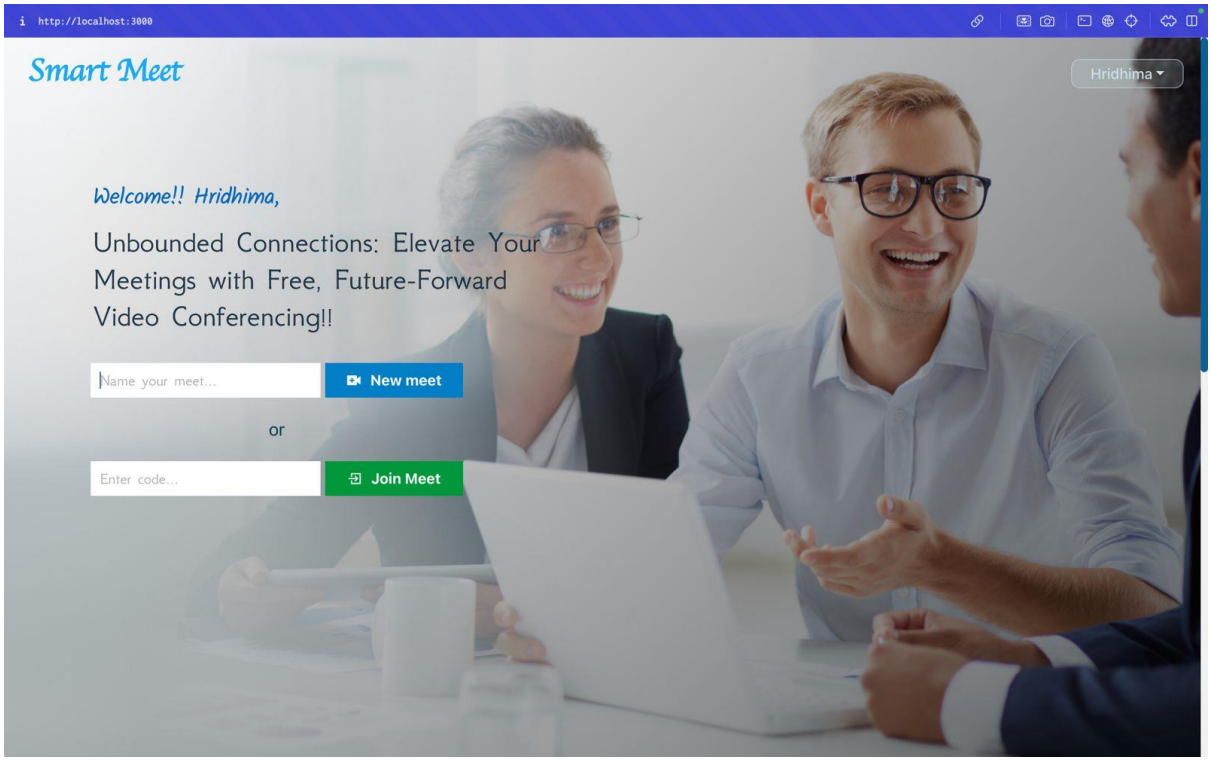
Technologies Used

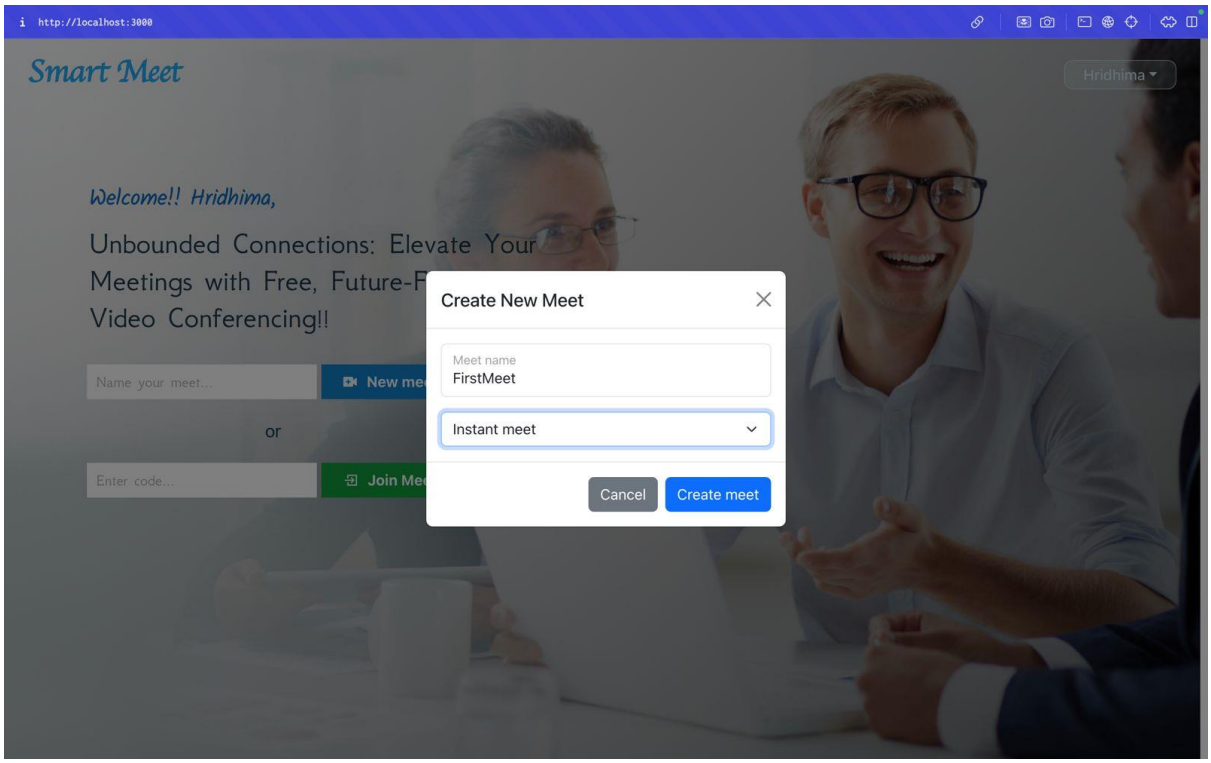
- **Frontend Framework:** React.js
- **State Management:** Context API
- **UI Framework/Libraries:** Material-UI
- **API Libraries:** Axios

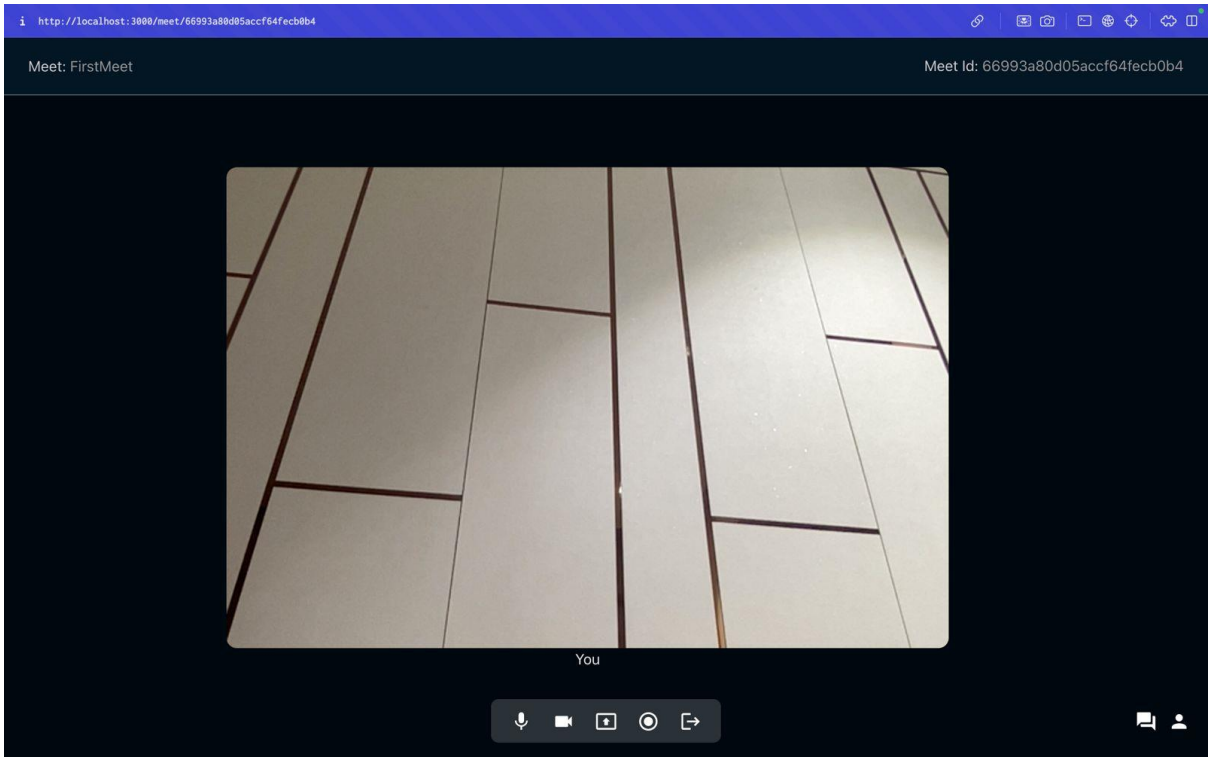
Project Structure:

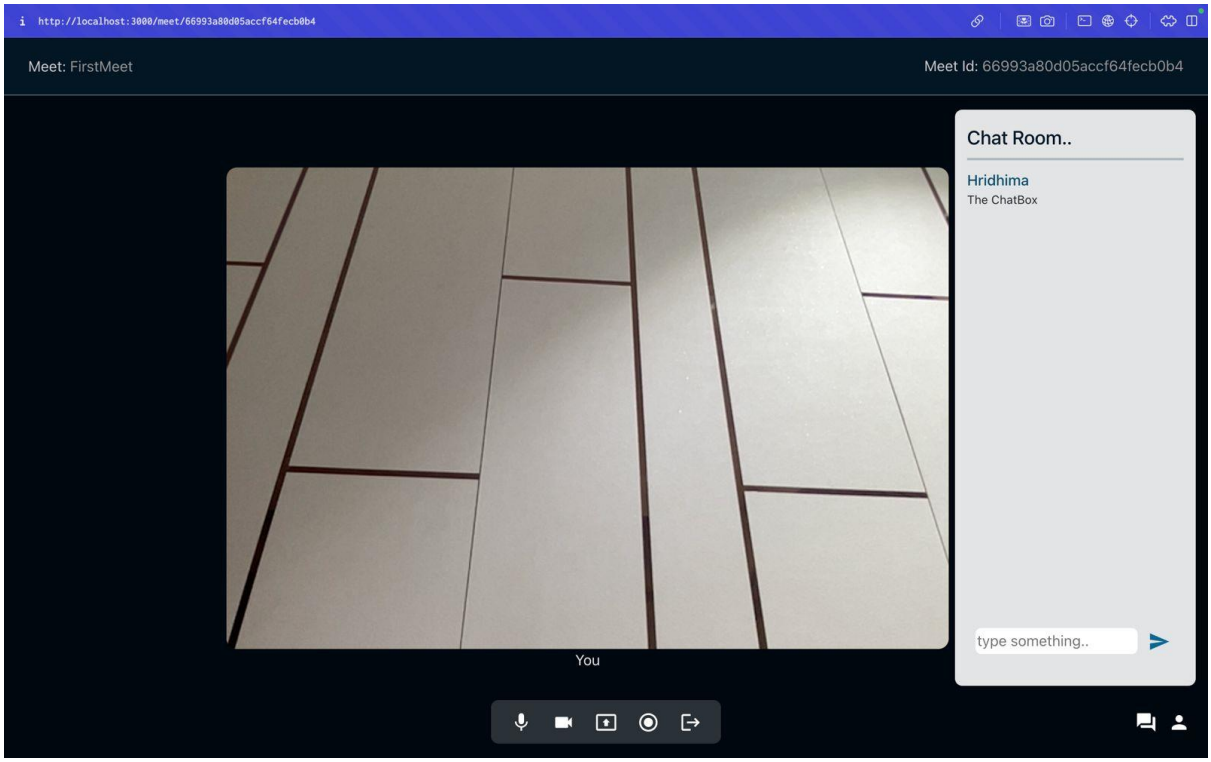

http://localhost:3000/login
Smart Meet
Login
Email address
Hridhima
Password
Login
Not registered? Register now!


http://localhost:3000
Smart Meet
login
Unbounded Connections: Elevate Your Meetings with Free, Future-Forward Video Conferencing!!
Revolutionize your meetings with our cutting-edge, future-forward video conferencing platform. Experience seamless collaboration, crystal-clear audio, and HD video, all at zero-cost..!! Elevate your virtual communication and connect without boundaries today!
Join Now..


http://localhost:3000
Smart Meet
Hridhima
Welcome!! Hridhima,
Unbounded Connections: Elevate Your Meetings with Free, Future-Forward Video Conferencing!!
Name your meet...
New meet
or
Enter code...
Join Meet


http://localhost:3000
Smart Meet
Hridhima
Welcome!! Hridhima,
Unbounded Connections: Elevate Your
Meetings with Free, Future-F
Video Conferencing!!
Name your meet...
New me
or
Enter code...
Join Me
Create New Meet
Meet name
FirstMeet
Instant meet
Cancel
Create meet





Overview of the web application:

The Smart Meet video conferencing web application features a structured and user-friendly front-end, developed with the MERN stack.

Upon accessing the application, users are greeted by the Landing Page, which serves as an introduction and provides navigation options to sign up or log in. This page is designed to welcome both new and returning users, offering a brief overview of the application's capabilities.

The Sign-Up and Login Pages are essential for user account management, allowing individuals to create a new account or log in to an existing one. These pages include input fields for credentials, submit buttons, and error handling to ensure valid entries.

Once logged in, users are directed to the Front page o of the application. This serves as the main control center where users can join existing meetings or create new ones. The dashboard also displays upcoming meetings, facilitating easy access to various functionalities.

The core of the application is the Meeting Room, where video conferencing takes place. This room is equipped with features such as video and audio controls, participant management, real-time chat, and screen sharing. Additionally, users can record the meeting.

Navigation across the application is facilitated by a consistent Navigation Bar, ensuring quick access to primary features. The application's design is also responsive, ensuring usability across devices like desktops, tablets, and mobile phones.

Overview of the Coding part:

The frontend architecture of the 'Smart Meet' video conferencing application is meticulously designed to enhance modularity, scalability, and maintainability. At its core is the src/ directory, serving as the central hub for all application logic and components. Within src/, distinct subdirectories are strategically organized: assets/ houses static resources like images and icons essential for the application's visual elements, while components/ contains modular UI elements such as Navbar.js and Footer.js, each encapsulating specific functionality and styling. In parallel, the context/ directory manages global state through UserContext.js, ensuring consistent data access across various parts of the application. The pages/ directory holds pivotal views like Home.js and Dashboard.js, each dedicated to presenting different aspects of user interaction and data visualization. Meanwhile, the services/ directory facilitates seamless integration with backend services via api.js, streamlining data fetching and management through efficient API calls.

Critical to the application's structure, App.js orchestrates the overall layout and routing, while index.js initializes the React application, rendering it into the browser's DOM. Complementing these, setupTests.js configures the testing environment, supporting robust quality assurance practices.

Key Components

1. **App.js**
    - Responsible for routing and main application layout.

2. **/components**
    - Contains reusable UI components such as Navbar, Footer and Video Call components
3. **/pages**
    - Includes different pages like Home, Dashboard, Profile and Meeting.

Routing

Routing is managed using React Router. Here are the main routes:

- **/home** - Landing page of the application.
- **/dashboard** - Dashboard displaying user data and meeting statistics.
- **/profile** - User profile management.
- **/meeting**- Video conference page

State Management (If Applicable)

State management is achieved using Context API. It helps in managing global state like user authentication and meeting details.

Integration with Backend

The frontend communicates with the backend APIs hosted on [backend URL]. Key endpoints include:

- **GET /api/meetings** - Retrieves meeting data for display.
- **POST /api/user/login** - Handles user authentication.
- **POST/api/meeting/create** - Creates a new meeting session.

User Interface (UI) Design

- The UI design follows modern and intuitive design principles:

    - ✓ Consistency: Uniform look and feel across the application.
    - ✓ Responsiveness: Adapts to various screen sizes and devices.
    - ✓ Accessibility: Follows accessibility guidelines for usability by all.
    - ✓ User-Centric: Focuses on ease of use and user satisfaction.

- Implemented using material UI:

    - ✓ Material-UI: A React UI framework that follows Google's Material Design guidelines.
    - ✓ Customization: Allows extensive customization to match Smart Meet’s design requirements.

- ✓ Component Library: Uses a rich set of pre-designed components like buttons, forms, and dialogs.
- ✓ Performance: Optimized for a fast and smooth user experience.

Third-Party Integrations (If any)

- Socket.IO: Used for real- time communication during video conferences.