**Project Design Phase-II**

**Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 15 JUNE 2025 |
| Team ID | LTVIP2025TMID46320 |
| Project Name | DOC SPOT |
| Maximum Marks | 4 Marks |

**Technical Architecture:**

The DocSpot project follows a MERN Stack (MongoDB, Express.js, React.js, Node.js) architecture that enables a full-stack, scalable, and efficient web application for healthcare appointment booking.Example: Order processing during pandemics for offline mode.

**Frontend (Client-Side):**

React.js is used to build a dynamic, single-page application (SPA) for a smooth user experience.

Axios handles HTTP requests to communicate with the backend.

Material UI / Bootstrap provides responsive and modern UI components.

**Backend (Server-Side):**

Node.js with Express.js manages API routing, user authentication, appointment scheduling, and data validation.

Role-based access control is implemented for patients, doctors, and admins.

**Table-1 : Components & Technologies:**

|  |  |  |
| --- | --- | --- |
| **S.No** | **Component** | **Description** |
|  | User Interface | Web and mobile-friendly interface for patients and providers |
|  | Application Logic-1 | Appointment booking, calendar management, reminders |
|  | Application Logic-2 | Admin panel, provider management, reporting |
|  | Database | Stores user profiles, appointments, provider data |

**Table-2: Application Characteristics:**

| **S.No** | **Characteristics** | **Description** | **Technology** |
| --- | --- | --- | --- |
|  | Open-Source Frameworks | |  | | --- | | Frontend frameworks |  |  | | --- | |  | | React.js, Node.js, Bootstrap, Tailwind CSS |
|  | Scalable Architecture | 3-tier architecture with RESTful APIs | Microservices |