**Project Design Phase-II**

**Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 20 June 2025 |
| Team ID | SWUID20250177139 |
| Project Name | Social Media App |
| Maximum Marks | 4 Marks |

**Technical Architecture:**

The application follows a modular full-stack architecture using the MERN stack and Socket.io for real-time communication. It uses a client-server model with a RESTful API and WebSocket connections.

Client (React.js) ←→ REST API / WebSocket (Socket.io) ←→ Server (Node.js + Express.js) ←→ Database (MongoDB)

* Frontend communicates with the backend via:
  + REST API (for standard data exchange)
  + WebSockets (for real-time features like chat and notifications)
* Backend server handles:
  + Routing and logic (Express.js)
  + Real-time communication (Socket.io)
  + Data modeling and access (Mongoose + MongoDB)

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

| **Component** | **Technologies Used** |
| --- | --- |
| Frontend | React.js, Redux, Axios, Socket.io-client |
| Backend | Node.js, Express.js, Socket.io |
| Database | MongoDB, Mongoose |
| Authentication | JWT (JSON Web Tokens), bcrypt |
| Real-Time Communication | Socket.io (WebSockets) |
| Media Handling | Firebase Storage / Cloudinary |
| State Management | Redux (Frontend), Session/JWT (Backend) |
| UI Frameworks | Bootstrap, CSS Modules |
| API Communication | RESTful APIs, WebSockets (Socket.io) |
| Deployment | Vercel/Netlify (frontend), Render/Heroku (backend), MongoDB Atlas |
| Development Tools | Git, GitHub, Postman, VS Code, Chrome DevTools |
| Security | HTTPS, CORS, dotenv (.env), Input Validation, Token-based Auth |

### **Table-2: Application Characteristics**

| **Characteristic** | **Description** |
| --- | --- |
| Architecture | Modular, Client-Server (MERN Stack) with REST APIs and WebSockets |
| Scalability | Horizontally scalable using MongoDB and stateless backend |
| Real-time Communication | Enabled through Socket.io (WebSocket-based messaging and updates) |
| Security | JWT Authentication, bcrypt for password hashing, HTTPS, CORS |
| Responsiveness | Fully responsive UI for mobile, tablet, and desktop |
| Usability | User-friendly interface with intuitive navigation and interactive components |
| Performance | Optimized API routes, minimal latency in messaging and notifications |
| Data Storage | NoSQL structure with MongoDB for flexible, document-based storage |
| Media Support | Supports image/video upload via Firebase or Cloudinary |
| Extensibility | Easily extendable to include features like video calls, AI suggestions, etc. |
| Deployment Readiness | Frontend (Vercel/Netlify), Backend (Render/Heroku), DB (MongoDB Atlas) |
| Maintainability | Clean codebase with modular folders, version control via Git |