

Project Design Phase-II
Technology Stack (Architecture & Stack)

| | |
|---------------|------------------|
| Date | 26-05-2025 |
| Team ID | SWUID20250172526 |
| Project Name | SocialeX |
| Maximum Marks | 4 Marks |

Technical Architecture:

The **SocialeX** platform is designed with a scalable, real-time **3-tier architecture**, including:

- Presentation Layer (Frontend):** Built with React.js and Bootstrap, providing a responsive and modern UI.
- Business Logic Layer (Backend):** Node.js and Express.js handle core logic, API routing, authentication, and chat integration.
- Data Storage Layer:** MongoDB stores user profiles, posts, stories, and message logs. Firebase handles media storage.
- Real-time Layer:** Socket.io enables live messaging and notifications.

The architecture emphasizes **real-time interaction**, **security**, and **modularity**, enabling easy integration of future features (e.g., reactions, voice chat, analytics).

Table-1: Components & Technologies

| S. No | Component | Description | Technology |
|-------|---------------------|---|--|
| 1 | User Interface | Web interface for social interaction | HTML, CSS, JavaScript, React.js, Bootstrap |
| 2 | Application Logic-1 | Post/story creation, feed rendering, profile management | Node.js, Express.js |
| 3 | Application Logic-2 | Chat system, notifications, story expiry automation | React.js, Socket.io, Node.js |
| 4 | Database | Stores users, posts, stories, messages | MongoDB (with Mongoose ODM) |
| 5 | Media Storage | Secure file storage and access | Firebase Storage |

Table-2: Application Characteristics

| S. No | Characteristics | Description | Technology |
|-------|---------------------------|--|--|
| 1 | Open-Source Frameworks | Core development frameworks | React.js, Node.js, Bootstrap, Tailwind CSS |
| 2 | Scalable Architecture | RESTful APIs, real-time services, modular services | Microservices architecture, Socket.io |
| 3 | Real-time Communication | Live messaging, notifications, story sync | Socket.io, WebSocket Protocol |
| 4 | Authentication & Security | Secure login and session handling | JWT, Bcrypt, HTTPS, Firebase Rules |

References:

- React.js Official Documentation
- Node.js Best Practices
- MongoDB + Mongoose Guides
- Firebase Storage & Access Rules
- [How to Draw Technical Architecture Diagrams – Medium](#)