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

**Date:** 26-05-2025  
**Team ID:**

**Team Members: Yuvraj Singh and Amrutansh Panda**   
**Project Name:** SocialeX  
**Maximum Marks:** 4 Marks

**Technical Architecture**

The **SocialeX** platform is designed with a scalable **3-tier architecture**, consisting of:

* **Presentation Layer (Frontend):** Provides the user interface for messaging, media sharing, stories, calls, and settings.
* **Business Logic Layer (Backend):** Handles authentication, real-time chat, media storage, stories management, calls, and user settings.
* **Data Storage Layer:** Manages storage of user data, messages, media, and call logs securely.

This architecture ensures **real-time communication, security (including encryption and app lock), scalability (microservice-ready), and rich media support**.

**Table-1: Components & Technologies**

| **S.No** | **Component** | **Description** | **Technology** |
| --- | --- | --- | --- |
| 1 | User Interface | Web/mobile-friendly interface for chat, stories, calls, settings | React.js, HTML, CSS, JavaScript, Tailwind CSS |
| 2 | Application Logic-1 | Chat service, media sharing, app lock, mood-based themes | Node.js, Express.js |
| 3 | Application Logic-2 | Stories feature, voice/video calling, settings management | React.js (frontend), Node.js (backend) |
| 4 | Database | Stores user profiles, messages, media files, stories, call logs | MongoDB |

**Table-2: Application Characteristics**

| **S.No** | **Characteristics** | **Description** | **Technology** |
| --- | --- | --- | --- |
| 1 | Open-Source Frameworks | Frontend and backend frameworks | React.js, Node.js, Tailwind CSS, Socket.io |
| 2 | Real-Time Communication | Enable live messaging and calls | WebSockets, Socket.io |
| 3 | Secure Architecture | Token-based authentication, app lock with PIN/biometric | JWT, HTTPS, bcrypt, device biometrics |
| 4 | Scalable Architecture | 3-tier architecture with RESTful and WebSocket APIs | Microservices, Load Balancers |
| 5 | Media Handling | Support image/video sharing in chat and stories | Cloud Storage (e.g., AWS S3 / Firebase Storage) |

**References**

* React.js Documentation – <https://react.dev>
* Node.js Best Practices – <https://nodejs.dev>
* Socket.io Documentation – <https://socket.io>
* JWT.io – <https://jwt.io>
* Example on Technical Architecture Diagrams – <https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d>