**Project Design Phase**

**Solution Architecture**

|  |  |
| --- | --- |
| Date | 6 June 2025 |
| Team ID | SWUID20250169335 and SWUID20250175086 |
| Project Name | Social Media App |
| Maximum Marks | 4 Marks |

**Solution Architecture:**

**Component Breakdown**

**Client (Frontend) – React.js**

* Handles UI, navigation, and rendering of:
  + Feeds, posts, stories
  + User profile and chats
* Uses **Axios** for REST API calls
* Uses **Socket.io-client** for real-time messaging

**Backend (Server) – Node.js + Express**

* **REST API**: Handles auth, post/story CRUD, user management
* **WebSocket (Socket.io)**: Enables live chat, typing indicators, real-time updates
* **Controllers**: Handle logic for user, post, story, chat, etc.
* **Middleware**: Auth verification, error handling, CORS

**Authentication Module**

* Uses **JWT** for secure token-based login
* **Bcrypt** for password hashing

**Real-Time Messaging Engine**

* **Socket.io server** listens for and emits events:
  + message, new-like, new-comment, story-upload, etc.
* Broadcasts updates to all connected clients

**Database – MongoDB**

* Stores:
  + User details, posts, stories, chats, followers
* Accessed using **Mongoose ODM**
* Supports relationships via schema referencing

**Security Components**

* Password hashing with **bcrypt**
* Token-based auth with **JWT**
* API protection with **CORS**, **dotenv**, and request validation

**Optional Add-ons**

* Media storage via **Firebase** or **Cloudinary**
* Deployment on **Vercel (client)** and **Render/Heroku (server)**

**Architecture:**

