**Project Design Phase-II**

**Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| **Date** | **31st, March 2025** |
| **Team ID** | **SWTID1742493942** |
| **Project Name** | **Connectify - Social Media App** |
| **Maximum Marks** | **4 Marks** |

**Technology Stack**

Here’s the chosen technology stack categorized by the development layer:

**Frontend**

**React.js**: For building a dynamic, component-based user interface.

**HTML5 & CSS3**: Standard markup and styling.

**Tailwind CSS**: For responsive and modern UI design.

**Fetch API**: To make HTTP requests to the backend.

**React Router**: For managing routing between pages like feed, profile, login, etc.

**Backend**

* **Node.js**: JavaScript runtime used to build the backend services.
* **Express.js**: For creating RESTful APIs and handling backend routing.
* **JWT (JSON Web Token)**: For secure user authentication and session handling.
* **Bcrypt.js**: For password hashing to enhance login security.

**Database**

**MongoDB (MongoDB Atlas)**: NoSQL database used to store user data, posts, comments, messages, and more.

**Users**

* Stores all user account details (used for both Login and Signup functionality)
* Fields: name, email, passwordHash, profilePic, bio, createdAt, isVerified, etc**.**

**Posts**

* User-generated content like status updates, media posts, etc.
* Fields: userId, caption, mediaURL, timestamp, likes, comments

**Comments**

* Contains user comments linked to posts
* Fields: postId, userId, text, timestamp

**Messages**

* For storing 1:1 or group chat messages
* Fields: senderId, receiverId, message, timestamp

**SetReminder**

* Stores user-defined reminders shown in the left-side panel
* Fields: userId, title, description, dateTime, status, createdAt

**Cloud & Storage**

* Firebase Storage or AWS S3: For storing music files and album art
* Cloudinary: Optional, for handling media transformations

**Deployment**

**Frontend**: Deployed using **Vercel** or **Netlify** for continuous deployment and ease of access.

**Backend**: Hosted using **Render** or **Railway**, allowing smooth integration with MongoDB Atlas and continuous integration.

**MongoDB Atlas**: Cloud-based database hosting.

**Testing Tools**

**Postman**: Used to test backend APIs during development.

**Browser-based Testing**: Manual testing of frontend UI, interactions, and responsiveness across devices.

**Analytics & Monitoring**

**Google Analytics**: For tracking user interactions and website traffic

**MongoDB Charts / Custom Admin Dashboard**: For visualizing post counts, user activity, and reports.