



COLLEGE CODE : 9623

COLLEGE NAME : Amrita College Of
Engineering And Technology

DEPARTMENT : Computer Science and
Engineering

STUDENT NM-ID :

F58A45CD199F65582904B377E24880E9

ROLL NO : 962323104078

DATE : 9-10-2025

Completed the project named as Phase_04_
ChatApplicationUI

NAME : Chat Application UI

SUBMITTED BY,

NAME : Raffrin Narmadh V M

MOBILE NO : 7845411725

Overview

- Focus: Designing, developing, and implementing a real-time Chat Application User Interface (UI).

- **Objectives:**

- Create an intuitive and user-friendly chat interface.
- Enable seamless communication between users.
- Ensure responsive and consistent UI across devices.

Features

- **Core Features:**

- Real-time messaging system.
- User authentication & profile management.
- Message delivery status (sent, delivered, seen).
- Chat history & media sharing support.

- **User Convenience:**

- Light/Dark theme support.
- Typing and online indicators.

UI/UX Design

- **Design Focus:**

- Clean, minimal, and modern interface.

- Easy navigation between chat lists and messages.

- Consistent color palette and typography.

- **Mockups / Wireframes:**

- Login screen, chat window, contact list.

- Responsive layout for mobile & desktop views.

Technology Stack

- Frontend: HTML, CSS, JavaScript, React.js
- Backend: Node.js with Express
- Database: MongoDB / Firebase
- Communication: Socket.io for real-time chat
- Tools Used: Figma (UI design), Postman (API testing)

Performance & Security

- Performance:
 - Optimized rendering for large message lists.
 - Efficient API calls to reduce latency.
- Security:
 - JWT Authentication for secure sessions.
 - End-to-end message encryption (conceptual).
 - Regular input validation to prevent XSS or injection attacks.

Testing

- Testing Methods:
 - Unit Testing (Frontend & Backend).
 - Integration Testing for message flow.
 - UI Testing using Cypress / Selenium.
- Results:
 - Smooth message synchronization.
 - Stable performance under multiple concurrent users.

Deployment

- Deployment Pipeline: Build → Test → Deploy → Monitor
- Platforms:
 - Frontend: Netlify / Vercel
 - Backend: Render / AWS Cloud
- Post-Deployment:
 - Real-time monitoring and error tracking.
 - Backup and rollback setup.

Summary

- Successfully developed a responsive and modern Chat UI.
- Improved real-time communication using Socket.io.
- Secure authentication and efficient message flow implemented.
- Ready for future scalability and mobile app adaptation.