



**COLLEGE CODE :- 9509**

**COLLEGE NAME :-** Holy Cross Engineering College

**DEPARTMENT :- CSE**

**STUDENT NM-ID:-**

**F56CFE83F6612DBB5FC42499263F8CA1**

**DATE :- 29/09/2025**

**Completed the project named as**

**Phase\_\_TECHNOLOGY PROJECT NAME :**

**CHAT UI APPLICATION**

**SUBMITTED BY,**

**NAME :- K.Murugesh**

**MOBILE NO :- 8300829750**

## Phase 5 — Project Demonstration & Documentation

### 1. Final Demo Walkthrough

#### Overview:

The Chat Application UI provides a clean and responsive interface that allows users to send and receive messages in real-time. The design focuses on simplicity, readability, and modern aesthetics using React + CSS.

#### Demo Features:

- Login / Join Page: User enters a name to join the chat room.
- Chat Window: Real-time message updates with smooth scrolling.
- Message Input Box: Send messages instantly.
- Active Users Section: Displays the list of users currently online.
- Timestamp: Shows when each message was sent.
- Responsive UI: Works across desktop and mobile devices.

### 2. Project Report

Title: Chat Application UI

Technology Stack: React.js, CSS, JavaScript, Node.js (for backend API), Socket.io (for real-time chat)

#### Objective:

To design a modern, responsive chat interface that supports smooth real-time communication and provides a visually appealing user experience.

#### Modules:

##### 1. Frontend (React):

- o Components: ChatWindow, Message, InputBox, UserList
- o State management using React Hooks

##### 2. Backend (Optional / Future Integration):

- o Node.js server with Socket.io for message broadcasting

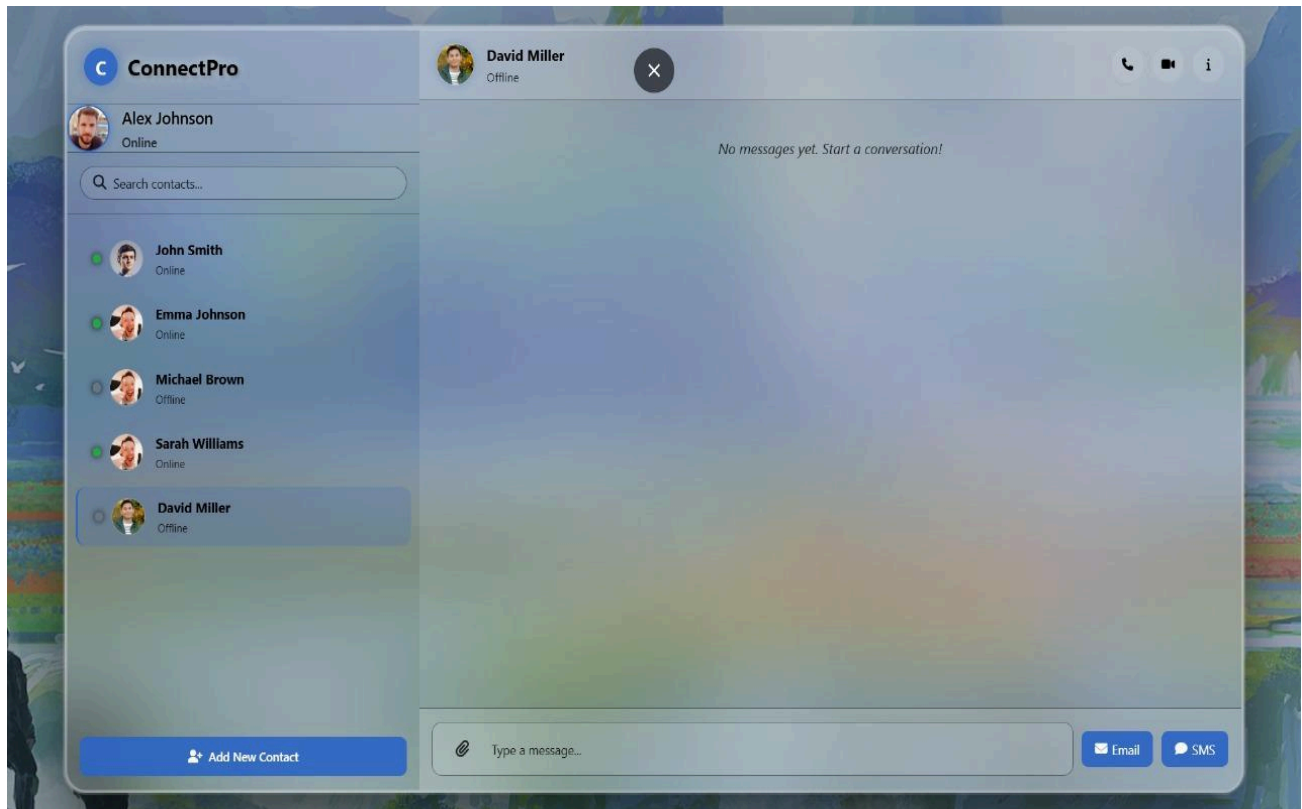
##### 3. Styling:

- o Custom CSS with Flexbox and Grid layout for responsiveness

#### Key Features:

- Live messaging
- Auto-scroll on new messages
- Minimalist design
- User status indicator

### 3. Screenshots / API Documentation.



### 4. Challenges & Solutions:

Challenge

Designing responsive layout

Auto-scroll for new messages

Managing chat states

Keeping UI consistent

Solution

Used Flexbox & media queries in CSS

Used useEffect with scrollToView()

Used React hooks (useState, useEffect) efficiently

Created reusable components for message and input box

· Final Submission:

REPO LINK:

<https://github.com/kmuruges325-gif/Chatifyy.git>

Deployment link : <https://chatify-ten-sand.vercel.app/>