System Design Document for Connect App

Name: Avani Gupta

University: IIT Hyderabad

Department: Mechanical and Aerospace Engineering

- Table of Contents

1. Introduction

This document outlines the system design for a Chat App(Connect App), a real-time messaging platform designed to facilitate communication between users. The Chat App aims to provide users a seamless and secure chatting experience across various platforms, including web and mobile.

2. System Overview

The Connect App system consists of three main components: the client-side application, the server-side application, and a database. Users interact with the client-side application to send and receive messages, while the server-side application handles message routing, user authentication, and data storage. The database stores user profiles, messages, and other relevant data.

3. System Architecture

3.1. Client-Side

The client-side application is responsible for the user interface and handling user interactions. It is implemented as a web application. Key components include:

- User Interface (UI): Provides a user-friendly interface for sending and receiving messages and friend requests, managing contacts, configuring settings, managing call logs, managing user profiles, and has a light/dark adjustable theme.

- Message Rendering: Displays messages in real-time, supports multimedia attachments, and organizes conversations.

User Authentication: Allows users to sign up, log in, and change their password. Links and OTPs for user verification are sent via mail for authentication.

- Real-Time Messaging Client: Connects to the server using Socket.io to enable real-time message delivery.

3.2. Server-Side

The server-side application acts as an intermediary between clients, managing message routing and handling core functionality. It includes:

- Message Routing: Direct messages to the appropriate recipients, ensuring real-time message delivery.

- User Authentication: Verifies user credentials and issues access tokens for secure communication.

- User Management: Handles user accounts, profiles, and contacts.

- Real-Time Messaging Server: Manages Socket.io connections, maintains presence information, and supports broadcasting.

3.3. Database

The database stores essential data, including user profiles, messages, contacts, friends lists, call logs, and settings. It consists of collections for:

- Users: Stores user account information, such as usernames, passwords (stored in an encrypted form), and profile details.

- Messages: Records chat messages, including sender, recipient, timestamps, and content.

- Contacts: Manages user contacts, friend lists, and call logs.

- Settings: Stores user preferences, such as notification settings and theme choices.

4. Communication Protocols

The Chat App employs the Socket.io protocol to ensure real-time communication and data synchronization between clients and servers. Additionally, HTTPS is used for secure communication during authentication and data transmission.

5. User Authentication and Authorization

User authentication is handled using industry-standard practices, including username and password authentication or third-party authentication providers (e.g., OAuth). Upon successful authentication, access tokens are issued and used for subsequent authorization. For creating an account, a 2-step verification process is followed in which the user first enters their credentials and then is sent an OTP for further verification. If a user forgets a password, the user is asked to provide his/her registered email to create a new password. The link for doing so is sent via email to their registered email address, after which they can successfully create a new password and login to their account.

6. Data Storage and Management

Data is stored in a MongoDB database, chosen for its flexibility to accommodate varying scalability and performance demands. Data management encompasses efficient indexing, caching strategies, and regular database backups to ensure data reliability and accessibility.

7. Real-Time Messaging

Real-time messaging is achieved through Socket.io connections between clients and the server. Messages are queued for delivery when a user is offline and delivered upon reconnection. Users not only have text-based messaging but can also share various forms of media.

8. User Interface (UI)

The user interface is designed for a user-friendly and intuitive experience. It includes features such as message threading, chat history, multimedia support, starred messages, and contact management. Additionally, users can check other users' online/offline status in real-time, enhancing their ability to engage in timely and responsive conversations within the Chat App.

9. Security

Security measures include:

- Data encryption in transit and at rest.
- Secure password
- Rate limiting and intrusion detection.
- Regular security audits and updates to address vulnerabilities.

10. Conclusion

The Connect App system design outlined in this document provides a solid foundation for building a secure, scalable, and user-friendly messaging platform. It addresses key architectural, security, and operational considerations to ensure a robust and reliable application.

11. References

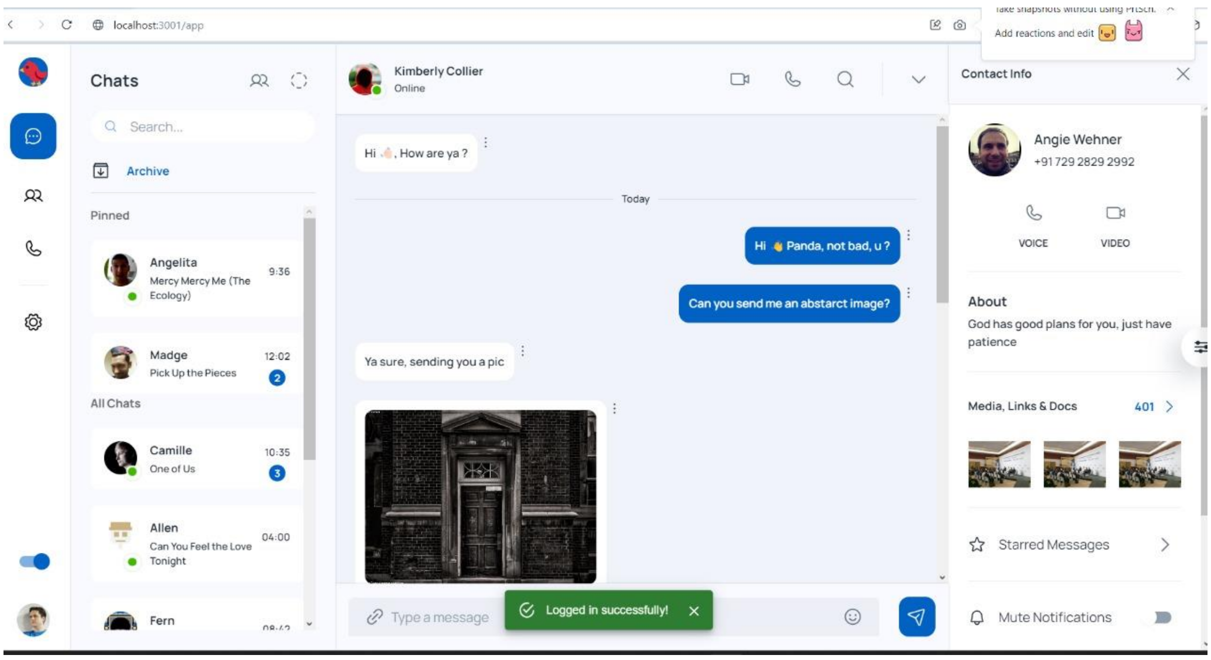
FrameWorks ssed in the project:

Frontend: React v18, Mui v5, and Redux.

Backend: NodeJs, ExpressJs, MongoDB, Mongoose, Socket.io, ZEGOCLOUD WebRTC API, SendGrid

Application Interface:

Home Page:



Angelita 9:36 Mercy Mercy Me (The Ecology)

Madge 12:02 Pick Up the Pieces 2

Camille 10:35 One of Us 3

Allen 04:00 Can You Feel the Love Tonight

Fern

Hi 👋, How are ya ?

Today

Hi 👋 Panda, not bad, u ?

Can you send me an abstarct image?

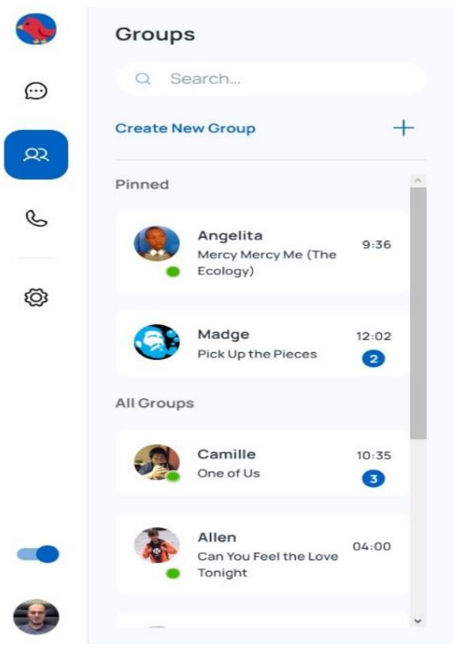
Ya sure, sending you a pic

About

God has good plans for you, just have patience

Logged in successfully!

Various Features Interface:


Groups

Search...

Create New Group

Pinned

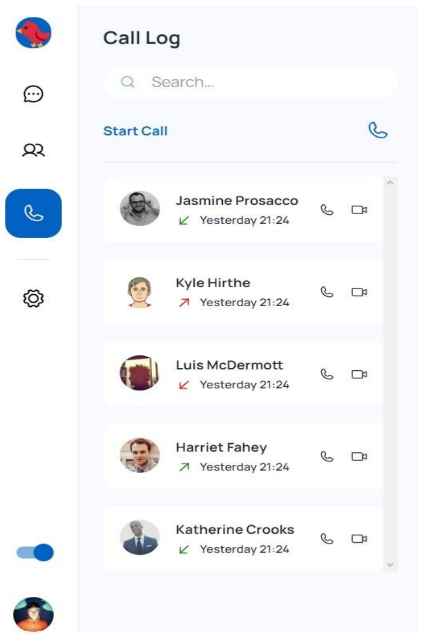
Angelita
Mercy Mercy Me (The Ecology)
9:36

Madge
Pick Up the Pieces
12:02
2

All Groups

Camille
One of Us
10:35
3

Allen
Can You Feel the Love Tonight
04:00


Call Log

Search...

Start Call

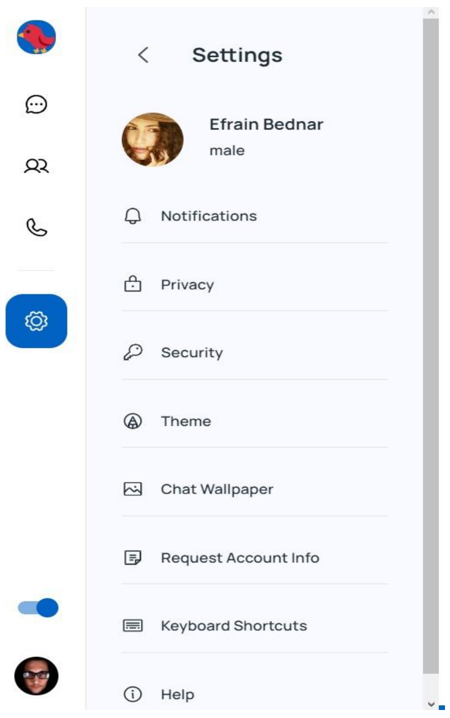
Jasmine Prosacco
Yesterday 21:24

Kyle Hirthe
Yesterday 21:24

Luis McDermott
Yesterday 21:24

Harriet Fahey
Yesterday 21:24

Katherine Crooks
Yesterday 21:24


Settings

Efrain Bednar
male

Notifications

Privacy

Security

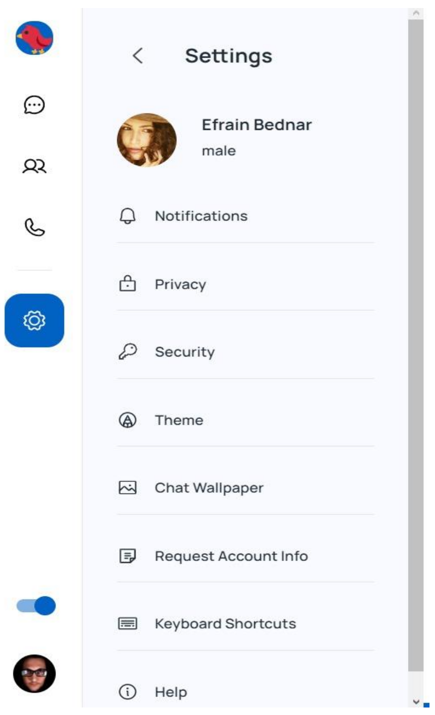
Theme

Chat Wallpaper

Request Account Info

Keyboard Shortcuts

Help


Settings

Efrain Bednar
male

Notifications

Privacy

Security

Theme

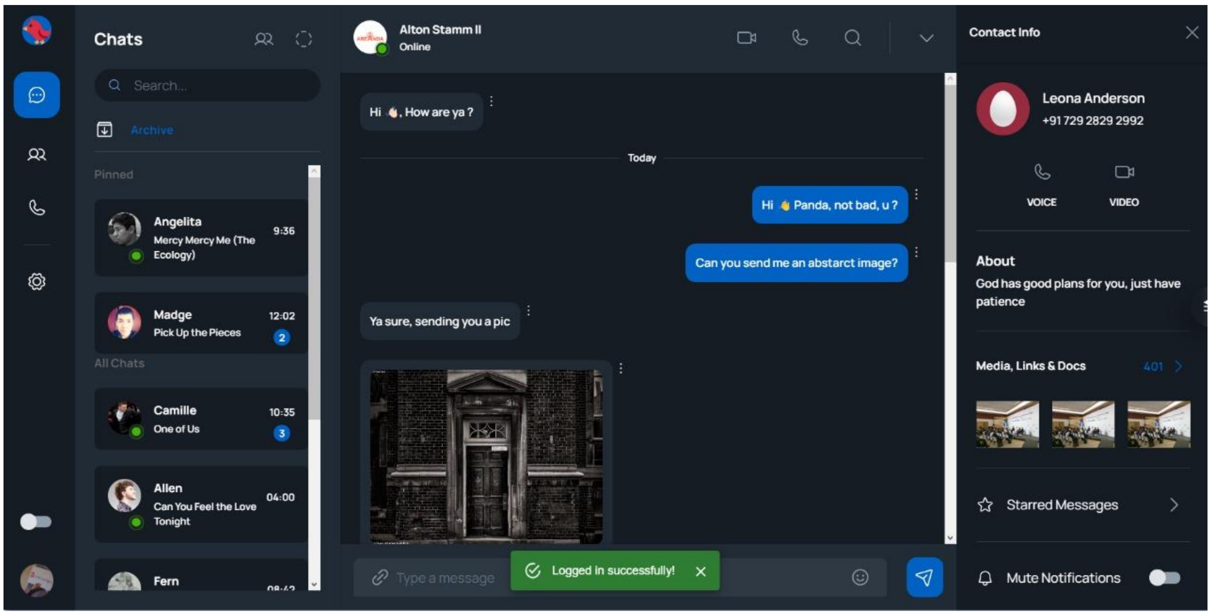
Chat Wallpaper

Request Account Info

Keyboard Shortcuts

Help

Dark and Light Theme:



Chats

Search...

Archive

Pinned

Angelita — Mercy Mercy Me (The Ecology) — 9:36

Madge — Pick Up the Pieces — 12:02 — 2

All Chats

Camille — One of Us — 10:35 — 3

Allen — Can You Feel the Love Tonight — 04:00

Fern — 08:42

Alton Stamm II
Online

Hi 👋, How are ya ?

Today

Hi 👋 Panda, not bad, u ?

Can you send me an abstarct image?

Ya sure, sending you a pic

Type a message

Logged in successfully!

Contact Info

Leona Anderson
+91 729 2829 2992

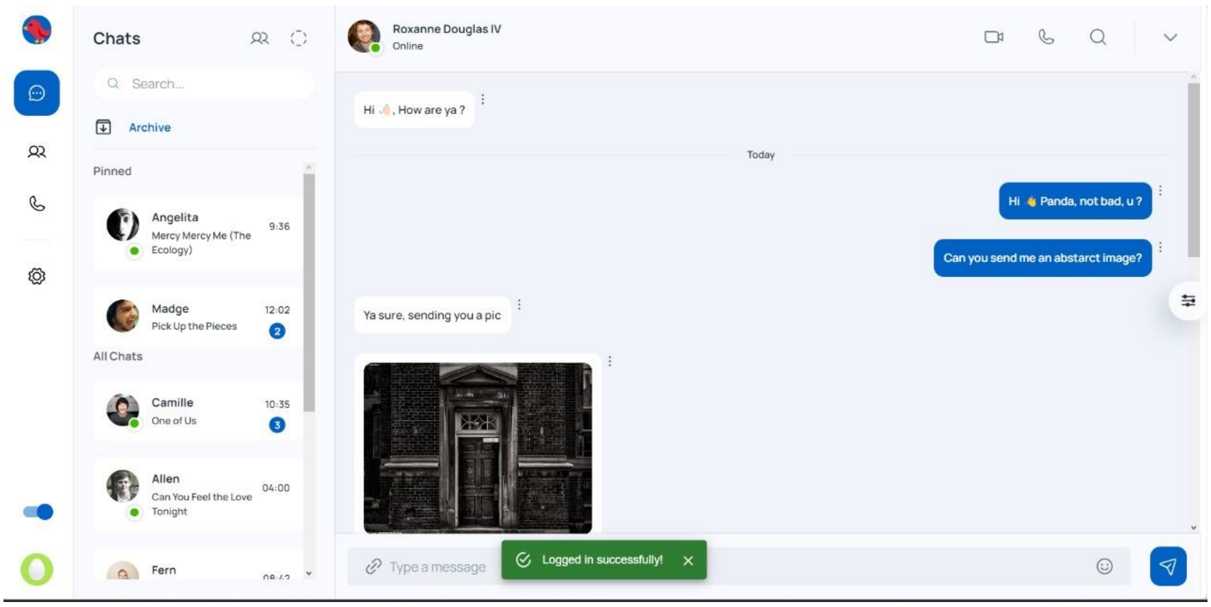
VOICE VIDEO

About
God has good plans for you, just have patience

Media, Links & Docs 401

Starred Messages

Mute Notifications



Chats

Search...

Archive

Pinned

Angelita — Mercy Mercy Me (The Ecology) — 9:36

Madge — Pick Up the Pieces — 12:02 — 2

All Chats

Camille — One of Us — 10:35 — 3

Allen — Can You Feel the Love Tonight — 04:00

Fern — 08:42

Roxanne Douglas IV
Online

Hi 👋, How are ya ?

Today

Hi 👋 Panda, not bad, u ?

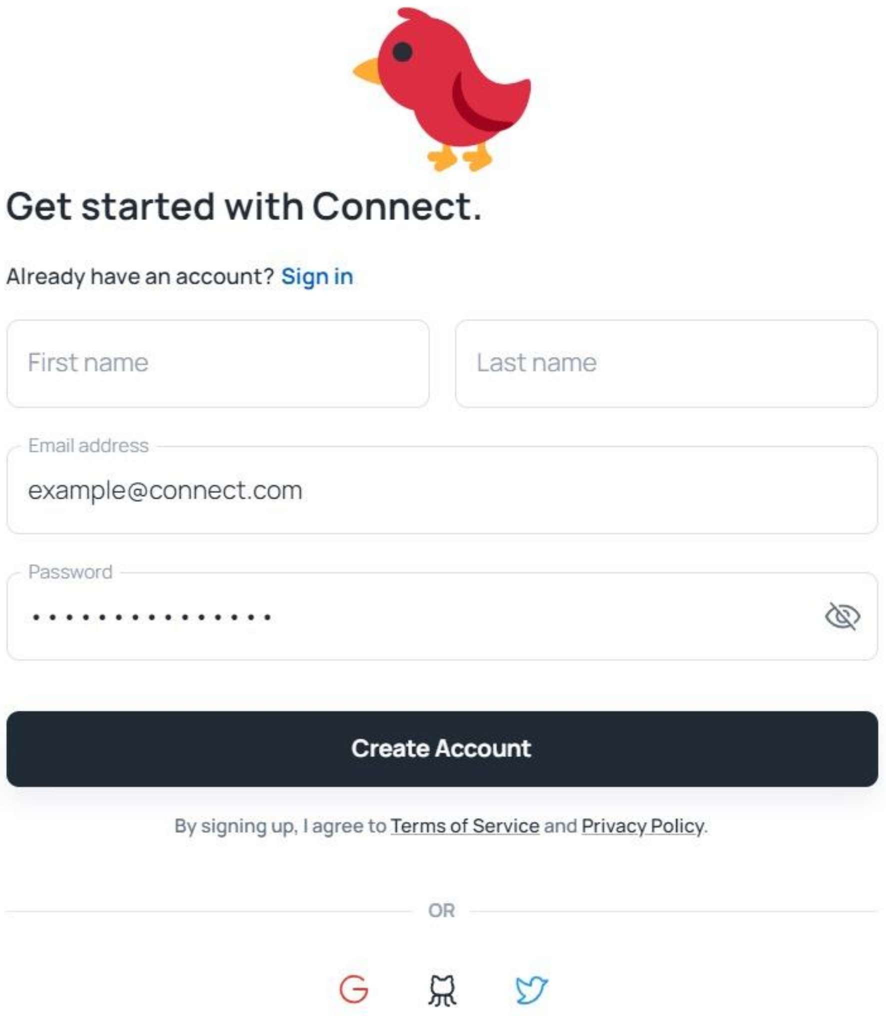
Can you send me an abstarct image?

Ya sure, sending you a pic

Type a message

Logged in successfully!

Auth Page:



Get started with Connect.

Already have an account? Sign in

First name

Last name

Email address

example@connect.com

Password

•••••••••••••••

Create Account

By signing up, I agree to Terms of Service and Privacy Policy.

OR