Harsha Vardhan

IIT Bhubaneswar
ECE

EchoChat

September 28, 2023

Introduction

1. **Purpose of the Document:**

   The purpose of this document is to provide a comprehensive overview of the system design for EchoChat, a web-based chat application.

2. **Document Scope:**

   This document covers the architecture, components, dependencies, and design decisions of EchoChat.

3. **Document Audience:**

   This document is intended for developers and evaluators involved in EchoChat's development.

Project Overview

1. **Description:**

   EchoChat is a real-time chat application that allows users to communicate via text, audio, and video in both one-on-one and group conversations.

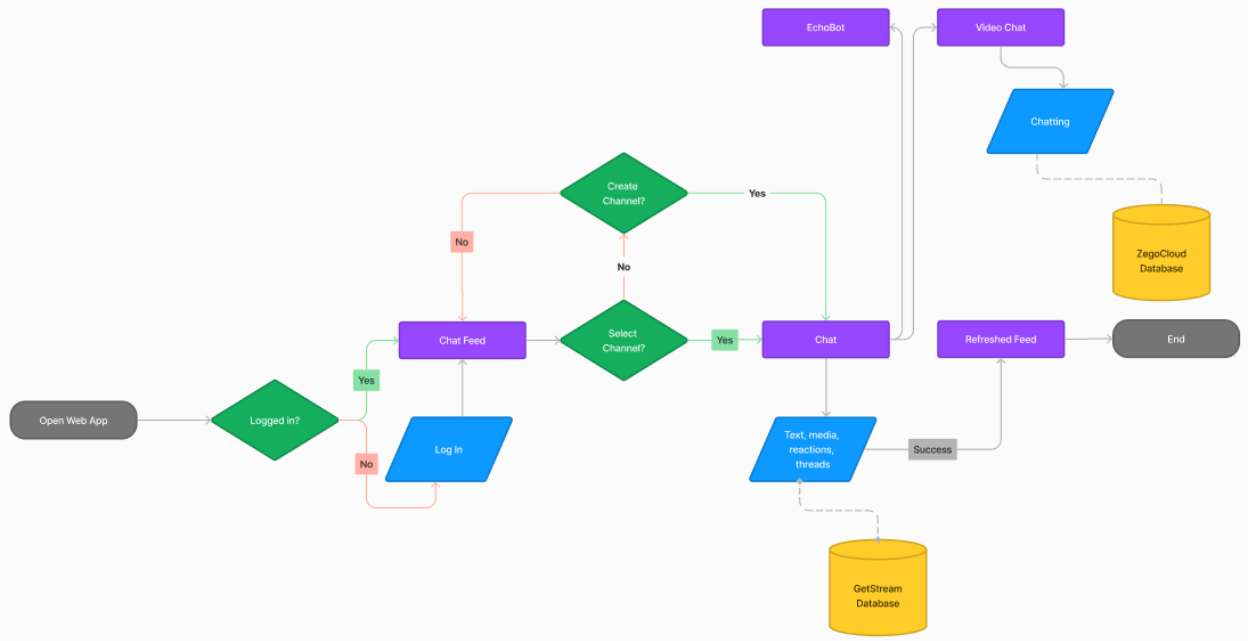
2. **Goals and Objectives:**
   - Create a user-friendly chat interface with real-time messaging capabilities.
   - Implement end-to-end encryption for user privacy and security.
   - Enable text and video messaging.

System Architecture

1. Overview

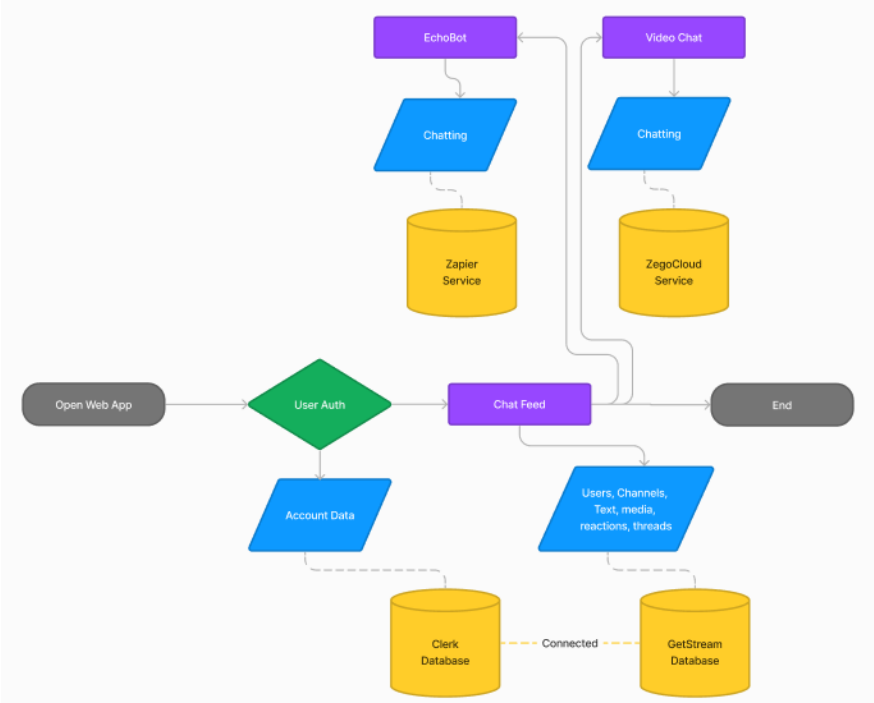
   This project uses Stream's chat SDK to provide a fully functional live chat with attachments, reactions, threads, and more. It also uses the web push API to send push notifications about new chat messages even if the browser window is closed.

   This project uses ZegoCloud's video SDK to provide a fully functional live video chat with many features such as screen sharing, chatting and layout options with sharable meeting link.



High-level Diagram

2. Client Application:
    - The front-end application used by end-users to create channels/groups, send and receive messages, connect to video meet, connec through EchoMeet and connect to EchoBot for chatting with AI powered with gpt-3.5
    - .UI components are inspired from GetStream(for text), ZegoCloud(for video meet).
3. Backend Application:
    - Most of the backend application is managed by services like GetStream(for text) and ZegoCloud(for video meet).
    - User Authentication is provided by Clerk Auth.
    - EchoBot is powered by gpt-3.5 with help of Zapier.



Data-flow Diagram

4. Dependencies & Libraries:
    - All project dependencies and there motive of usage can be found here.
    - External APIs & Integrations:

      > Zapier Interfaces for integrating EchoBot(chatbot).

        - Due to lack of resources to purchase OpenAI API key, had to use external service like Zapier.
        - Therefore could not embed the chatbot in the website itself.

      > ZegoCloud Video Call SDK for integrating EchoMeet.

        - Feature packed SDK for video calls providing easy integration and maintenance.

      > GetStream Chat SDK for integrating realtime chatting and messaging features.

        - Feature packed SDK for chat services providing easy integration and maintenance.
        - Scalable and effective.
        - GetStream vs MongoDB

      > Web-Push for integrating real-time notifications.

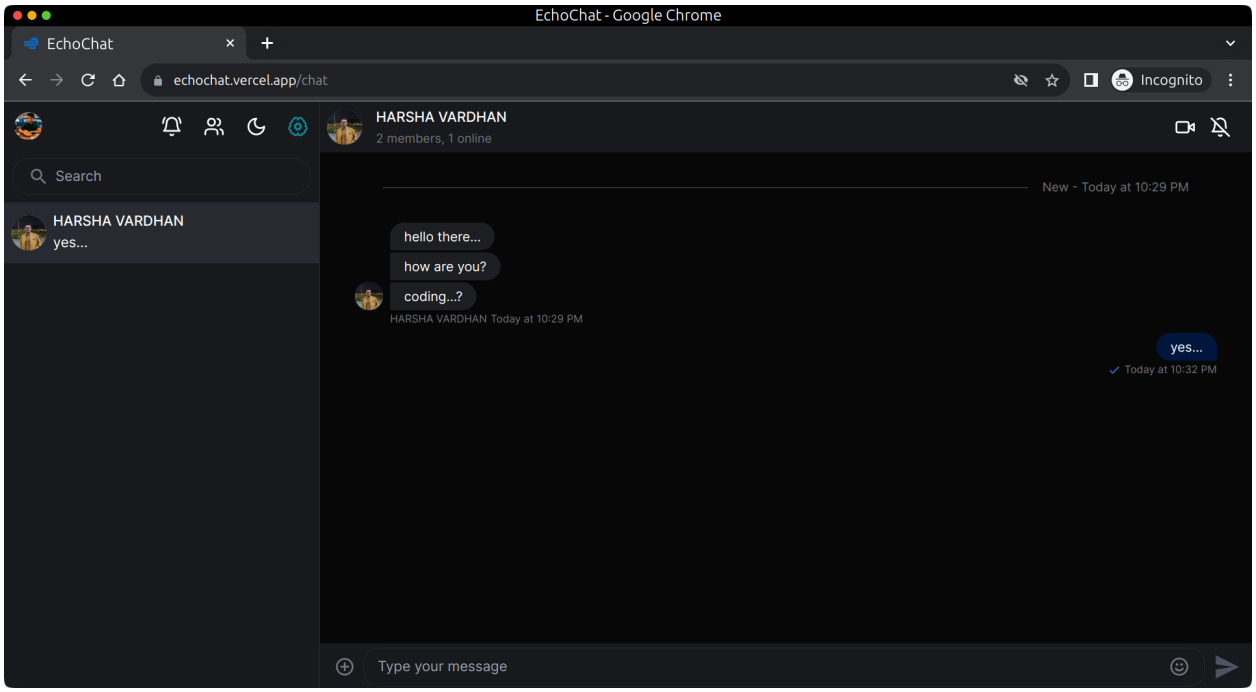
      > Clerk Auth for integrating User Authentication.

        - Provides fast and easily integrable user authentication services.

5. Data Models:
    - Clerk Auth: > user.id, user.name, user.image
    - GetStream:

      > users, channel type(messaging), channel id, channel members, messages, last_message.

6. User Interface Design:
    - EchoChat follows a minimalist and intuitive design, prioritizing ease of use.



7. Security Measures:
    - Users are authenticated using Clerk tokens.
    - Messages are end-to-end encrypted rooted in GetStream.
    - More about GetStream encryption can be found here.

8. Deployment:

Deployment is done using Vercel.

Necessary Environment Variables need to be provided at the time of deployment:

- NEXT_PUBLIC_CLERK_PUBLISHABLE_KEY
- CLERK_SECRET_KEY
- CLERK_WEBHOOK_SECRET
- NEXT_PUBLIC_STREAM_KEY
- STREAM_SECRET
- NEXT_PUBLIC_WEB_PUSH_PUBLIC_KEY
- WEB_PUSH_PRIVATE_KEY

9. Documentation:

Complete project documentation can be found here.

- Clerk Docs
- GetStream Chat Docs
- ZegoCloud Docs

10. Future Enhancements:

Add more real life features like -

- Status
- Stories
- Adding more AI features.
- Embedding chatbot inside the website.
- Adding 1-1 and Group Audio/Video Call.

CONCLUSION

EchoChat is designed to provide secure and real-time communication features to users.

Special thanks to the recruitment team at I'mbesideyou for assigning me with this project, which allowed me to delve into various aspects of creating chat platforms and gain valuable hands-on experience.

Project Link: EchoChat

End of Document