

22AIE204 INTRODUCTION TO COMPUTER NETWORKS

“ CHAT APPLICATION [WE_CHAT] ”

A report submitted in partial fulfillment of the requirements for 3rd semester

Bachelor in Technology

In

Artificial Intelligence

By

Team -D10

VADA GOURI HANSIKA REDDY	CB.SC.U4AIE23304
MALAVIKA S PRASAD	CB.SC.U4AIE23315
KATIKALA DEDEEPPYA	CB.SC.U4AIE23349
GESHNA B	CB.SC.U4AIE23360

Under the guidance of

Mr. Jaisooraj



Department of Artificial Intelligence

Amrita Vishwa Vidyapeetham

Coimbatore

INTRODUCTION

We Chat is an innovative, versatile application that is designed to facilitate communication between two users as well as between the user and the chatbot. Be it We Chat in the 'social' or 'business' mode, the app deploys a strong service where one can expect continuous messaging even as one creates sophisticated texts, powered by Google Deep Mind's Gemini AI. With the aid of the Gemini AI technology integrated into WeChat, users are able to communicate in a more engaging way by generating humanlike and contextually appropriate responses as conversations progress. Where other AIs may focus on how something looks, Gemini is all about how it reads, and writes, specifically drawing one's attention to language.

WORKING METHODS

- **User Authentication**
 - **Sign-up/Sign-in:** Firebase Authentication securely verifies user login and signup credentials.
 - **Token Generation:** Each logged-in user receives a unique access token for controlled access.
 - **Session Management:** Authenticated sessions allow prolonged access without re-login until logout.
- **Real-Time Messaging**
 - **Message Sync:** Firebase Firestore enables real-time file storage and retrieval.
 - **Data Flow:** Messages are stored and instantly displayed to all active users.
 - **Data Mode:** Messages are stored with timestamps and sender info for organized retrieval.
- **Push Notifications**
 - **FCM Setup:** Firebase Cloud Messaging (FCM) sends notifications for new messages.
 - **Notification Triggers:** Notifications alert recipients of new messages.
 - **Display & Response:** Notifications are visible even when minimized, enabling quick replies.
- **Database Management**
 - **Structured Data and Security:** Firebase Firestore organizes data in collections (e.g., "Users" and "Messages") with custom security rules to protect user privacy and enable role-based access.
 - **Real-Time Sync and Offline Access:** Data synchronizes automatically across devices and supports offline access, ensuring smooth user experience even with unstable networks.
 - **Scalability:** Firestore scales effortlessly with user demand, handling increasing data and user requests without performance drops.
- **Deployment and Optimization**

- **Efficient APK Packaging:** Android Studio creates an optimized APK, making app installation and distribution seamless across Android devices.
- **Performance Enhancements:** Implements caching, image compression, and efficient API calls to reduce load times, data usage, and conserve battery life.
- **Comprehensive Testing:** Rigorous cross-device testing, debugging tools, and crash analytics ensure stability and compatibility across various Android versions and devices.

SOFTWARE TOOLS USED

- **Android Studio**

Purpose: Comes with emulators, connects to Firebase, and offers great bug-fixing assistance.

- **Firebase Authentication**

Purpose: Provides various types of authentications (email, googling/signing up, etc.) and can generate unique tokens for the ongoing session controls.

- **Firebase Firestore**

Purpose: It processes data synchronously without users noticing the difference and even allows users to store data in various forms and moreover outside or without the use of the internet.

- **Firebase Cloud Messaging (FCM)**

Purpose: It ensures that different devices carry the messages and that sent messages are carried out using Firestore.

- **Kotlin (or Java)**

Purpose: Kotlin and Java serve as core languages in We Chat's development, enabling efficient, high-performance coding for Android compatibility and seamless backend functionality across devices.

- **Emulators and physical devices**

Purpose: The devices are used to test the design, functionality and scalability of the application on different screen sizes and various android versions.