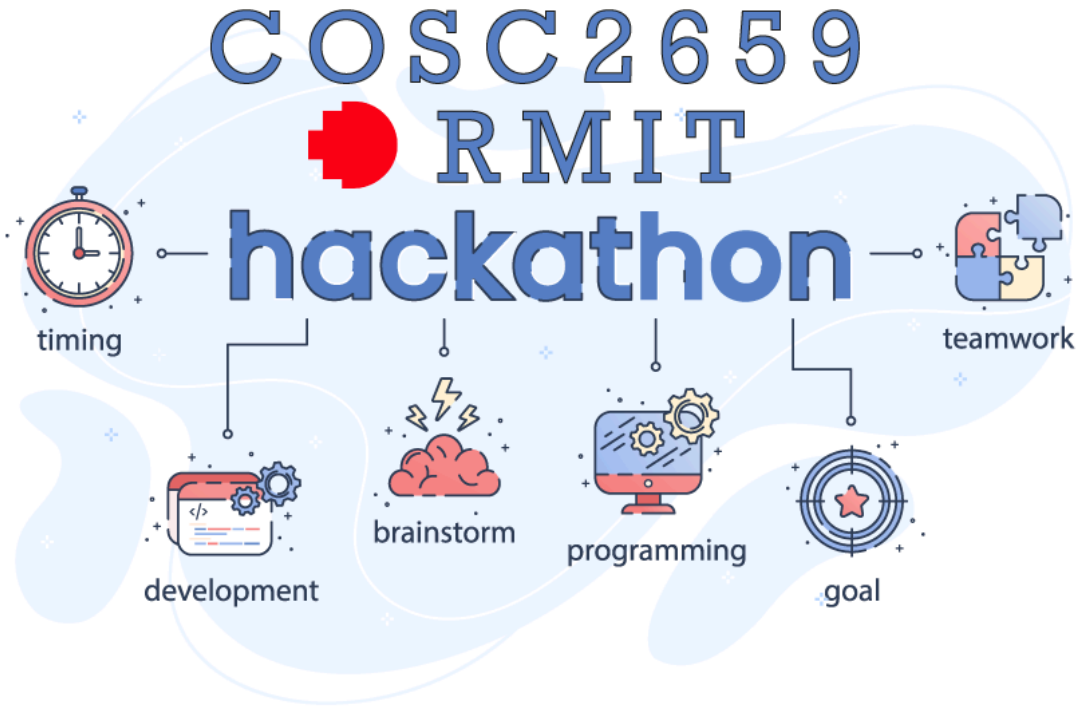
RMIT UNIVERSITY

TEAM REPORT



Course – Course code	iOS Development - COSC 2659|COSC 2813
Title of Assignment	Assignment 3
Student name - Student ID	Nguyen Huy Anh - s3956092 Tran Xuan Hoang Dat - s3651550 Nguyen The Anh - s3927195 Tran Vinh Trong - s3863973
Lecturer Name	Tom Huynh
Group Number	SG-Group19 Project Groups
Topic	Mental Health
Date of Submission	23.09.2024

RMIT UNIVERSITY

Table of Contents

I. Introduction

For this assignment, our team has chosen to develop FriendlyBot, an AI-powered chatbot that allows its users to have conversations with the application, sharing their thoughts and feelings with a friendly partner. By using this application, FriendlyBot's users will be able to chat with the AI about anything they want, including asking for advice, talking about a recent experience, or sometimes simply venting about a problem they are having. The slogan of the application is "Want someone to listen to you? FriendlyBot is here."

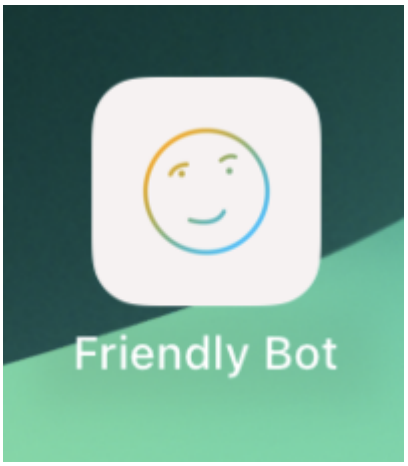


Figure 1: App icon and name

II. Inspiration

1. Relevance to Vietnam context

   Mental health issues are a growing concern in Vietnam, where the stigma surrounding mental illness often prevents people from seeking help. Cultural norms in Vietnam tend to view mental health problems as a sign of personal weakness, which discourages individuals from discussing their emotions openly. Additionally, access to mental health services is limited, particularly in rural areas. According to a report by the World Health Organization, the prevalence of common mental disorders in Vietnam was around 14.2% in 2014, with depressive disorders affecting about 2.45% of the population. Suicide rates were also significant, with 5.87 per 100,000 people in 2015,

highlighting the urgent need for better mental health resources [1].

To address these issues, FriendlyBot serves as a friend whom users will be able to open up to, without the fear of being judged or ridiculed for their emotions, as the application provides an AI-based partner who is willing to listen to any thoughts that users want to share in a friendly and supportive manner. By utilizing the rapidly developing technology of AI, FriendlyBot aims to contribute to improving the overall mental well-being of Vietnamese users, offering a friendly companion and also an emotion-tracking system.

2. App idea

Unlike other existing chatbots, FriendlyBot uses sentiment analysis to detect and evaluate its user's mood, therefore tailoring its responses to better fit the current tone of the conversation, while also logging and tracking the user's emotional state over time. It also provides its services in both English and Vietnamese, accommodating Vietnamese users who are not fluent in English, making the app more culturally relevant.

III. Competitors

- Competitors of FriendlyBot

Several mental health apps already exist that provide chatbot-based therapy. Two notable competitors are:

- **Woebot**: A mental health chatbot that delivers cognitive behavioral therapy (CBT) techniques. It helps users manage stress, anxiety, and depression by providing interactive conversations [3].
- **Replika**: An AI chatbot designed for emotional support, and mental health reflection. It enables users to have customized chats and provides a variety of interactions based on the user's mood and emotional condition. [4].

- FriendlyBot's "killer feature"

What makes FriendlyBot stands out from other chatbot apps is its **real-time mood detection system**, which evaluates the user's emotional state by analyzing the tone and wording of their messages. This feature sets the

application apart from its competitors like Woebot or Replika, by helping its users observe their emotional trends. Additionally, the application's Vietnamese language support furthermore expands its target user base which are Vietnamese users.

IV. Features

FriendlyBot offers a variety of different features for its users to utilize while using the application.

- Splash screen:

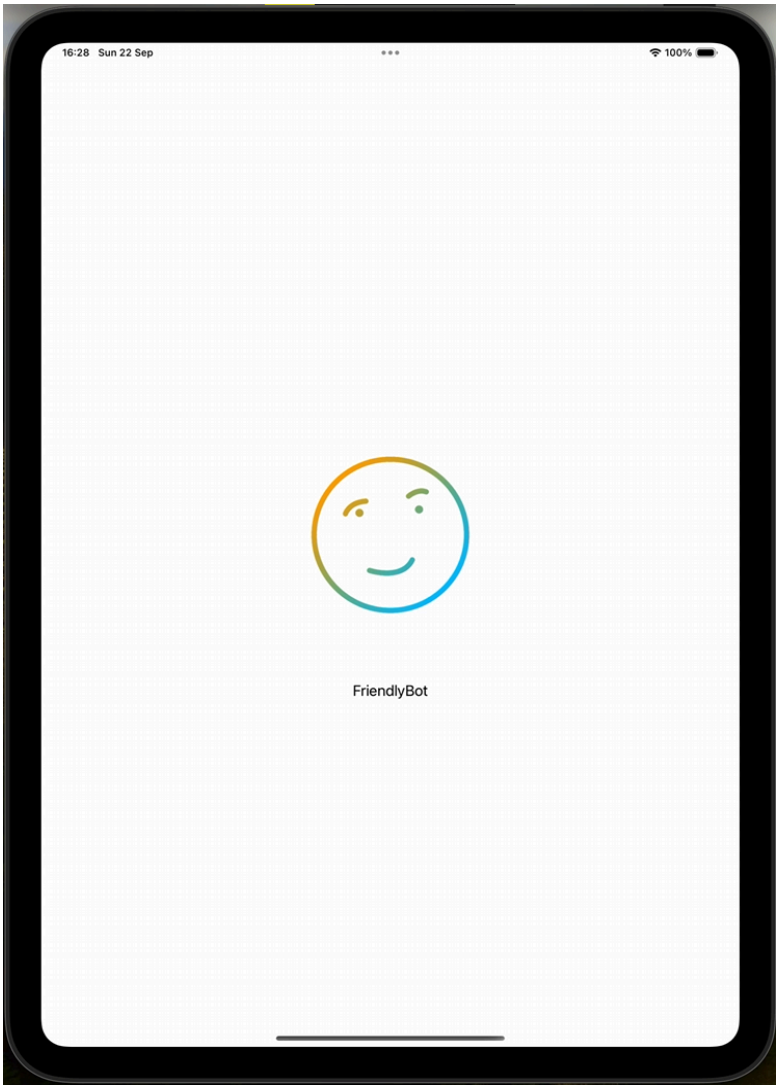


Figure 2: Startup splash screen

When users first opens the application, they will be greeted with a splash screen as it boots, which features the logo of the application and its name. After that, they will be able to use the application's other features from now on.

- Sign in:

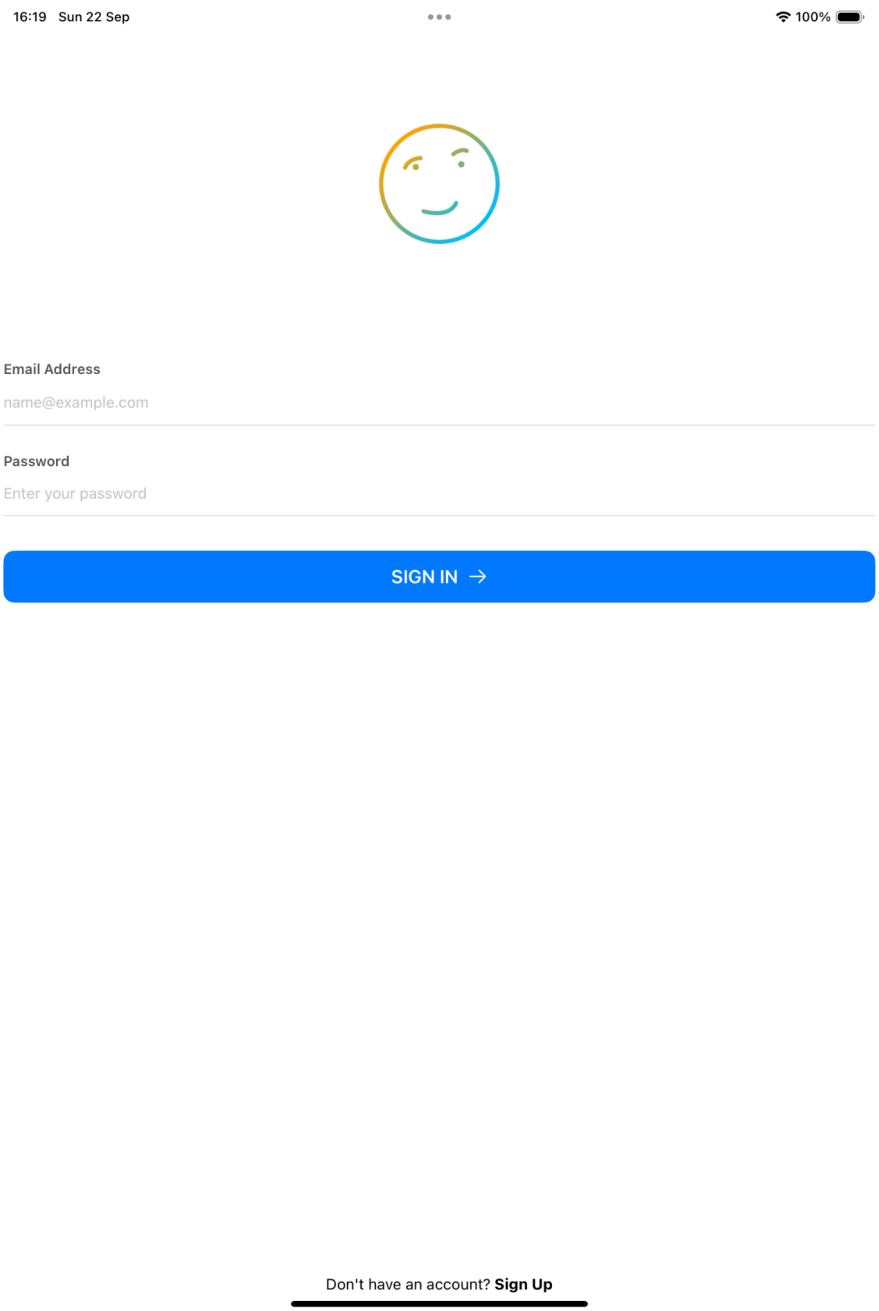


Figure 3: Sign in screen

After opening the application, if the user is not already signed-in from their last session, the application will show the user the sign in screen, where they will be able to enter their account credentials to sign in to the system in order to use the application's main features. Alternatively, if the user does not have an existing account, there is a prompt asking them "Don't have an account?" in the bottom of the screen, which when selected will redirect them to another view, which is the sign up screen.

- Sign up:

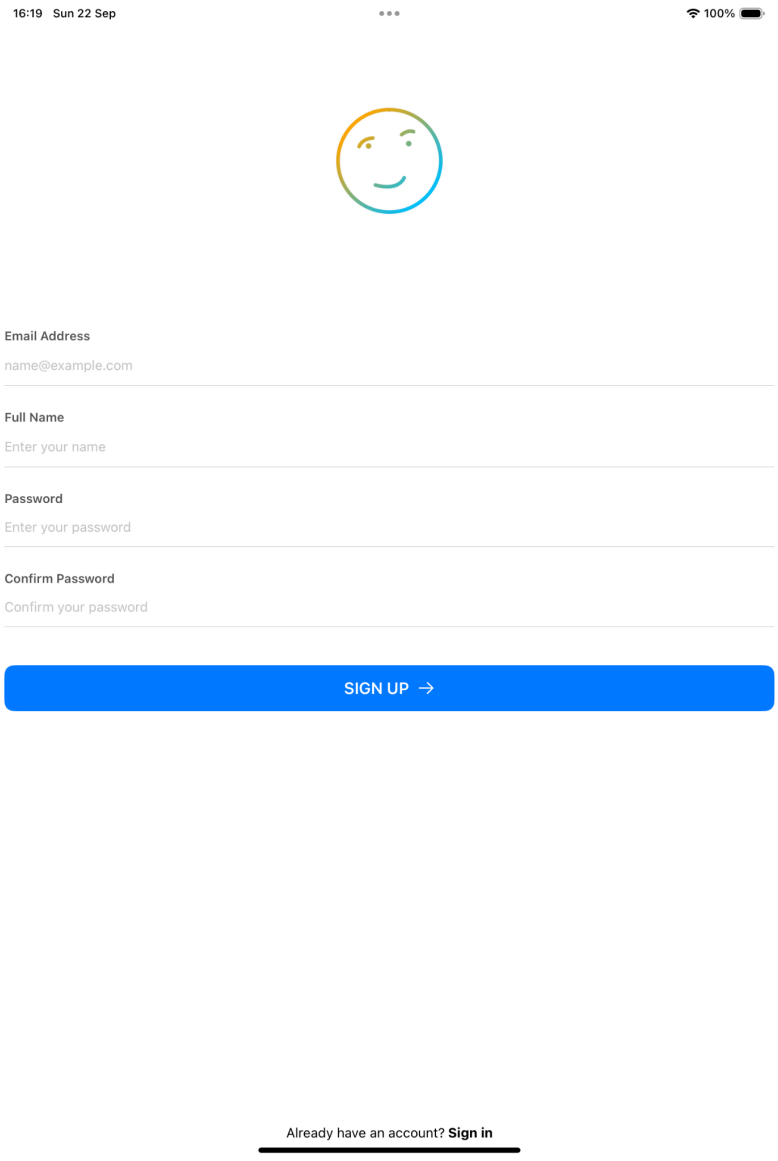


Figure 4: Sign up screen

For users without an existing account, they will be able to register for a new account in the system using this feature. In this screen, they will be asked to input their email address, name, as well as a password to create a new account. After selecting "SIGN UP", they will then be automatically logged-in and redirected to the profile view.

- Profile view:

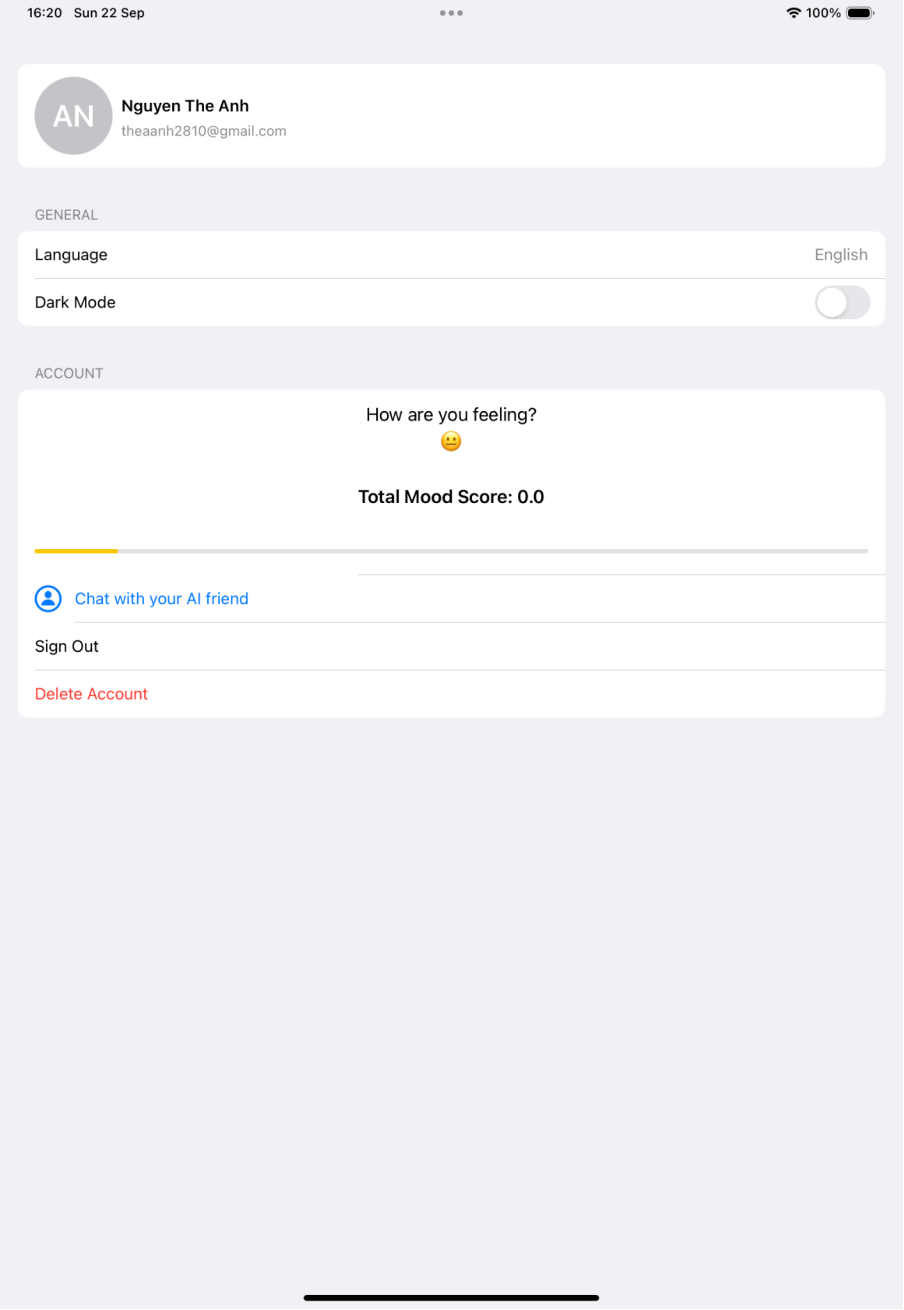


Figure 5: Profile view

After logging in, the user will now be able to see their profile, as well as customizing the application's settings and using the chatbot itself. This view provides the user various options to use, including the language switch, dark mode toggle, mood tracker, chat view button, sign out button, and delete account button:

- Language switch:

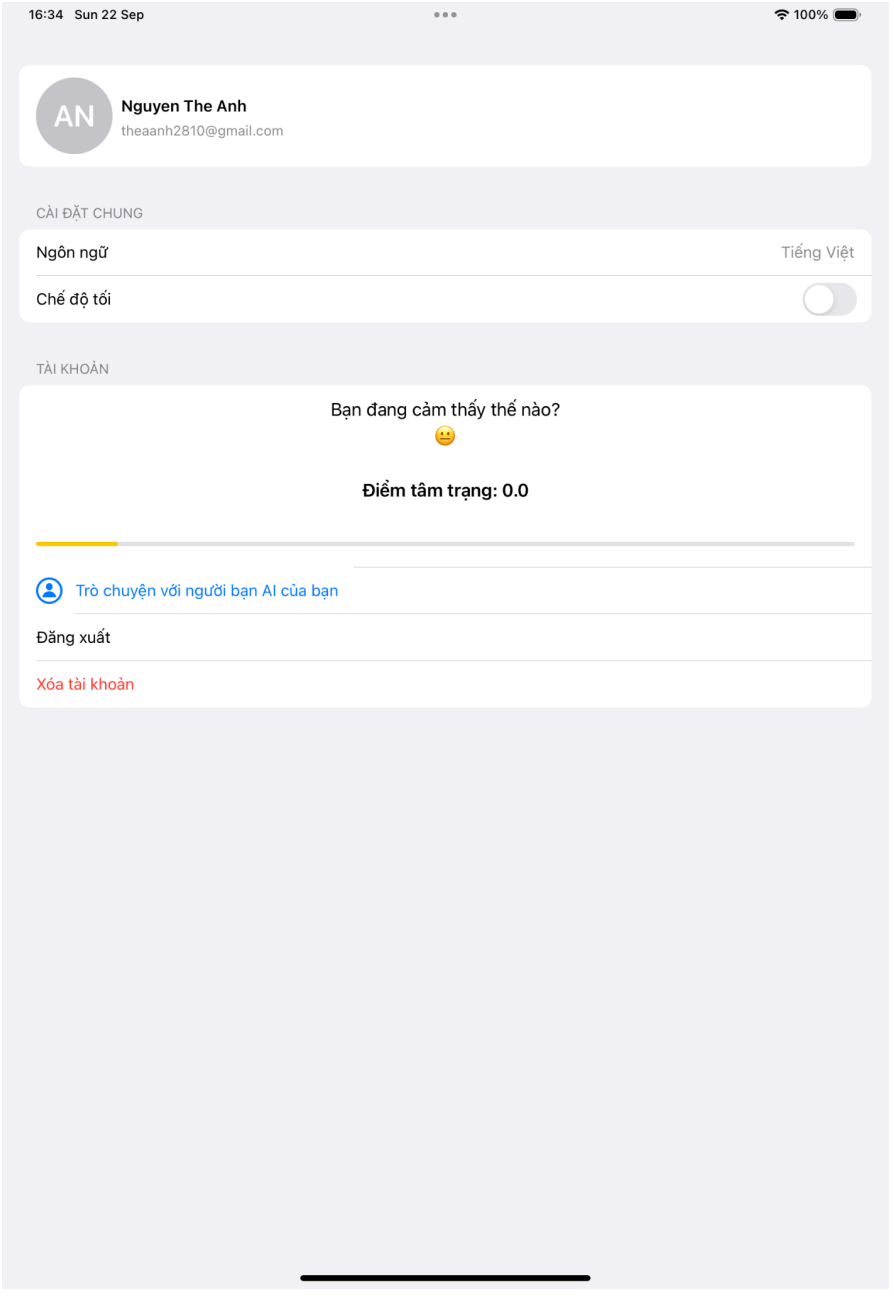


Figure 6: The application in Vietnamese mode

As mentioned, FriendlyBot features a bilingual interface, meaning that users will be able to select between an English interface and a Vietnamese interface. They will be able to do so by selecting the "Language" button, which instantly changes the whole application's language. This setting option also affects the sign in and sign up features, giving them an UI in Vietnamese as well.

- Dark mode:

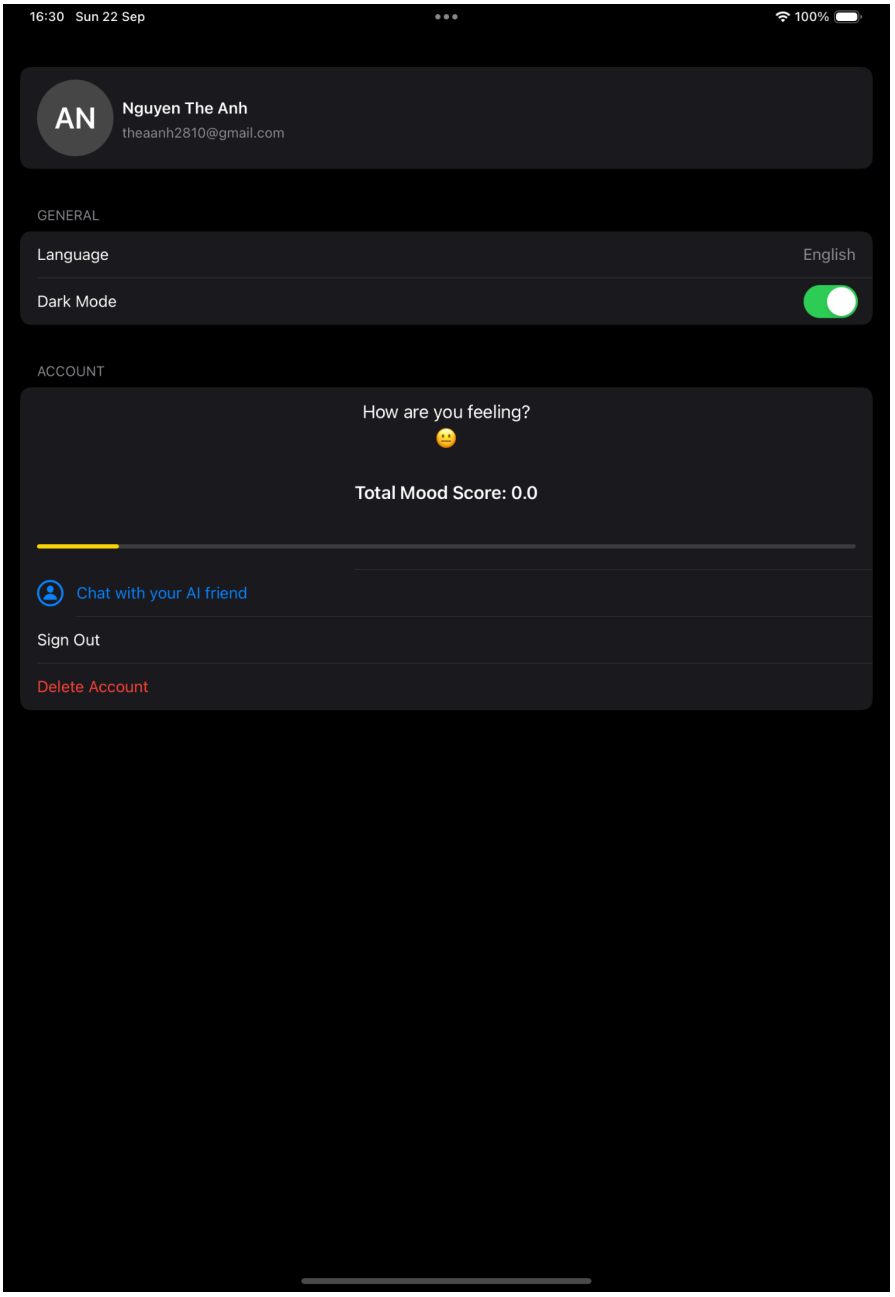
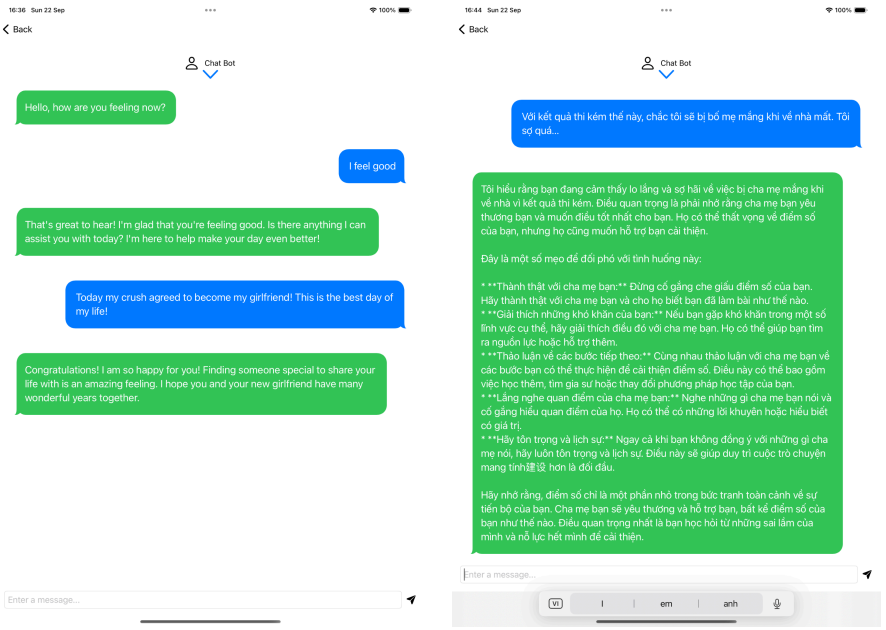


Figure 7: The application in dark mode

By default, the application's theme is set to light mode. However, if preferred, users can switch the theme of the application to dark mode by using the dark mode toggle below the language switch option.

- Chat view:



Figures 8, 9: Chat view in English and Vietnamese

The main feature of the application is the chat view, where the user can have conversations with the chatbot. To access this feature, the user should select the "Chat with your AI friend button", where they will be redirected into the chat view. Here, users will be able to interact with the chatbot as if they are texting with another person using messages. As mentioned, the app offers its features in both English and Vietnamese, meaning that the user can communicate with the chatbot in both languages.

- Mood tracker:

  As the user interacts with the chatbot, it will actively analyze the user's choice of words to determine if they are feeling happy, neutral, or sad. This will in

turn affect the mood score of the user, which will then be displayed outside the profile view for the user to see.

- Sign out:

  If the user wishes to sign out of their account, they will be able to do so by selecting “Sign out”, which in turn will redirect them back to the Sign in screen.

- Delete account:

  In case the user wants to permanently delete their account, below the “Sign out” option is the “Delete account” option, where they will be able to easily delete their FriendlyBot account by simply selecting this button.

V. Data Persistence

For the development of FriendlyBot, we have chosen to use **Firebase** and **UserDefaults** for data persistence to ensure a seamless user experience and efficient management of both user data and preferences.

- **Firebase**

  Firebase plays a crucial role in FriendlyBot’s backend for the following reasons:

  - **Real-Time Data Synchronization**: Firebase Firestore provides real-time syncing of chat conversations and mood tracking across devices. This is essential for FriendlyBot, where users need to access up-to-date chat history and mood analytics, regardless of the device they use [5].
  - **User Authentication**: Firebase Authentication simplifies user management by securely handling sign-ups, logins, and session maintenance. This service ensures that users can log in from any device and access their personalized profiles, chat data, and mood records seamlessly.
  - **Scalability**: Firebase is a cloud-based solution that scales automatically with the user base. It can handle a large number of concurrent users without the need for complex server-side management, making it an ideal choice for future growth [5].

- **UserDefaults**

  UserDefaults is utilized for managing simple, local data such as user preferences. Here's why it was chosen:

  - **Local Data Storage**: UserDefaults is a lightweight, fast solution for storing user settings like dark mode preferences and language selections. These settings don't need to be synced across devices, making UserDefaults a perfect fit for this use case [6].
  - **Efficiency**: UserDefaults offers fast read and write operations for small pieces of data, ensuring a smooth user experience without the overhead of cloud storage or syncing. This efficiency is essential for UI-related settings that need to be accessed instantly.

By combining **Firebase** for data synchronization and **UserDefaults** for local settings storage, FriendlyBot is optimized for both real-time data handling and swift preference management. This architecture supports the app's core functionalities and enhances the overall user experience.

VI. Application Flow

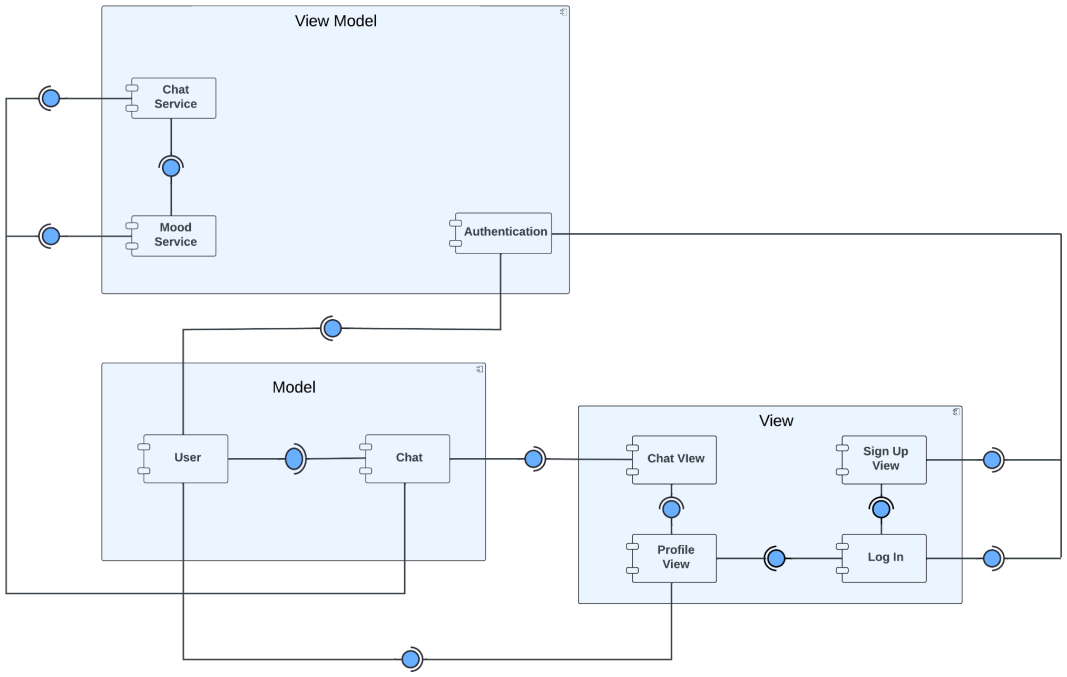


Figure 10: Diagram representing the application's flow

When first launching the app, the users will be met with a splash screen then encounter a login view. In the login view, there is also an option to navigate to the sign up view. Whether the user either logs into an old account or signs up a new account, their inputs will be transferred to AuthModel for verification. If there is a failure in the verifying process like login with the wrong account/password, or the confirmed password does not match to the initial one, the system will refuse to let the user proceed with the progress. If a user succeeds one of those checks, the user will be navigated to the profile view which has 2 main features of group 19 are the chat bot and mood tracking score displaying on this view. Since mood relies on the user ID so that each user has a different chat history and mood score, mood does not need a model to distinguish. Moreover, a mood service is needed and its purpose is to detect certain user input words so that the service can fetch numbers which will be displayed in sum of the amount within the range it is given in the program. For the chat, each chat needs to be distinguished as well as reviewable for each user hence that is the purpose for chat Service.

VII. Technologies

1. Swift and SwiftUI Components

- **Swift 5.8**: Used for the core logic and integration with iOS APIs.
- **SwiftUI Framework**: The declarative UI framework used to create a responsive user interface. Key components include:
    - **Text**: Displays user messages and system responses.
    - **TextField**: Allows users to input their messages.
    - **Button**: Triggers sending messages and other user actions.
    - **List**: Displays the chat history in a scrollable list.
    - **ScrollView**: Handles scrolling behavior, such as auto-scrolling to the latest message.
    - **State and Binding**: Manages the app's reactive state, ensuring that the UI updates based on changes in the mood analysis.
    - **@ObservedObject / @EnvironmentObject**: Enables data sharing and state management across multiple views.

- **ProgressView**: Shows ongoing network operations, such as sending messages or retrieving data.
- **Alert**: Provides feedback for critical issues, such as failed API calls.

2. CRUD Feature with Mood Analysis for the Chatbot

- **Create**:
  - Users send messages to the chatbot, and these messages are stored in **Firebase Firestore**.
  - The chatbot analyzes the message content and assigns a "mood score" based on the sentiment of the words. For example, positive words like "happy" add +1, while negative words like "sad" subtract -1.
- **Read**:
  - Chat history is retrieved from Firebase and displayed in chronological order.
  - The app automatically scrolls to the bottom when a new message is received, ensuring a seamless user experience.
- **Update**:
  - When users edit a message, the chatbot updates the message in the Firebase database.
  - The mood score is recalculated based on the updated content, reflecting the new sentiment.
- **Delete**:
  - Users can delete previously sent messages, and the chatbot removes them from the Firebase database.
  - The mood score is recalculated, if necessary, based on the deletion.
- **Additional Features**:
  - **Auto-scroll**: Automatically scrolls to the latest message when a new message is added to the chat.
  - **Pin message to the top**: Users can pin important messages for easy reference and management within the chat.

3. External Libraries

- **Firebase**:
  - **Firebase Firestore**: A NoSQL cloud database used to store chat messages, user data, and mood scores.

  - **Firebase Auth**: Manages user authentication for secure access to the app.
  - **Firebase Analytics**: Tracks user interaction and behavior for performance insights.
- **Google Gemini AI**:
  - Powers the chatbot's natural language processing and sentiment analysis to detect mood from user messages.
- **Alamofire**: Used to manage network requests, including those for external API calls to Google Gemini AI and Firebase.
- **Lottie for iOS**: Handles the display of animations within the chatbot, improving the user interface with interactive feedback.

4. Other Tools

- **Xcode 15.4**: The development environment used for coding, testing, and debugging.
- **Stripe API**: Enables secure payment processing for any future monetization features or donations.

VIII. Design Elements and User Experience

FriendlyBot's design follows a minimalistic approach, using a simple color palette of white/gray for light mode and black/gray for dark mode. This clean, understated design ensures that the focus remains on the user's interactions with the chatbot rather than being distracted by overly complex visuals. The choice of using the default text font across the entire UI further supports this minimalism, providing a consistent and easy-to-read experience. By opting for simplicity in both color and typography, the app creates a calm, distraction-free environment, which aligns with its goal of offering mental health support in a straightforward and accessible way.

Overall, FriendlyBot's user interface is easy to use for the average user, as its functions are straightforward and prominently placed, making them hard to go unnoticed by users. Furthermore, the inclusion of the Vietnamese interface helps accommodate Vietnamese users who may not know or be fluent in English, serving a wide demographic range of users.

IX. Conclusion

Throughout this project, our team gained valuable experience in both technical development and collaboration. We enhance our skills in SwiftUI, Firebase

integration, and chatbot functionality, particularly with the CRUD feature and mood analysis. Dividing tasks helped us improve time management, and we learned to handle real-time data, message persistence, and AI integration. Additionally, we strengthened soft skills such as communication, problem-solving, and adaptability, preparing us for future app development challenges.

X. Project Responsibilities

Name	Responsibilities
Nguyen Huy Anh - s3956092	Delete account, chatbot, reset mood score, set realtime moodscore and report
Tran Xuan Hoang Dat - s3651550	Firebase setup, sign in /up, delete account, chatbot,readme,report and slide.
Nguyen The Anh - s3927195	Front-end elements, startup splash screen, app logo and icon, app colors, dark mode, Vietnamese language, iPad responsive design, report and slide
Tran Vinh Trong - s3863973	Algorithm for chatbot, mood tracking, algorithm for mood tracking, delete chat history, report, diagram, slide.

XI. References

[1] World Health Organization (2018) 'Mental health in Viet Nam', *WHO Regional Office for the Western Pacific*, available at: https://www.who.int/vietnam/health-topics/mental-health (Accessed: 22 September 2024).

[2] Do, M., Pham, N.N.K., Wallick, S., & Nastasi, B.K. (2014) 'Perceptions of mental illness and related stigma among Vietnamese populations: Findings from a mixed method study', *Journal of Immigrant and Minority Health*, 16(6), pp. 1294-1298, doi: 10.1007/s10903-014-0018-7.

[3] Fitzpatrick, K.K., Darcy, A., & Vierhile, M. (2017) 'Delivering cognitive behavior therapy to young adults with symptoms of depression and anxiety using a fully automated conversational agent (Woebot)', *JMIR Mental Health*, 4(2), e19, doi: 10.2196/mental.7785.

[4] Kuyda, E. (2020) 'Replika: AI companion to improve mental health', *Replika*, available at: https://replika.ai/ (Accessed: 22 September 2024).

[5] Google Firebase (2024) 'Cloud Firestore Documentation', *Firebase Documentation*, available at: https://firebase.google.com/docs/firestore (Accessed: 22 September 2024).

[6] Apple Developer (2024) 'UserDefaults Documentation', *Apple Developer*, available at: https://developer.apple.com/documentation/foundation/userdefaults (Accessed: 22 September 2024).

[7] **T. Videnov,** "How I Became a Software Engineer Without a Computer Science Degree," *YouTube,* 2023. [Online]. Available: https://www.youtube.com/watch?v=GewyZgQbsIw. [Accessed: Sep. 23, 2024].