Team Members:

Aya Ehab Salama Rizk Hamada

Donia Ahmed Mahmoud

Eman Arafa Abdel Hamid

Salma Mohamed Tarek

Shrouk Hassan Saad Sayed

Nermeen Mohamed Hassanein

Nour Ibrahim Ziden Shemis

Supervisor:

Dr. Ahmed Mustafa

Faculty of Science
Computer Science Department
Alexandria University

Contents

1. Introduction

This project aims to develop an intelligent application capable of translating sign language into text or speech and vice versa. The app leverages artificial intelligence and deep learning techniques to recognize hand signs and convert them into text. Additionally, it uses a 3D character to display sign language when text or speech is input.

1.1. Motivation

Communication between individuals who use sign language and the rest of society is a significant challenge. Many deaf and hard-of-hearing individuals struggle to interact with others due to the limited knowledge of sign language among the general public. This project aims to provide an innovative solution to bridge this linguistic gap and enhance daily interactions for people with hearing impairments.

1.2. Problem Statement

Despite technological advancements, there are still considerable difficulties in communication between sign language users and others. Most available applications either translate text into sign language or provide sign language dictionaries, but they lack an interactive and seamless experience that enables users to perform real-time translation both ways. Thus, there is a need for a comprehensive application that integrates bidirectional translation with AI and 3D animations.

1.3. Goals of the Project

The project aims to achieve the following objectives:

- Develop an app that recognizes sign language and converts it into text or speech.
- Support both Arabic and English, with letters and numbers for each.
- Provide features for copying, listening to, and sharing translated text.
- Enable users to share text or audio within or outside the app .
- Include a simple chat feature for users to exchange messages.
- Support video calling between users.
- Offer a specialized keyboard with sign language images for letters, along with their names and some common signs in GIF format.
- Translate text or speech into sign language using a 3D character(called Moushira).
- Allow users to scan images to extract text and convert it into sign language movements.
- Enable sharing sign animations through a link, where the recipient can view them on a web page with the same theme as the app.

1.4. Currently Available Solutions

Several applications aim to facilitate communication for the deaf and hard-of-hearing community by providing instant sign language translation or converting text and speech into sign language. Below are some of the prominent applications in this field, along with their strengths and limitations:

- **Signly (UK)**: A browser extension that translates text into sign language videos but has limited coverage.
- **HandTalk (Brazil)**: Translates text into sign language using a virtual avatar but lacks gesture recognition and OCR capabilities.
- **Sign Language Translator (India)**: Converts text and speech into sign language but is limited to specific languages and signs.
- **ProDeaf (Brazil)**: Translates text into sign language using a virtual avatar but lacks comprehensive sign support.
- **GSL Translator (Greece)**: Focuses on translating text into Greek Sign Language, making it geographically limited.
- **Tarjuman (Jordan)**: An Arabic sign language translator that converts text and speech into sign language using 3D avatars but lacks gesture recognition from the camera.
- **Tawasal (UAE)**: Supports multilingual translation but does not fully focus on sign language and translate it into text or speech.

Compared to these solutions, our application offers a more comprehensive set of features, integrating real-time sign recognition via camera, text and speech translation into sign language, link sharing, chat and video calls, and OCR capabilities.

1.5. Features Matrix

The proposed application is compared with other existing solutions as shown in **Table 1.1**

Feature	Signly	HandTalk	Sign Language Translator	ProDeaf	GSL Translator	Tarjuman	Tawasal	Gesture Vox(our project)
Converts Text to Sign Language	✓	✓	✓	✓	✓	✓	✓	✓✓
Converts Speech to Sign Language	×	✓	✓	✓	×	✓	×	✓✓
OCR (Scans Images for Text)	×	×	×	×	×	×	×	✓
Chat and Video Call Support	×	×	×	×	×	×	×	✓
Gesture Recognition (Sign to Text)	×	×	×	×	×	×	×	✓
Gesture Recognition (Sign to Voice)	×	×	×	×	×	×	×	✓
3D Character for Sign Language	×	✓	×	✓	×	✓	×	✓✓
Share Sign Animations	×	✓	×	×	×	×	×	✓✓

Table 1.1 :Feature Comparison Matrix

1.6. Software Development Methodology Used

Given the nature of the project, which integrates multiple technologies such as gesture recognition via camera, text processing, and speech-to-sign translation using a 3D character, we have chosen the **Agile** methodology with the **Scrum** framework for development.

Why did we adopt this methodology?

1. **Flexibility and adaptability to changes**
   - Since our project involves AI and image processing, it requires continuous updates and modifications based on testing results and user feedback.
   - Agile allows us to adjust project requirements easily based on new findings during development.
2. **Dividing work into small phases (Sprints)**
   - Scrum helps us break the project into short development cycles (Sprints), ensuring continuous progress and frequent improvements.
   - This enables us to develop separate features such as sign recognition, chat, and multilingual support in parallel.
3. **Collaboration between different teams**
   - The project involves multiple teams (AI, front-end, back-end, and 3D character design), requiring regular meetings and updates to ensure smooth integration.
4. **Releasing early versions (MVP) and incremental improvements**
   - Agile allows us to launch early versions of the app, test features with real users, and gather feedback to refine the final product before the official release.

   Compared to other methodologies like Waterfall, Agile is more flexible and allows continuous adaptation, whereas Waterfall is better suited for projects with fixed and well-defined requirements, which does not apply to our project due to the evolving nature of AI and gesture recognition technologies.

2. Requirements Analysis

In this section, we analyze user requirements to ensure that the application meets their actual needs. To achieve this, we collected data through a survey targeting potential users, analyzed the results using graphical representations, and discussed our findings.

2.1. Survey

To gather user requirements, we designed a survey using Google Forms, targeting various user groups, including individuals interested in AI-based translation solutions.

Target Audience:

- Deaf and hard-of-hearing individuals
- Deaf and hard-of-hearing individuals
- People interested in AI-based translation technology

Number of Participants:

A total of 76 participants responded to the survey.

Data Collection Method:

The survey was distributed through social media platforms and university student groups.

Key Questions Asked:

1. **What is your age?**
    - Under 18 years old
    - 18-24 years old
    - 25-34 years old
    - over 34 years old
2. **What is your Gender?**
    - Female
    - Male
3. **Do you have any knowledge of sign language?**
    - Yes
    - No
4. **Do you have relatives or friends who are deaf or mute?**
    - Yes

- No

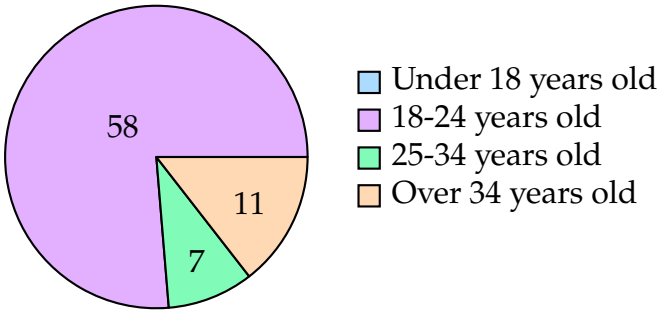
5. **Have you ever had difficulty communicating with a deaf or mute person?**
   - Yes, many times
   - Yes, but rarely
   - No
6. **How important is it to facilitate communication between people who use sign language and others?**
   - Extremely important
   - Somewhat important
   - Neutral
   - Somewhat not important
   - Extremely not important
7. **Do you think that having an application that can translate sign language into text and voice and vice versa will facilitate communication with deaf and mute people?**
   - Yes
   - No
8. **Do you think this app can be useful in public places like banks, hospitals, or service centers?**
   - Yes
   - No
9. **What feature do you think would be most useful in the app?(You can choose more than one option)**
   - Translating sign language into text
   - Translating sign language into voice
   - Translate text into sign language using a 3D character
   - Translate voice into sign language using a 3D character
10. **In your opinion, how can this application help improve the lives of people who use sign language?(you can choose more than one option)**
    - Improve communication with others
    - Improving employment and education opportunities
    - Promoting social integration
    - Reducing social isolation
    - Another
11. **Would you be interested in trying this app if it was available on your phone?**

- Yes
- No

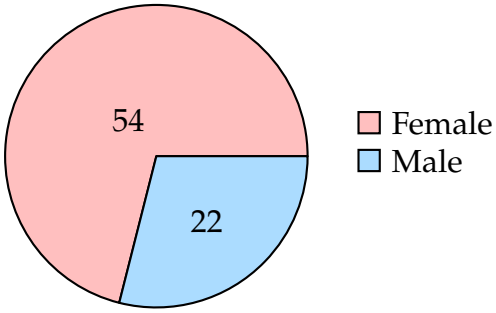
12. **Do you have any suggestions to improve the application?**

2.2. Results of Survey

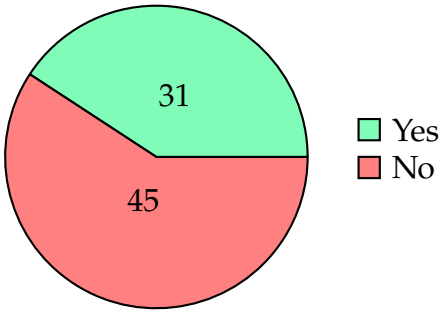
What is your age?



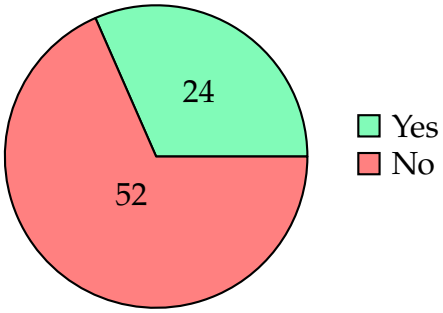
What is your Gender?



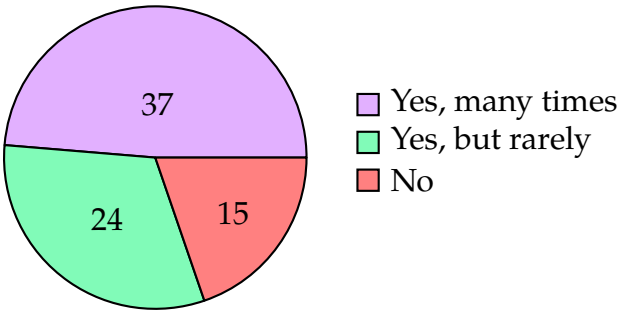
Do you have any knowledge of sign language?



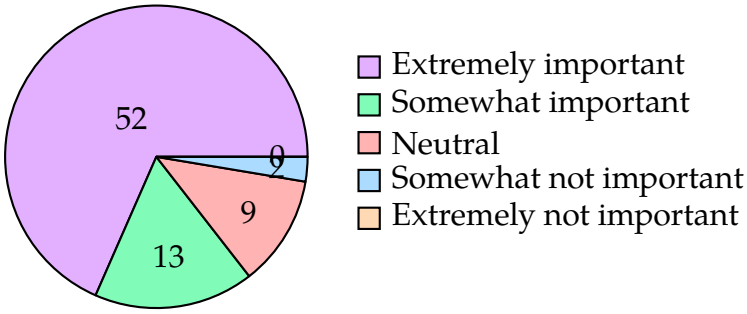
Do you have relatives or friends who are deaf or mute?



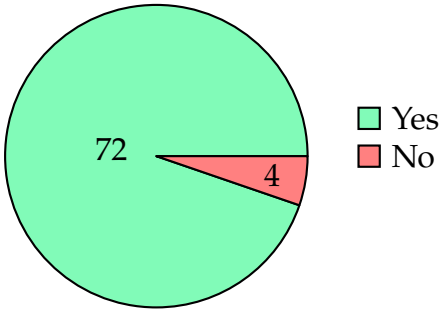
Have you ever had difficulty communicating with a deaf or mute person?



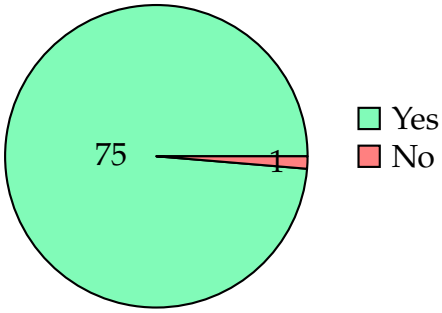
How important is it to facilitate communication between people who use sign language and others?



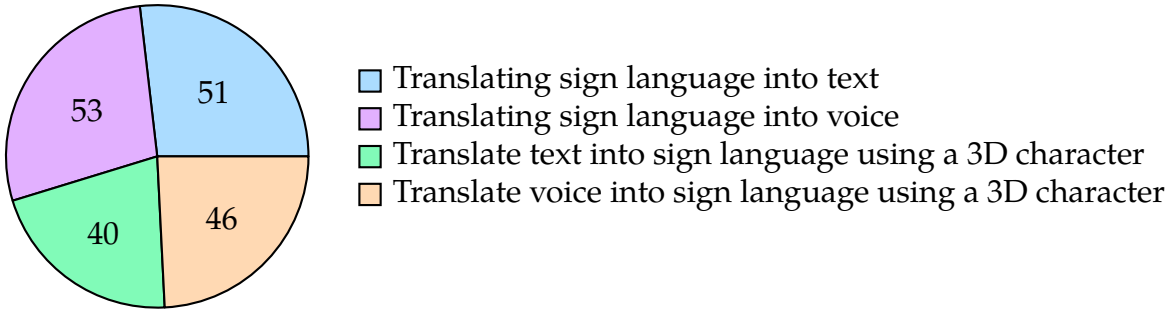
Do you think that having an application that can translate sign language into text and voice and vice versa will facilitate communication with deaf and mute people?



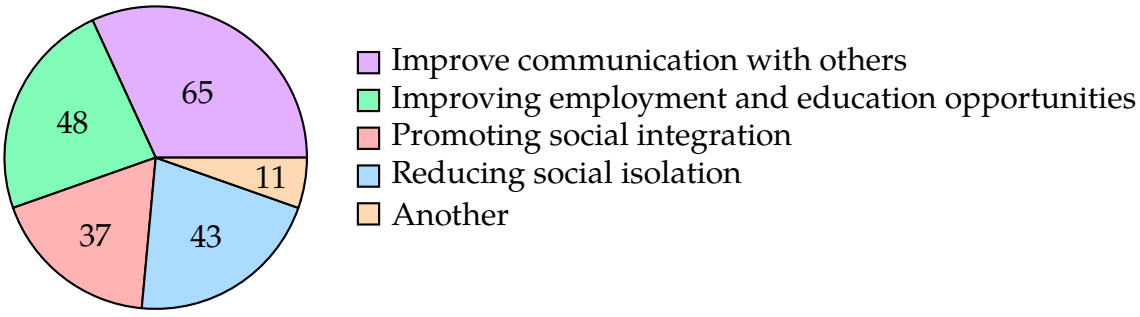
Do you think this app can be useful in public places like banks, hospitals, or service centers?



What feature do you think would be most useful in the app?(You can choose more than one option)



In your opinion, how can this application help improve the lives of people who use sign language?(you can choose more than one option)



2.3. Discussion of results in Results of Survey

The survey results showed that **61% of participants find communication difficult**, indicating a market gap that requires innovative solutions. Additionally, **99% of participants stated that the app would be suitable or useful in public places**, and a significant **93% expressed interest in trying the app upon its release**.

Based on these findings, we have decided to focus on the following features in our application:

- **Core Features (Based on Survey Results):**
    1. Accurate real-time translation using AI.
    2. Support for both text and voice output, depending on user preferences.
    3. Convert text and speech into sign language using a 3D character for an interactive translation experience.
    4. A user-friendly interface to ensure a smooth experience for all users.
- **Additional Features to Enhance User Experience:**
    1. Integrated chat feature that allows users to share translated text and send any converted content with ease.
    2. Support for video calls to facilitate real-time communication using sign language.
    3. A dedicated sign language keyboard to make typing and interaction more intuitive.
    4. A collection of GIFs for the most common signs, helping users learn and communicate faster.
    5. Scan images to extract text, which can then be translated directly using a 3D character.

2.4. Functional Requirements

1. **Burger Menu**

   As shown in the following **Table 2.1**

Req. ID	Requirement	Req. Description	Priority
F1.1	Navigate to Scan	Navigate the user to the Scan feature when clicked.	High
F1.2	Display Translation History	Display the user's previous translation history.	Medium
F1.3	Change Language	Change the app language to the user's preferred language.	Medium
F1.4	Enable Dark Mode	Enable dark mode to change the app's appearance.	Low
F1.5	Support	Redirect the user to the support page.	Low
F1.6	Logout	Log the user out of the app.	High

Table 2.1: Burger Menu

2. **Login & Signup Pages**

   As shown in the following **Table 2.2**

Req. ID	Requirement	Req. Description	Priority
F2.1	Login with Email	Allow login using email and password.	High
F2.2	Login with Google	Allow login using a Google account.	Medium
F2.3	Login with Facebook	Allow login using a Facebook account.	Medium
F2.4	Show/Hide Password	Show or hide the password when the icon is clicked.	Low
F2.5	Remember Me	Remember the user's credentials for automatic login.	Low
F2.6	Forgot Password	Send a password reset link to the user's email.	Medium
F2.7	Create New Account	Create a new account using full name, email, and password.	High

Table 2.2 : Login & Signup Pages

3. **Scan Feature**

   As shown in the following **Table 2.3**

Req. ID	Requirement	Req. Description	Priority
F3.1	Scan Images	Scan images using the camera or gallery.	High
F3.2	Extract Text	Extract text from scanned images.	High
F3.3	Redirect to Translation	Redirect the user to translate the extracted text into sign language.	Medium

Table 2.3 : Scan Feature

4. **Settings**

   As shown in the following **Table 2.4**

Req. ID	Requirement	Req. Description	Priority
F4.1	Edit Profile	Allow editing of profile picture, name, and password.	Medium
F4.2	Report Problem	Allow users to report issues they encounter.	Low
F4.3	Contact Us	Redirect users to the contact page.	Low
F4.4	Share App	Allow users to share the app with others.	Low
F4.5	Rate App	Redirect users to rate the app on the store.	Low
F4.6	Dark Mode	Enable or disable dark mode.	Medium
F4.7	Privacy Policy	Display the app's privacy policy.	Low
F4.8	Terms and Conditions	Display the app's terms and conditions.	Low
F4.9	Logout	Log the user out of the app.	High

Table 2.4 : Settings

5. **Chat Feature**

   As shown in the following **Table 2.5**

Req. ID	Requirement	Req. Description	Priority
F5.1	Search for Users	Search for users using their email.	Medium
F5.2	Send/Receive Messages	Send and receive text messages.	High
F5.3	Sign Language Keyboard	Use a sign language-specific keyboard.	Medium
F5.4	Send/Receive GIFs	Send and receive GIFs.	Low
F5.5	Video Call	Make video calls with other users.	Medium

Table 2.5 : Chat Feature

6. **Sign Translator**

   As shown in the following **Table 2.6**

Req. ID	Requirement	Req. Description	Priority
F6.1	Select Language	Select language (Arabic/English).	High
F6.2	Select Letters/Numbers	Select letters or numbers for translation.	Medium
F6.3	Use Camera	Use the camera to recognize signs.	High
F6.4	Display Text	Display the recognized text in a text field.	Medium
F6.5	Reset	Clear the text field and start over.	Low
F6.6	Send Text	Send the text to be displayed on another page.	Medium
F6.7	Copy/Listen to Text	Copy or listen to the recognized text.	Low
F6.8	Share Text/Audio	Share the text or audio within or outside the app.	Low

Table 2.6 : Sign Translator

7. **Word Translator**

As shown in the following **Table 2.7**

Req. ID	Requirement	Req. Description	Priority
F7.1	Input Text/Speech	Input text or record speech.	High
F7.2	Convert Speech to Text	Convert recorded speech to text.	High
F7.3	Display Text on 3D Character	Display the text on a 3D Character (Mushira).	High
F7.4	Translate Text to Signs	Translate the text into sign language using the 3D Character.	High
F7.5	Share Translation	Share the translation as a link.	Low

Table 2.7 : Word Translator

2.5. Non-functional Requirements

1. **Burger Menu**

   As shown in the following **Table 2.8**

Req. ID	Requirement	Req. Description	Priority
NF1.1	Easy Access	Ensure easy access to the menu from any page in the app.	High
NF1.2	Quick Response	Ensure quick navigation between menu options.	High
NF1.3	System Compatibility	Automatically enable dark mode if the system is in dark mode.	Medium

Table 2.8 : Burger Menu

2. **Login & Signup Pages**

   As shown in the following **Table 2.9**

Req. ID	Requirement	Req. Description	Priority
NF2.1	High Security	Securely store passwords using encryption.	High
NF2.2	Fast Login/Signup	Ensure quick login and account creation processes.	High
NF2.3	Simple UI	Provide a simple and clear interface for login and signup.	Medium

Table 2.9 : Login & Signup Pages

3. **Scan Feature**

   As shown in the following **Table 2.10**

Req. ID	Requirement	Req. Description	Priority
NF3.1	High Accuracy	Ensure high accuracy in text extraction from images.	High
NF3.2	Fast Scanning	Ensure fast scanning and conversion processes.	High
NF3.3	Image Compatibility	Support various image formats (JPEG, PNG, etc.).	Medium

Table 2.10 : Scan Feature

4. **Settings**

   As shown in the following **Table 2.11**

Req. ID	Requirement	Req. Description	Priority
NF4.1	Easy Access	Ensure easy access to settings from any page.	High
NF4.2	Quick Response	Ensure quick response to changes in settings.	High
NF4.3	Clear UI	Provide a clear and easy-to-understand interface.	High

Table 2.11 : Settings

5. **Chat Feature**

   As shown in the following **Table 2.12**

Req. ID	Requirement	Req. Description	Priority
NF5.1	Fast Messaging	Ensure fast sending and receiving of messages.	High
NF5.2	Simple UI	Provide a simple and user-friendly interface.	High
NF5.3	Device Compatibility	Ensure compatibility with various devices (phones, tablets).	Medium

Table 2.12 : Chat Feature

6. **Sign Translator**

   As shown in the following **Table 2.13**

Req. ID	Requirement	Req. Description	Priority
NF6.1	High Accuracy	Ensure high accuracy in sign recognition.	High
NF6.2	Fast Translation	Ensure fast translation processes.	High
NF6.3	Clear UI	Provide a clear and easy-to-understand interface.	High

Table 2.13 : Sign Translator

7. **Word Translator**

   As shown in the following **Table 2.14**

Req. ID	Requirement	Req. Description	Priority
NF7.1	High Accuracy	Ensure high accuracy in speech-to-text conversion.	High
NF7.2	Fast Translation	Ensure fast translation processes.	High
NF7.3	Attractive UI	Provide an attractive and user-friendly interface.	High

Table 2.14 : Word Translator

2.6. Use Case diagrams

- **Use Case Diagram**

  As shown in the following **Figure 2.1**

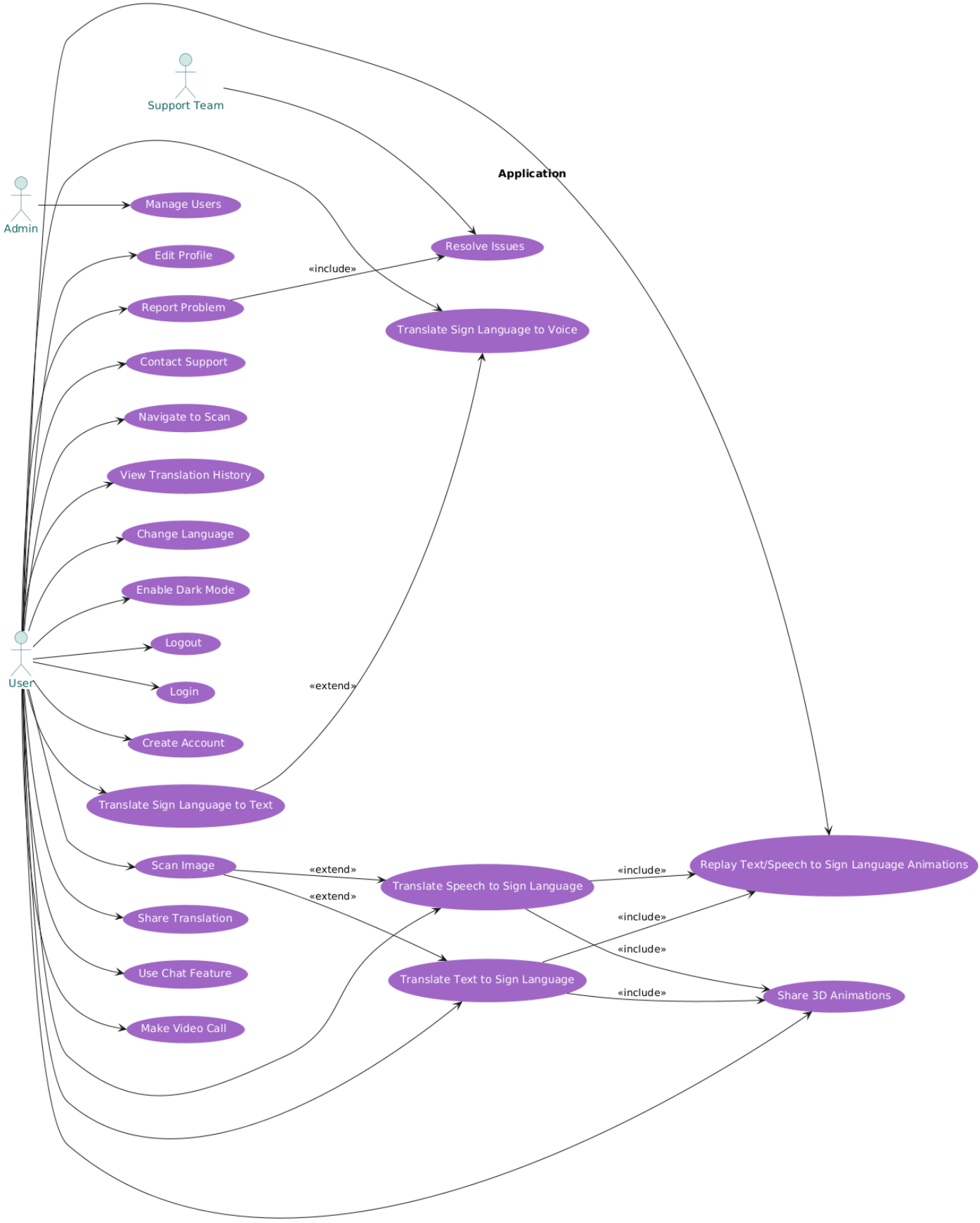


Figure 2.1 : Use Case Diagram

- **Use Case Description**

1. **Navigate to Scan (UC1)**

As shown in the following **Table 2.15**

Use Case Name	Navigate to Scan
Primary Actors	User
Description	The user navigates to the Scan feature through the burger menu.
Precondition	The app must be open.
Trigger	Clicking on the Scan option in the burger menu.
Scenario	1. The user opens the burger menu. 2. Clicks on the Scan option. 3. The user is redirected to the Scan page.
Possible Exceptions	- Failed to load the Scan page.
Postconditions	The Scan page is opened successfully.

Table 2.15 : Navigate to Scan

2. **View Translation History (UC2)**

As shown in the following **Table 2.16**

Use Case Name	View Translation History
Primary Actors	User
Description	The user views their previous translation history.
Precondition	The user must be logged in.
Trigger	Clicking on the History option in the burger menu.
Scenario	1. The user opens the burger menu. 2. Clicks on the History option. 3. The translation history is displayed.
Possible Exceptions	- No previous translations. - Failed to load history.
Postconditions	The translation history is displayed successfully.

Table 2.16 : View Translation History

3. **Change Language (UC3)**

   As shown in the following **Table 2.17**

Use Case Name	Change Language
Primary Actors	User
Description	The user changes the app language to their preferred language.
Precondition	The app must be open.
Trigger	Clicking on the Change Language option in the burger menu.
Scenario	1. The user opens the burger menu. 2. Clicks on the Change Language option. 3. Selects their preferred language.
Possible Exceptions	- The selected language is not supported.
Postconditions	The app language is changed successfully.

Table 2.17: Change Language

4. **Enable Dark Mode (UC4)**

   As shown in the following **Table 2.18**

Use Case Name	Enable Dark Mode
Primary Actors	User
Description	The user enables dark mode to change the app's appearance.
Precondition	The app must be open.
Trigger	Clicking on the Dark Mode option in the burger menu.
Scenario	1. The user opens the burger menu. 2. Clicks on the Dark Mode option. 3. Dark mode is enabled.
Possible Exceptions	- Failed to enable dark mode.
Postconditions	Dark mode is enabled successfully.

Table 2.18: Enable Dark Mode

5. **Logout (UC5)**

As shown in the following **Table 2.19**

Use Case Name	Logout
Primary Actors	User
Description	The user logs out of the app.
Precondition	The user must be logged in.
Trigger	Clicking on the Logout option in the burger menu .
Scenario	1. The user opens the burger menu. 2.Clicks on the Logout option. 3. The user is logged out.
Possible Exceptions	- Failed to log out.
Postconditions	The user is logged out successfully and redirected to the login page.

Table 2.19 : Logout

6. **Login (UC6)**

As shown in the following **Table 2.20**

Use Case Name	Login
Primary Actors	User
Description	The user logs into the app using their email and password or via Google/Facebook.
Precondition	The user must have a registered account.
Trigger	Clicking on the Login button on the login page.
Scenario	1. The user enters their login credentials. 2. Clicks on the Login button. 3. The user is logged in.
Possible Exceptions	-Invalid login credentials. -No internet connection.
Postconditions	The user is logged in successfully and redirected to the home page.

Table 2.20 : Login

7. **Create Account (UC7)**
   As shown in the following **Table 2.21**

Use Case Name	Create Account
Primary Actors	User
Description	The user creates a new account using their full name, email, and password.
Precondition	The user must not be logged in.
Trigger	Clicking on the Create Account button on the login page.
Scenario	1. The user enters their account details. 2. Clicks on the Create Account button. 3. The account is created.
Possible Exceptions	- Email is already registered. - Invalid data.
Postconditions	The account is created successfully, and the user is redirected to the login page.

Table 2.21 : Create Account

8. **Scan Image (UC8)**
   As shown in the following **Table 2.22**

Use Case Name	Scan Image
Primary Actors	User
Description	The user scans an image using the camera or gallery to extract text.
Precondition	The app must be open, and the user must be in the Scan section.
Trigger	Clicking on the Scan button.
Scenario	1. The user opens the Scan section. 2. Clicks on the Scan button. 3. Text is extracted from the image.
Possible Exceptions	-The image is unclear. -Failed to extract text.
Postconditions	The text is extracted successfully and displayed to the user.

Table 2.22 : Scan Image

9. **Translate Text to Sign Language (UC9)**

   As shown in the following **Table 2.23**

Use Case Name	Translate Text to Sign Language
Primary Actors	User
Description	The user translates text into sign language using a 3D character.
Precondition	The text must be extracted from an image or entered manually.
Trigger	Clicking on the Translate button.
Scenario	1. The user enters the text. 2. Clicks on the Translate button. 3. The translation is displayed using the 3D character. 4.The user can share the animations or replay them.
Possible Exceptions	- The text is not supported. - Failed to load the 3D character.
Postconditions	The translation is displayed successfully, and the user can share or replay the animations.

Table 2.23 : Translate Text to Sign Language

10. **Translate Speech to Sign Language (UC10)**

    As shown in the following **Table 2.24**

Use Case Name	Translate Speech to Sign Language
Primary Actors	User
Description	The user records speech, which is converted to text and then translated into sign language.
Precondition	The app must be open, and the user must be in the Translate section.
Trigger	Clicking on the Record button.
Scenario	1. The user clicks on the Record button. 2. Records their speech. 3. The speech is converted to text and translated. 4. The user can share the animations or replay them.
Possible Exceptions	-The speech is unclear. - Failed to convert speech to text.
Postconditions	The translation is displayed successfully, and the user can share or replay the animations.

Table 2.24: Translate Speech to Sign Language

11. **Use Chat Feature (UC11)**
    As shown in the following **Table 2.25**

Use Case Name	Use Chat Feature
Primary Actors	User
Description	The user uses the chat feature to communicate with other users.
Precondition	The user must be logged in.
Trigger	Clicking on the Chat button on the home page.
Scenario	1. The user opens the Chat section. 2. Searches for another user. 3.Starts chatting.
Possible Exceptions	- No internet connection. - Failed to load the chat.
Postconditions	The chat is started successfully.

Table 2.25: Use Chat Feature

12. **Make Video Call (UC12)**
    As shown in the following **Table 2.26**

Use Case Name	Make Video Call
Primary Actors	User
Description	The user makes a video call with another user.
Precondition	The user must be logged in and in the Chat section.
Trigger	Clicking on the Video Call button in the chat.
Scenario	1. The user opens the chat. 2. Clicks on the Video Call button. 3. The video call starts.
Possible Exceptions	-No internet connection. - Failed to start the video call.
Postconditions	The video call is started successfully.

Table 2.26: Make Video Call

13. **Edit Profile (UC13)**
    As shown in the following **Table 2.27**

Use Case Name	Edit Profile
Primary Actors	User
Description	The user edits their profile (picture, name, password).
Precondition	The user must be logged in.
Trigger	Clicking on the Edit Profile option in the settings.
Scenario	1. The user opens the settings. 2. Clicks on the Edit Profile option. 3. Makes changes and saves them.
Possible Exceptions	- Invalid data. - Failed to save changes.
Postconditions	The changes are saved successfully.

Table 2.27: Edit Profile

14. **Report Problem (UC14)**
    As shown in the following **Table 2.28**

Use Case Name	Report Problem
Primary Actors	User
Description	The user reports a problem they are facing in the app.
Precondition	The user must be logged in.
Trigger	Clicking on the Report Problem button.
Scenario	1. The user opens the settings. 2. Clicks on the Report Problem button. 3. Fills out the form and submits it.
Possible Exceptions	- No internet connection. - Failed to submit the report.
Postconditions	The report is submitted successfully to the support team.

Table 2.28: Report Problem

15. **Contact Support (UC15)**
    As shown in the following **Table 2.29**

Use Case Name	Contact Support
Primary Actors	User
Description	TThe user contacts the support team.
Precondition	The app must be open.
Trigger	Clicking on the Contact Support option in the settings.
Scenario	1. The user opens the settings. 2. Clicks on the Contact Support option. 3. Sends a message.
Possible Exceptions	- No internet connection. - Failed to send the message.
Postconditions	The message is sent successfully to the support team.

Table 2.29: Contact Support

16. **Manage Users (UC16)**
    As shown in the following **Table 2.30**

Use Case Name	Manage Users
Primary Actors	Admin
Description	The admin manages user accounts (delete, edit, etc.).
Precondition	TThe admin must be logged in.
Trigger	Clicking on the Manage Users option in the admin panel.
Scenario	1. The admin opens the admin panel. 2. Clicks on the Manage Users option. 3. Manages user accounts.
Possible Exceptions	- Failed to load the user list.
Postconditions	The user accounts are managed successfully.

Table 2.30: Manage Users

17. **Translate Sign Language to Text (UC17)**

As shown in the following **Table 2.31**

Use Case Name	Translate Sign Language to Text
Primary Actors	User
Description	The user translates sign language gestures into text.
Precondition	The app must be open, and the user must be in the Translate section.
Trigger	Clicking on the Translate Sign Language to Text button.
Scenario	1. The user performs sign language gestures in front of the camera. 2. The app recognizes the gestures and converts them to text. 3. The text is displayed on the screen. 4. The user can share the animations or replay them.
Possible Exceptions	- Gestures are not recognized. - Failed to convert gestures to text.
Postconditions	The text is displayed successfully, and the user can share or replay the animations.

Table 2.31: Translate Sign Language to Text

18. **Translate Sign Language to Voice (UC18)**

As shown in the following **Table 2.32**

Use Case Name	Translate Sign Language to Voice
Primary Actors	User
Description	The user translates sign language gestures into voice.
Precondition	The app must be open, and the user must be in the Translate section.
Trigger	Clicking on the Translate Sign Language to Voice button.
Scenario	1. The user performs sign language gestures in front of the camera. 2. The app recognizes the gestures and converts them to voice. 3. The voice is played through the app. 4. The user can share the animations or replay them.
Possible Exceptions	- Gestures are not recognized. - Failed to convert gestures to voice.
Postconditions	The voice is played successfully, and the user can share or replay the animations.

Table 2.32: Translate Sign Language to Voice

19. **Share Translation (UC19)**
    As shown in the following **Table 2.33**

Use Case Name	Share Translation
Primary Actors	User
Description	The user shares the translated text or voice with others.
Precondition	The user must have a completed translation (text or voice).
Trigger	Clicking on the Share button.
Scenario	1. The user completes a translation. 2. Clicks on the Share button. 3. Chooses a sharing method (e.g., chat, email, social media).
Possible Exceptions	- No internet connection. - Failed to share the translation.
Postconditions	The translation is shared successfully.

Table 2.33: Share Translation

20. **Share 3D Animations (UC20)**
    As shown in the following **Table 2.34**

Use Case Name	Share 3D Animations
Primary Actors	User
Description	The user shares the 3D character animations as a link that opens in a browser with the same animations.
Precondition	The user must have completed a translation (text, speech, or sign language).
Trigger	Clicking on the Share Animations button.
Scenario	1. The user completes a translation. 2. TClicks on the Share Animations button. 3. A link is generated and shared. 4. The link opens in a browser with the same animations.
Possible Exceptions	-No internet connection. - Failed to generate the link.
Postconditions	The animations are shared successfully as a link.

Table 2.34: Share 3D Animations

21. **Replay Text/Speech to Sign Language Animations (UC21)**

As shown in the following **Table 2.35**

Use Case Name	Replay Text/Speech to Sign Language Animations
Primary Actors	User
Description	The user replays the 3D character animations for translations from text or speech to sign language.
Precondition	The user must have completed a translation from text or speech to sign language.
Trigger	Clicking on the Replay button.
Scenario	1. The user completes a translation from text or speech to sign language. 2. Clicks on the Replay button. 3. The animations are replayed.
Possible Exceptions	- Failed to load the animations.
Postconditions	The animations are replayed successfully.

Table 2.35: Replay Text/Speech to Sign Language Animations

22. **Resolve Issues (UC22)**

As shown in the following **Table 2.36**

Use Case Name	Resolve Issues
Primary Actors	Support Team
Description	The support team resolves issues reported by users.
Precondition	The support team must be logged in.
Trigger	Receiving a problem report from a user.
Scenario	1. The support team receives the report. 2. Analyzes the issue. 3. Resolves the issue.
Possible Exceptions	-No solution is available.
Postconditions	The issue is resolved successfully.

Table 2.36: Resolve Issues

3. Design

3.1. Sequence diagrams

1. **Navigate to Scan (UC1)**

   As shown in the following **Figure 3.1**

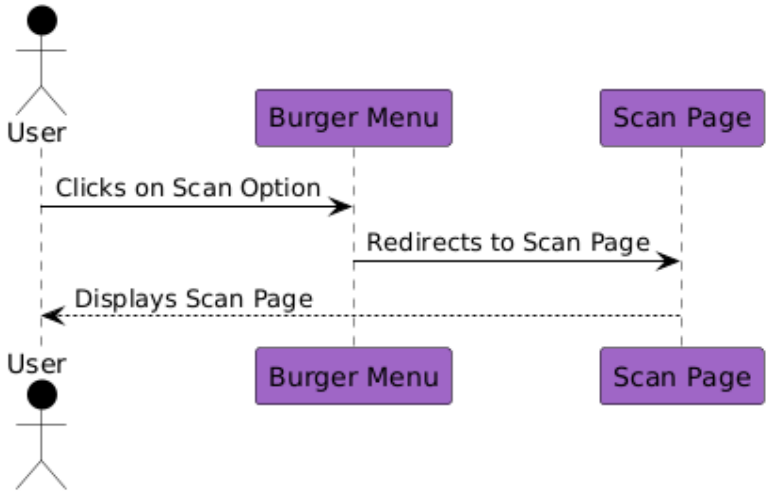


Figure 3.1 : Navigate to Scan

2. **View Translation History (UC2)**

   As shown in the following **Figure 3.2**

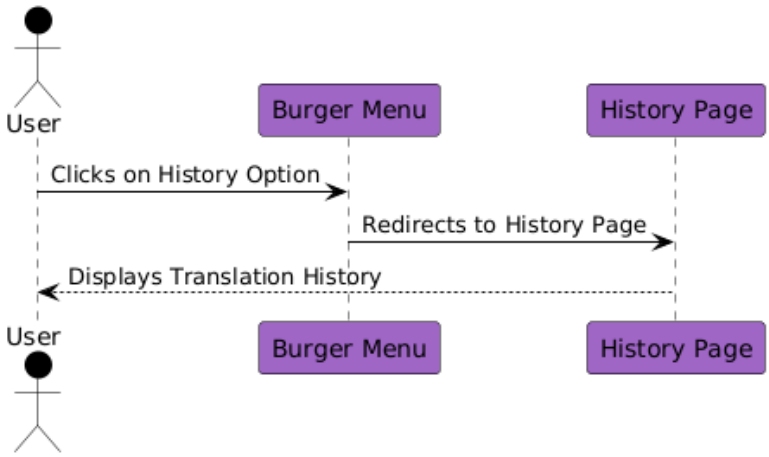


Figure 3.2 : View Translation History

3. **Change Language (UC3)**

As shown in the following **Figure 3.3**

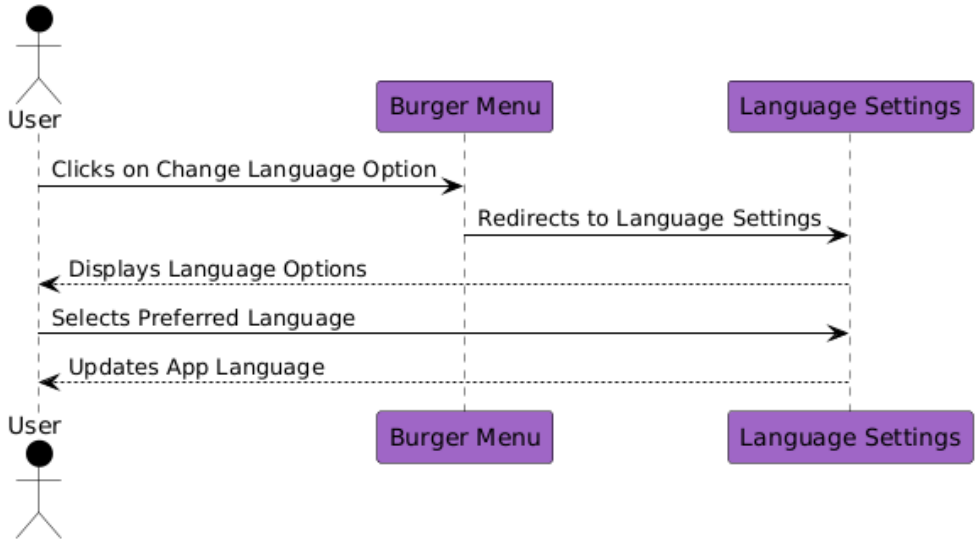


Figure 3.3 : Change Language

4. **Enable Dark Mode (UC4)**

As shown in the following **Figure 3.4**

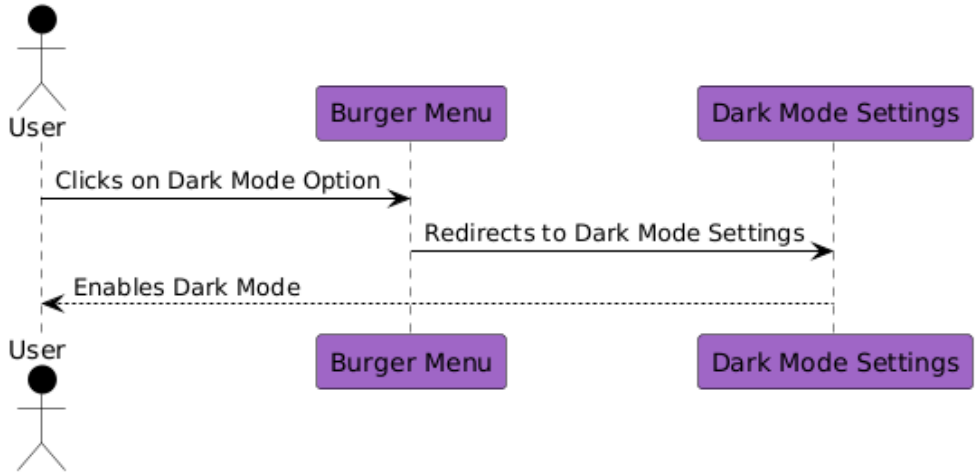


Figure 3.4 : Enable Dark Mode

5. **Logout (UC5)**

   As shown in the following **Figure 3.5**

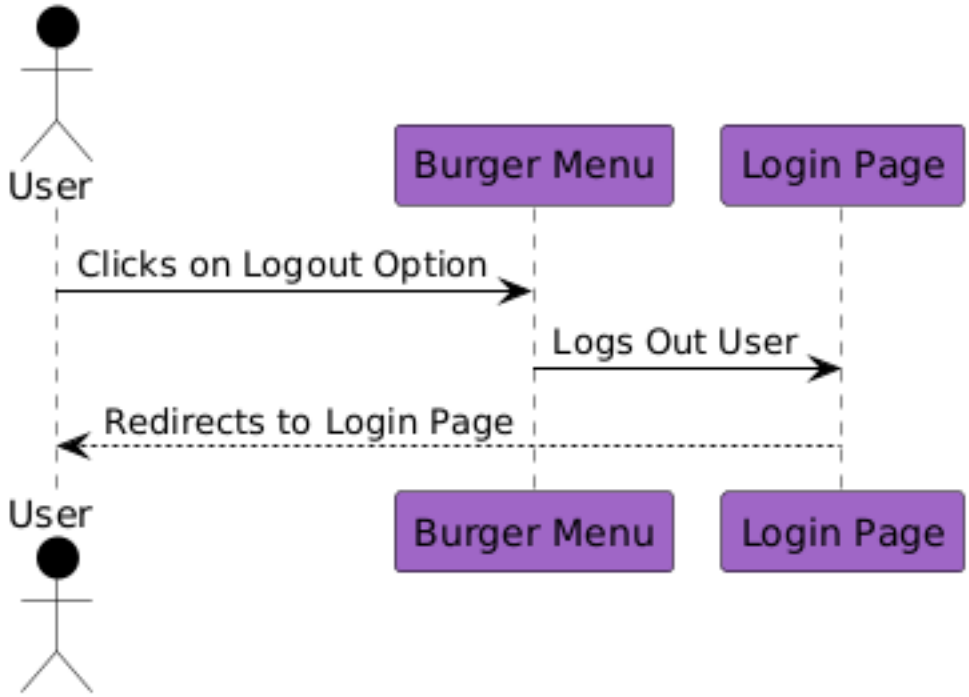


Figure 3.5 : Logout

6. **Login (UC6)**

   As shown in the following **Figure 3.6**

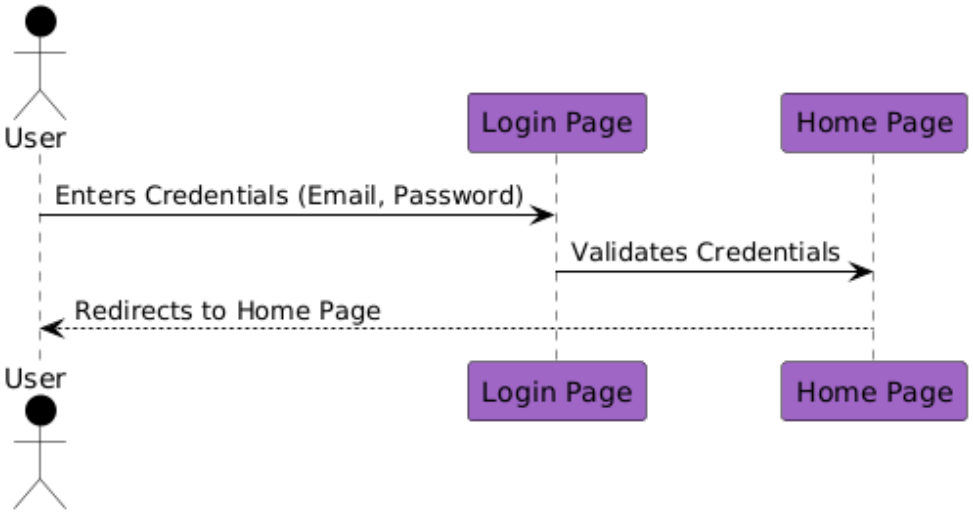


Figure 3.6 : Login

7. **Create Account (UC7)**

As shown in the following **Figure 3.7**

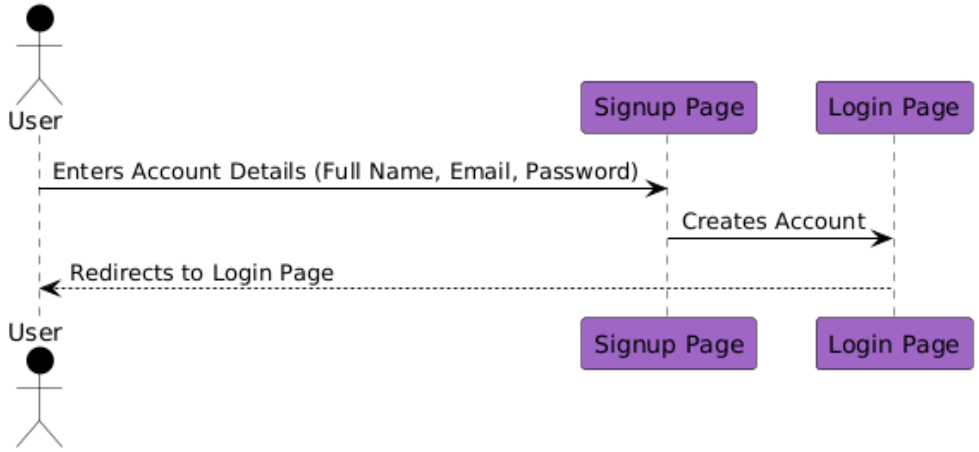


Figure 3.7 : Create Account

8. **Scan Image (UC8)**

As shown in the following **Figure 3.8**

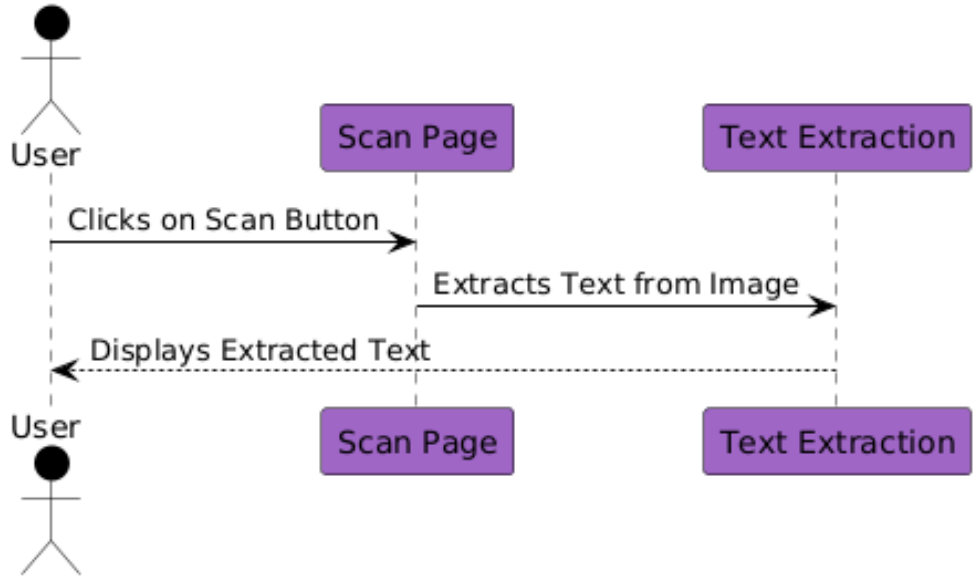


Figure 3.8 : Scan Image

9. **Translate Text to Sign Language (UC9)**

As shown in the following **Figure 3.9**

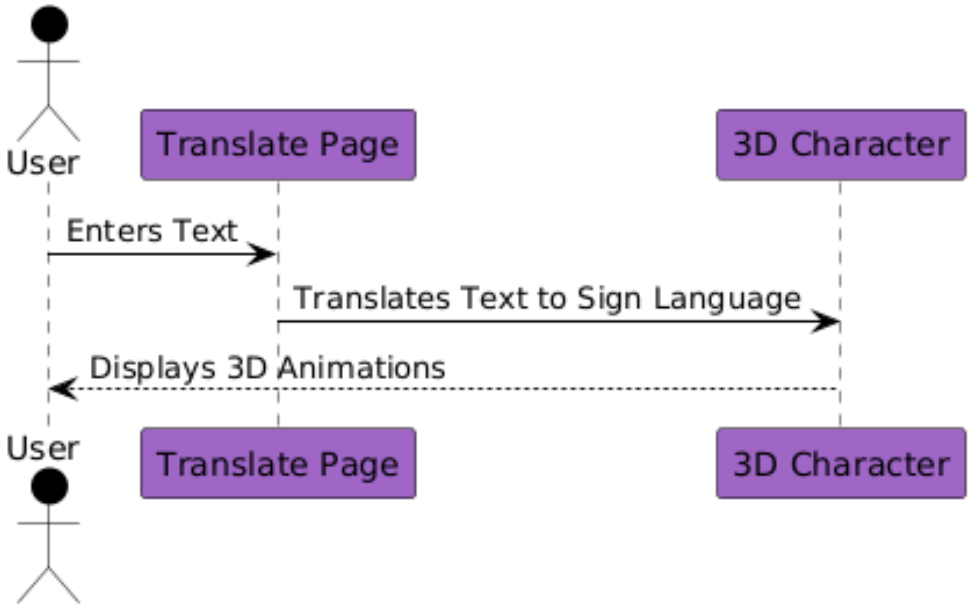


Figure 3.9 : Translate Text to Sign Language

10. **Translate Speech to Sign Language (UC10)**

As shown in the following **Figure 3.10**

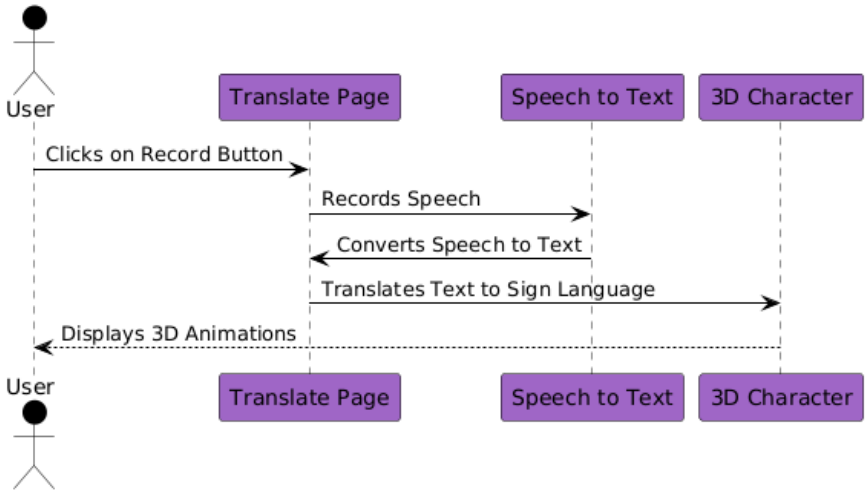


Figure 3.10 : Translate Speech to Sign Language

11. **Use Chat Feature (UC11)**
    As shown in the following **Figure 3.11**

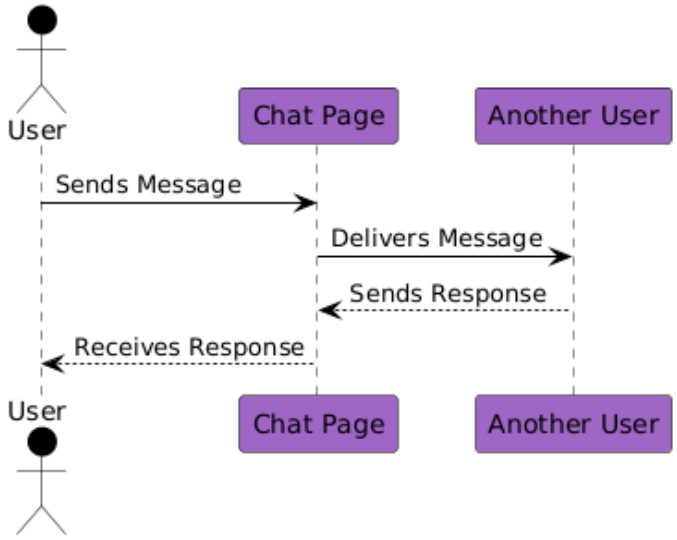


Figure 3.11 : Use Chat Feature

12. **Make Video Call (UC12)**
    As shown in the following **Figure 3.12**

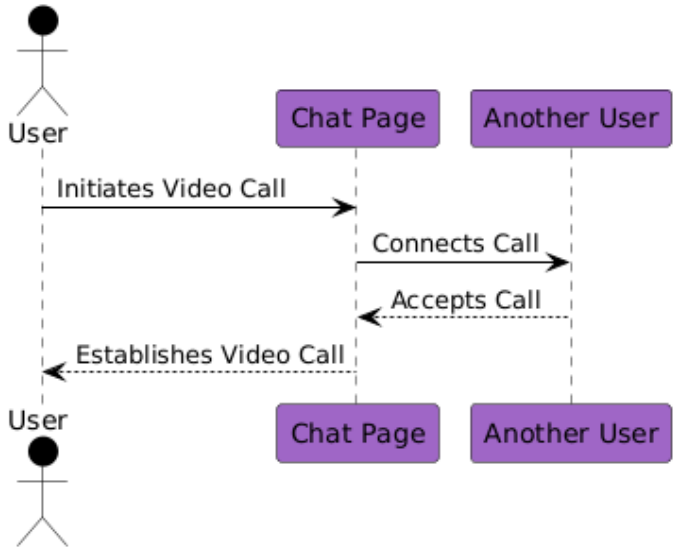


Figure 3.12 : Make Video Call

13. **Edit Profile (UC13)**
As shown in the following **Figure 3.13**

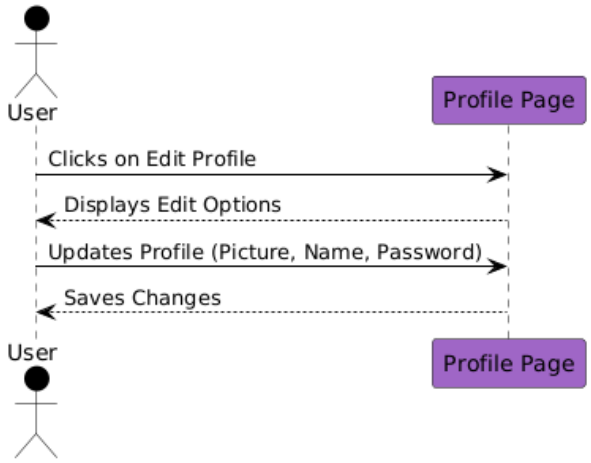


Figure 3.13 : Edit Profile

14. **Report Problem (UC14)**
As shown in the following **Figure 3.14**

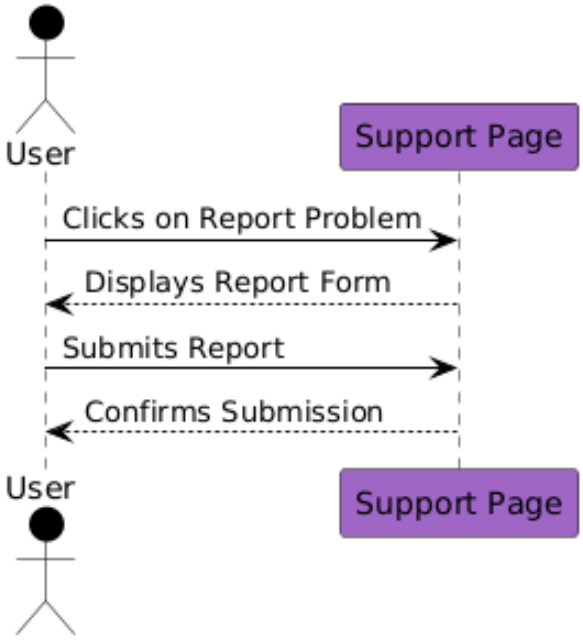


Figure 3.14: Report Problem

15. **Contact Support (UC15)**
As shown in the following **Figure 3.15**

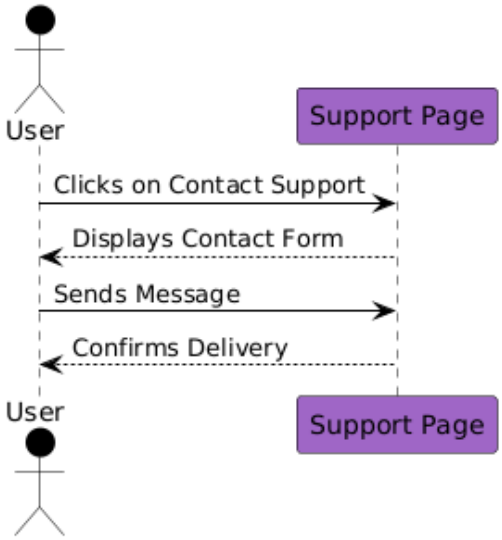


Figure 3.15 : Contact Support

16. **Manage Users (UC16)**
As shown in the following **Figure 3.16**

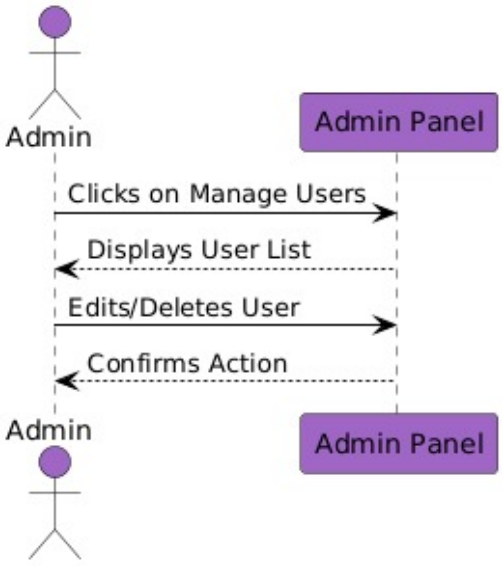


Figure 3.16 : Manage Users

17. **Translate Sign Language to Text (UC17)**

As shown in the following **Figure 3.17**

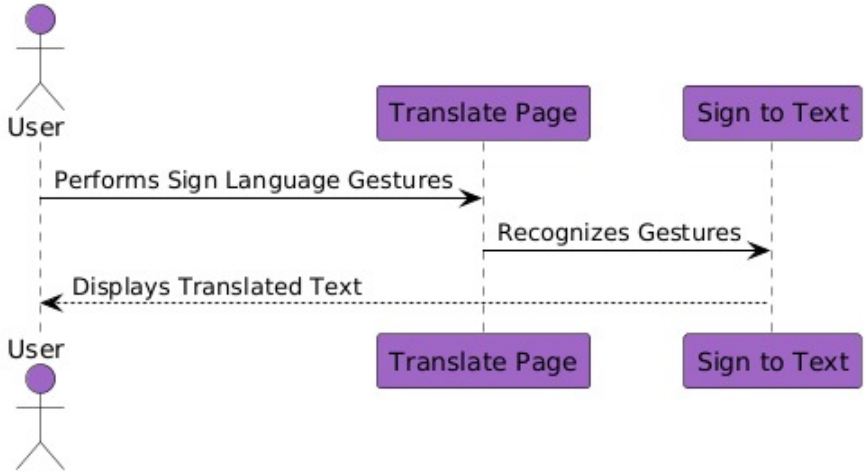


Figure 3.17 : Translate Sign Language to Text

18. **Translate Sign Language to Voice (UC18)**

As shown in the following **Figure 3.18**

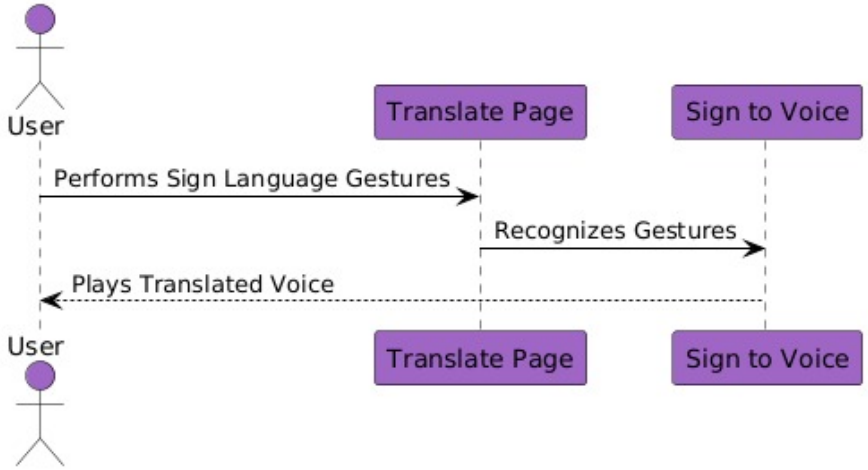


Figure 3.18 : Translate Sign Language to Voice

19. **Share Translation (UC19)**

As shown in the following **Figure 3.19**

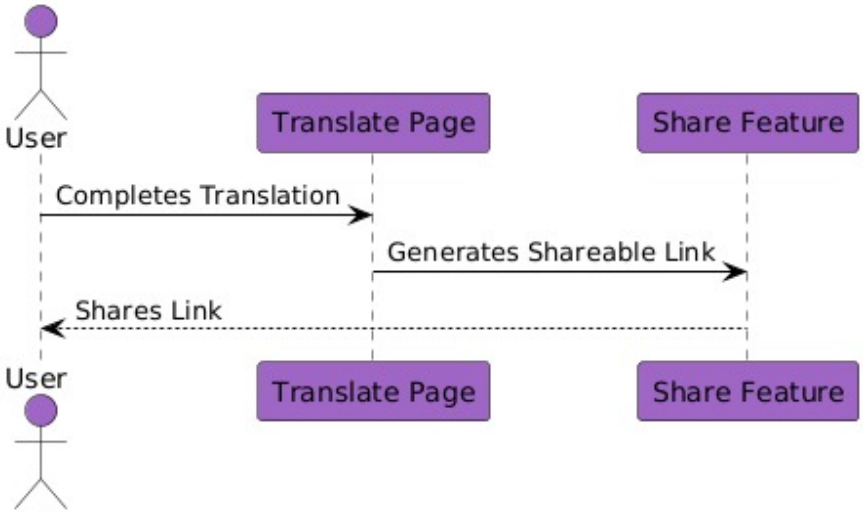


Figure 3.19 : Share Translation

20. **Share 3D Animations (UC20)**

As shown in the following **Figure 3.20**

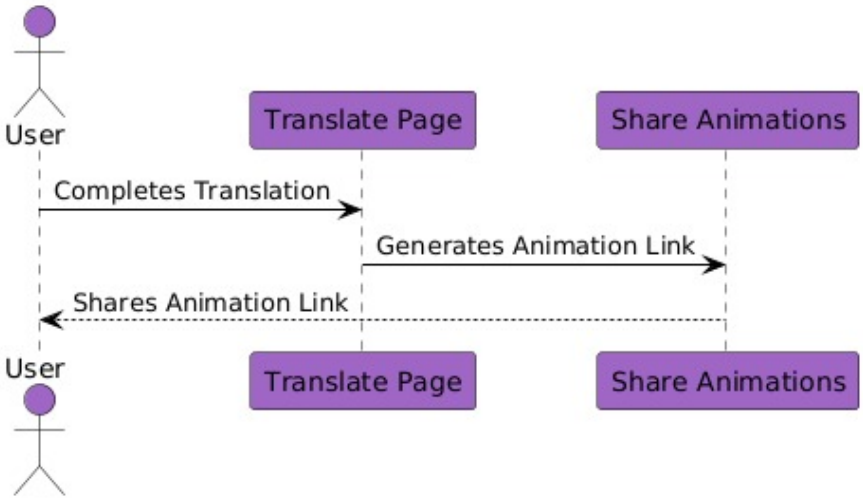


Figure 3.20 : Share 3D Animations

21. **Replay Text/Speech to Sign Language Animations (UC21)**

As shown in the following **Figure 3.21**

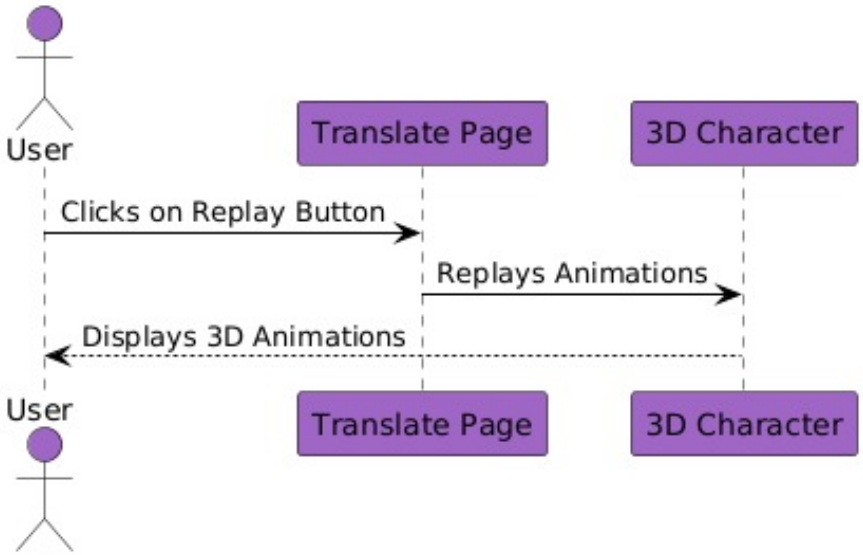


Figure 3.21 : Replay Text/Speech to Sign Language Animations

22. **Resolve Issues (UC22)**

As shown in the following **Figure 3.22**

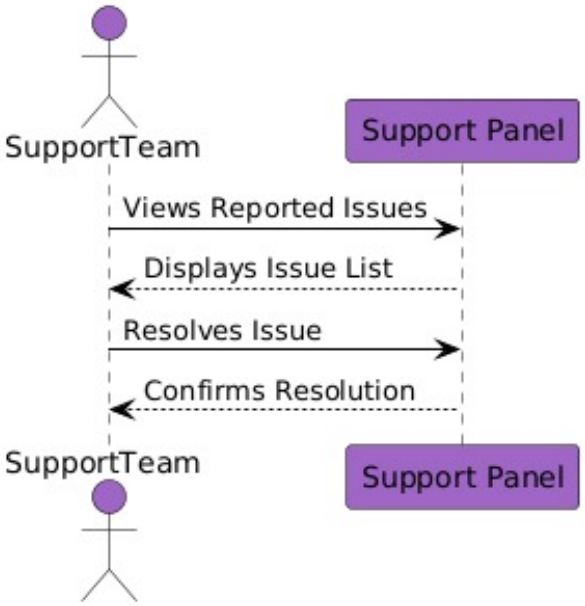


Figure 3.22 : Resolve Issues

3.2. ERD

As shown in the following **Figure 3.23**

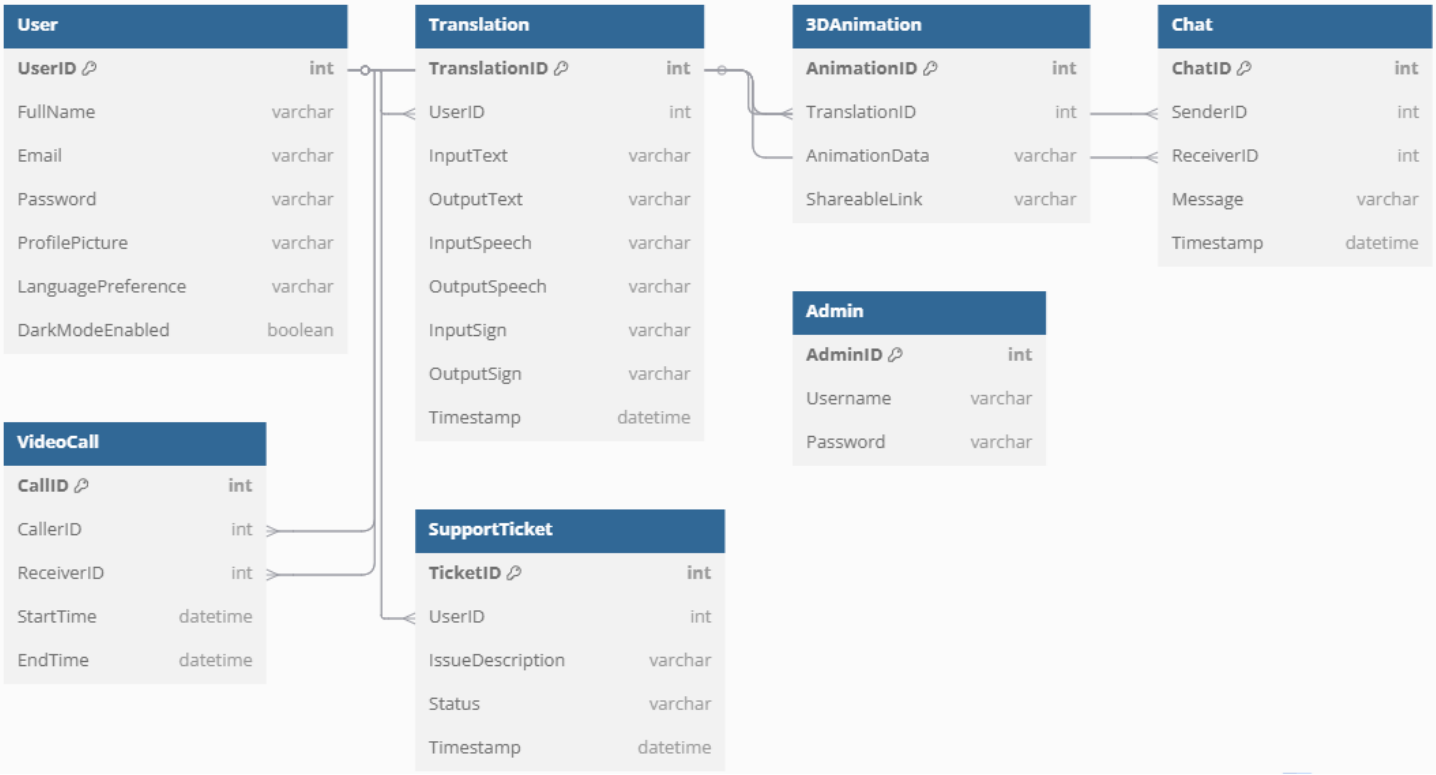


Figure 3.23 : ERD

3.3. Data Flow Diagrams

- DFD-level 0

As shown in the following **Figure 3.24**

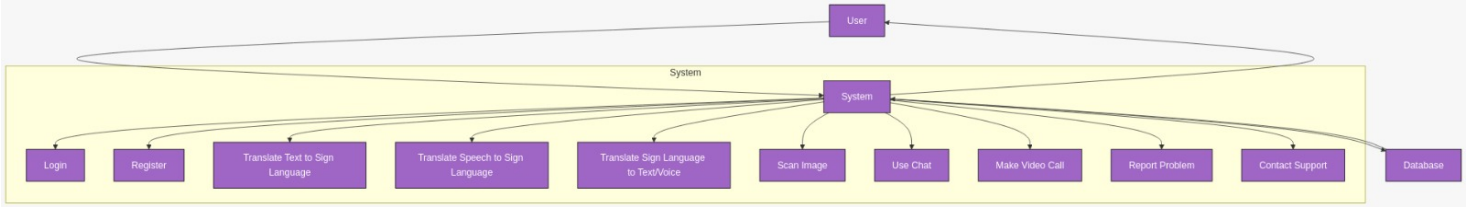


Figure 3.24: DFD-level 0

- DFD-level 1

As shown in the following **Figure 3.25**

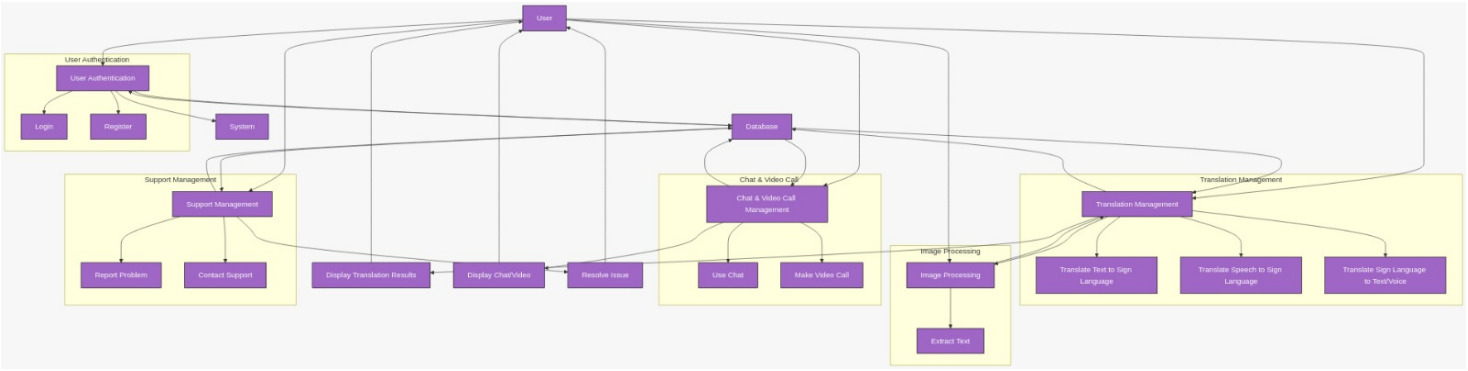


Figure 3.25 : DFD-level 1

3.4. State Diagram

As shown in the following **Figure 3.26**

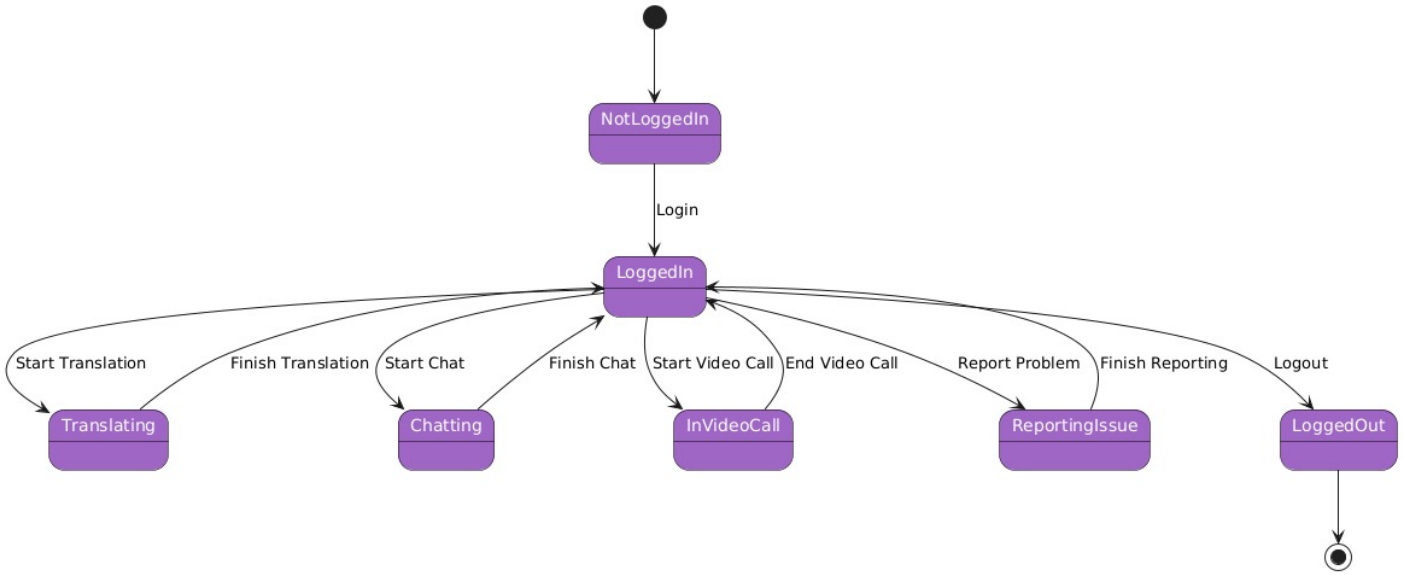


Figure 3.26: State Diagram

3.5. Flow Chart Diagram

As shown in the following **Figure 3.27**

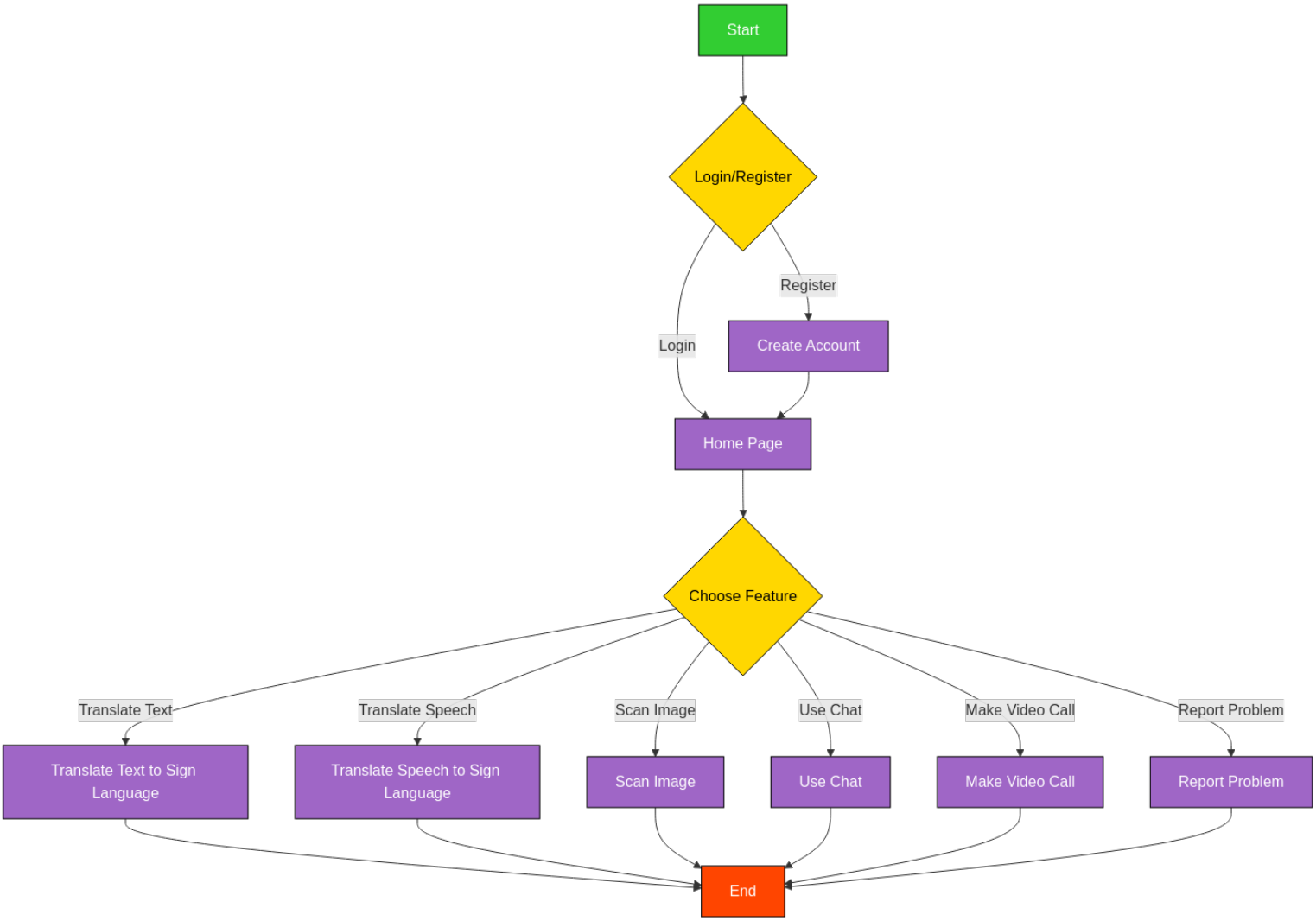


Figure 3.27 : Flow Chart Diagram

3.6. Algorithms or Pseudocode

1. **Login**

```
FUNCTION login(email, password):
    IF email EXISTS IN Users AND password MATCHES:
        SET currentUser TO User WITH email
        DISPLAY "Login Successful"
        REDIRECT TO Home Page
    ELSE:
        DISPLAY "Invalid Email or Password"
```

2. **Register**

```
FUNCTION register(fullName, email, password):
    IF email NOT EXISTS IN Users:
        CREATE NEW User WITH fullName, email, password
        SET currentUser TO NEW User
        DISPLAY "Account Created Successfully"
        REDIRECT TO Home Page
    ELSE:
        DISPLAY "Email Already Exists"
```

3. **Translate Text to Sign Language**

```
FUNCTION translateTextToSign(text):
    IF text IS VALID:
        CONVERT text TO Sign Language USING 3D Character
        DISPLAY 3D Animation
        GENERATE shareableLink FOR Animation
        DISPLAY "Translation Successful"
    ELSE:
        DISPLAY "Invalid Text Input"
```

4. **Translate Speech to Sign Language**

```
FUNCTION translateSpeechToSign(audio):
    IF audio IS VALID:
        CONVERT audio TO text USING Speech-to-Text API
        CONVERT text TO Sign Language USING 3D Character
        DISPLAY 3D Animation
        GENERATE shareableLink FOR Animation
        DISPLAY "Translation Successful"
    ELSE:
        DISPLAY "Invalid Audio Input"
```

5. **Scan Image**

```
FUNCTION scanImage(image):
    IF image IS VALID:
        EXTRACT text FROM image USING OCR
        DISPLAY Extracted Text
        REDIRECT TO Translate Text to Sign Language WITH Extracted Text
    ELSE:
        DISPLAY "Invalid Image"
```

6. **Use Chat**

```
FUNCTION useChat(senderId, receiverId, message):
    IF senderId AND receiverId ARE VALID:
        SEND message FROM senderId TO receiverId
        DISPLAY message IN Chat Window
    ELSE:
        DISPLAY "Invalid User IDs"
```

7. **Make Video Call**

```
FUNCTION makeVideoCall(callerId, receiverId):
    IF callerId AND receiverId ARE VALID:
        INITIATE Video Call BETWEEN callerId AND receiverId
        DISPLAY Video Call Interface
    ELSE:
        DISPLAY "Invalid User IDs"
```

8. **Report Problem**

```
FUNCTION reportProblem(userId, issueDescription):
    IF userId IS VALID:
        CREATE NEW SupportTicket WITH userId, issueDescription
        DISPLAY "Issue Reported Successfully"
    ELSE:
        DISPLAY "Invalid User ID"
```

9. **Logout**

```
FUNCTION logout():
    SET currentUser TO NULL
    DISPLAY "Logged Out Successfully"
    REDIRECT TO Login Page
```

10. **Change Language**

```
FUNCTION changeLanguage(userId, newLanguage):
    IF userId IS VALID:
        SET user.languagePreference TO newLanguage
        DISPLAY "Language Changed Successfully"
    ELSE:
        DISPLAY "Invalid User ID"
```

11. **Enable Dark Mode**

```
FUNCTION enableDarkMode(userId):
    IF userId IS VALID:
        SET user.darkModeEnabled TO TRUE
        DISPLAY "Dark Mode Enabled"
    ELSE:
        DISPLAY "Invalid User ID"
```

12. **Share Translation**

```
FUNCTION shareTranslation(translationId):
    IF translationId IS VALID:
        GENERATE shareableLink FOR translationId
        DISPLAY "Translation Shared Successfully"
    ELSE:
        DISPLAY "Invalid Translation ID"
```

13. **Replay Translation**

```
FUNCTION replayTranslation(translationId):
    IF translationId IS VALID:
        RETRIEVE 3D Animation FOR translationId
        DISPLAY 3D Animation
    ELSE:
        DISPLAY "Invalid Translation ID"
```

14. **Edit Profile**

```
FUNCTION editProfile(userId, newName, newPassword):
    IF userId IS VALID:
        SET user.fullName TO newName
        SET user.password TO newPassword
        DISPLAY "Profile Updated Successfully"
    ELSE:
        DISPLAY "Invalid User ID"
```

15. **View Translation History**

```
FUNCTION viewTranslationHistory(userId):
    IF userId IS VALID:
        RETRIEVE ALL translations FOR userId
        DISPLAY Translation History
    ELSE:
        DISPLAY "Invalid User ID"
```

16. **Contact Support**

```
FUNCTION contactSupport(userId, message):
    IF userId IS VALID:
        SEND message TO Support Team
        DISPLAY "Message Sent Successfully"
    ELSE:
        DISPLAY "Invalid User ID"
```

17. **Manage Users**

```
FUNCTION manageUsers(adminId):
    IF adminId IS VALID:
        RETRIEVE ALL users
        DISPLAY User List
        ALLOW admin TO EDIT/DELETE users
    ELSE:
        DISPLAY "Invalid Admin ID"
```

18. **Resolve Issues**

```
FUNCTION resolveIssue(adminId, ticketId):
    IF adminId AND ticketId ARE VALID:
        SET ticket.status TO "Resolved"
        DISPLAY "Issue Resolved Successfully"
    ELSE:
        DISPLAY "Invalid Admin ID or Ticket ID"
```

19. **Translate Sign Language to Text**

```
FUNCTION translateSignToText(signGestures):
    IF signGestures ARE VALID:
        CONVERT signGestures TO text USING Sign-to-Text API
        DISPLAY Translated Text
        DISPLAY "Translation Successful"
    ELSE:
        DISPLAY "Invalid Sign Gestures"
```

20. **Translate Sign Language to Voice**

```
FUNCTION translateSignToVoice(signGestures):
    IF signGestures ARE VALID:
        CONVERT signGestures TO text USING Sign-to-Text API
        CONVERT text TO voice USING Text-to-Speech API
        PLAY Translated Voice
        DISPLAY "Translation Successful"
    ELSE:
        DISPLAY "Invalid Sign Gestures"
```

21. **Share 3D Animations**

```
FUNCTION share3DAnimations(animationId):
    IF animationId IS VALID:
        GENERATE shareableLink FOR animationId
        DISPLAY "Animation Shared Successfully"
    ELSE:
        DISPLAY "Invalid Animation ID"
```

3.7. UML class diagrams

As shown in the following **Figure 3.28**

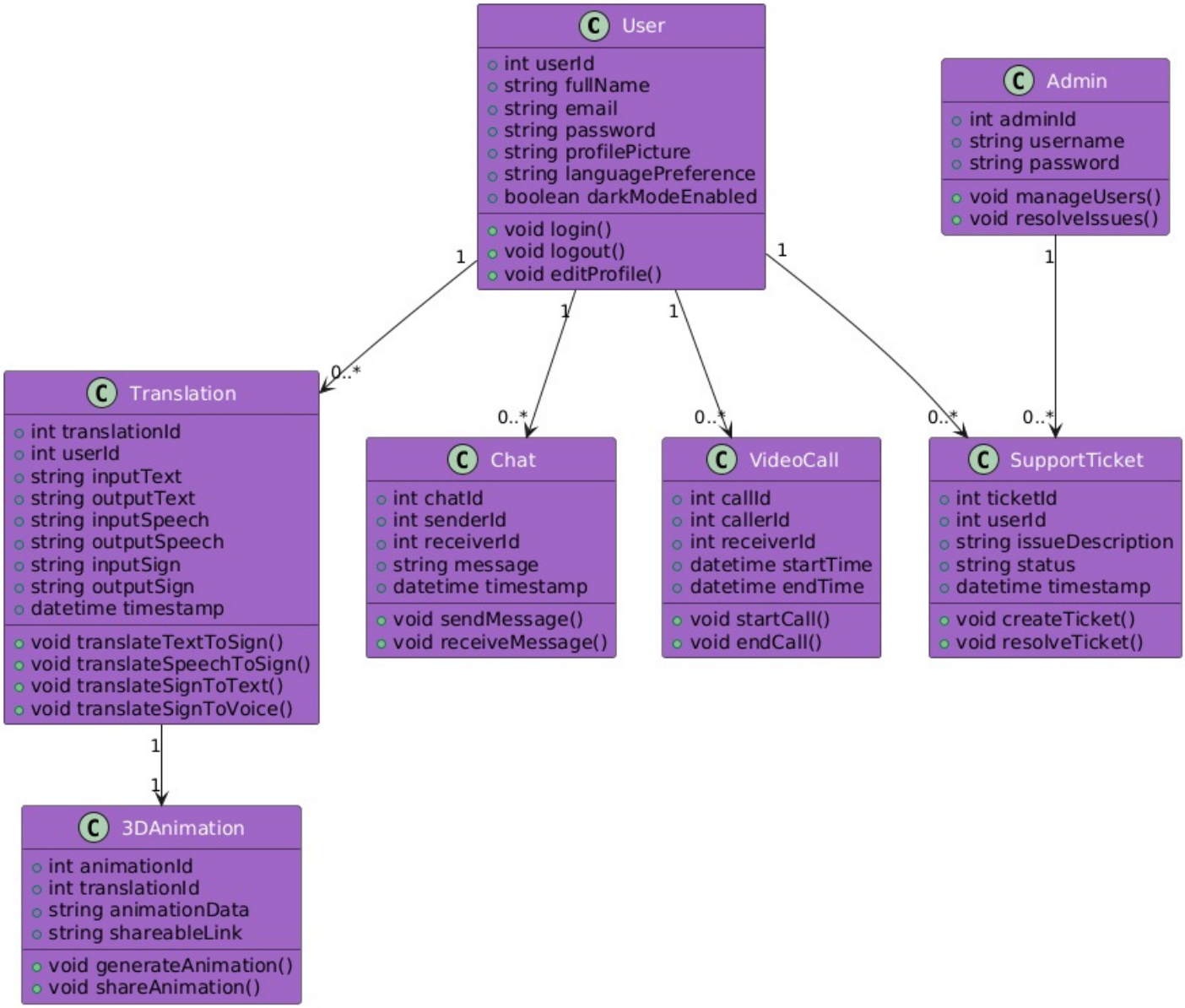


Figure 3.28 : UML class diagrams

4. Implementation Aspects

4.1. Blocks diagram

As shown in the following **Figure 4.1**

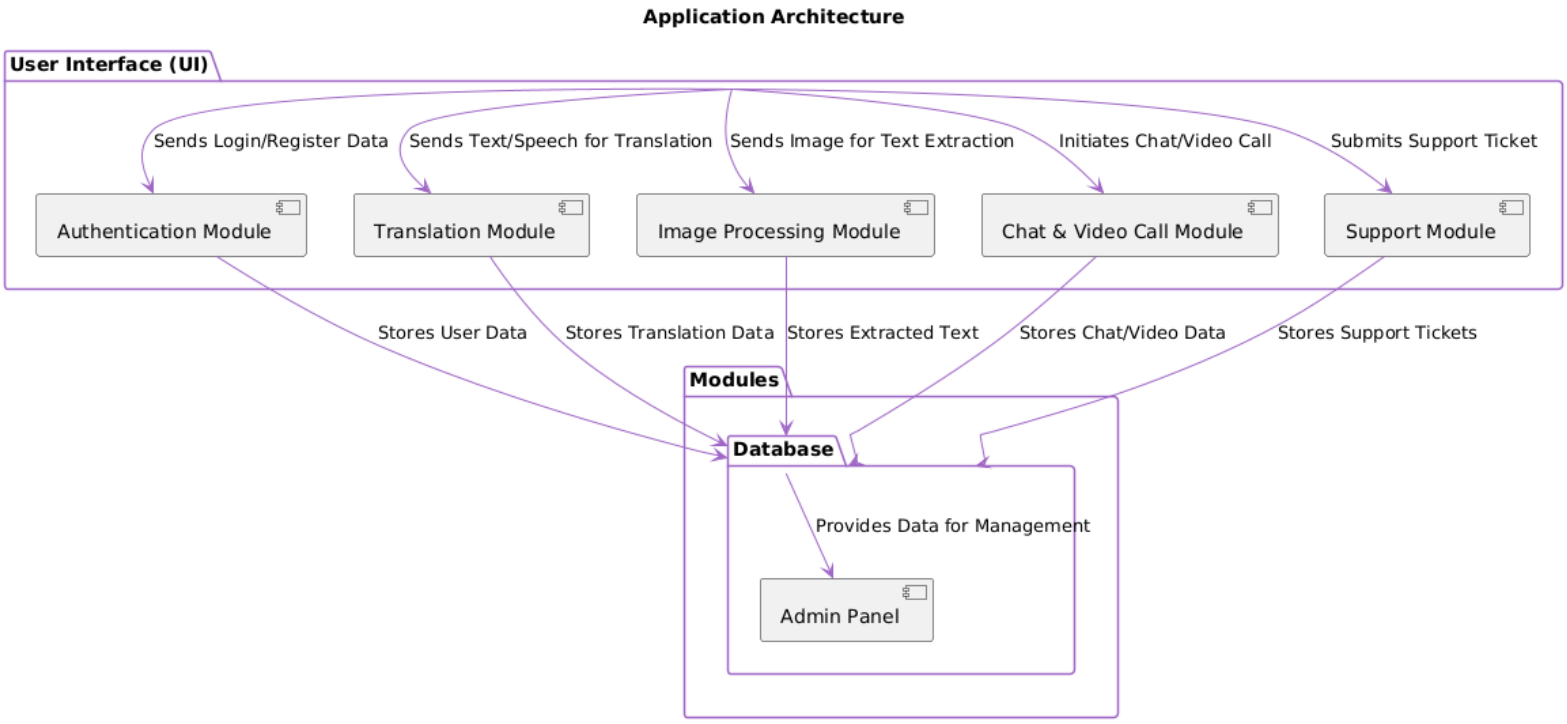


Figure 4.1 : Blocks diagram

4.2. Tools,Technologies and Programming Languages

As shown in the following **Figure 4.2** and **Figure 4.3**

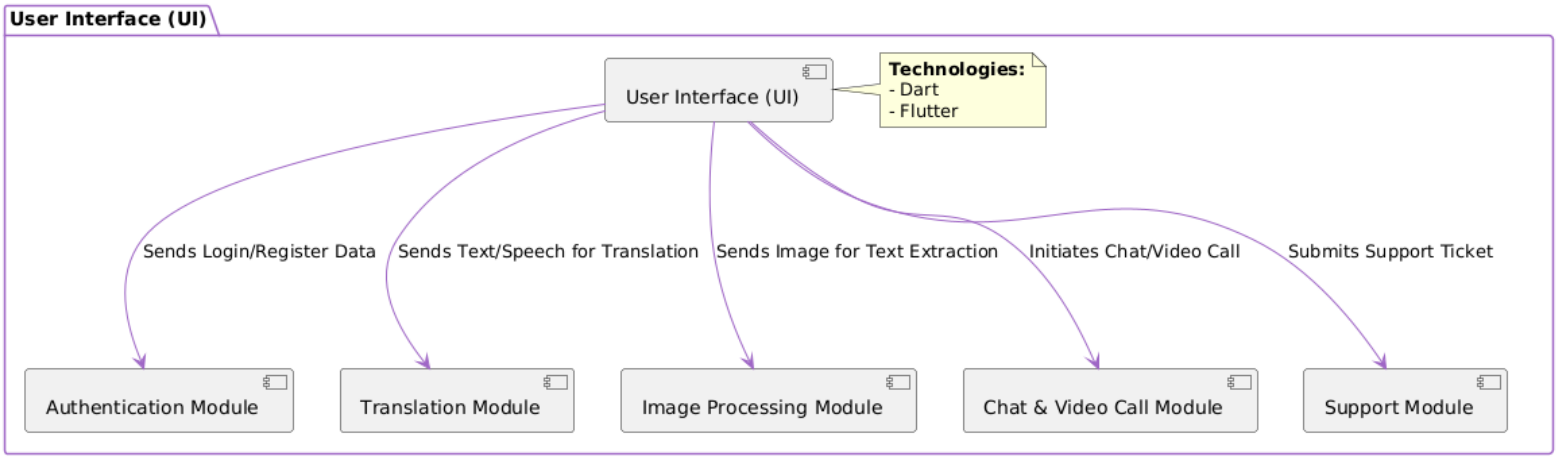


Figure 4.2 : User Interface (UI)

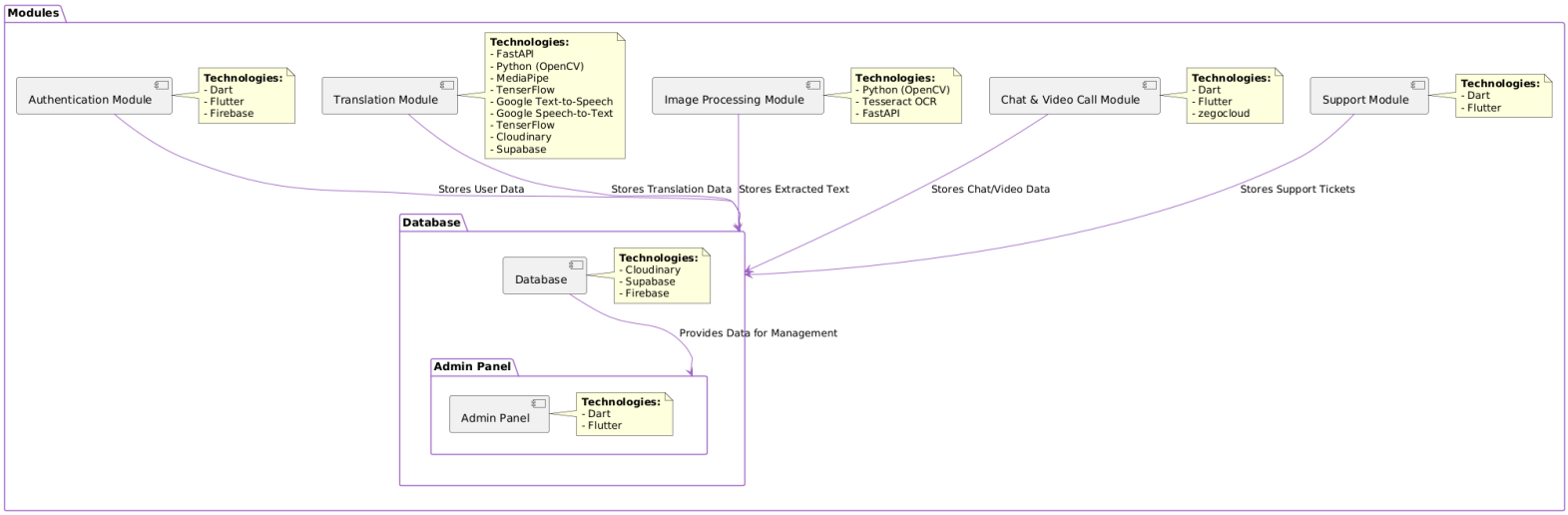
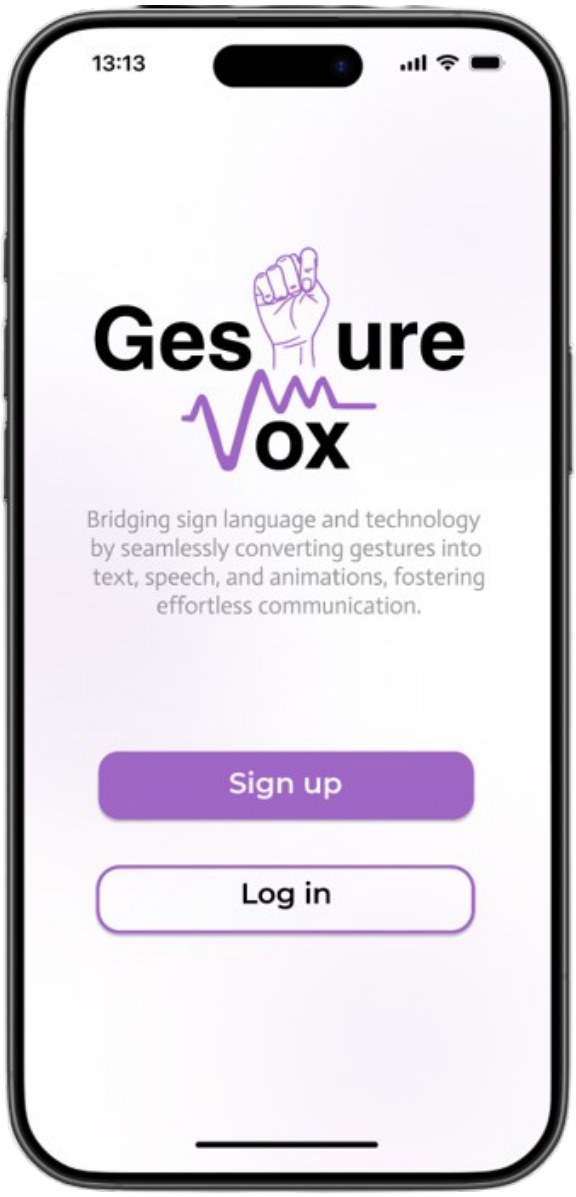


Figure 4.3: Modules

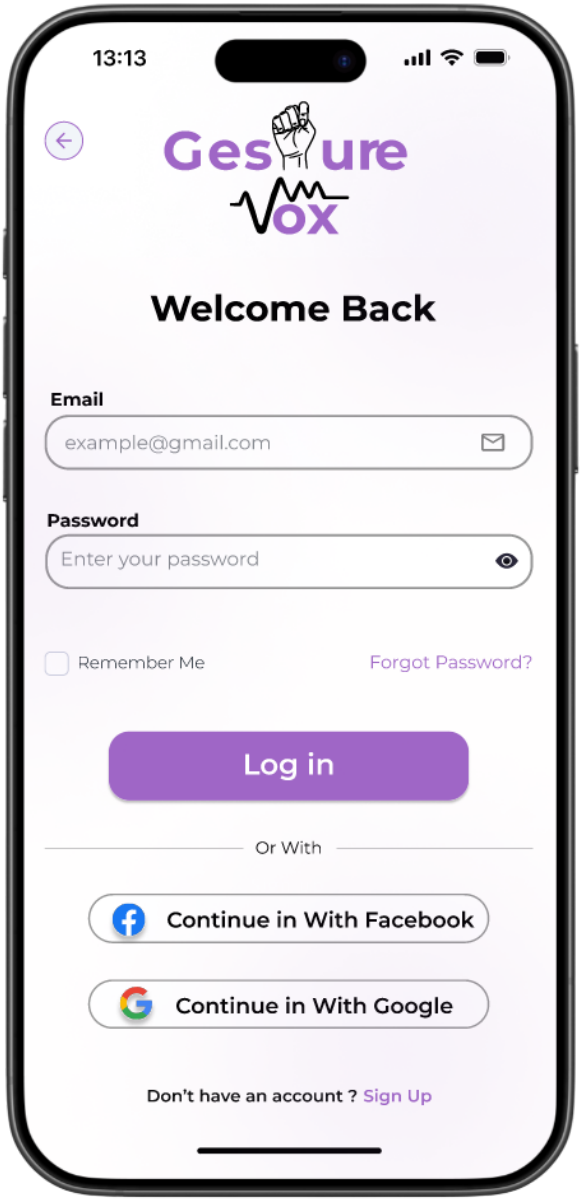
4.3. Prototype

- **Authentication**

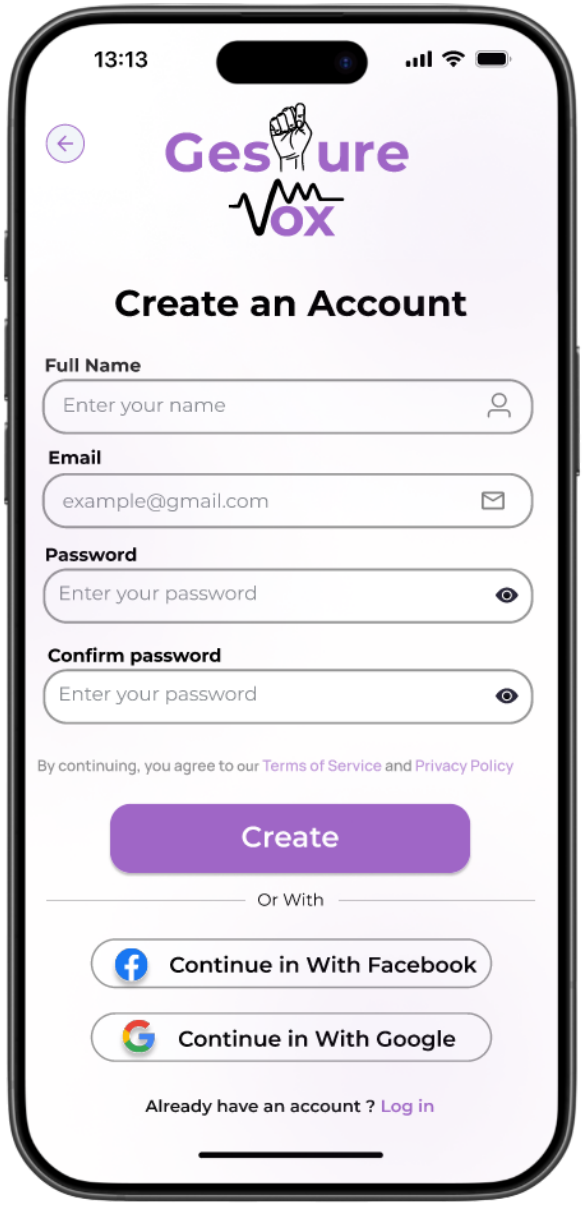
Welcome Page



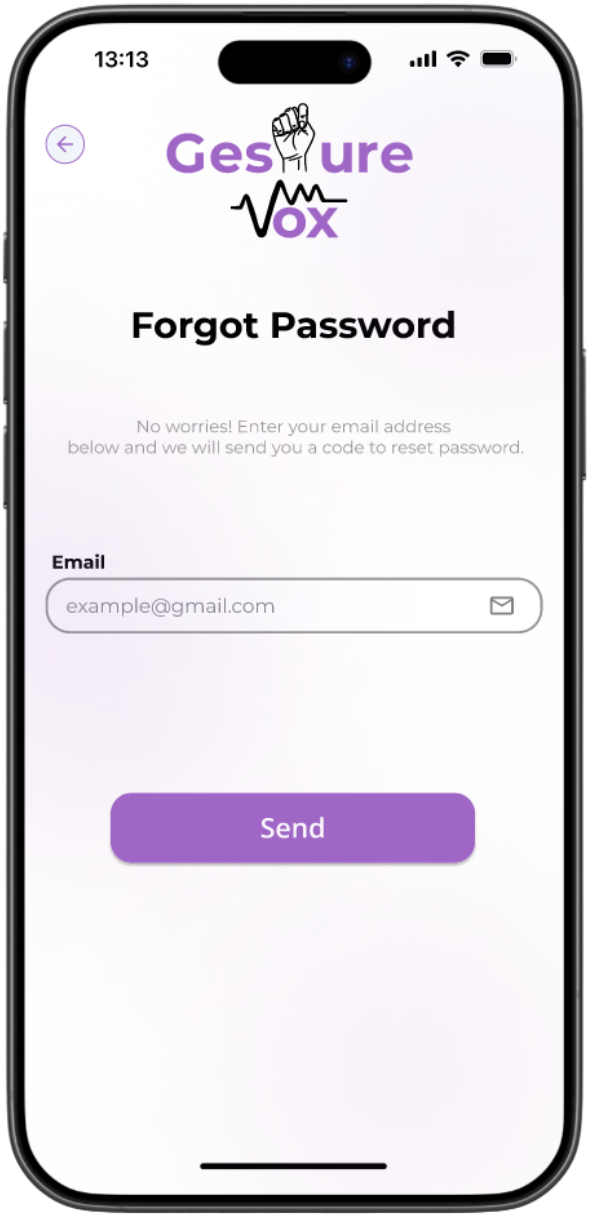
Log in Page



Sign Up Page

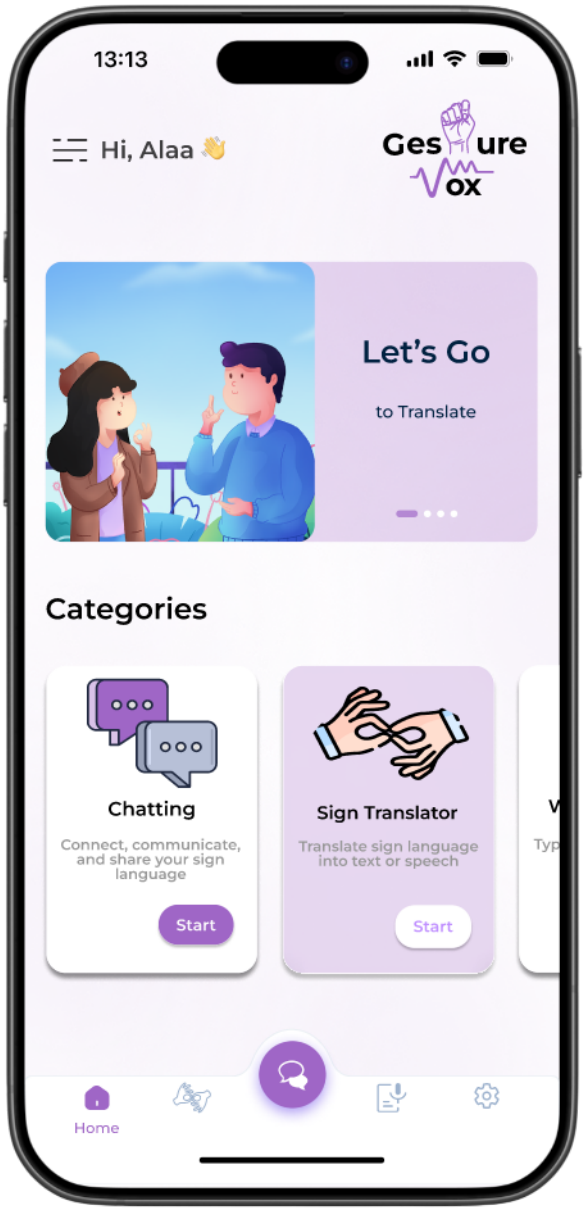


Forgot Password Page

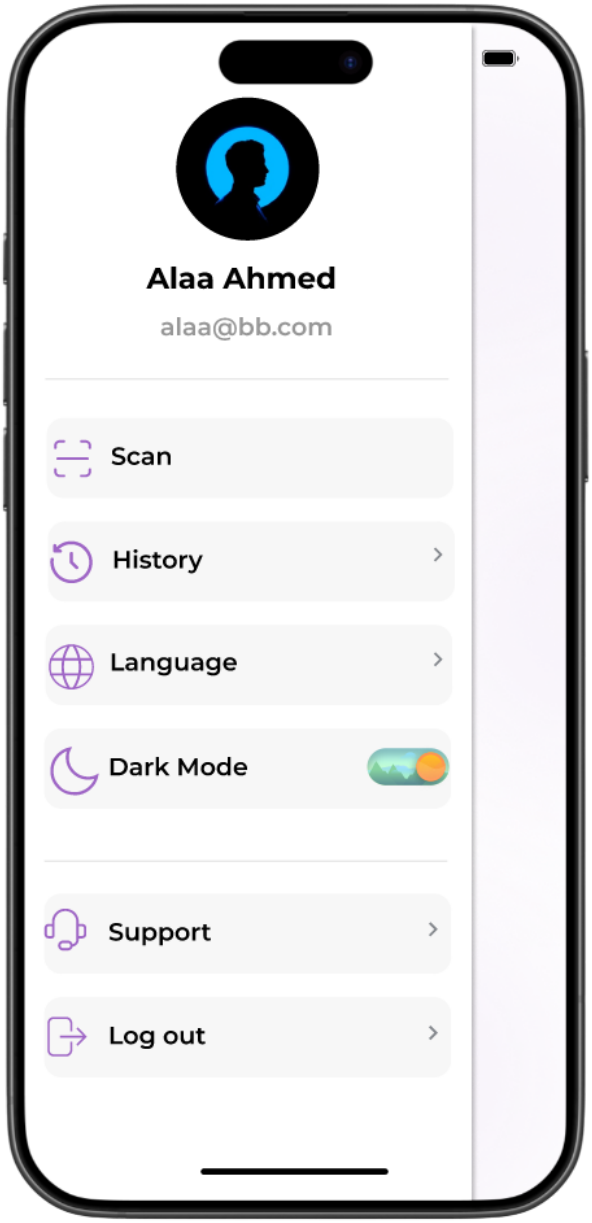


- **Home**

Home Page

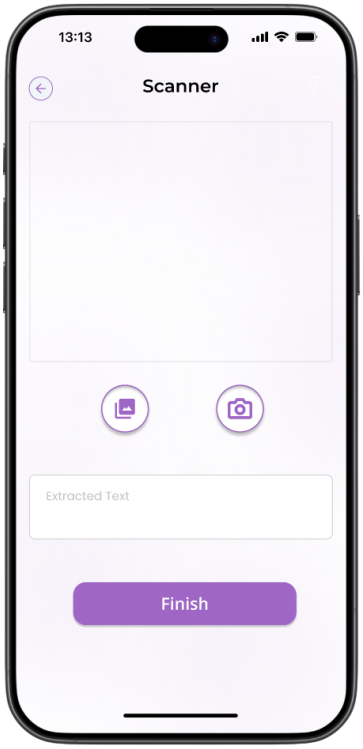


Burger Menu

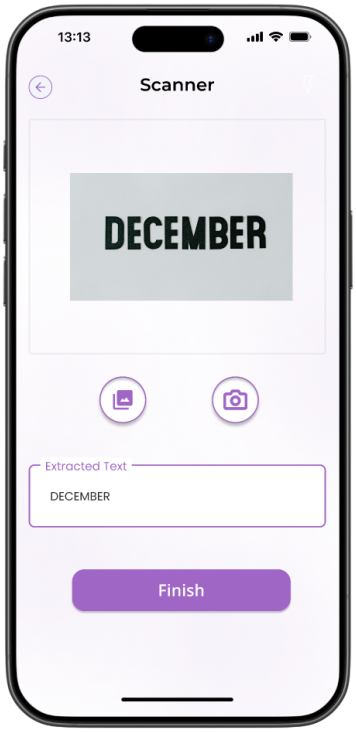


- **Burger Menu**

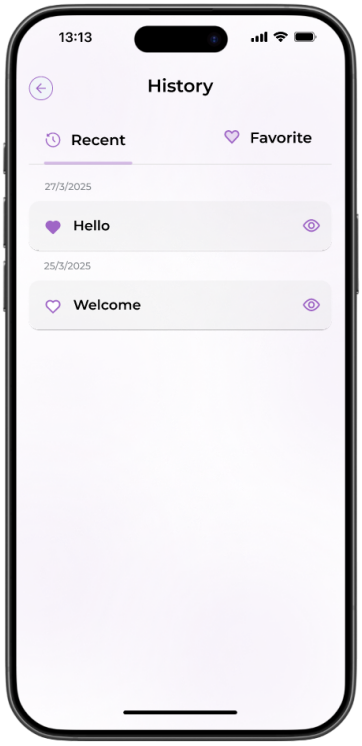
Scanner Page



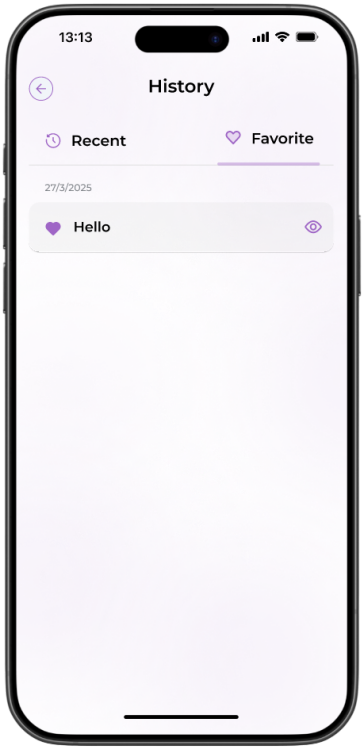
Extracted Text Page



History

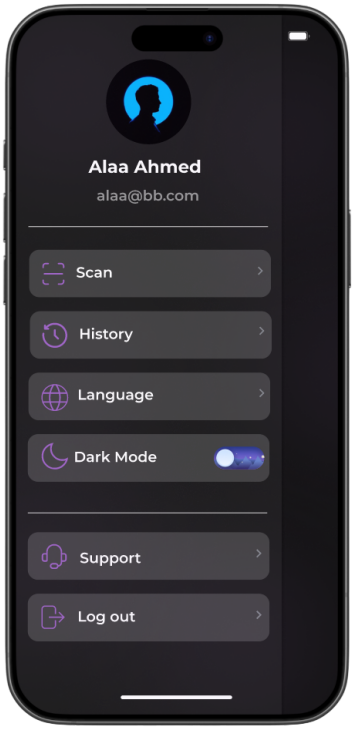


Favorite



Language

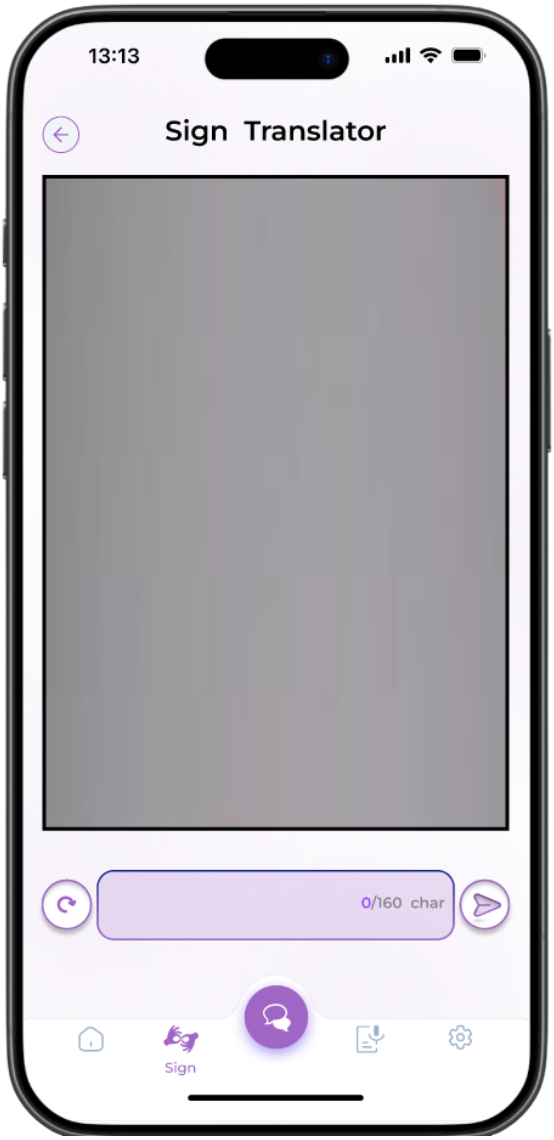
Dark Mode



- **Sign Translator**

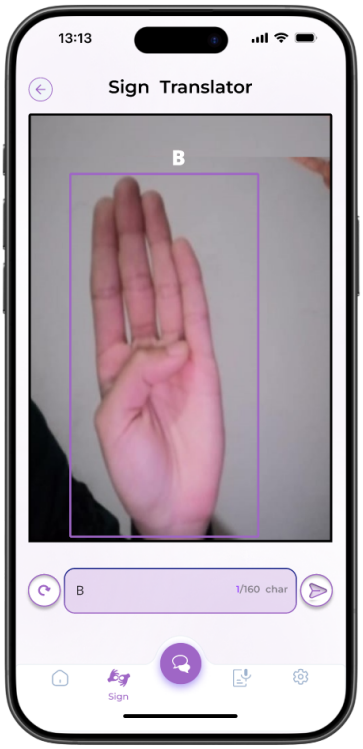
Start Sign Translator Page

Sign Translator

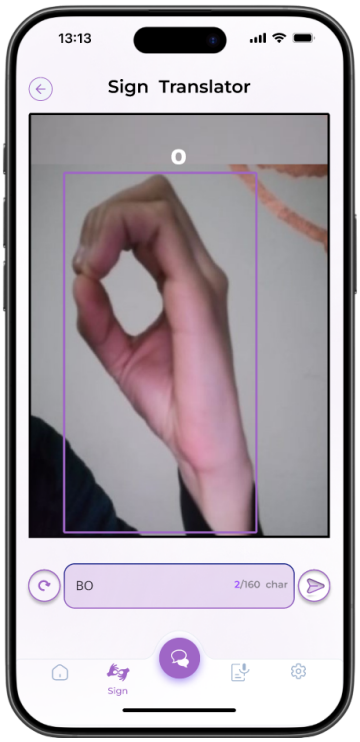


- **Example : Sign Translator**

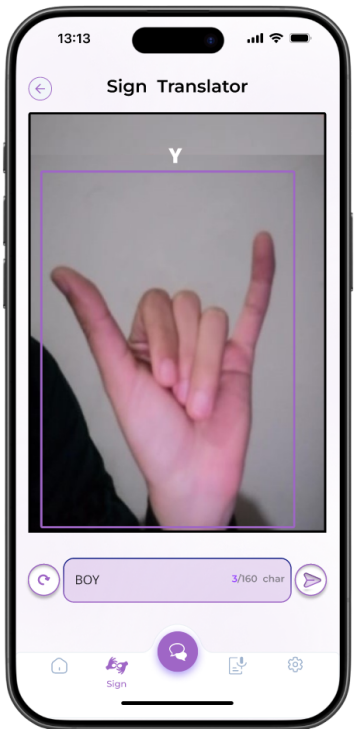
Letter B



Letter O



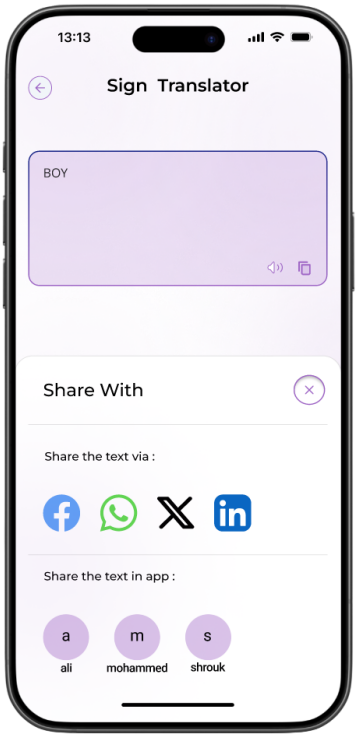
Letter Y



Translation

Options

Share



- **Word Translator**

Word Translator Page

Word Translator

Use Keyboard

Record Voice

Pause Voice

- **Example : Word Translator**

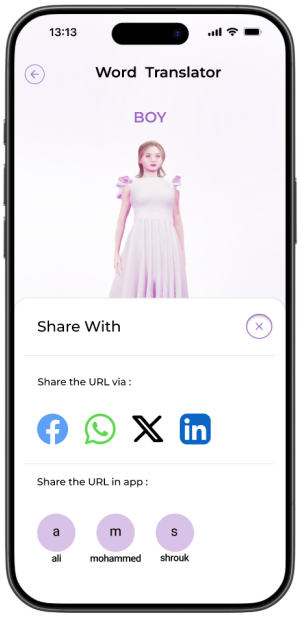
Letter B

Letter O

Letter Y

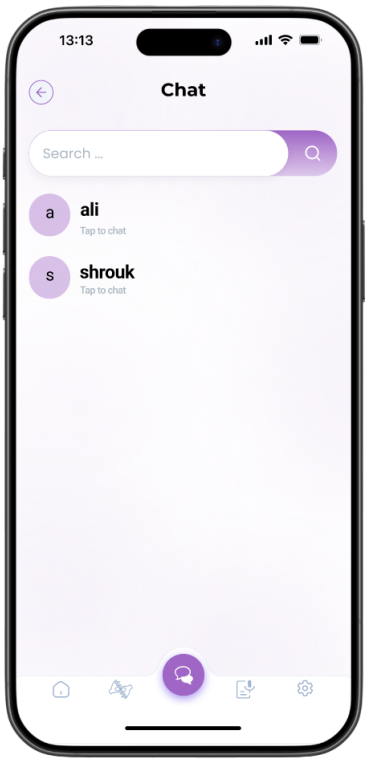
Translation

Share

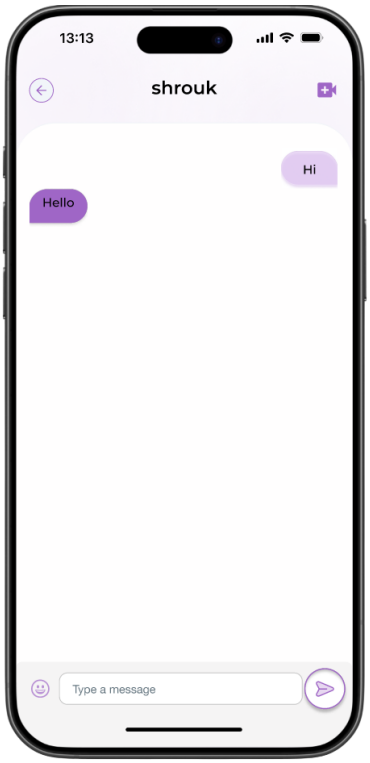


- **Chat**

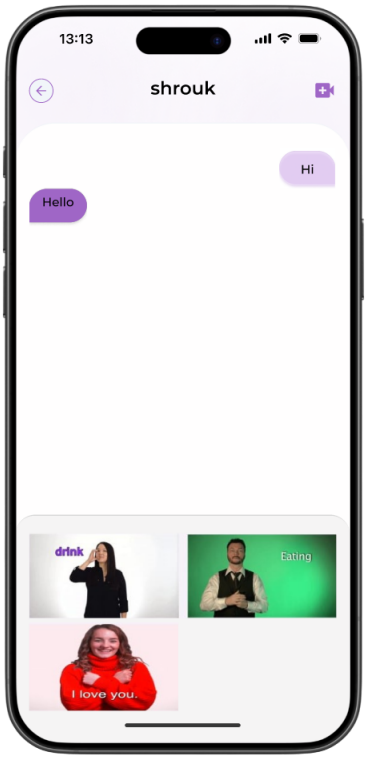
Search Page



Chatting Page



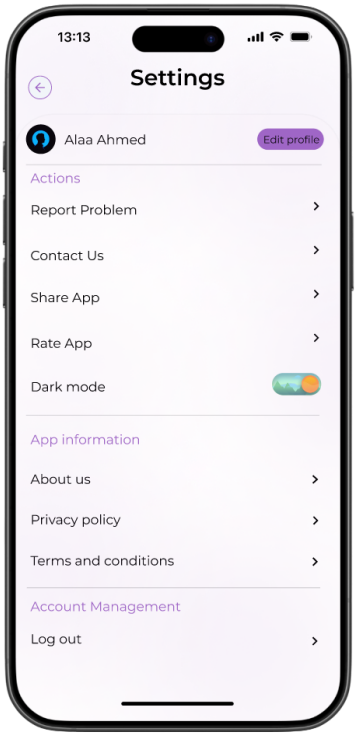
Gif in Chat



Video Call Page

- **Settings**

Settings Page



Edit Profile Page

Log out Page

Log out Page