**­­HANZIFY: A Sign Language Translation App**

**MINI PROJECT**

***Submitted******to***

**ASSAM DON BOSCO UNIVERSITY**

***by***

**SHIDHARTH LAISHRAM**

**DC2022BTE0131**

**KH DEEPAK KUMAR SINGHA**

**DC2022BTE0118**

**LAMJINGBA NINGOMBAM**

**DC2022BTE0095**

***in partial fulfilment for the award of the degree***

***of***

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**SCHOOL OF TECHNOLOGY**

**ASSAM DON BOSCO UNIVERSITY**

**AZARA, GUWAHATI 781017**

**ASSAM, INDIA.**

**BATCH (2022- 2026)**

**ACKNOWLEDGEMENT**

We extend our heartfelt gratitude and appreciation to all those who have successfully contributed to our project, **"Hanzify: A Sign Language Translation App"**.

First and foremost, we would like to express our sincere thanks to our project guide, **Mrs. Smriti Priya Medhi, Assistant Professor (Stage II), Department of Computer Science and Engineering,** for unwavering support, expert guidance, and invaluable mentorship throughout this project. Ma’am’s profound knowledge, patience, and dedication have been instrumental in shaping our ideas into a reality.

We are also immensely thankful to our project coordinators, **Mrs. Smriti Priya Medhi** and **Dr. Nupur Choudhury** for helping us in our project.

We would also like to thank **Dr. Bobby Sharma, Head of Department, Computer Science and Engineering, School of technology, Assam Don Bosco University** for providing us this opportunity and their support and encouragement throughout this project.

Thank you

Shidharth Laishram

DC2022BTE0131

Kh Deepak Kumar Singha

DC2022BTE0118

Lamjingba Ningombam

DC2022BTE0095

**DECLARATION**

We hereby declare that the project work entitled “**Hanzify: A Sign Language Translation App**” submitted to the Assam Don Bosco University, Guwahati, Assam, in partial fulfilment of the requirement for Mini Project of 6th semester of Bachelor of Technology. It is an original work done by me under the guidance of name of Mrs. Smriti Priya Medhi **(Assistant Professor (Stage II)**, Department of Computer Science and Engineering*, School of Technology, Assam Don Bosco University)* and has not been submitted for the award of any degree.

(Signature of the student)

Shidharth Laishram

DC2022BTE0131

Department of Computer Science and Engineering

(Signature of the student)

Kh Deepak Kumar Singha

DC2022BTE0118

Department of Computer Science and Engineering

(Signature of the student)

Lamjingba Ningombam

DC2022BTE0095

Department of Computer Science and Engineering

**CERTIFICATE**

This is to certify that the Project Report entitled “**Hanzify: A Sign Language Translation App**” submitted by **Shidharth Laishram (DC2022BTE0131), Kh Deepak Kumar Singha (DC2022BTE0118) and Lamjingba Ningombam (DC2022BTE095)** to Assam Don Bosco University, Guwahati, Assam, in partial fulfilment of the requirement for Mini project of 6th semester of Bachelor of Technology. It is a bonafide record of the project work carried out by them under my supervision during the year 2025

(Signature of the Internal Guide)

**Mrs. Smriti Priya Medhi,  
Assistant Professor (Stage II),  
Department of Computer Science and Engineering,   
School of Technology**

**Assam Don Bosco University**

Date: .......................................................

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| Sl. No. | Topic | Page No. |
| 1. | List of Figures | 1 |
| 2. | List of Tables | 2 |
| 3. | Abstract of the project | 3 |
| 4. | Introduction | 4 |
| 5. | Objectives | 6 |
| 6. | Background and Existing System/Literature | 7 |
| 7. | Problem Definition | 10 |
| 8. | Proposed Methodology | 12 |
| 9. | SDG Mapped | 13 |
| 10. | Feasibility Study | 14 |
| 11. | Hardware & Software Requirements | 16 |
| 12. | Proposed Plan | 18 |
| 13. | Dataset and AI Models For ML | 20 |
| 14. | Design Diagrams | 23 |
| 15. | Implementation | 26 |
| 16. | Future Scope and Limitations | 37 |
| 17. | Conclusion | 38 |
| 18. | References | 39 |
| 19. | Initial Comments & Updates | 40 |

**1. LIST OF FIGURES**

1. FIGURE 10.1: WORK BREAKDOWN STRUCTURE (WBS) – Pg No: 14
2. FIGURE 10.2: GANTT CHART – Pg No: 15
3. FIGURE 13.1: ASL (DATASET) – Pg No: 20-21
4. FIGURE 13.2: WORD SIGN LANGUAGE – Pg No: 21
5. FIGURE 13.3: NUMBERS ASL – Pg No: 21
6. FIGURE 14.1: USE CASE DIAGRAM – Pg No: 23
7. FIGURE 14.2: ACTIVITY DIAGRAM – Pg No: 24
8. FIGURE 14.3: BLOCK DIAGRAM – Pg No: 25
9. FIGURE 15.1: LOADING PAGE – Pg No: 26
10. FIGURE 15.2: HOME PAGE – Pg No: 26
11. FIGURE 15.3: WORD TO SIGN PAGE – Pg No: 27
12. FIGURE 15.4: SIGN TO WORD PAGE – Pg No: 27
13. FIGURE 15.5: REVIEW PAGE – Pg No: 28
14. FIGURE 15.6: CODE SNIPPET (APP DEVELOPMENT) – Pg No: 29-30
15. FIGURE 15.7: MODEL ACCURACY & MODEL LOSS OF SEQ. MODEL FOR ASL ALPHABETS – Pg No: 31
16. FIGURE 15.8: CONFUSION MATRIX OF SEQUENTIAL MODEL (ASL ALPHABETS) – Pg No: 31
17. FIGURE 15.9: MODEL ACCURACY & MODEL LOSS OF MOBILENETV2 FOR ASL ALPHABETS – Pg No: 32
18. FIGURE 15.10: CONFUSION MATRIX OF MOBILENETV2 (ASL ALPHABETS) – Pg No: 32
19. FIGURE 15.11: MODEL ACCURACY & MODEL LOSS OF MOBILENETV2 FOR WORDS – Pg No: 33
20. FIGURE 15.12: CONFUSION MATRIX OF MOBILENETV2 FOR WORDS –   
    Pg No: 33
21. FIGURE 15.13: MODEL ACCURACY & MODEL LOSS OF MOBILENETV2 FOR NUMBERS – Pg No: 34
22. FIGURE 15.14: CONFUSION MATRIX OF MOBILENETV2 FOR NUMBERS –  
    Pg No: 34

**2. LIST OF TABLES**

1. TABLE 6.1: TABLE OF EXISTING SYSTEMS – Pg No: 9
2. TABLE 13.4: CHALLENGES AND SOLUTION – Pg No: 22
3. TABLE 15.15: EXAMPLES OF SENTENCES AND ITS SIGN FORMAT – Pg No: 35-36
4. **ABSTRACT OF THE PROJECT**

Communication barriers faced by individuals with hearing and speech impairments remain a significant challenge in inclusive human interaction. This project presents **"Hanzify"**, a real-time Sign Language Translation mobile application designed to facilitate seamless communication between sign language users and non-signers. The application focuses on recognizing and translating **American Sign Language (ASL)** gestures into textual outputs using a deep learning approach.

Built on **TensorFlow** and employing the **MobileNetV2** architecture, the system achieves efficient and accurate hand gesture recognition while maintaining compatibility with mobile devices. The use of **MediaPipe Hands** enhances hand landmark detection and gesture tracking, enabling the system to interpret dynamic and static signs with minimal latency. The app is designed with an intuitive user interface, making it accessible for daily use and educational purposes. By combining machine learning, computer vision, and mobile technology, Hanzify contributes to breaking down communication barriers, fostering inclusivity, and empowering the hearing-impaired community through innovative technological solutions.

1. **INTRODUCTION**

In today’s fast paced, interconnected world, effective communication remains at the heart of human interaction. However, the linguistic barrier between individuals who use spoken language and those who rely on sign language for communication often limits understanding. Recognizing this critical challenge, we present Hanzify – a cutting-edge sign language translation app designed to bridge this communication gap and foster an inclusive society where everyone’s voice is heard, irrespective of their mode of expression.

This project focuses on creating a sophisticated, versatile application that not only simplifies the communication between individuals but also enhances the overall experience and bring upon inclusiveness among people.

Key Features of Hanzify:

1. **Real-Time Translation**: Translates sign language into spoken or written text and vice versa instantly, enabling seamless communication.
2. **Multi-Language Support**: Supports multiple sign languages, including **ASL (American Sign Language), BSL (British Sign Language)**, and more, catering to a global audience.
3. **Advanced AI and ML Technology:** Utilizes cutting-edge **gesture recognition, computer vision,** and **natural language processing** to ensure accurate and contextually relevant translations.
4. **User-Friendly Interface:** Designed for intuitive navigation, making it accessible for users of all technical skill levels.
5. **Multi-Platform Compatibility**: Works across various platforms, including smartphones and tablets, ensuring flexibility and convenience for users.
6. **Versatile Applications**: Suitable for diverse environments such as schools, workplaces, healthcare, facilities and public services.
7. **High Translation Accuracy**: Delivers precise translations with contextual understanding, setting it apart from other tools.
8. **Offline Mode**: Provides the ability to function in offline mode for areas with limited or no internet connectivity, ensuring continuous usability.
9. **Continuous Learning and Updates**: The app evolves with updates, enhancing recognition accuracy and incorporation user feedback.

By providing tools that promote accessible education, inclusivity, and equitable opportunities, the app contributes to building a society where no one is left behind. It enables individuals with hearing impairments to access information, participate in conversations, and engage in societal activities, thereby fostering a more sustainable and inclusive future.

With Hanzify, harnessing the power of machine learning and artificial intelligence technologies, communication has no bounds. It is not just a tool but a statement of commitment to innovation, and the belief that technology can make the world a better place for all.

1. **OBJECTIVES**

The objectives of Hanzify app mini project is:

1. To develop a robust machine learning model for real-time sign language detection.
   1. This includes preprocessing sign language gestures, extracting features, and training machine learning or deep learning algorithms to accurately classify gestures into corresponding meanings.
   2. The study will involve exploring image processing, pose estimation, and gesture recognition techniques to improve detection accuracy under diverse conditions (e.g., lighting, camera angles, and occlusions).
2. To create an application integrating the sign language detection model.
3. The application will transform detected sign language gestures into text and speech outputs, providing accessibility for both hearing and non-hearing users.
4. The system will include features like sign language translation, customizable language settings, and real-time feedback for enhancing user experience.

**6. BACKGROUND AND EXISTING SYSTEMS/LITERATURE**

**6.1 Background:**

Sign languages are natural languages that use manual gestures, facial expressions, and body postures to convey meaning. For individuals who are deaf or hard of hearing, sign language serves as a primary means of communication. However, the majority of the population, including often lack knowledge of sign language, creating a significant communication barrier that limits social inclusion, access to services, and educational opportunities for the hearing-impaired community.

The traditional methods of overcoming this barrier include the use of human sign language interpreters. It may be effective, but are often expensive, not always available, and can introduce latency in real-time communication. In recent years, technological advancements in **computer vision**, **machine learning**, and **mobile computing** have opened new avenues for building real-time translation systems that can automatically interpret sign language into spoken or written language.

The development of lightweight deep learning models like **MobileNetV2**, and frameworks like **MediaPipe**, has made it feasible to implement gesture recognition on smartphones and edge devices. These tools enable accurate detection and tracking of hand landmarks in real time with minimal computational overhead, making mobile-based solutions both practical and scalable.

This project seeks to address this need by developing a mobile application that captures hand gestures using the device’s camera, processes them through an optimized neural network, and translates the gestures into text or audio in real time.

**6.2 Existing Systems:**

Several systems and tools exist in the domain of sign language recognition and translation, each having their own strength and limitations. Some prominent existing systems are given below:

1. Google Translate (ASL Keyboard Integration)
   1. Google Translate provides limited support for sign languages, including features like an ASL keyboard for text input.
   2. Limitation: It lacks real-time video-based recognition and comprehensive sign language-to-speech capabilities.
2. ASL Translator by AppMySite
   1. A mobile app that translates text into sign language animations (signs).
   2. Limitation: Primarily one-directional and lacks real-time communication features.
3. Microsoft Azure Cognitive Services
   1. Includes features for gesture and speech recognition but does not directly provide end-to-end sign language translation.
   2. Limitation: Requires extensive customization and technical expertise for implementation.
   3. **Literature Review:**

The following literature outlines the current research, methodologies and gaps in the field of sing language translation:

1. “SIGN LANGUAGE TRANSLATOR - A NOVEL APPROACH FOR REAL-TIME SIGN LANGUAGE TRANSLATOR USING PYTHON AND MACHINE LEARNING” by Prof. Sujata Dake
   1. Summary: The paper introduces a real-time sign language translator using machine learning and computer vision to bridge communication gaps for the hearing impaired. The system employs advanced neural networks to recognize gestures and translate them into written language for better inclusivity.
   2. Findings: The model demonstrates high accuracy in recognizing diverse sign language gestures and effectively translates them in real time, enhancing accessibility and fostering communication.
   3. Research Gap: The study identifies a need for broader dataset inclusivity, improved temporal gesture recognition, and better integration across varying regional sign languages.
2. “ADVANCEMENTS IN SIGN LANGUAGE RECOGNITION: A COMPREHENSIVE REVIEW AND FUTURE PROSPECTS” by Bashaer A. AI Abdullah
   1. Summary: The paper provides a comprehensive review of advancements in sign language recognition technologies, emphasizing machine learning and deep learning models for gesture and context understanding.
   2. Findings: The study highlights the progress in accuracy, multi-language recognition, and adaptability across diverse environments, demonstrating potential for real-world applications.
   3. Research Gap: The gaps include limited datasets for training diverse sign languages, lack of contextual interpretation, and challenges in real-time integration for dynamic and complex environments.
3. “HAND GESTURE RECOGNITION IN INDIAN SIGN LANGUAGE USING DEEP LEARNING” by Harsh Kumar Vashisth, Tuhin Tarafdar
   1. Summary: The study explores hand gestures recognition for Indian Sign Language using deep learning, emphasizing the importance of accurate, real-time translation to improve communication for the deaf and mute communities.
   2. Findings: High Accuracy was achieved using CNNs. A dataset specific to Indian Sign Language gestures was developed. The proposed models were performed well compared to traditional methods.
   3. Research Gap: Limited focus on dynamic gestures and real-world lighting conditions. Lack of scalability for diverse sign language systems.

TABLE 6.1: TABLE OF EXISTING SYSTEMS

|  |  |  |
| --- | --- | --- |
| SL. NO. | SYSTEM NAME | CONS |
| 1. | Google Translate (ASL Keyboard) | Limited support for sign languages, including features like an ASL keyboard for text input. |
| 2. | ASL Translator | Primarily one-direction and lacks real-time features. |
| 3. | Microsoft Azure Cognitive Services | Requires extensive customization and technical expertise for implementation. |

* + - 1. **PROBLEM DEFINITION**

The communication gap between sign language users and non-sign language users is a persistent challenge that has far-reaching consequences in various spheres of life. Despite advancements in technology and awareness, this gap continues to create significant barriers, hindering equal opportunities and social integration. The lack of effective tools to facilitate real-time, contextually accurate translation between sign language and spoken languages exacerbates these issues.

Key problems include:

1. **Lack of Universal Access to Sign Language Translation**: While some tools exist, they are often region-specific, catering to a single sign language, such as ASL (American Sign Language) or BSL (British Sign Language). This limitation excludes users who rely on other sign languages, leaving a large portion of the global population underserved.
2. **Inadequate Real-Time Communication Solutions**: Effective communication often requires real-time interaction. Existing technologies struggle to provide instantaneous, context-aware translations that mimic natural human conversation. This shortfall makes communication cumbersome and less effective, particularly in dynamic environments like classrooms, hospitals, or customer service interactions.
3. **High Costs and Limited Availability**: Many available solutions are prohibitively expensive, making them inaccessible to individuals and organizations with limited financial resources. Moreover, these tools often require specialized hardware or training, further limiting their adoption.
4. **Exclusion in Critical Environments**: The absence of reliable translation tools means that individuals who rely on sign language face exclusion in crucial settings such as education, healthcare, and employment. For example, students may struggle to access quality education due to the lack of interpreters, while patients may face difficulties conveying their needs in medical emergencies.
5. **Lack of Awareness and Cultural Understanding**: Beyond technological barriers, a significant problem lies in societal attitudes and awareness. Many people lack basic knowledge about sign language and its importance, perpetuating misunderstandings and stigmas surrounding deaf communities.

Hanzify addresses these challenges by:

* Providing a comprehensive platform that supports multiple sign languages, making it accessible to a global audience.
* Leveraging cutting-edge machine learning and artificial intelligence to ensure real-time, contextually accurate translations.
* Offering an affordable, user-friendly solution that works across multiple devices, eliminating the need for specialized hardware.
* Bridging gaps in critical environments, empowering individuals to participate fully in education, employment, and healthcare.
* Promoting awareness and inclusivity by fostering better understanding and interaction between sign language users and non-users.

By tackling these problems head-on, Hanzify is not just a technological solution but a step toward a more inclusive and equitable society where communication is a right, not a privilege.

**8. PROPOSED METHODOLOGY**

The development of Hanzify, a real-time sign language translation application, involves the integration of computer vision and deep learning techniques for gesture recognition and natural language translation. The proposed methodology is structured in the following phases:

8.1 DATA COLLECTION & PREPROCESSING

* A dataset of American Sign Language (ASL) hand gestures is compiled using publicly available datasets and custom recordings.
* Each gesture sample is captured as an image frame and labelled accordingly.
* Resizing of images.
* Normalization of pixel values.
* Data augmentation (rotation, flip, zoom) to increase dataset diversity and improve model generalization.

8.2 HAND LANDMARK DETECTION USING MEDIAPIPE AND CV2

* **MediaPipe Hands** framework is employed to detect and track 21 hand landmarks in real-time from the camera feed.
* This lightweight pipeline enables high-performance gesture detection with minimal computational load.
* The (x, y, z) coordinates of each landmark are extracted and used as input features for the gesture recognition model.

8.3 FEATURE EXTRACTION AND MODEL TRAINING

* The extracted hand landmarks are passed into a deep learning model for classification.
* **MobileNetV2**, a lightweight and efficient CNN architecture, is fine-tuned for real-time classification on mobile devices.
* The model is trained using **TensorFlow**.
* A **Convolutional Neural Network (CNN)** or **LSTM** model is trained to recognize both static and dynamic signs based on sequences of landmark coordinates.

8.4 REAL-TIME TRANSLATION AND INTERFACE INTEGRATION

* The trained model is integrated into a **Flutter**-based mobile application.
* The app captures live camera input, detects hand gestures using MediaPipe, classifies them using the model, and translates them into textual output.

8.5 TESTING AND EVALUATION

* The system is evaluated on accuracy, user satisfaction/user experience.

**9. SDG MAPPED**

Beyond addressing communication barriers, Hanzify serves as a transformative tool that contributes meaningfully to multiple United Nations Sustainable Development Goals (SDGs):

* SDG 4 – Quality Education: Hanzify fosters inclusive and equitable quality education by enabling specially-abled individuals. Through real-time sign language translation and intuitive interaction interfaces, the app empowers these learners to fully participate in classrooms, digital learning environments, and informal education. This bridges the gap in access to quality education for individuals who are often left behind by traditional systems.
* SDG 10 – Reduced Inequalities: The app plays a vital role in reducing inequalities within and among populations by breaking down the communication divide between non-impaired and impaired users. Hanzify encourages inclusion, understanding, and collaboration, allowing differently-abled individuals to engage equally in academic, professional, and social contexts. By normalizing accessibility features and making them a part of everyday digital communication, the app helps eliminate systemic barriers and promotes social equity.
* SDG 11 – Sustainable Cities and Communities: By promoting accessibility and inclusivity at scale, Hanzify contributes to the development of sustainable, inclusive, and resilient communities. Its scalable architecture can be adapted across various urban services—from public transport to healthcare and education—making smart cities truly inclusive.

**10. FEASIBILITY STUDY**

**10.1 Work Breakdown Structure (WBS)**

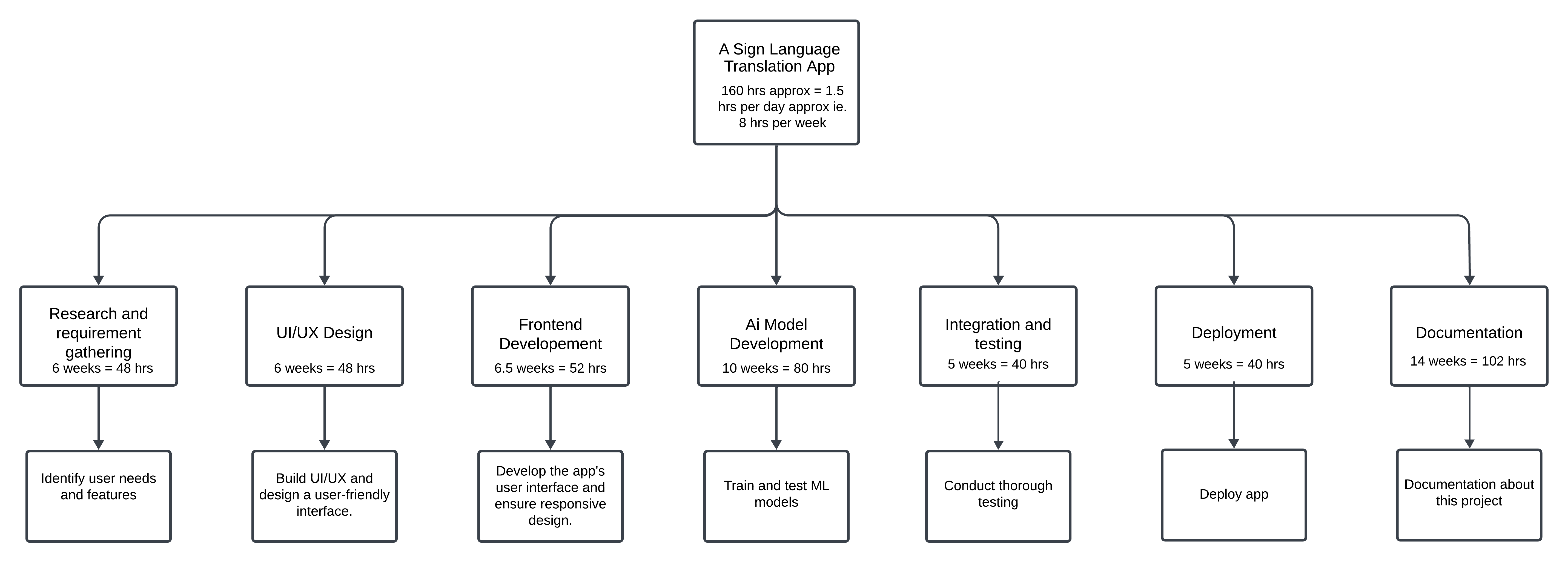


FIGURE 10.1: WORK BREAKDOWN STRUCTURE

The Work Breakdown Structure (WBS) for the Hanzify App provides a clear representation of all the essential phases involved in the development of the project, with corresponding working hours allocated to each phase. This structured approach helps to break down the project into manageable tasks and provides a comprehensive roadmap to ensure timely completion.

10.1.1 Calculation of total working hours:

1. Number of weeks in 5 months (January to May) = 20 Weeks

2. Number of working hours per week = 7-8 hours/week (let’s take 8)

3. Total working hours available = 8 hours/week x 20 weeks = 140 hours

4. No. Of working hour per day = 1 - 1.5 hours/day

**10.2 GANTT CHART**

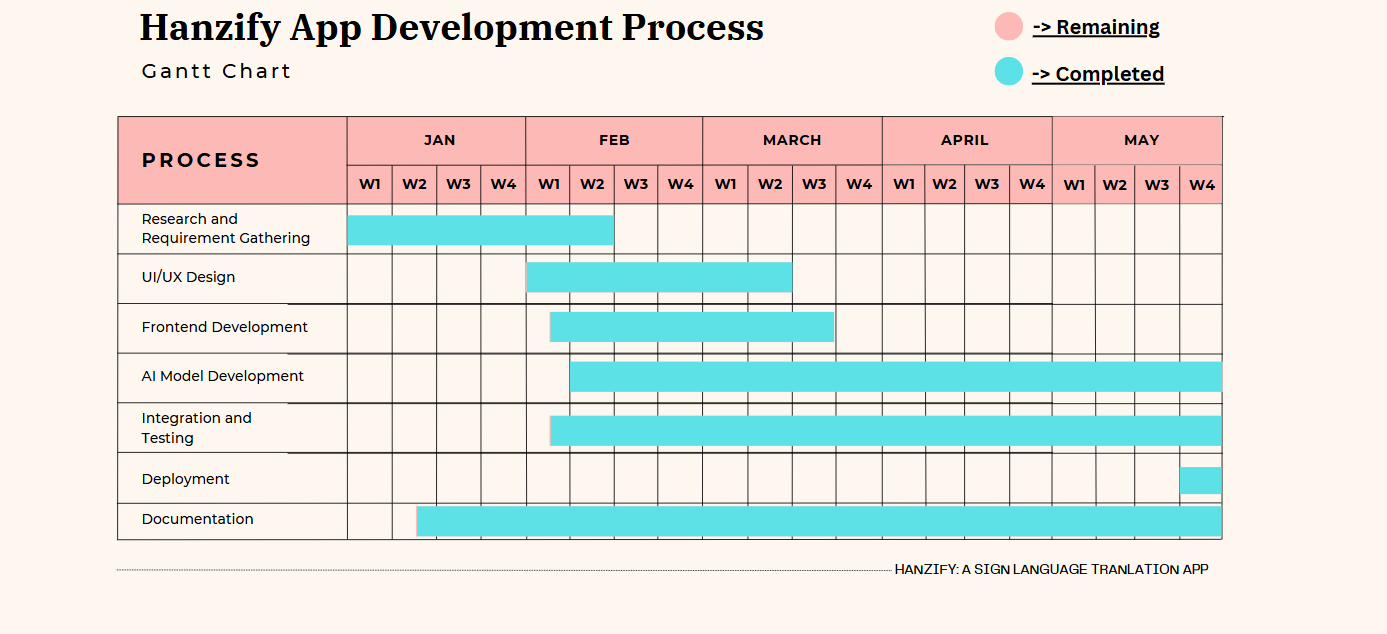


FIGURE 10.2: GANTT CHART

The diagram above illustrates the Gantt Chart for the Hanzify App Development Process, which represents the schedule and timeline for completing the various phases of the project. This chart serves as a visual tool for tracking progress, identifying dependencies, and ensuring that each task is completed within the allocated time frame.

1. **HARDWARE & SOFTWARE REQUIREMENTS**

**11.1 HARDWARE REQUIREMENTS**

* 1. For Development:
     + Processor: Intel i7 9th Gen or higher, AMD Ryzen 5 or higher.
     + RAM: 8 GB (minimum).
     + Storage: 512 GB SSD (recommended).
     + Graphics: Dedicated GPU for training machine learning models.
     + Display: 1920 x 1080 resolution.
     + Camera: Webcam for gesture and sign detection testing.
  2. For End Users:
     + Mobile Device: Android (7.0 or above).
     + Processor: Dedicated processor.
     + RAM: 2 GB (minimum).
     + Camera: Built-in camera for sign language detection.
     + Storage: 100 MB of free space for the app.

**11.2 SOFTWARE REQUIREMENTS**

1. For Development:
   * + Operating System: Windows 10/11.
     + Development Tools:
       1. IDE (Integrated Development Environment): VS Code, Android Studio & Google Collab for training model.
       2. OpenCV for computer vision processing.
       3. TensorFlow or PyTorch for AI model development.
     + Programming Languages: Dart (for app development) & Python (for AI & ML).
     + Framework & Libraries: Flutter.
     + Database: Firebase.
     + Version Control: Git & GitHub.
     + APIs & Libraries:
       1. TensorFlow.js for deploying AI models in mobile apps.
       2. MediaPipe for gesture detection.
       3. Natural Language Toolkit (NLTK) for NLP tasks.
2. For End-Users:
   * + Platform Compatibility: Android or iOS.
     + Permissions: Storage, camera, microphone.
     + Internet access: For real-time language updates (if required).
3. **PROPOSED PLAN**

**Phase 1: Research and Requirement Analysis (January)**

* + - * + Research Existing Solutions
  + Study current sign language translation tools
  + Identify their limitations and areas for improvement.
    - * + Define requirements:
  + Gather user requirements.
  + Finalize the list of supported sign languages (e.g., ASL, BSL etc.).
    - * + Finalize features:
  + Define key functionalities such as real-time translation, multi-language support, and contextual understanding.

**Phase 2: Design & Planning (February)**

* UI/UX Design:
  + Create wireframes of the app.
  + Ensuring the design is intuitive and accessible for all user abilities.
* Project Timeline:
  + Define milestones and allocate tasks using a Gantt Chart or WBS.
* System Architecture Design:
* Plan the app’s architecture, including front-end, back-end, and AI model components.

**Phase 3: Development (February-April)**

* AI/ML Model Development:
  + Build and train machine learning models for gesture recognition, natural language processing, and contextual analysis.
  + Ensure models are accurate and capable of handling diverse sign languages.
* Database Setup:
  + Design a database to store language data, user preferences, and translations.
* Core Application Development:
  + Develop front-end and back-end components.
  + Implement AI/ML models into the app.
  + Ensure compatibility with Android and iOS platforms.

**Phase 4: Integration and Testing (April-May)**

* Integration:
  + Combine AI/ML components, the database, and the app interface.
  + Ensure seamless interaction between modules.
* Testing:
  + Perform unit testing, integration testing, and user acceptance testing (UAT).
  + Address bugs and optimize performance for real-time functionality.

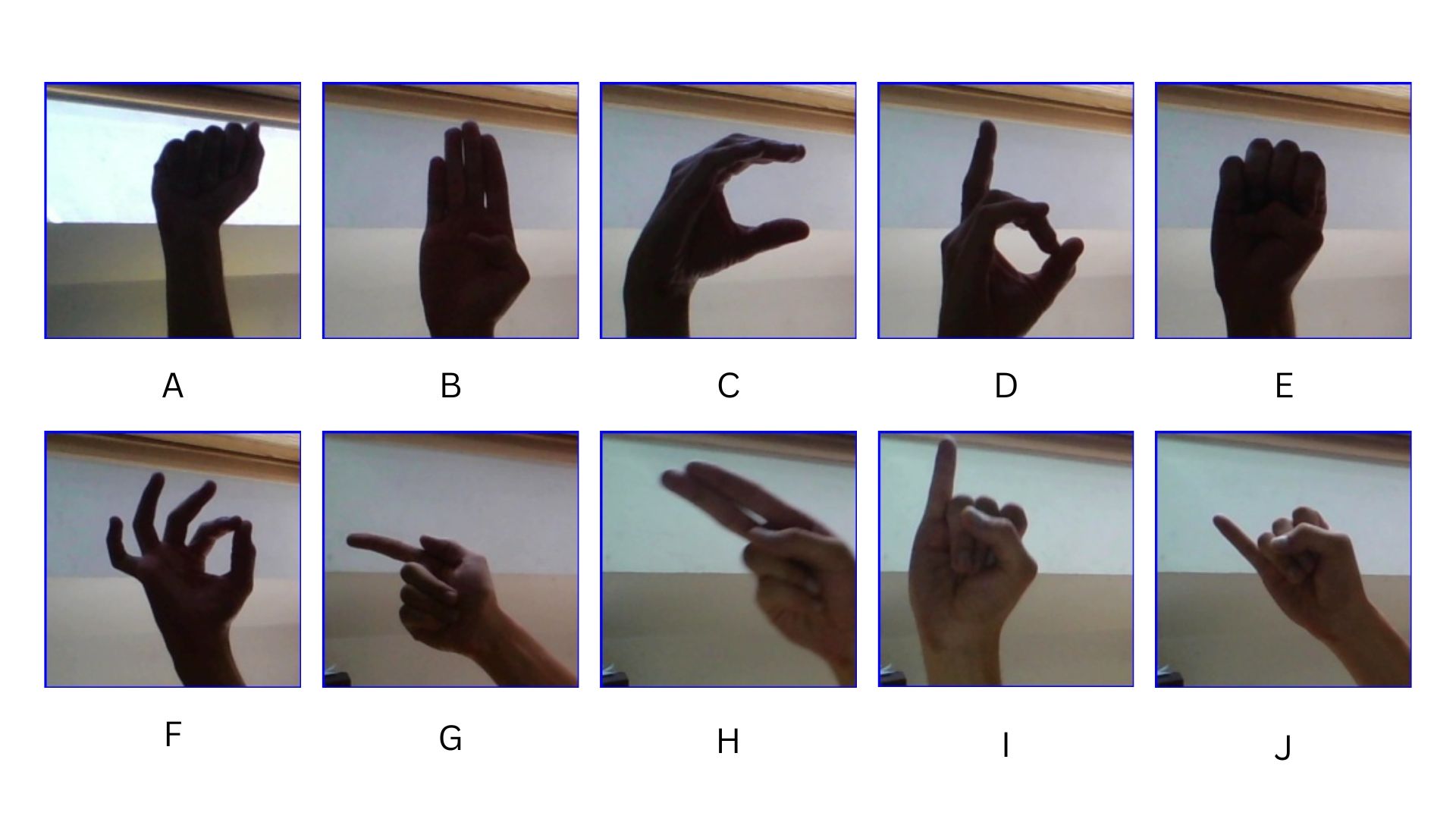
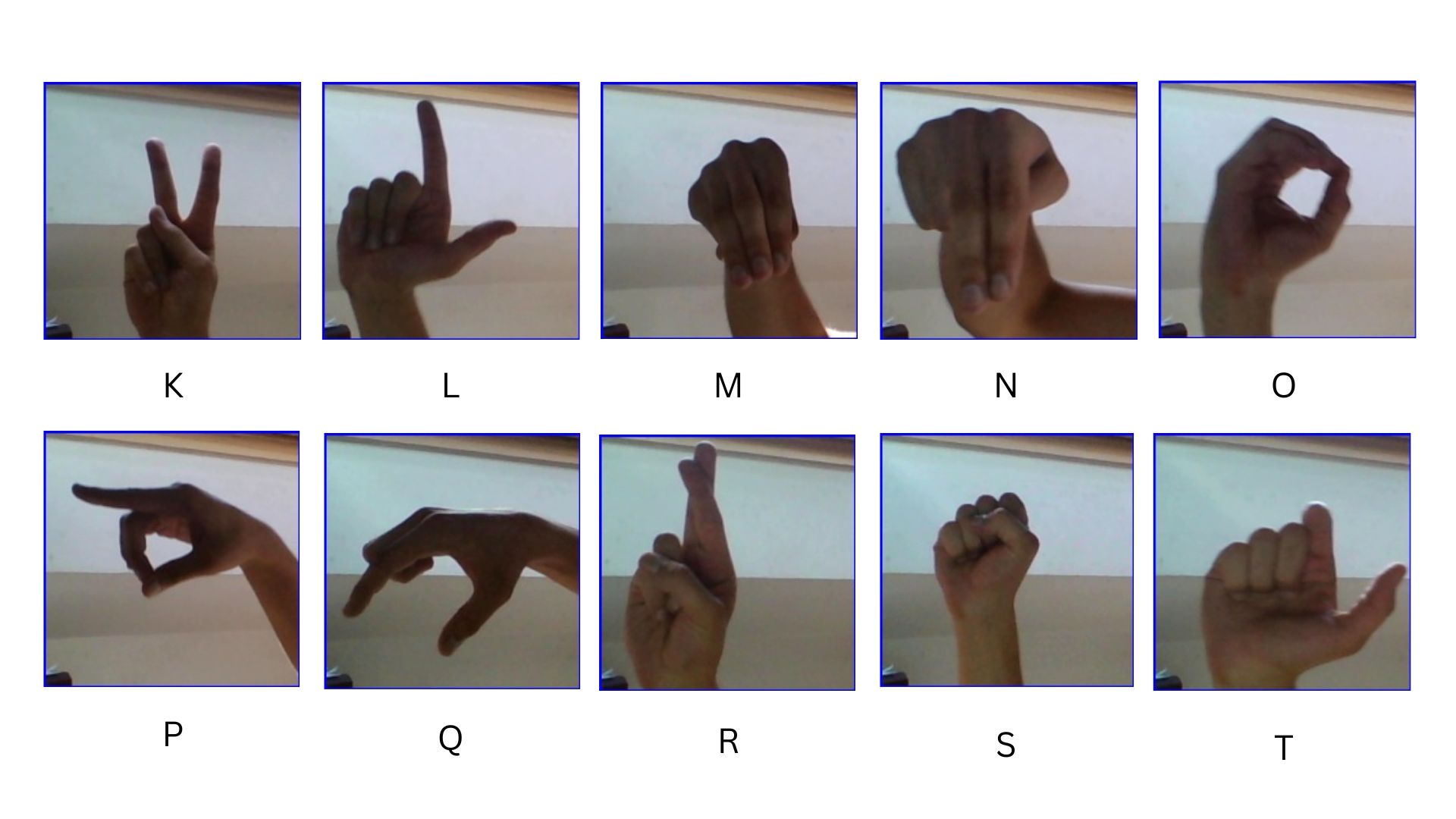
**Phase 5: Deployment (May)**

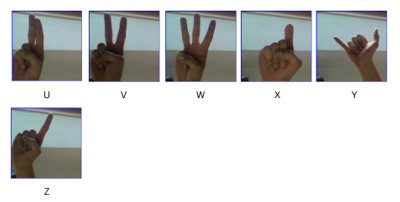
* Deployment:
  + Launch the app-on-app store (Google Play Store).
  + Ensure proper hosting and back-end support for real-time translation services.
* Feedback:
  + Gather user feedback through reviews.
  + Identify areas for improvement and plan future updates.

**13. DATASET AND AI-MODELS FOR ML**

13.1 Dataset of ASL (American sign Language)

This is the sample of the dataset for American Sign Language.



A white line drawing of a hand

AI-generated content may be incorrect.FIGURE 13.1: ASL DATASET

A set of hands with fingers extended

AI-generated content may be incorrect.FIGURE 13.2: WORDS SIGN LANGUAGE

FIGURE 13.3: NUMBERS ASL

13.2 AI Model for Sign Language Detection

Real-time translation of sign language into text has challenges, especially on low to mid spec devices. This project aims to develop a sign language detection system using MobileNetV2, an efficient deep-learning model optimized for mobile applications.

MobileNetV2 is a lightweight convolutional neural network (CNN) that provides high accuracy with minimal computational resources. The key reasons for choosing MobileNetV2 include:

* Efficiency**:** Optimized for mobile and edge devices with low latency.
* Small Model Size**:** Uses depthwise separable convolutions, reducing the number of parameters.
* Fast Inference**:** Provides real-time performance on mobile devices.
* High Accuracy**:** Maintains a balance between efficiency and accuracy, making it suitable for sign language recognition.

TABLE 13.4: CHALLENGES AND SOLUTION

|  |  |
| --- | --- |
| CHALLENGES | SOLUTION |
| High computational requirements | Use MobileNetV2, which is lightweight and optimized for mobile |
| Real-time processing delay | Apply quantization and use TensorFlow Lite for faster inference |
| Limited dataset availability | Augment existing datasets with variations in lighting, hand position, and background |

**14. DESIGN DIAGRAM**

14.1 Use Case Diagram

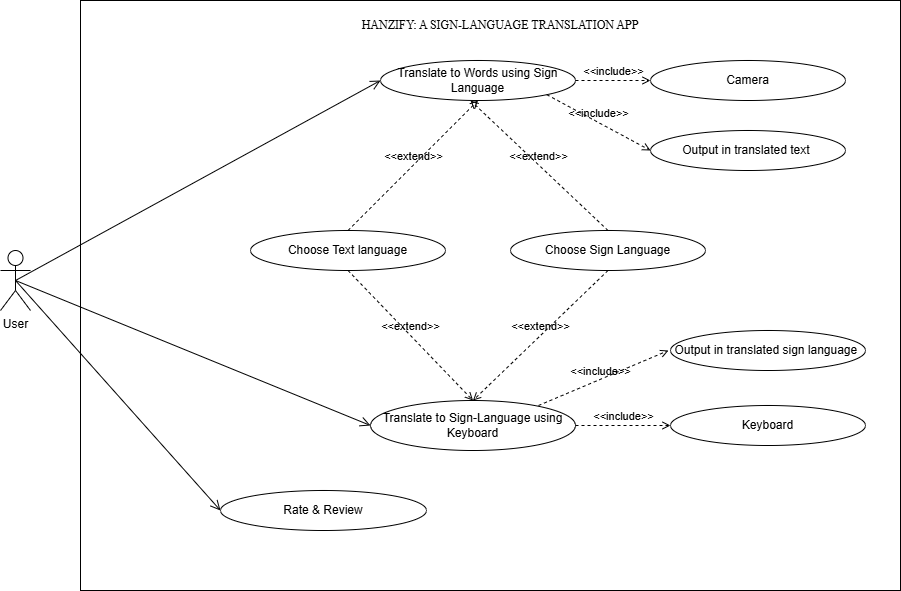


FIGURE 14.1: USE CASE DIAGRAM

14.2 Activity Diagram

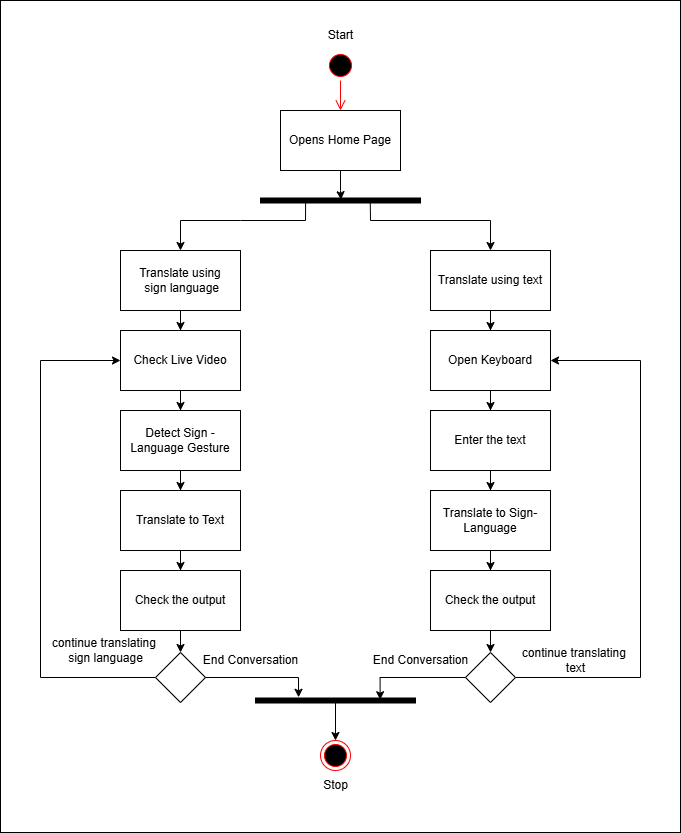


FIGURE 14.2: ACTIVITY DIAGRAM

14.3 Block Diagram

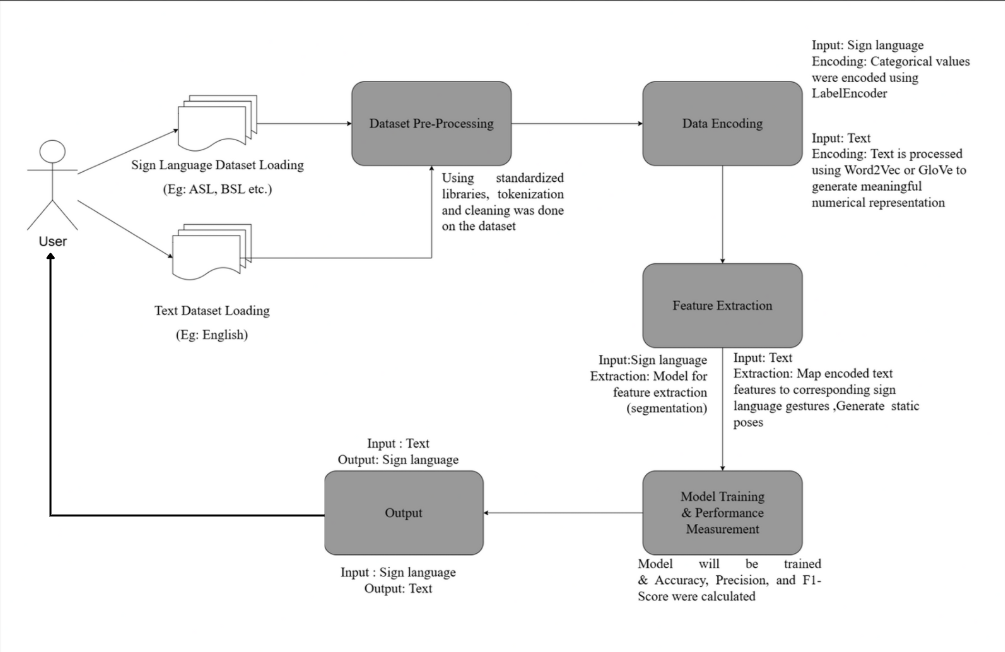


FIGURE 14.3: BLOCK DIAGRAM

**15.IMPLEMENTATION**

15.1 UI/UX Design

Created mock-ups for:

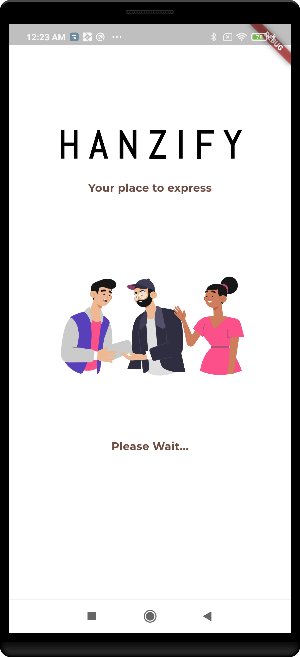
1. **Loading Page**:

FIGURE 15.1: LOADING PAGE

This is the starting page when the app opens.

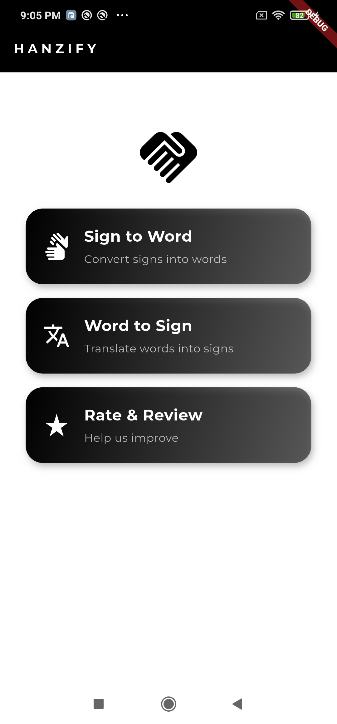
1. **Home page:**

FIGURE 15.2: HOME PAGE

* + This is the home page.
  + The user will be able to access the features – Sign to Word, Word to Sign and Rate & Review.

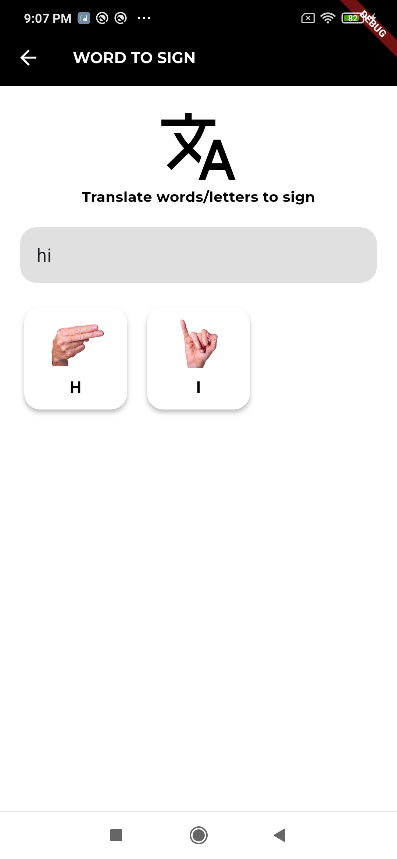
1. **Word To Sign page:**

FIGURE 15.3: WORD TO SIGN PAGE

1. **Sign To Word page:**

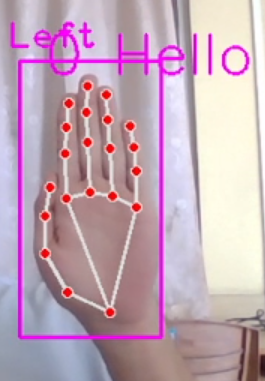


FIGURE 15.4: SIGN TO WORD PAGE

1. **Review page**:

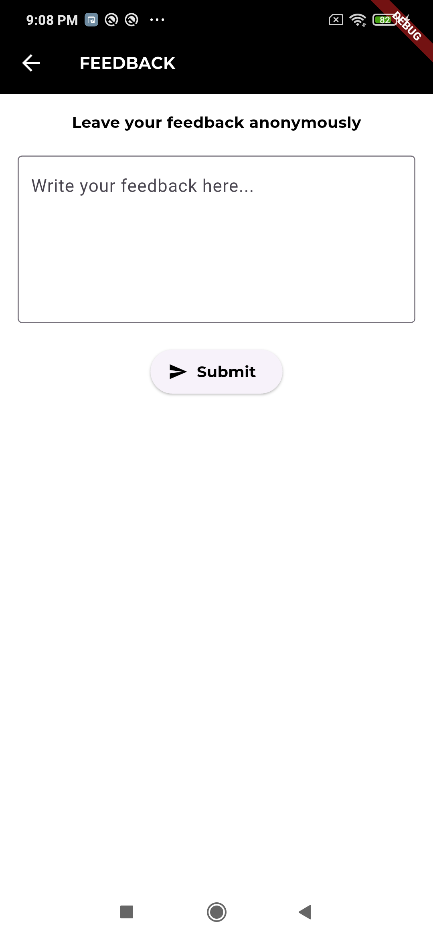
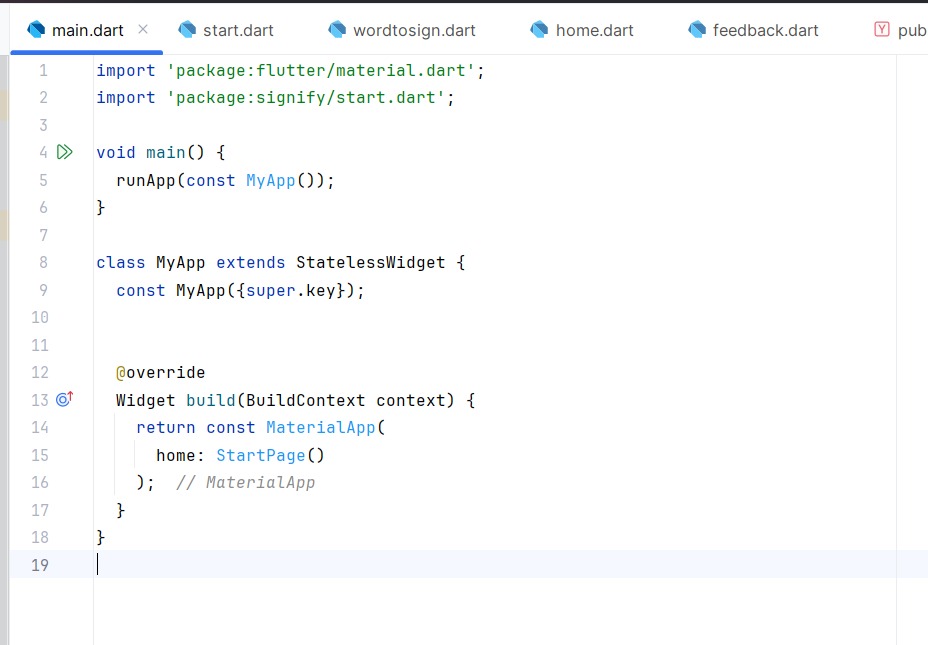
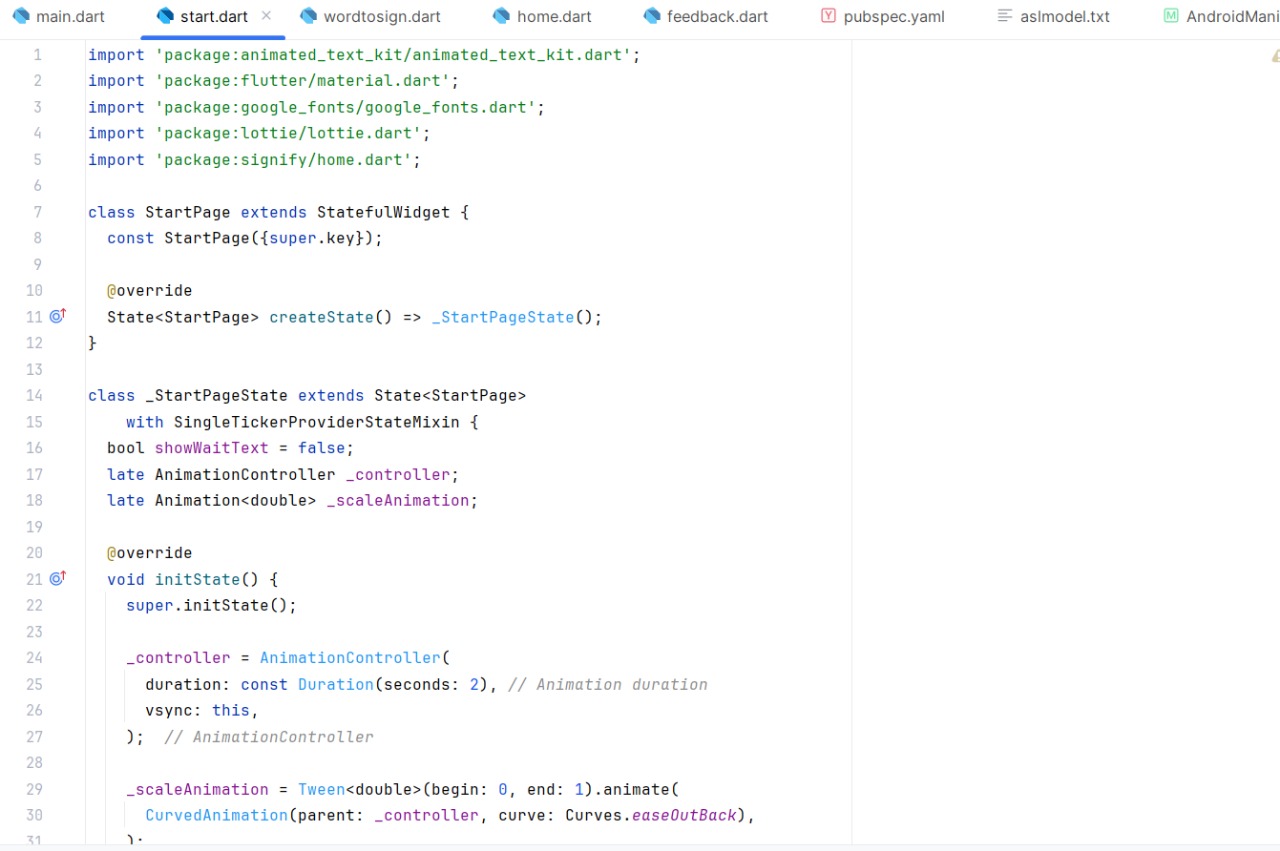


FIGURE 15.5: REVIEW PAGE

15.2 Code Snippet





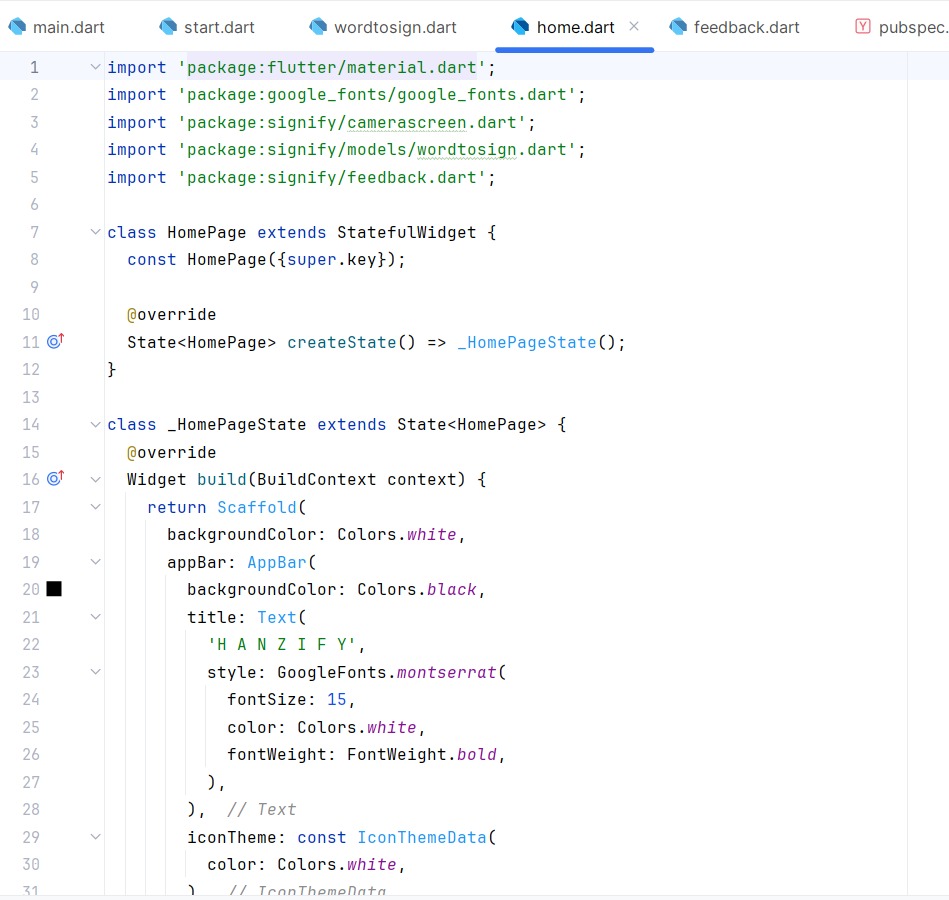


FIG 15.6: CODE SNIPPET (APP DEVELOPMENT)

15.3 MACHINE LEARNING MODEL:

We have chosen two models – Sequential Model and MobilenetV2 Model. We used and train the models using the ASL (American Sign Language) Dataset & Digits Dataset(0-9) from Kaggle.

Following is the outcomes or results for ASL Dataset, we have achieved after training the model:

A graph of a function

AI-generated content may be incorrect.15.3.1: SEQUENTIAL MODEL:

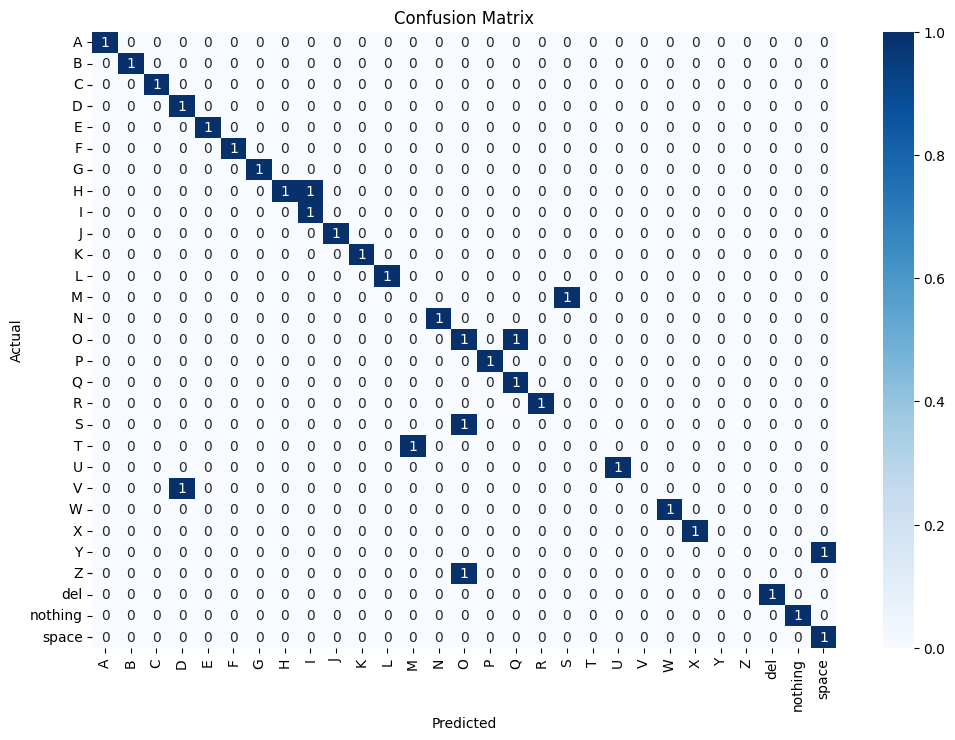
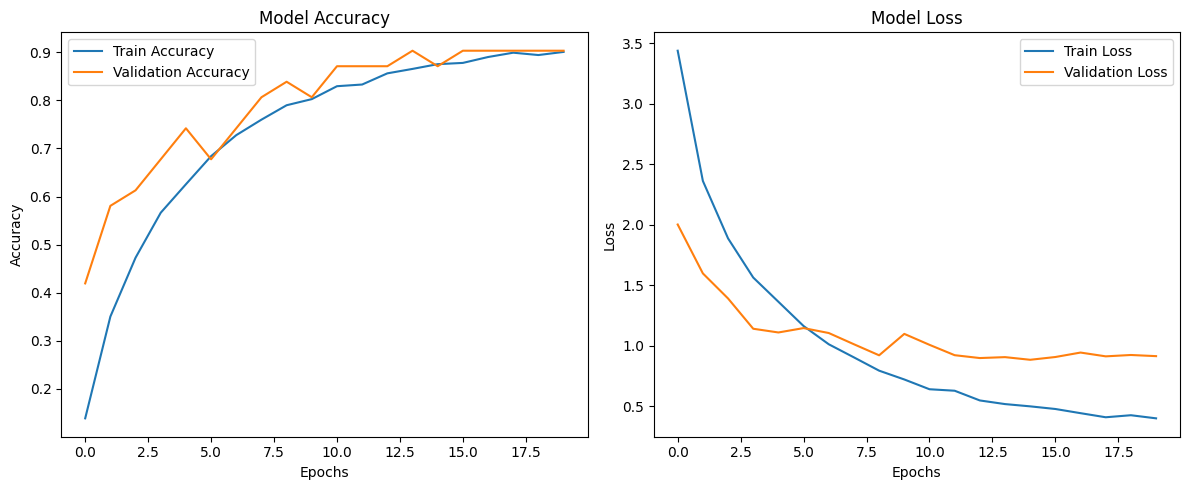
FIGURE 15.7: MODEL ACCURACY & MODEL LOSS OF SEQUENTIAL MODEL FOR ASL ALPHABETS

FIGURE 15.8: CONFUSION MATRIX OF SEQUENTIAL MODEL (ASL ALPHABETS)

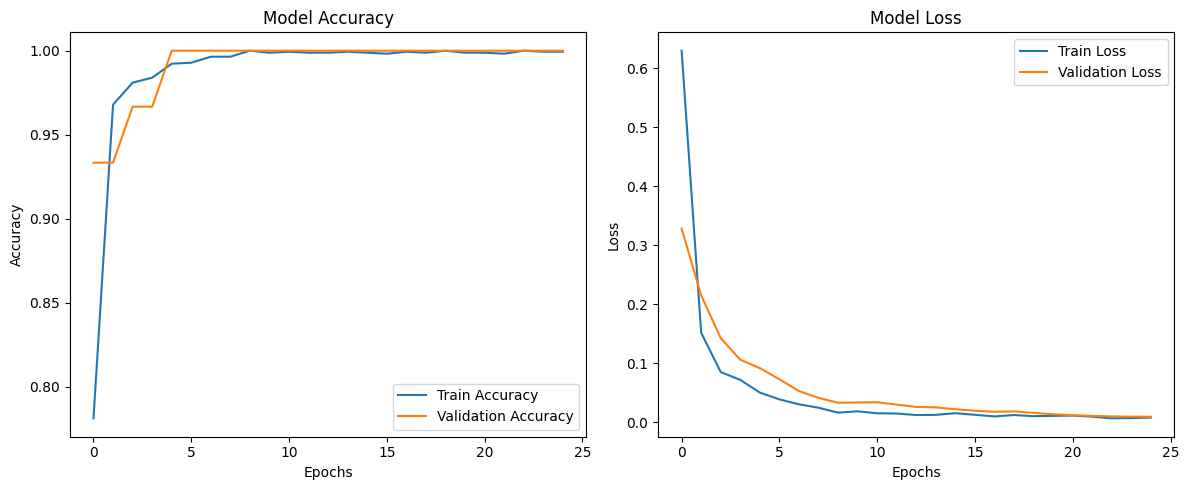
15.3.2: MOBILENETV2 MODEL:



A graph of confusion matrix

AI-generated content may be incorrect.FIGURE 15.9: MODEL ACCURACY & MODEL LOSS OF MOBILENETV2 MODEL FOR ASL ALPHABETS

FIGURE 15.10: CONFUSION MATRIX OF MOBILENETV2 (ASL ALPHABETS)



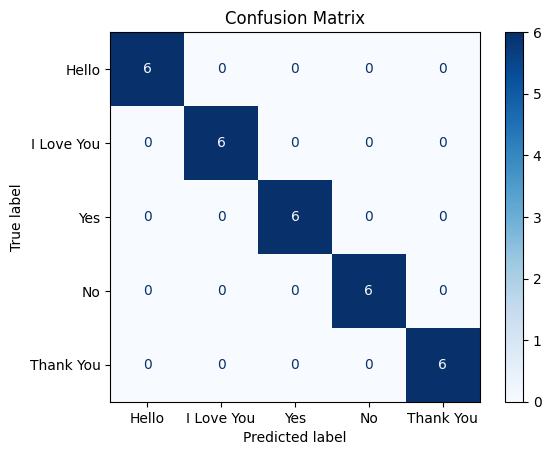
FIG 15.11: MODEL ACCURACY & MODEL LOSS OF MOBILENETV2 FOR WORDS

FIG 15.12: CONFUSION MATRIX OF MOBILENETV2 FOR WORDS

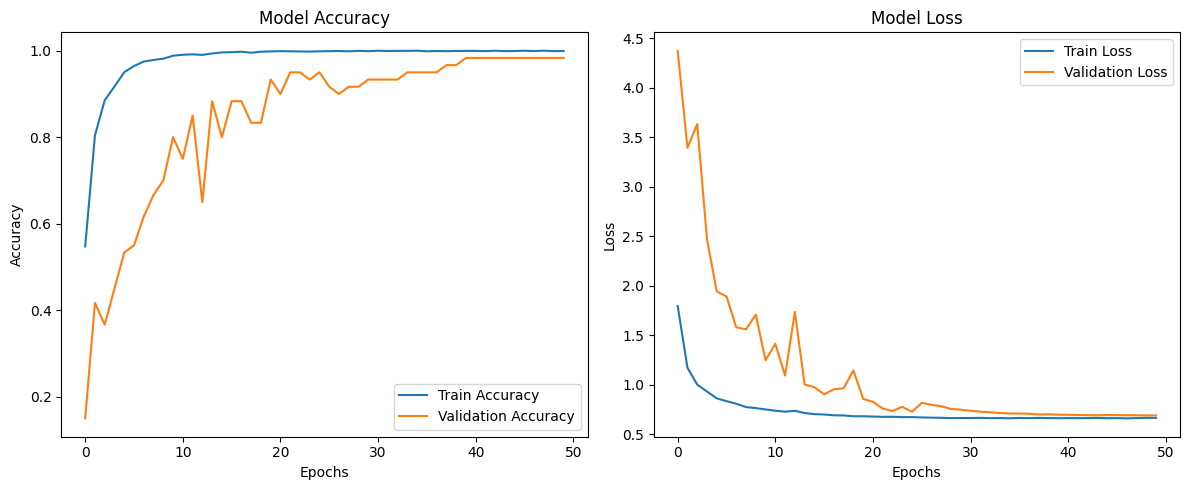
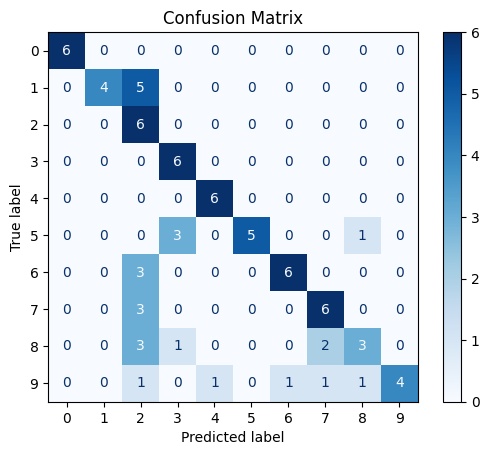
FIG 15.13: MODEL ACCURACY & MODEL LOSS OF MOBILENETV2 FOR NUMBERS

FIG 15.14: CONFUSION MATRIX OF MOBILENETV2 FOR NUMBERS

Upon training and observation of the accuracy-loss graph, and the confusion matrix, the MobileNetV2 Model is performing better than Sequential Model. Hence, the MobilenetV2 Model will be carried forward with the development.

TABLE 15.15: EXAMPLES OF SENTENCES AND IT’S SIGN FORMAT

|  |  |  |
| --- | --- | --- |
| S.NO. | LETTER/WORD/NUMBER | SIGN FORMAT |
| 1. | 0(ZERO) | A person holding a hand  AI-generated content may be incorrect. |
| 2. | 3(THREE) | A hand with red dots and points on it  AI-generated content may be incorrect. |
| 3. | HELLO | A hand with red dots on it  AI-generated content may be incorrect. |
| 4. | YES | A hand with red dots and white lines  AI-generated content may be incorrect. |
| 5. | THANK YOU | A person with a hand over their mouth  AI-generated content may be incorrect. |
| 6. | I LOVE YOU | A hand with red dots and a white border  AI-generated content may be incorrect. |
| 7. | NO | A hand with red dots and a white line  AI-generated content may be incorrect. |
| 8. | 9(NINE) |  |
| 9. | ADBU |  |
| 10. | IS |  |

**16. FUTURE SCOPE & LIMITATIONS**

FUTURE SCOPE:

The Hanzify application presents a promising foundation for inclusive communication. Its future enhancements can significantly increase its usability, accuracy, and global impact:

1. Multi-language Sign Support: Extending beyond American Sign Language (ASL).
2. Facial Expression and Body Pose Integration: Many sign languages rely on facial expressions and body posture.
3. Animated Sign Avatars: Implement module that generates animated signing avatars.

LIMITATIONS:

Despite its strengths, the current version of Hanzify faces several limitations that need to be addressed in future iterations:

Limited Vocabulary: The model currently supports a restricted set of signs, which limits its usefulness in complex conversations.

Gesture Ambiguity: Similar gestures or poor lightning conditions can lead to misclassification especially in dynamic or overlapping hand movements.

Dependence on Camera Quality: The accuracy of gesture recognition is affected by the camera resolution, background clutter and lighting.

Single-Handed Recognition Focus: Initial versions may only support single-hand gestures.

Real-time Performance Trade-offs: Optimizing for mobile real-time performance may involve compromises on model complexity and accuracy.

**17. CONCLUSION**

In essence, the Hanzify project represents a significant step toward enhancing inclusivity, accessibility, and social integration through innovative technology. As a sign language translation app, it addresses critical communication barriers faced by the deaf and hard-of-hearing communities while promoting a more equitable society. Below are the key takeaways and achievements from this project:

1. Bridging the communication gap
   1. Hanzify successfully facilitates real-time translation between sign language and spoken or written text, enabling seamless communication between sign language users and non-users.
   2. The app supports multiple sign languages, catering to a global audience and fostering universal inclusivity.
2. Technological Innovation
   1. By leveraging advanced technologies such as machine learning, gesture recognition, and natural language processing, Hanzify achieves high levels of accuracy and contextual relevance in translations.
   2. The app’s ability to function on standard devices ensures accessibility without requiring specialized hardware.
3. Practical Impact Across Domains
   1. Hanzify empowers users in critical sectors such as education, healthcare, and workplaces, allowing for greater participation and opportunities.
   2. It provides an affordable, scalable solution that can be widely adopted by individuals, organizations, and public services.
4. Impactful Contribution to Society
   1. Through Hanzify, the project demonstrates how technology can break down barriers, create opportunities, and empower marginalized communities.
   2. It stands as a testament to the power of innovation in addressing real-world challenges.
5. Scalability and future potential
   1. The modular and future-proof design of Hanzify allows for continuous updates, including the integration of additional sign languages, enhanced features, and broader applications.

At last, the completion of Hanzify is not just the realization of a technical achievement but a meaningful contribution toward building an inclusive society. This project underscores the importance of combining technology with empathy to create solutions that truly matter. As it evolves, Hanzify has the potential to inspire broader adoption of inclusive practices and technologies, transforming lives and fostering equality for generations to come.

**18. REFERENCES**

**[1]** R. Mall, *Introduction to Software Engineering*. New Delhi, India: PHI Learning, 2018.

**[2]** S. Haykin, *Neural Networks and Learning Machines*, 3rd ed. Upper Saddle River, NJ, USA: Pearson, 2009.

**[3]** S. Uma, *Fundamentals of Machine Learning*. Chennai, India: McGraw Hill Education, 2021.

**[4]** A. G. Howard *et al*., “MobileNetV2: Inverted Residuals and Linear Bottlenecks,” *arXiv preprint arXiv:1801.04381*, 2018. [Online]. Available: <https://arxiv.org/abs/1801.04381>

**[5]** Grassknoted, “ASL Alphabet Dataset,” Kaggle, [Online]. Available: <https://www.kaggle.com/datasets/grassknoted/asl-alphabet>

**[6]** A. Arda Mavi, “Sign Language Digits Dataset,” GitHub, [Online]. Available: <https://github.com/ardamavi/Sign-Language-Digits-Dataset/tree/master/Dataset>

**19. INITIAL COMMENTS & UPDATES**

Confusion matrix in Fig. 24 (previously Fig. 21) has data for 1, 5, 7.

Title for Fig. 18 and Fig. 19 (previously Fig. 19 and 21) is updated.

Fig. 20 (previously Fig. 17) is learning alphabet T.

Included Examples of Sentences and it’s Sign Format in a table (refer TABLE 15.15)