Sign2Text

Project Idea:

We will develop a system that recognizes hand signs and converts them into text subtitles. Later, we plan to enhance the system to support real-time camera input

Github repo

Updated-Github repo

Tools and libraries:

- Python
- Jupyter Notebook
- NumPy, Pandas
- Matplotlib
- OpenCV
- TensorFlow / Keras
- Scikit-learn
- Git & GitHub

....

Project Team Members:

- Zeinab Osama
- Hana Ibrahim
- Abdelrahman Darwish
- Ahmed Saeed
- Amr Fouad
- Mahmoud Hallul

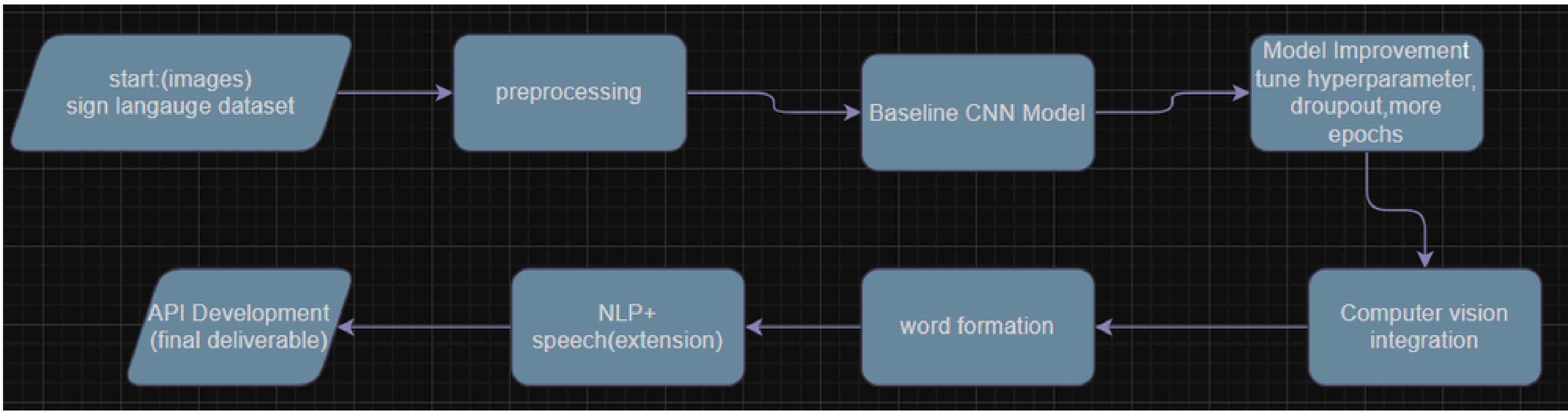
Team Leader:

- **Zeinab Osama**

Final Project Deliverable

- The final deliverable will be a web-based application (website).
- The website will integrate computer vision to capture images in real-time (via webcam) or accept uploaded images.
- Captured/Uploaded images will be preprocessed and passed to the trained CNN model to recognize sign language letters.
- Recognized letters will then be combined into words using simple rules, and the final output will be displayed directly on the website interface

Machine Learning Pipeline:



start:(images)
sign langauge dataset

preprocessing

Baseline CNN Model

Model Improvement
tune hyperparameter,
droupout,more
epochs

Computer vision
integration

word formation

NLP+
speech(extension)

API Development
(final deliverable)

Dataset source

Timeline Deliverables

Week 1 & 2

Weeks deliverables(project-wise):

- Dataset collected & confirmed suitability
- Environment setup (Python, TensorFlow, Keras, OpenCV, Scikit-learn)
- Initial preprocessing experiments (resize, grayscale, normalization, augmentation)
- Dataset split into train/test sets
- Evaluation metrics decided (accuracy, confusion matrix)

Team Members' Roles:

- Hana, Zeinab, Amr :
  - Work together on preprocessing.
  - Try different methods (resize, grayscale, normalization, augmentation) and compare which one gives better results.
- Mahmoud :
  - Handle dataset splitting (train/test) and prepare evaluation setup (accuracy, confusion matrix).
- Abdelrahman & Ahmed :
  - Focus mainly on making sure the environment is fully set up and running smoothly for everyone (Python, TensorFlow, Keras, OpenCV, Scikit-learn). They can guide the rest of the team so we all get started quickly.

Week 3&4

Weeks deliverables(project-wise):

- Trained CNN model (saved & documented).
- Live webcam demo showing predictions.
- Updated GitHub repo with clear documentation

Team Members’ Roles:

- Hana, Zeinab, Abdelrahman , Ahmed:
  - Computer vision (webcam + model + predictions)
  - Deliverable: demo with live predictions
- Mahmoud &□ Amr :
  - Model training & evaluation
  - Save best-performing models (.h5 + .keras)
  - Document results (accuracy/loss comparison)
- Team Leader (Zeinab)
  - Organize GitHub repo and add notes for each experiment
  - Maintain README with instructions
  - Ensure deliverables are tracked and ready for review

Week 5&6

Team Members’ Roles:

- Mahmoud, Amr, Ahmed :
  - Will built a simple static web page
- Zenab, Hana, Abdelrahman:
  - Still working on the accuracy and predictions in computer vision

Week 7&8 :

Weeks Deliverables (project-wise):

- Improve model performance and prediction stability.
- Fix the random behavior in the video capture module.
- Update GitHub repository with final working version and documentation.

Team Members’ Roles:

- Mahmoud & Amr:
    - Fine-tune the CNN model to enhance accuracy and reduce prediction errors.
    - Re-evaluate model performance on validation and real-world data.
    - Save and document the updated model versions.
- Zeinab, Hana, Abdelrahman, Ahmed:
    - Debug and optimize the video capture and live prediction modules.
    - Integrate the updated model into the web interface.
    - Conduct testing and prepare a final demo showing stable, accurate real-time results.

Week 9&10 :

Weeks Deliverables (project-wise):

- Collected our own dataset manually instead of using a ready-made one.
- Successfully solved the overfitting problem in the model.
- Deployed the final working version of the model on a static webpage.

Team Members’ Roles:

- Mahmoud & Amr:
    - Data collection and preparation
- Zeinab, Hana:
    - Model tuning and evaluating
- Ahmed & Abdelrahman:
    - Web deployment and testing

Weeks 11 & 12: Summary of Achievement and Deployment

Weeks Deliverables : FINALLY

Our Custom Data is Ready: We finished collecting and preparing our entire custom dataset. We are no longer relying on any external data.
Model Excellence: We successfully solved the Overfitting problem. The model now delivers a very high and reliable Accuracy on all data.
Robust Management with MLflow: We implemented MLflow to organize all our experiments. The final model is saved
Final Launch: The final, working version of the model has been deployed on a simple static webpage

Team Members' Roles:

- Mahmoud & Amr

Data & MLflow Setup

Completed all Data Collection and Preparation. Set up and assisted with MLflow Tracking.

- Zeinab & Hana

Model Optimization

Tuned the model and resolved the Overfitting issue. Ensured high model accuracy

- Ahmed & Abdelrahman

Deployment & Testing

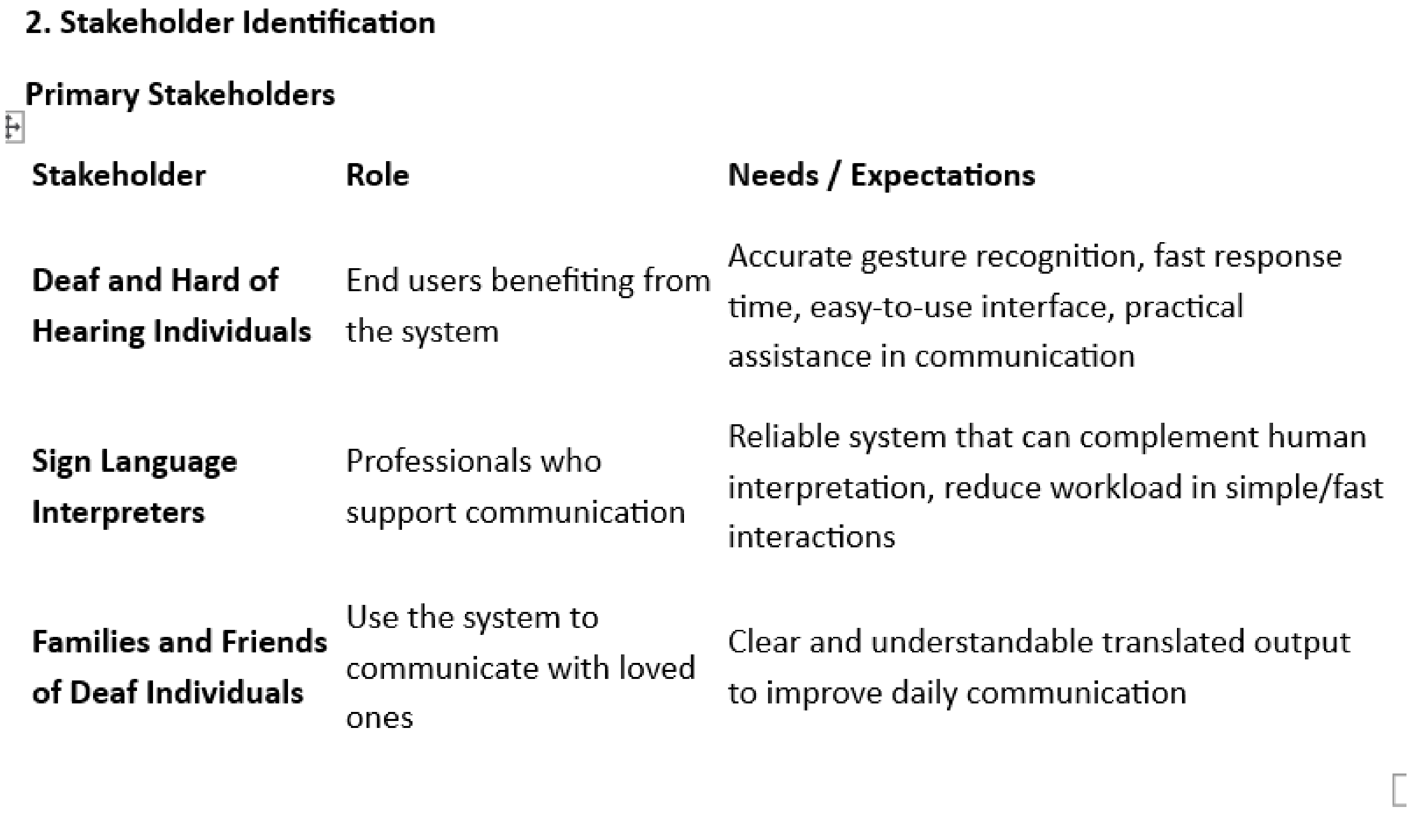
Handled the model's Deployment onto the webpage. Ensured everything works efficiently on the site.

Stakeholder Analysis

1. Overview

The Sign Language Detection System aims to recognize hand gestures representing alphabet letters in order to support individuals who are deaf or hard of hearing. The project provides an accessible communication tool that translates sign language into readable text, helping improve interaction between signers and non-signers.

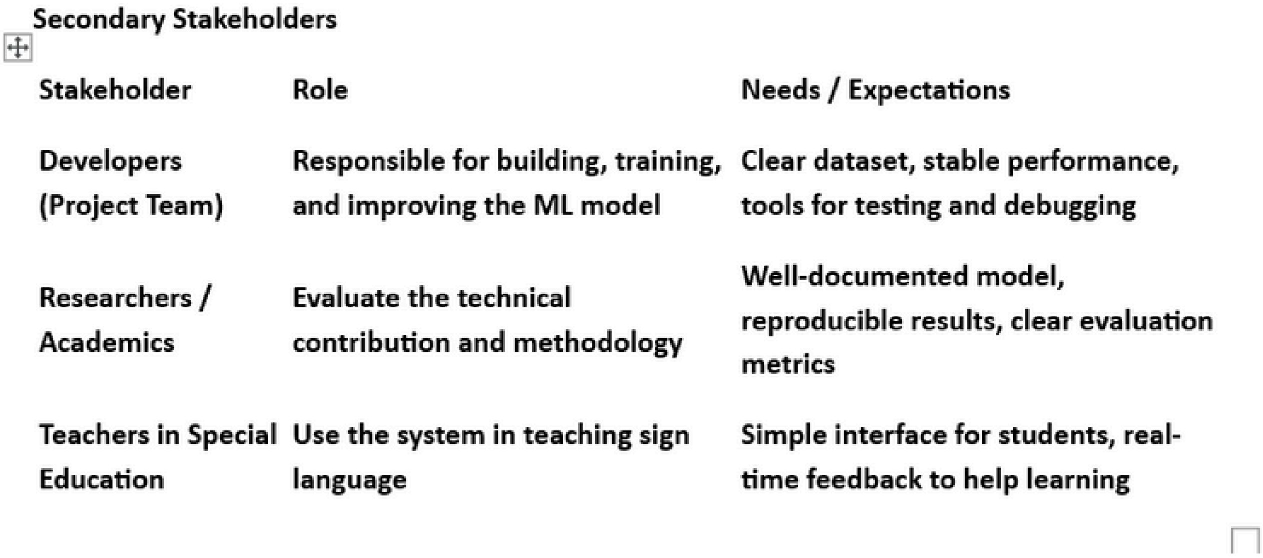
In this section, the stakeholders involved in the project are identified, along with their roles, needs, and expectation



2. Stakeholder Identification

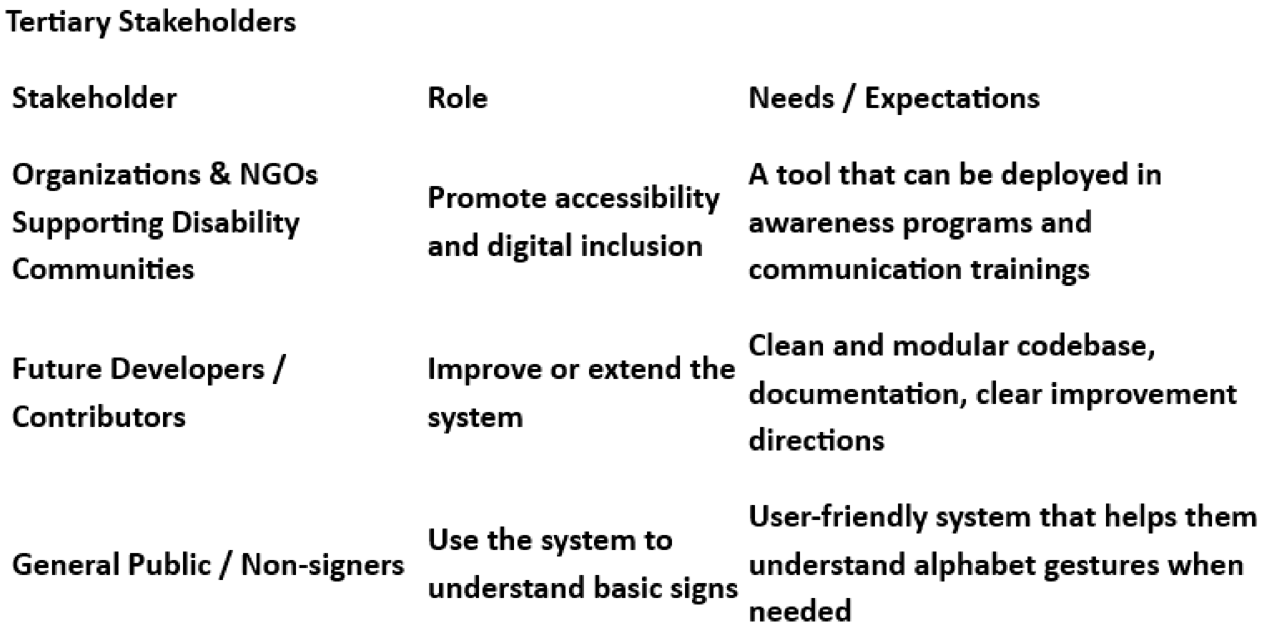
Primary Stakeholders

Stakeholder	Role	Needs / Expectations
Deaf and Hard of Hearing Individuals	End users benefiting from the system	Accurate gesture recognition, fast response time, easy-to-use interface, practical assistance in communication
Sign Language Interpreters	Professionals who support communication	Reliable system that can complement human interpretation, reduce workload in simple/fast interactions
Families and Friends of Deaf Individuals	Use the system to communicate with loved ones	Clear and understandable translated output to improve daily communication



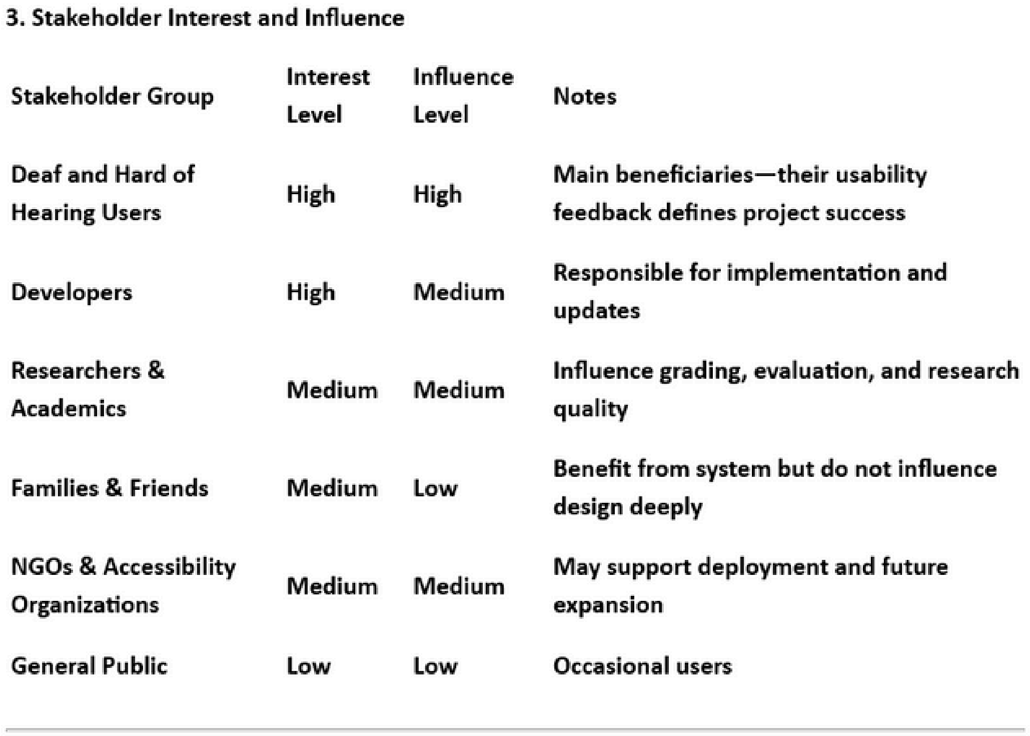
Secondary Stakeholders

Stakeholder	Role	Needs / Expectations
Developers (Project Team)	Responsible for building, training, and improving the ML model	Clear dataset, stable performance, tools for testing and debugging
Researchers / Academics	Evaluate the technical contribution and methodology	Well-documented model, reproducible results, clear evaluation metrics
Teachers in Special Education	Use the system in teaching sign language	Simple interface for students, real-time feedback to help learning



Tertiary Stakeholders

Stakeholder	Role	Needs / Expectations
Organizations & NGOs Supporting Disability Communities	Promote accessibility and digital inclusion	A tool that can be deployed in awareness programs and communication trainings
Future Developers / Contributors	Improve or extend the system	Clean and modular codebase, documentation, clear improvement directions
General Public / Non-signers	Use the system to understand basic signs	User-friendly system that helps them understand alphabet gestures when needed



3. Stakeholder Interest and Influence

Stakeholder Group	Interest Level	Influence Level	Notes
Deaf and Hard of Hearing Users	High	High	Main beneficiaries—their usability feedback defines project success
Developers	High	Medium	Responsible for implementation and updates
Researchers & Academics	Medium	Medium	Influence grading, evaluation, and research quality
Families & Friends	Medium	Low	Benefit from system but do not influence design deeply
NGOs & Accessibility Organizations	Medium	Medium	May support deployment and future expansion
General Public	Low	Low	Occasional users

4. Stakeholder Needs Summary

- Accessibility — A system that is simple, intuitive, and works for a wide range of users.
- Accuracy — Correct detection of alphabet gestures with minimal errors.
- Speed — Real-time or near-real-time gesture recognition.
- Reliability — Stable performance across different lighting, backgrounds, and hand shapes.

Scalability — Potential future upgrade to full word/phrase recognition