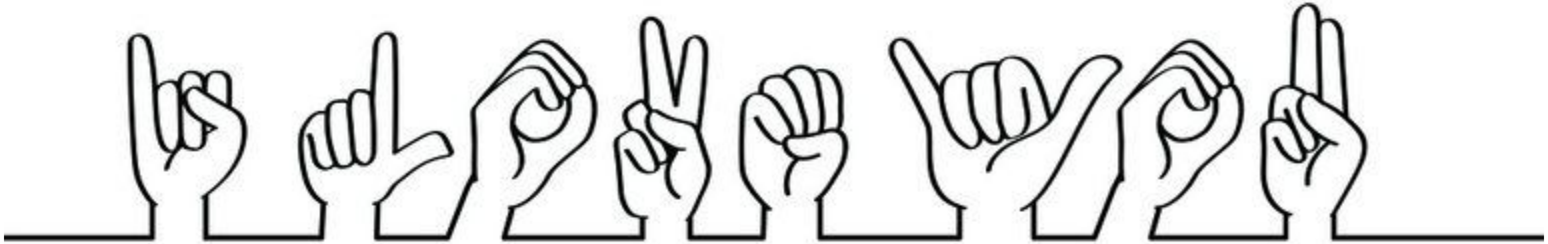
Team EI:

Give Me A Sign

Leia Park, Ran Fang, Sejal Madan

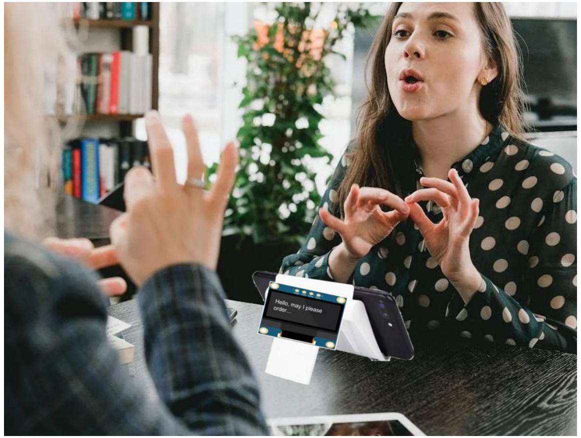


Use Case / Application

There exists communication barriers between the deaf community and those who are not familiar with sign language.

Our Solution:

Real-time ASL Translator App + Phone Attachment



Quantitative Design Requirements

Requirement	Quantitative/Qualitative Specifications
Person must be near camera so gestures are visible and tracked	Distance📏: 1.0 - 3.9 ft + Brightness💡: 10 - 500 lux [1] → Process image (resize, grayscale, normalize) and reduce noise using temporal/spatial filtering and/or background subtraction
Gesture recognition should be accurate	Accuracy🎯: >= 95% → MediaPipe hand & pose recognition (21*2+22 = 64 landmarks)[2]
Translation should be accurate	Accuracy🎯: >= 95% → Use hybrid of CNN for static and LSTM for dynamic signing
Translation should be relatively immediate to work as "live subtitles"	Latency🏃‍♀️: 1 - 3s → CV frame rate: 10-15 fps + ML processing + NLP correction
Good accessibility for positive user experience for both parties involved	Satisfaction rate😄: >=90% → Minimalistic mobile app UI/UX design + near-random sampling

Solution Approach

Inclusivity of ASL users and for people to actively engage in conversations even with communication barriers

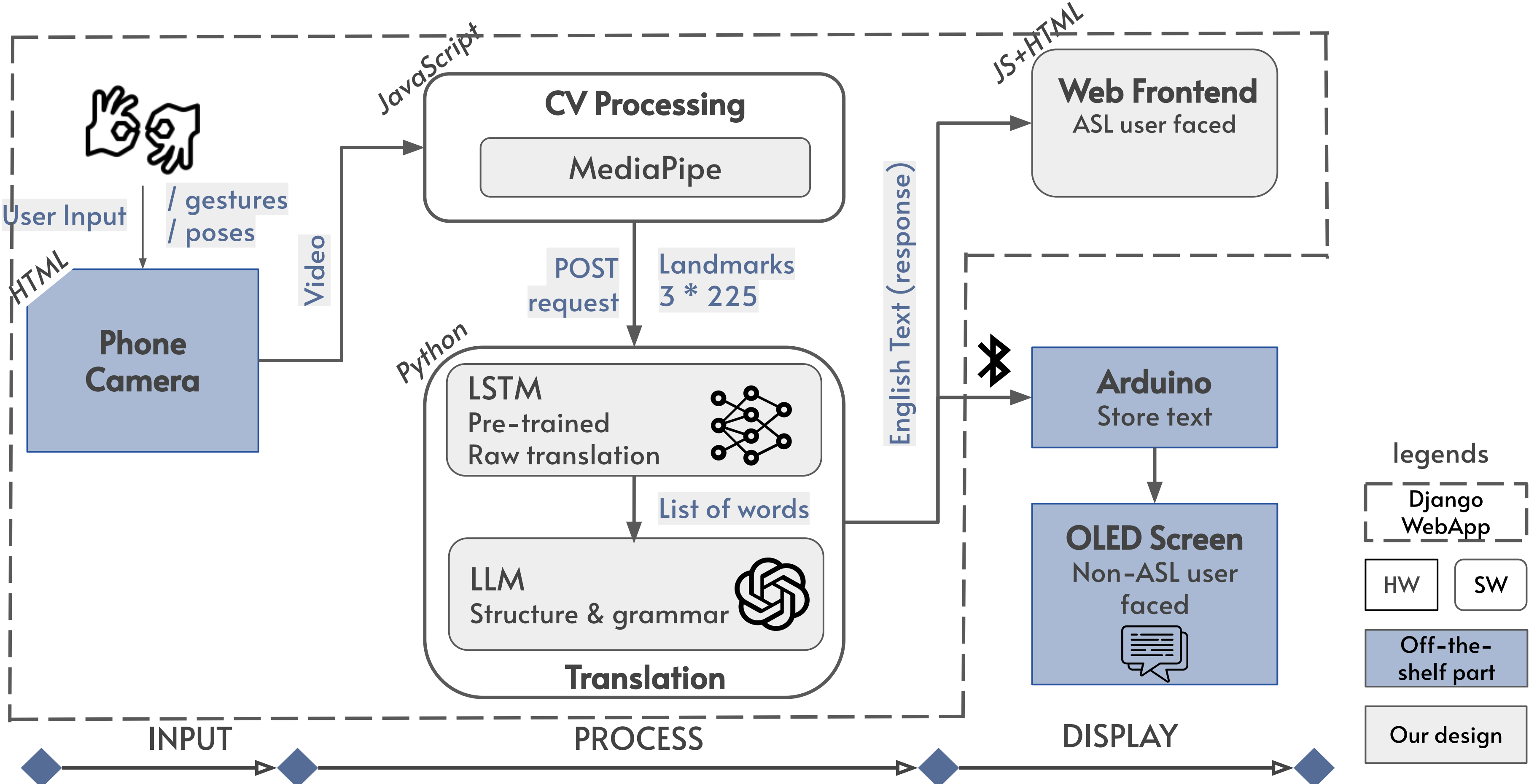
Our product aims to promote:

✓ Public Health & Welfare
✓ DEI & Social Support
✓ Accessibility

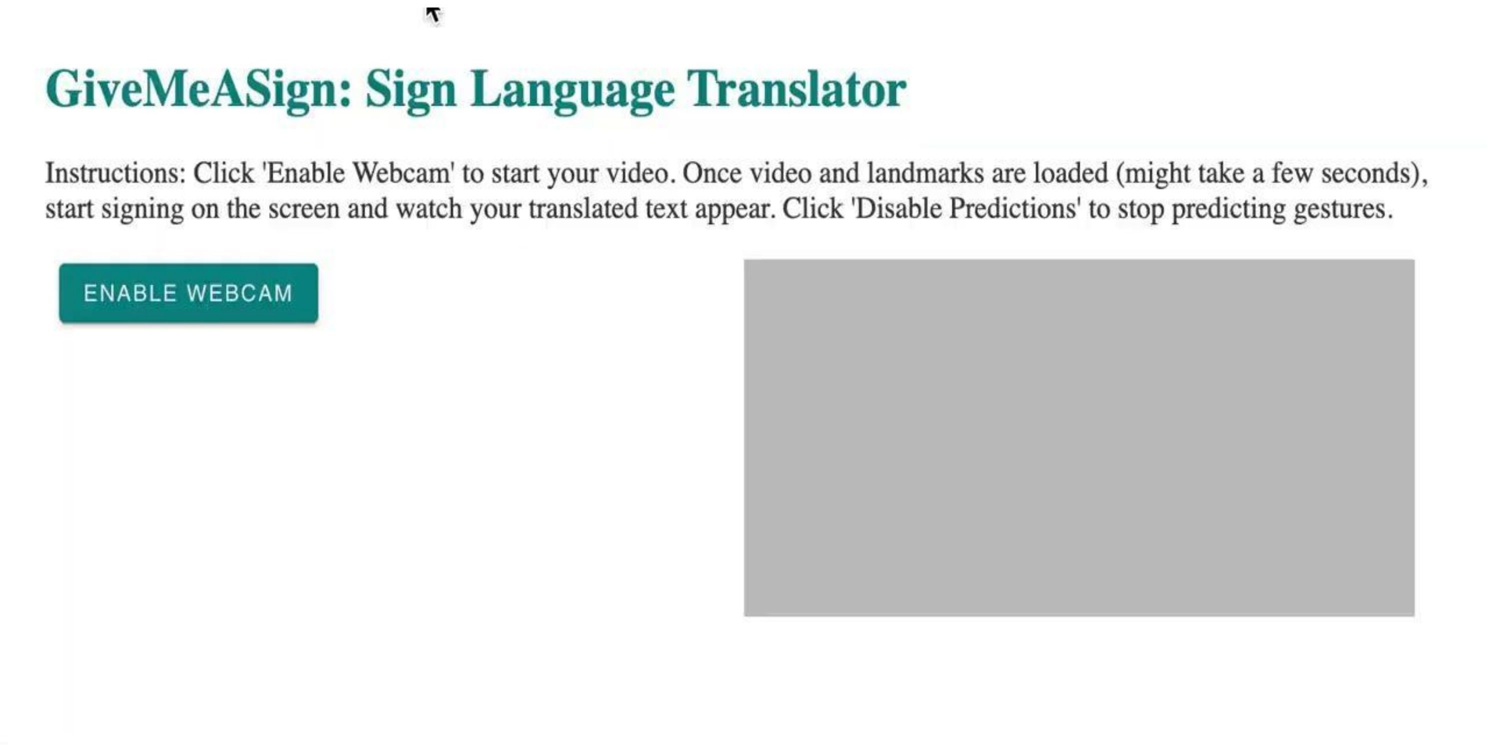
New Developments:

- Web App Conversion
- Versatility
- Local Packaging

System Specification



Complete Solution



GiveMeASign: Sign Language Translator

Instructions: Click 'Enable Webcam' to start your video. Once video and landmarks are loaded (might take a few seconds), start signing on the screen and watch your translated text appear. Click 'Disable Predictions' to stop predicting gestures.

https://drive.google.com/file/d/1mlokBxdzZHNwwxahjLRvPVLH6H23mJe4/view?usp=sharing

Final demo will include web app integrated with bluetooth + second screen

Test, Verification & Validation

Use-Case Metric	How we tested	Passing Metric	Results
Signing to occur 1-3.9ft from the camera	→] Different distances **5 samples** at each interval: 1ft, 1.5ft, 2.0ft, 2.5ft, 3ft, 3.5ft, 4ft	Proper landmarks should appear at <3.9 ft	Proper landmarks appear **100%** of the time at distances between 1-3.9ft
High accuracy (~95%) for gesture detection	→] Different BG settings: **2 samples** at each trial: **1-10 distractors** in the background ←] landmarks	CV/MediaPipe should display proper landmarks of the hands and upper body 95% of the time	Landmarks are drawn on the target subject **95%** of time Exceptions: humans in background
High accuracy (~95%) for sign language translation	→] Different signings: **3 members** sign each phrase **3 times** and **3 complex sentences** ←] English text	English text should appear and be 95% accurate in semantic meaning	Phrases translation accuracy: **91.1%** Sentences translation accuracy: **88.9%**

Test, Verification & Validation (continued)

Use-Case Metric	How to test	Passing Metric	Results
Low latency (1-3s) in translation	→] each team member signs each phrase 3 times ←] time (ms) elapsed before the translation appears	Translation should appear 1000-3000ms after a gesture	Prediction of words appear an average of **1100ms** after a gesture Translation appeared an average of **2900ms** after a sentence
Product user satisfaction >= 90%	→] invite sign language users ←] oral feedback & survey results	90% user satisfaction	Our fully integrated product is still in progress, so user satisfaction is unavailable now
Ease of phone attachment use	→] invited sign language users ←] oral feedback & survey results	90% user satisfaction	Our fully integrated product is still in progress, so user satisfaction is unavailable now

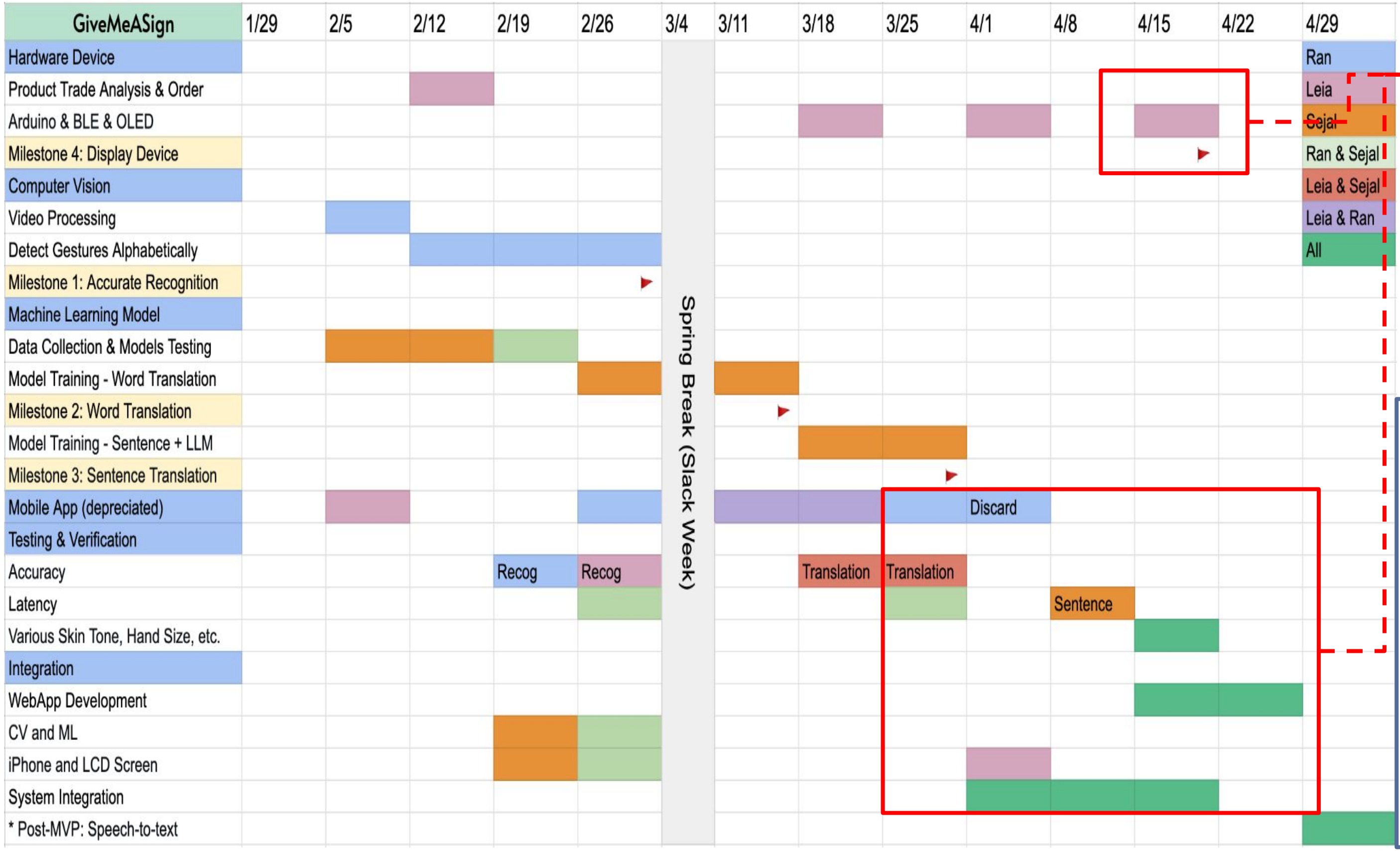
Design Trade-offs

- Mobile app → Web app
  - **Bottleneck 1**: Real-time video transmission between phone camera and cloud server
    - **Mitigation**: Discard cloud server, process everything locally in mobile app
    - **Trade-off(s):**
      - Increased programming complexity: 33% → 100% in Swift ☹
  - **Bottleneck 2**: Integration issues w/ keras-trained ML model (CoreML is tricky!)
    - **Mitigation**: Convert to a web app that can be opened up via url on the phone
    - **Trade-off(s)**:
      - Reduced development time due to design change: 4 → 2 weeks ☹
      - Decreased accuracy due to separation of codelayers for CV and ML ☹
      - Decreased complexity, no major compromise in functionality ☺

Design Trade-offs

- Limited Amount of Phrases
  - **Bottleneck 1**: Minimal translations able to be generated
    - **Mitigation**: Focus on correctness of translations, quality over quantity
    - **Trade-off(s)**:
      - Increased accuracy with focused batch of words: 30% → 90% 🙂
      - Decreased scope of phrases app is capable of: 20 → 10 phrases 🙁
- NLP → LLM
  - **Bottleneck 1**: NLP requires extensive training of another model
    - **Mitigation**: Switch to LLM with OpenAI API integration instead
    - **Trade-off(s)**:
      - Greater compatibility with our needs and current implementation 🙂
      - Decreased complexity, no major compromise in functionality 🙂

Project Management



GiveMeASign	1/29	2/5	2/12	2/19	2/26	3/4	3/11	3/18	3/25	4/1	4/8	4/15	4/22	4/29
Hardware Device						Spring Break (Slack Week)								Ran
Product Trade Analysis & Order			🟪											Leia
Arduino & BLE & OLED								🟪		🟪		🟪		Sejal
Milestone 4: Display Device												▶		Ran & Sejal
Computer Vision														Leia & Sejal
Video Processing		🟦												Leia & Ran
Detect Gestures Alphabetically			🟦	🟦	🟦									All
Milestone 1: Accurate Recognition					▶									
Machine Learning Model														
Data Collection & Models Testing		🟧	🟧	▩										
Model Training - Word Translation					🟧		🟧							
Milestone 2: Word Translation							▶							
Model Training - Sentence + LLM								🟧	🟧					
Milestone 3: Sentence Translation									▶					
Mobile App (depreciated)		🟪			🟦		▦	▦	🟦	Discard				
Testing & Verification														
Accuracy				Recog	Recog			Translation	Translation					
Latency					▩				▩		Sentence			
Various Skin Tone, Hand Size, etc.												🟩		
Integration														
WebApp Development												🟩	🟩	
CV and ML				🟧	▩									
iPhone and LCD Screen				🟧	▩					🟪				
System Integration										🟩	🟩	🟩		
* Post-MVP: Speech-to-text														🟩

Changes due to design change

Major changes happened when we decided to switch from mobile app to web app.

Remaining tasks before public demo:

- Integrate OLED screen
- Improve web app UI
- Collect more user experience feedback

*Speech-to-text

Conclusion

❗ Lessons Learned ❗

1. Overestimation/Underestimation 🤸
2. Time Allocation 🕓
3. What We Want vs. What Is Possible ⚖️
4. Being Resourceful 🛠️

Through a **simple and sleek** phone attachment and combined web app, we can **break down language barriers** and ensure **accessibility** for deaf and hard of hearing community