

SignSpeak: Filipino Sign Language to Speech Translator

Aira Joy Q. Austria¹, John Rey M. Gulapa², Eloiza Mae G. Sigue³, Joshua M. Soliman⁴,
and Sarah Jane M. Ferrer⁵

^{1,2,3,4,5}Pangasinan State University - Bayambang Campus
Bayambang, Pangasinan, Philippines

Article Info:

Received: 01 Feb 2025; Revised: 09 April 2025; Accepted: 21 July 2025; Available Online: 20 Aug 2025

Abstract- SignSpeak: Filipino Sign Language to Speech Translator is an assistive Android mobile application developed to facilitate communication between deaf students and hearing individuals with limited knowledge of Filipino Sign Language (FSL), particularly in an educational setting. The project combined an Artificial Intelligence Modeling Cycle with a Software/System Engineering Lifecycle. A Multi-Layer Perceptron (MLP) was used for static alphabet and number gestures, while a Long Short-Term Memory (LSTM) network was used for dynamic word gestures. MediaPipe supported landmark extraction; static data were augmented through horizontal flipping, while dynamic sequences were normalized relative to the shoulder landmark. The datasets consisted of 62,400 alphabet images, 24,000 number images, and 9,000 word-gesture videos. Two domain experts, including a certified FSL interpreter and a Deaf validator, evaluated a validation subset, producing a Cohen's Kappa of 0.4949, interpreted as moderate agreement. The best-performing 80/10/10 split achieved test accuracies of 97.22% for dynamic words, 99.89% for alphabets, and 99.96% for numbers. The system was evaluated by 15 respondents composed of 12 students and 3 teachers/instructors. The Software Usability Measurement Inventory-based assessment produced an overall weighted mean of 4.51, interpreted as acceptable and usable. Economic, technical, and operational assessments further supported the feasibility of the developed application.

Keywords- Filipino Sign Language, sign language recognition, assistive technology, LSTM, MLP, MediaPipe, mobile application, SUMI

INTRODUCTION

Communication enables people to exchange ideas, emotions, and information and to participate in social and educational relationships. For deaf and hard-of-hearing individuals, sign language serves as an important mode of communication when spoken communication is limited or unavailable (Ganmote, 2019; Yin et al., 2021). The source manuscript also frames Filipino Sign Language as a critical tool for inclusive participation in Philippine education, supported by Republic Act No. 11106 (Rogers, 2020).

Deaf students may encounter difficulty interacting with hearing classmates and teachers when the latter have limited knowledge of FSL. The manuscript identifies communication barriers such as reliance on lip-reading, delayed exchanges through writing or texting, limited FSL learning materials, and the absence of immediate translation technologies in

classrooms. Similar educational and accessibility challenges are discussed in the literature cited by the project (Cahill, 2023; Levesque et al., 2025).

Machine learning and mobile computing provide a possible means of reducing these barriers. Sign-language recognition systems can use visual hand and body landmarks together with trained neural networks to classify static and dynamic gestures. In the present project, static FSL alphabets and numbers were recognized through a Multi-Layer Perceptron, while dynamic words were handled through a Long Short-Term Memory network. The approach reflects the distinction between spatially defined static signs and temporally sequenced dynamic gestures described in related studies cited in the manuscript.

SignSpeak was therefore developed as a real-time mobile system that combines FSL learning functions, gesture recognition, text or speech output, and response features intended to support communication between deaf and nondeaf users. The project also examined dataset reliability, machine-learning performance, system design, feasibility, and user acceptability and usability.

OBJECTIVE OF THE STUDY

The project aimed to create an easy-to-use application named SignSpeak: Filipino Sign Language to Speech Translator that can convert Filipino Sign Language gestures into spoken language through trained machine-learning models and help connect deaf and hearing individuals, particularly in an educational environment.

1. Identify the communication challenges affecting daily interactions between deaf students and hearing individuals.
2. Implement trained models, specifically Long Short-Term Memory (LSTM) and Multi-Layer Perceptron (MLP), for real-time recognition of Filipino Sign Language gestures and their translation into spoken language.
3. Identify the features of the system in terms of the description of the initial prototype, the proposed implementation plan, and the technical specifications for hardware and software.
4. Assess the feasibility of the system in terms of economic feasibility, technical feasibility, and operational feasibility based on SUMI dimensions of efficiency, affect, helpfulness, control, and learnability.

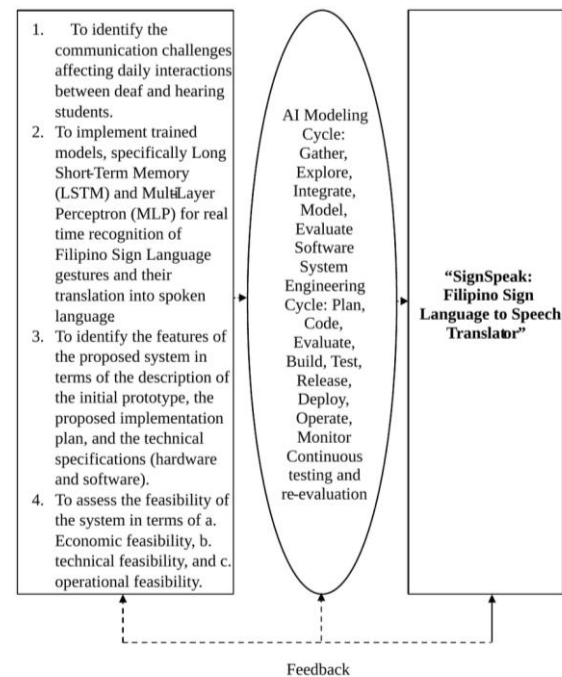


Figure 1. Conceptual framework of SignSpeak: Filipino Sign Language to Speech Translator.

Figure 1 summarizes how the project's four objectives are processed through the AI Modeling Cycle and the Software/System Engineering Cycle. The framework links the identified communication problem, model implementation, system features, and feasibility assessment to the final SignSpeak mobile application, with feedback supporting continuing refinement.

MATERIALS AND METHODS

Research and Development Design

The project used a developmental system approach organized around two complementary processes: an Artificial Intelligence Modeling Cycle and a Software/System Engineering Lifecycle. The AI cycle covered gathering, exploring, integrating, modeling, and evaluating the data, while the software cycle covered planning, coding, evaluation, building, testing, release, deployment, operation, monitoring, monitoring, and iterative improvement.

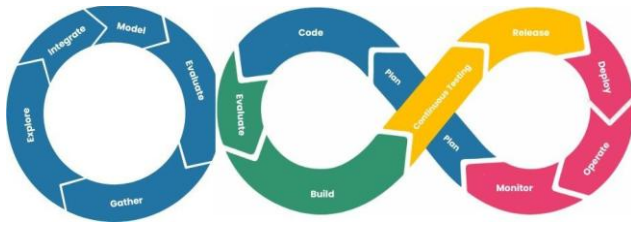


Figure 2. Artificial Intelligence Modeling and Software/System Engineering Lifecycle used in the project.

Figure 2 presents the two development cycles used to organize the creation of SignSpeak. The AI Modeling Cycle governed preparation and evaluation of the machine-learning models, while the Software/System Engineering Lifecycle guided implementation of the Android application, integration of the trained models, testing, release, deployment, operation, and monitoring.

AI Model Development and Training

The developers first consulted the SNED teacher/coordinator at Bayambang National High School regarding real-world communication challenges. A dataset was then collected using a webcam and included 62,400 images of FSL alphabets, 24,000 images of FSL numbers, and 9,000 videos of dynamic word gestures. A certified FSL interpreter and a Deaf validator assessed the correctness and authenticity of the signs.

MediaPipe was used to extract gesture landmarks. Static signs were processed with a Multi-Layer Perceptron containing dense layers of 128 and 64 neurons and Dropout regularization. Horizontal flipping was used for static-data augmentation. Dynamic signs were processed using an LSTM with sequences of 30 frames and a stacked architecture of 64, 128, and 64 units. Dynamic landmarks were normalized using the shoulder as the reference point. Model performance was evaluated using accuracy, precision, recall, and F1-score, and the trained models were converted to TensorFlow Lite for mobile deployment.

System Development and Deployment

The Android application was developed in Android Studio using user-centered design principles.

The MLP and LSTM models were integrated for real-time recognition of static and dynamic FSL gestures. The application was tested for recognition accuracy, responsiveness, translation quality, user-interface behavior, detection time, and resource consumption. A signed Android Package Kit was released through GitHub and installed on target devices for pilot testing and feedback.

Population and Locale

The project was implemented in collaboration with the Special Needs Education (SNED) Program of Bayambang National High School. Fifteen respondents participated in system testing and evaluation: 12 students and 3 teachers/instructors who were active users of Filipino Sign Language. Interviews, demonstrations, and consultations were also used to gather information about communication needs and system requirements.

Data-Gathering Instruments

For the first objective, an interview with the SNED coordinator was used to identify communication problems encountered by deaf students. For the second objective, webcams were used for gesture-data collection, and model evaluation used accuracy, precision, recall, F1-score, confusion-matrix analysis, and inter-rater validation. For the third objective, an initial prototype and deployed application were used to demonstrate the Learn and Translate functions and other application features. For the fourth objective, cost estimation supported economic feasibility analysis, documentary analysis supported technical feasibility, and an adapted Software Usability Measurement Inventory questionnaire measured operational feasibility.

Data Analysis

Thematic analysis was used to organize the communication challenges reported during interviews, and an Ishikawa diagram was used to group root causes. Cohen's Kappa was used to assess agreement between the Deaf validator and certified FSL interpreter. Machine-learning models were compared across 80/10/10, 70/15/15, and 50/25/25 training-validation-test splits. A use-case diagram represented

system functions and user interactions. Economic feasibility was assessed using cost-benefit analysis, while usability responses were summarized using average weighted means.

Table 1. Likert Scale for Usability Testing

Numerical Equivalent	Weighted	Descriptive Equivalent	Descriptive Interpretation
5	4.51-5.00	Strongly Agree	Acceptable/Usable
4	3.51-4.50	Agree	Acceptable/Usable
3	2.51-3.50	Moderate Agree	Acceptable/Usable
2	1.51-2.50	Slightly Agree	Not Acceptable/Usable
1	1.00-1.50	Disagree	Not Acceptable/Usable

Table 1 shows the five-point interpretation scale used for the operational-feasibility evaluation. Scores from 2.51 to 5.00 were treated as positive indicators and interpreted as acceptable/usable, while scores from 1.00 to 2.50 were interpreted as not acceptable/usable.

RESULTS AND DISCUSSION

Communication Challenges Affecting Daily Interactions

Interviews conducted with the SNED program identified recurring communication difficulties between deaf students and hearing individuals. The manuscript reports limited FSL knowledge among hearing students and teachers, low confidence among deaf students, reliance on lip-reading, dependence on writing or texting, limited FSL learning materials, one-way communication, and a lack of real-time translation technologies.

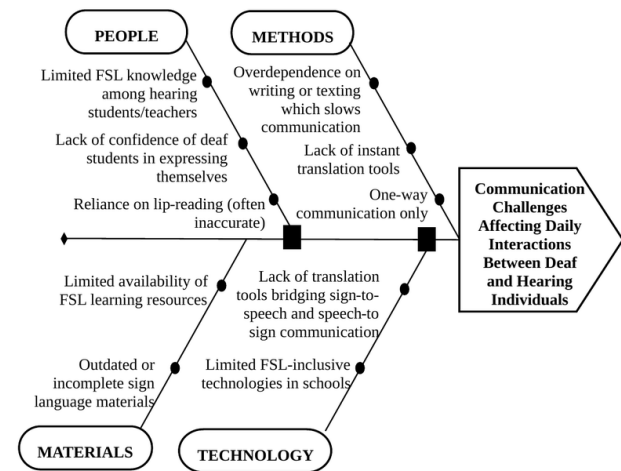


Figure 3. Ishikawa diagram of communication challenges affecting daily interactions between deaf and hearing individuals.

Figure 3 groups the identified communication barriers into four categories: People, Methods, Materials, and Technology. The diagram shows that communication problems arise not from a single factor but from the interaction of limited FSL knowledge, slow communication methods, inadequate learning resources, and limited translation technologies in the school environment.

Dataset Reliability

Table 2. Confusion Matrix of Validator Ratings

Validator 2 / Validator 1	1	0	Total
Validator 2 = 1	94 (Agreement)	5 (Disagreement)	99
Validator 2 = 0	5 (Disagreement)	6 (Agreement)	11
Total	99	11	110

Table 2 presents the distribution of ratings from the two domain validators. The validators agreed that 94 samples were valid and that 6 samples were invalid, while 10 samples received different ratings. The pattern indicates a high level of observed agreement, although disagreement remained for a small subset of the validation samples.

Table 3. Inter-Rater Reliability Analysis Results

Metric	Value
Number of samples validated	110
Observed Agreement (Po)	0.9091 (90.91%)
Chance of Agreement (Pe)	0.8200 (82.00%)
Cohen's Kappa (κ)	0.4949
Interpretation	Moderate Agreement

Table 3 shows a Cohen's Kappa of 0.4949, interpreted in the source manuscript as moderate agreement. Although observed agreement reached 90.91%, the high prevalence of samples classified as valid increased the expected chance agreement and reduced the Kappa value. The result was considered sufficient to establish a reasonable level of dataset validity and consistency for subsequent model training.

Machine-Learning Model Performance

Table 4. Performance of LSTM Model on Validation Set

Data Split	Accuracy	Precision	Recall	F1-Score
80-10-10	0.9622	0.9644	0.9622	0.9623
70-15-15	0.9519	0.9567	0.9519	0.9516
50-25-25	0.9391	0.9426	0.9391	0.9390

Table 4 shows that the LSTM model performed best on the validation set using the 80/10/10 split. This configuration obtained 0.9622 accuracy and a 0.9623 F1-score, while performance declined as the proportion of training data was reduced, supporting the importance of sufficient sequential training data for dynamic gesture recognition.

Table 5. Performance of LSTM Model on Test Set

Data Split	Accuracy	Precision	Recall	F1-Score
80-10-10	0.9722	0.9744	0.9722	0.9720
70-15-15	0.9519	0.9569	0.9519	0.9510

50-25-25	0.9293	0.9334	0.9293	0.9290
----------	--------	--------	--------	--------

Table 5 confirms the same pattern on the LSTM test set. The 80/10/10 split achieved the highest test accuracy at 0.9722 and F1-score at 0.9720. The 50/25/25 split produced the lowest test performance, indicating that dynamic word recognition benefited from a larger training partition.

Table 6. Performance of the MLP on Alphabets in the Validation Set

Data Split	Accuracy	Precision	Recall	F1-Score
80-10-10	0.9994	0.9994	0.9994	0.9994
70-15-15	0.9983	0.9983	0.9983	0.9983
50-25-25	0.9980	0.9980	0.9980	0.9980

Table 6 presents the validation performance of the MLP for static FSL alphabets. All three splits produced very high scores, with the 80/10/10 configuration obtaining the highest reported value of 0.9994. The results indicate that static alphabet landmarks were highly separable using the MLP architecture.

Table 7. Performance of the MLP on Alphabets in the Test Set

Data Split	Accuracy	Precision	Recall	F1-Score
80-10-10	0.9989	0.9989	0.9989	0.9989
70-15-15	0.9982	0.9982	0.9982	0.9982
50-25-25	0.9977	0.9977	0.9977	0.9977

Table 7 shows the MLP test results for alphabets. The 80/10/10 split again obtained the strongest result at 0.9989, while the other splits also remained above 0.997. These results demonstrate consistently high static alphabet-recognition performance across the evaluated data splits.

Table 8. Performance of the MLP on Numbers in the Validation Set

Data Split	Accuracy	Precision	Recall	F1-Score
80-10-10	1.0000	1.0000	1.0000	1.0000
70-15-15	0.9997	0.9997	0.9997	0.9997
50-25-25	0.9982	0.9982	0.9982	0.9982

Table 8 presents the MLP validation results for FSL numbers. The 80/10/10 split achieved 1.0000 across the reported metrics, while the 70/15/15 and 50/25/25 splits also produced very high results. The manuscript attributes this strong performance to the clear visual distinctions among many number gestures.

Table 9. Performance of the MLP on Numbers in the Test Set

Data Split	Accuracy	Precision	Recall	F1-Score
80-10-10	0.9996	0.9996	0.9996	0.9996
70-15-15	0.9986	0.9986	0.9986	0.9986
50-25-25	0.9982	0.9982	0.9982	0.9982

Table 9 shows that the MLP retained near-perfect performance on the number test set. The 80/10/10 split produced the highest reported result of 0.9996. Together with the alphabet and word results, the findings led the project to select the 80/10/10 split as the preferred configuration.

System Design and Major Application Features

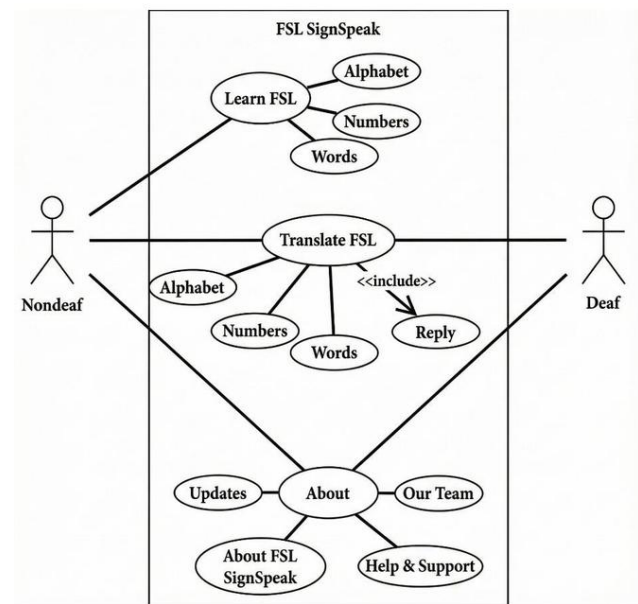


Figure 4. Use-case diagram of SignSpeak.

Figure 4 presents the functional relationship between the two primary user groups, Deaf and nondeaf users, and the application's major modules. The system groups its functions into Learn FSL, Translate FSL, and About, with subfunctions for alphabets, numbers, words, reply to features, updates, team information, FSL information, and help and support.



Figure 5. Welcome page of the deployed SignSpeak application.

Figure 5 shows the Welcome page, which introduces the application and provides the entry point to its primary functions. The screen is designed to orient users before they proceed to learning, translation, and support features.



Figure 6. Learn page of the deployed SignSpeak application.

Figure 6 presents the Learn page. It introduces the instructional component of SignSpeak and

provides access to lessons covering FSL alphabets, numbers, and basic words.



Figure 7. Translate page of the deployed SignSpeak application.

Figure 7 shows the Translate page, which represents the application's principal communication feature. From this screen, users can access the available recognition and translation modes for Filipino Sign Language.

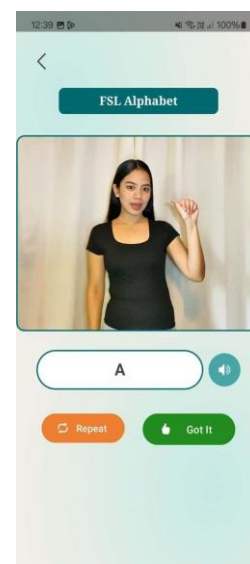


Figure 8. Learning screen for a selected FSL item.

Figure 8 presents the learning screen displayed after a user selects an FSL item. The interface provides the instructional content and

controls needed to review a sign and move through the learning module.

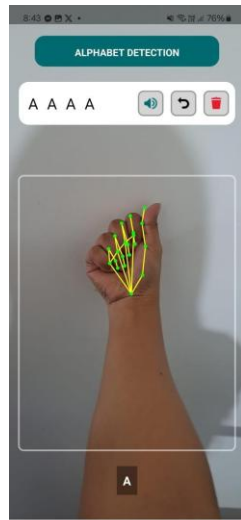


Figure 9. Alphabet-detection interface.

Figure 9 shows the alphabet-detection interface. The screen guides the user before activating the camera and supports real-time recognition of static alphabet gestures using the integrated MLP model.

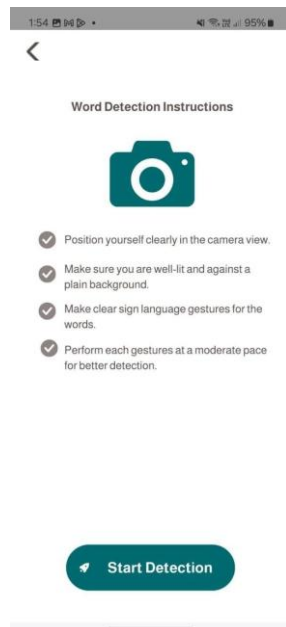


Figure 10. Word-detection interface.

Figure 10 presents the word-detection interface. This mode captures dynamic gestures through the camera and uses the LSTM model to recognize word-level FSL sequences.

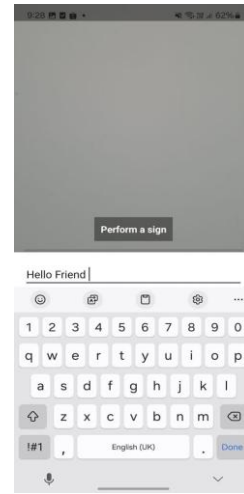


Figure 11. Text-to-FSL feature.

Figure 11 shows the Text-to-FSL feature, where a hearing user can type a message and view the corresponding FSL presentation. This feature provides a reverse communication path that can help a Deaf user understand typed input.

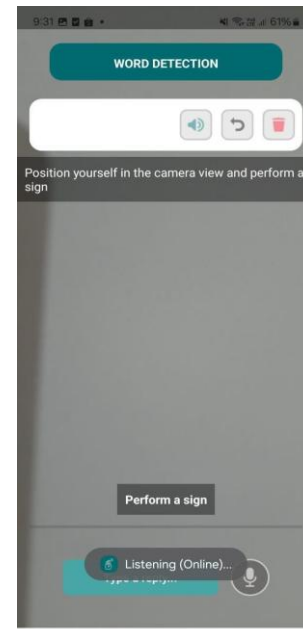


Figure 12. Speech-to-sign response feature.

Figure 12 illustrates the speech-response feature used to capture spoken input and present a sign-language response. The feature complements gesture-to-speech translation by supporting two-way communication between Deaf and hearing users.

Technical Specifications

Table 10. Minimum Hardware Requirements for System Development

Hardware	Specifications
Processor	AMD Radeon R7
Memory	8 GB
Device Storage	20 GB
Graphics Card	Integrated GPU
Internet Connection	2 Mbps
Webcam	720p-1080p
Deployment/Maintenance Platform	GitHub

Table 10 summarizes the hardware identified in the source manuscript for development. The listed configuration combines memory, storage, graphics, internet connectivity, and a webcam capable of capturing hand gestures. These resources were intended to support model training, application development, real-time testing, and deployment activities.

Table 11. Minimum Software Requirements for System Development

Software	Specifications
Operating System	Windows 10
Program Development System	Visual Studio Code
Application Form	Android Studio
Programming Language	Java, Python, XML
Design Editor	Microsoft Word, Figma
System Design	Android Studio

Table 11 presents the software environment used for development. Windows 10, Visual Studio Code, Android Studio, Java, Python, XML, Microsoft Word, and Figma supported application coding, machine-learning integration, interface design, documentation, and prototyping. The manuscript also notes the use of GitHub for deployment, version control, and maintenance.

Table 12. Minimum Hardware Requirements for Mobile App Implementation

Hardware	Specifications
Android Studio	10.0

RAM	4 GB
Storage	3 GB
Camera	Any Supported

Table 12 reproduces the implementation requirements reported in the source table. The accompanying manuscript text describes the first item as Android 10.0 or higher and identifies at least 4 GB of RAM, 3 GB of available storage, and a supported camera as the minimum conditions for deployment and gesture detection on a mobile device.

Economic Feasibility

Table 13. Development Capital Costs

Costs	Capital Costs / One-Time Purchases
Hardware Components (Laptop, Smartphone, Web Camera)	₱77,300
Software Components (MLP, LSTM, and GitHub)	₱0 (open source)
Deaf Consultant	₱500
Other Expenses (Food and Print)	₱2,000
Total Costs	₱79,800

Table 13 shows a total estimated project cost of ₱79,800. Hardware represented the largest cost component, while the machine-learning models and GitHub were treated as open-source resources with no software purchase cost. Consultant and miscellaneous project expenses were also included.

Table 14. Estimated Benefits

Benefits	Value	Frequency	Annual
Interpreter Cost	₱4,000	5	₱20,000
Seminars and Trainings	₱35,000	1	₱35,000
Deaf Validator	₱600	4	₱2,400
Total Benefits			₱57,400

Table 14 reports estimated annual benefits of ₱57,400. Using the source manuscript's cost-benefit

computation, the system was projected to reach break-even at approximately 1.39 years, or about one year and five months. By the second year, cumulative benefits of ₱114,800 were reported to exceed the ₱79,800 project cost, producing an estimated Year 2 ROI of about 43.9%.

Acceptability and Usability

Table 15. Profile of Respondents in Terms of Respondent Category

Respondent Category	Frequency	Percentage
Student	12	80%
Teacher/Instructor	3	20%
Total	15	100%

Table 15 shows that the operational-feasibility evaluation involved 15 respondents, composed of 12 students and 3 teachers/instructors. Students represented 80% of the sample, reflecting the project's emphasis on intended student end-users, while instructors contributed an educational-use perspective.

Table 16. Profile of Respondents in Terms of Age Range

Age Range	Frequency	Percentage
13-17 years old	5	33%
18 years old and above	10	67%
Total	15	100%

Table 16 presents the age profile of the respondents. Five respondents were 13-17 years old, while 10 were 18 years old or above. The two age groups provided user feedback from both younger student users and older respondents involved in testing the application.

Table 17. Acceptability and Usability Assessment in Terms of Efficiency

Items	Mean	Descriptive Equivalent
1. This software responds to fast inputs.	4.53	Strongly Agree

2. It is easy to make the software do exactly what you want.	4.60	Strongly Agree
3. This software is a great help to my work/studies.	4.53	Strongly Agree
4. The software presents alternative facilities that aid in my work/studies.	4.47	Agree
5. The software is consistent.	4.33	Agree
6. The speed of the software is fast enough.	4.33	Agree
7. 7. User needs have been fully taken into consideration.	4.40	Agree
8. 8. The system does not sometimes stop unexpectedly.	4.33	Agree
9. 9. I do not have to look for assistance most times when I use this software.	4.13	Agree
10. 10. It is relatively easy to move from one part of a task to another.	4.40	Agree
11. Average Weighted Mean	4.41	Agree

Table 17 shows an average weighted mean of 4.41 for efficiency, interpreted as Agree and acceptable/usable under the study's scale. The highest item mean was 4.60 for the system's ability to perform what users wanted, while the lowest item mean of 4.13 remained within the positive range.

Table 18. Acceptability and Usability Assessment in Terms of Affect

Items	Mean	Descriptive Equivalent
1. I know what to do next with this software.	4.53	Strongly Agree
2. Working with this software is satisfying.	4.73	Strongly Agree

3. I would like to use this software every day.	4.67	Strongly Agree
4. I would recommend this software to others.	4.60	Strongly Agree
5. Using this software is not frustrating.	4.13	Agree
Average Weighted Mean	4.53	Strongly Agree

Table 18 presents an average weighted mean of 4.53 for affect. Respondents reported a highly positive emotional response to the software, with the highest mean of 4.73 assigned to satisfaction in working with the application.

Table 19. Acceptability and Usability Assessment in Terms of Helpfulness

Items	Mean	Descriptive Equivalent
1. The instructions and prompts are helpful.	4.80	Strongly Agree
2. The way that system information is presented is clear and understandable.	4.67	Strongly Agree
3. There is enough information on the screen when it is needed.	4.60	Strongly Agree
4. Error prevention messages are adequate.	3.93	Strongly Agree
5. I am confident to use all the functions.	4.80	Strongly Agree
Average Weighted Mean	4.56	Strongly Agree

Table 19 shows an average weighted mean of 4.56 for helpfulness. The highest ratings of 4.80 were given to the usefulness of instructions and prompts and to respondents' confidence in using the system's functions, indicating that the application was perceived as supportive and understandable.

Table 20. Acceptability and Usability Assessment in Terms of Control

Items	Mean	Descriptive Equivalent
1. This software did not stop unexpectedly under different conditions.	4.07	Agree
2. There are familiar commands or operations to do a task.	4.73	Strongly Agree
3. I can understand and act on the information provided by the software.	4.53	Strongly Agree
4. Tasks can be performed in a straightforward manner using this software.	4.47	Agree
5. Fewer steps are required to get something done for work/studies.	4.67	Strongly Agree
6. It is easy to learn the software functions.	4.60	Strongly Agree
Average Weighted Mean	4.51	Strongly Agree

Table 20 presents an average weighted mean of 4.51 for control. Familiar commands received the highest item mean of 4.73, while stability under different conditions received 4.07. Overall, respondents indicated that they could understand, direct, and operate the application effectively.

Table 21. Acceptability and Usability Assessment in Terms of Learnability

Items	Mean	Descriptive Equivalent
1. Learning to operate this software is initially easy.	4.40	Agree
2. It is easy to learn the software commands and functions.	4.60	Strongly Agree

3. The commands lead to a proper step.	4.47	Agree
4. There is not much to read before using the software.	4.67	Strongly Agree
5. The organization of menus or information lists seems logical.	4.33	Agree
6. Learning how to use new functions is very easy.	4.80	Strongly Agree
7. It does not take too long to learn the software functions.	4.53	Strongly Agree
8. I know what to do next with this system.	4.67	Strongly Agree
Average Weighted Mean	4.56	Strongly Agree

Table 21 shows an average weighted mean of 4.56 for learnability. Respondents rated learning new functions highest at 4.80, while menu organization received the lowest means of 4.33. Both values remained in the positive range, supporting the conclusion that the application was learnable.

Table 22. Overall Acceptability and Usability Assessment for SignSpeak

Category	Mean	Descriptive Equivalent	Descriptive Interpretation
Efficiency	4.41	Agree	Acceptable/Usable
Affect	4.53	Strongly Agree	Acceptable/Usable
Helpfulness	4.56	Strongly Agree	Acceptable/Usable
Control	4.51	Strongly Agree	Acceptable/Usable
Learnability	4.56	Strongly Agree	Acceptable/Usable
Average Weighted Mean	4.51	Strongly Agree	Acceptable/Usable

Table 22 summarizes the operational-feasibility results. The application obtained an overall weighted mean of 4.51, interpreted as Strongly Agree and acceptable/usable. Helpfulness and learnability received the highest category means at 4.56, followed by affect at 4.53, control at 4.51, and efficiency at 4.41.

CONCLUSION AND RECOMMENDATIONS

CONCLUSION

SignSpeak demonstrates the integration of mobile application development and machine-learning models to address communication barriers between deaf students and hearing individuals. The project identified limitations in FSL knowledge, available learning materials, communication methods, and classroom translation technologies and translated these findings into the system's learning and translation functions.

The dual-model approach was technically effective within the tested datasets. The LSTM achieved its strongest test accuracy of 97.22% for dynamic word gestures using the 80/10/10 data split, while the MLP achieved 99.89% test accuracy for alphabets and 99.96% for numbers. The 80/10/10 split consistently produced the strongest reported results.

The system was also evaluated positively by its target users. The overall SUMI-based acceptability and usability mean was 4.51, interpreted as acceptable/usable. Economic analysis projected break-even after approximately 1.39 years, while the identified hardware, software, and Android requirements supported the project's technical feasibility.

RECOMMENDATIONS

1. Use mobile devices with greater processing power, high-resolution cameras, and at least 8 GB of RAM, with up to 16 GB recommended for smoother real-time recognition.
2. Use more capable development computers, including higher processing power, sufficient RAM, dedicated graphics, and SSD storage, to reduce model-training and testing time.
3. Improve recognition under low-light conditions, different camera angles, and varying gesture speeds to strengthen real-world performance.
4. Expand the training data using additional variations such as rotation, brightness changes, and environmental differences to improve generalization and reduce overfitting.

5. Continue expanding the dynamic-word and sentence structures supported by the application so that future versions can handle more natural and complex Filipino Sign Language communication.

REFERENCES

- Abdul, N. A., Valapa, N. A., & Khaivuzzaman, N. (2023). Enhancing accessibility with LSTM-based sign language detection. *International Journal of Scientific Research in Computer Science Engineering and Information Technology*.
https://www.researchgate.net/publication/375503571_Enhancing_Accessibility_with_LSTM-Based_Sign_Language_Detection
- Aina, S., & Ixomolere, A. (2025). Multi hand gesture recognition using a multilayer perceptron (MLP) model. *Journal of Engineering Research and Reports*.
<https://www.journaljerr.com/index.php/JERR/article/view/1694>
- Albala, S. A., Kasteng, F., Eide, A. H., & Kattel, R. (2021). Scoping review of economic evaluations of assistive technology globally. *Assistive Technology*, 33(sup1), 50–67.
<https://doi.org/10.1080/10400435.2021.1960449>
- Alberto, A. A. B., Mangampo, H. M. I., Presto, M. L. V., & Herradura, T. R. (2022). Senyas: A 3D animated Filipino sign language interpreter using speech recognition. *Proceedings of the Philippine Computing Science Congress (PCSC)* 2022.
https://www.researchgate.net/publication/361330423_Senyas_A_3D_Animated_Filipino_Sign_Language_Interpreter_Using_Speech_Recognition
- Ali, M., Hosseini, S. E., Pervez, S., & Ahmad, M. (2025). Real-time recognition of NZ sign language alphabets by optimal use of machine learning. *Bioengineering*, 12(10), 1068.
<https://www.mdpi.com/2306-5354/12/10/1068>
- Ananya, B. A., & Chaithra, U. C. (2024). Smart translation for deaf people. *International Journal of Advanced Research in Computer and Communication Engineering*, 13(9), Article 13848.
<https://ijarccce.com/wp-content/uploads/2024/09/IJARCCCE.2024.13848.pdf>
- Attar, R. K., Goyal, V., & Goyal, L. (2022). Open issues and challenges in automatic sign language translation systems. *AIP Conference Proceedings*, 2455, 040003.
<https://doi.org/10.1063/5.0100903>
- Boggaram, A., Boggaram, A., Sharma, A., Ramanujan, A. S., & R. B. (2022). Sign language translation systems. *International Journal of Software Science and Computational Intelligence*, 14(1), 1–33.
<https://doi.org/10.4018/ijssci.311448>
- Cahill, A. (2023, September 18). The challenges of being deaf in a hearing school. *Schreiber Center for Pediatric Development*.
<https://www.schreiberpediatric.org/2023/09/the-challenges-of-being-deaf-in-a-hearing-school/>
- Capitle, R. M., Ferrariz, G., Garcitos, A., Amacna, R., Caballero, J., & Diel, S. M. (2025). Breaking barriers: Sign language and visual aids in a regular English classroom for hearing-impaired students. *Journal of English as a Foreign Language Teaching and Research*, 5(2), 155–165.
<https://doi.org/10.31098/jefltr.v5i2.3393>
- Cassim, M. R., Parry, J., Pantanowitz, A., & Rubin, D. M. (2022b). Design and construction of a cost-effective, portable sign language to speech translators. *Informatics in Medicine Unlocked*, 30, 100927.
<https://doi.org/10.1016/j.imu.2022.100927>
- Cayme, K. J. F., Retutal, V. A. B., Salubre, M. E. P., Cañete, L. G. S., & Astillo, P. V. B. (2024). Gesture recognition of Filipino sign language using convolutional and longshort term memory neural network. In *Lecture Notes in Networks and Systems* (pp. 94–110).

- Springer. https://doi.org/10.1007/978-3-031-47724-9_8
- Chadha, S., Kamenov, K., & Cieza, A. (2021). The world report on hearing, 2021. Bulletin of the World Health Organization, 99(4), 242-242A. <https://doi.org/10.2471/blt.21.285643>
- De Sisto, M., Vandeghinste, V., Egea-Gómez, S., De Coster, M., Shterionov, D., & Saggion, H. (2022, June 1). Challenges with sign language datasets for sign language recognition and translation. In Proceedings of the 13th Language Resources and Evaluation Conference (LREC 2022) (pp. 2450–2457). European Language Resources Association. ACL Anthology. <https://aclanthology.org/2022.lrec-1.264/>
- Diaz, J. P. (2020). The lived experiences of the deaf and hard of hearing students in a mainstream education system. The Summit. <https://shorturl.at/9HDyN>
- Ekhsan, H. M., Khairi, M. D. M., & Hamid, J. N. (2022). Enhancing sign language learning with augmented reality. Journal of Computing Research and Innovation, 7(1), 139–146. <https://doi.org/10.24191/jcrinn.v7i1.277>
- Ganmote, D. P. (2019). Need for and importance of communication skills. Research Journey: International Multidisciplinary E-Research Journal. https://www.researchgate.net/publication/383038417_Need_Importance_of_Communication_Skills
- Halder, A., & Tayade, A. (2021). American Sign Language Recognition Using GRU and LSTM.SciTePress. <https://www.scitepress.org/PublishedPapers/2021/106195/pdf/index.html>
- Howerton-Fox, A., & Falk, J. L. (2019). Deaf children as ‘English learners’: The psycholinguistic turn in deaf education. Education Sciences, 9(2), 133. <https://doi.org/10.3390/educsci9020133>
- Hutchinson, C., Cleland, J., Williams, P. A. H., Manuel, K., & Laver, K. (2024). Calculating the social impact of home automation for people with disability: A social return on investment study. Australian Occupational Therapy Journal. <https://doi.org/10.1111/1440-1630.12977>
- Jenkins, P. (2023, September 22). Why do we communicate. Brilliantio. <https://brilliantio.com/why-do-we-communicate/>
- Jetomo, C. B., & De Lara, M. L. D. (2025). Filipino sign language alphabet recognition using Persistent Homology Classification algorithm. PeerJ Computer Science, 11, e2720. <https://doi.org/10.7717/peerj-cs.2720>
- Joshi, J., Gandhi, P., & Sawant, R. (2022). Sign language certification platform with action recognition using LSTM neural networks. In 2022 International Conference on Computing, Communication and Power Technology (ICCCPT). IEEE. <https://ieeexplore.ieee.org/document/9885321>
- Kim, J.-H., & Kang, E. (2023). An Empirical Research: Incorporation of User Innovativeness into TAM and UTAUT in Adopting a Golf App. Sustainability, 15(10), 8309. <https://doi.org/10.3390/su15108309>
- Kirakowski, J., & Corbett, M. (1993). SUMI: the Software Usability Measurement Inventory. British Journal of Educational Technology, 24(3), 210–212. <https://doi.org/10.1111/j.1467-8535.1993.tb00076.x>
- Landis, J. R., & Koch, G. G. (1977). The measurement of observer agreement for categorical data. Biometrics, 33(1), 159–174. <https://doi.org/10.2307/2529310>
- Leiva, V., Rahman, M. Z. U., Akbar, M. A., Castro, C., Huerta, M., & Riaz, M. T. (2025). A real-time intelligent system based on machine-learning methods for improving communication in sign language. IEEE Access. <https://doi.org/10.1109/ACCESS.2025.10839384>

- Levesque, E., Duncan, J., & Snoddon, K. (2025). Empowering deaf youth: cutting-edge technology for independence. *Deafness & Education International*, 27(1), 1–3. <https://doi.org/10.1080/14643154.2025.2452104>
- Madrid, G. K., Villanueva, R. G., & Caya, M. V. (2022, October 3). Recognition of dynamic Filipino sign language using MediaPipe and long short-term memory. In 2022 IEEE Conference Publication. IEEE. <https://ieeexplore.ieee.org/document/9984599>
- Madronio, J. (2022). Enhancing vocabulary among grade 4 deaf and hard-of-hearing (DHH) learners through an online sign language intervention: A mixed method study. *Interdisciplinary Research Review*, 17(4), 8–14. <https://ph02.tcithaijo.org/index.php/jtir/article/view/245980>
- Madronio, J. (2025, August 31). Utilization of Philippine School for the Deaf (PSD) ESign Library to improve the signing skills of deaf, hard-of-hearing (DHH) learners, and stakeholders. *Journal of Technology and Innovation Research*. <https://ph02.tcithaijo.org/index.php/jtir/article/view/260581>
- Mallikarjuna Rao, G., Sujasri, P. A., Sowmya, C., Anitha, S., Mamatha, D., & Alivela, R. (2023). Sign language recognition using LSTM and MediaPipe. In *Proceedings of the 7th International Conference on Intelligent Computing and Control Systems (ICICCS-2023)* (pp. 1086–1091). IEEE. http://www.cse.griet.ac.in/pdfs/Student_Publications/2022-23/sp22.pdf
- Mariana, M. (2025). English as a primary language of learning and literacy among deaf learners in formal education. *Asia Pacific Journal of Educators and Education*, 40(2), 63–78. http://ejournal.usm.my/apjee/article/view/apjee_40_2_2025_3
- Masbate, A. M. M., Ong, M. B. G., San Juan, F. R. T., Paglicawan, A. K. E., & Caballero, A. R. (2023). Text to English and text to Filipino sign language translator: An Android-based mobile application with text analysis using Knuth-Morris-Pratt naive algorithm. *Elementary Education Online*, 19(4), 78–84. <https://ilkogretimonline.org/index.php/pub/article/view/6833>
- Mendoza, M., Geroso, S., & Maguate, G. (2023). Hearing the unheard: Unveiling the untold stories of hearing-impaired students in inclusive education. *ResearchGate*, 6(8). https://www.researchgate.net/publication/373266345_Hearing_the_Unheard_Unveiling_the_Untold_Stories_of_Hearing_Impaired_Students_in_Inclusive_Education
- Mohammadi, M., et al. (2022). Static hand gesture recognition for American Sign Language using neuromorphic hardware. *Neuromorphic Computing and Engineering*. Advance online publication. <https://doi.org/10.1088/26344386/ac94f3>
- Montefalcon, M. D., Padilla, J. R., & Rodriguez, R. (2022). Filipino sign language recognition using long short-term memory and residual network architecture. In *Lecture Notes in Networks and Systems* (pp. 489–497). Springer. https://doi.org/10.1007/978-981-19-2397-5_45
- Murali, R. S. L., et al. (2022). Enhancing Sign Language Recognition Through LSTM Model. IEEE Xplore. <https://ieeexplore.ieee.org/abstract/document/10933396>
- Murillo, S. C. M., Villanueva, M. C. A. E., Tamayo, K. I. M., Apolinario, M. J. V., Lopez, M. J. D., & Edd. (2021). Speak the sign: A real-time sign language to text converter application for basic Filipino words and phrases. *Central Asian Journal of Mathematical Theory and Computer Sciences*, 2(8), 1–8. <https://cajmtcs.centralasianstudies.org/index.php/CAJMTCS/article/view/92>
- Najib, F. M. (2025). Sign Language Interpretation Using Machine Learning and Artificial

- Intelligence. *Neural Computing and Applications*, 37, 841–857. Springer Nature. <https://link.springer.com/article/10.1007/s00521-024-10395-9>
- Naraja, J. R., Leyson, C. J., & Fernandez, J. R. (2025). Real-time sign language recognition and translation: A mobile solution using convolutional neural network. *Isabela State University Linker Journal of Engineering Computing and Technology*, 2(1), 20. <https://doi.org/10.65141/ject.v2i1.n2>
- National Council on Disability Affairs. (2019). Directory of Sign Language Interpreters and Organizations (as of 3 June 2019) [PDF]. National Council on Disability Affairs <https://www.ncda.gov.ph/wp-content/uploads/2019/10/e-copy-SLIDirectory-2018-as-of-3-JUNE-2019.pdf>
- Paatsch, L., & Toe, D. (2020). The impact of pragmatic delays for deaf and hard of hearing students in mainstream classrooms. *Pediatrics*, 146(Suppl. 3), S292–S297. <https://doi.org/10.1542/peds.2020-0242i>
- Pilare, L. K. M., Mahinay, J. C. A., Degamo, A. C. C., & Laureano, E. S. (2024). Filipino sign language hand gesture recognition using MediaPipe and machine learning. *International Journal of Computing Sciences Research*, 8. <http://www.stepacademic.net/ijcsr/article/view/618>
- Pilare, L., Mahinay, J., Degamo, A., & Piner, B. (2024). Filipino Sign Language Hand Gesture Recognition Using MediaPipe and machine learning. *International Journal of Computing Sciences Research*, 8, 3252-3267. <http://www.stepacademic.net/ijcsr/article/view/618>
- Rajest, S. S., & Regin, R. (2024). Enhancing communication: Real-time sign language detection with deep learning and Python. *International Journal of Advanced Computer Science and Applications*, 15(5), 123–131. https://www.researchgate.net/publication/384860891_Enhancing_Communication_RealTime_Sign_Language_Detection_with_Deep_Learning_and_Python
- Ridwang, Ilham, A. A., Nurtanio, I., & Syafaruddin. (2023). Dynamic sign language recognition using Mediapipe library and modified LSTM method. *International Journal on Advanced Science, Engineering and Information Technology*, 13(6), 2171–2180. <https://doi.org/10.18517/ijaseit.13.6.19401>
- Rogers, A. (2020). The Filipino Sign Language Act in the Philippines: Policy analysis. IDA@Gallaudet. https://ida.gallaudet.edu/honors_capstones/73/
- Roh, K., Lee, H., Hwang, E. J., Cho, S., & Park, J. (2023). Preprocessing Mediapipe keypoints with keypoint reconstruction and anchors for isolated sign language recognition. In *Proceedings of the 10th Workshop on Sign Language Translation and Recognition* (pp. 1–6). <https://aclanthology.org/2024.signlang-1.36.pdf>
- Sakthivel, E. (2025). Indian sign language to text/speech translation. *International Journal for Research in Applied Science and Engineering Technology*, 13(5), 2964–2973. <https://doi.org/10.22214/ijraset.2025.70817>
- Salvador, M. F., Brito, E. G., Huyocan, C. C., Maro, M. W. R., Prado, J. B., & GonzagaGaliccia, S. M. J. (2026). Level of awareness and willingness of allied health students in basic Filipino Sign Language: Basis for program enhancement plan. *International Journal of Research and Scientific Innovation*, 9(11), 348–356.
- Shamitha, S. H., & Badarinath, K. (2023). Sign language recognition utilizing LSTM and Mediapipe for dynamic gestures of ISL. *International Journal for Multidisciplinary Research (IJFMR)*. <https://www.ijfmr.com/papers/2023/5/6868.pdf>
- Shams, K. A., Reaz, M. R., Rafi, M. R. U., Islam, S., Rahman, M. S., Rahman, R., Reza, M. T., Parvez, M. Z., Chakraborty, S., Pradhan, B.,

- & Alamri, A. (2024). MultiModal ensemble approach leveraging spatial, skeletal, and edge features for enhanced Bangla sign language recognition. *IEEE Access*, 12, 83638–83655.
<https://doi.org/10.1109/ACCESS.2024.XXX.XXX>
- Soberano, K. (2025). Sign-Bridge: A Dual Sign Language Translator App for enhanced communication and inclusivity. *Business, Education, Social Sciences, and Technology*, 1(2), 93–102.
<https://doi.org/10.69478/best2025v1n2a014>
- Srivastava, S., Gangwar, A., Mishra, R., & Singh, S. (2022). Sign language recognition System using TensorFlow object detection API. *Research Gate*, Volume 30, 2022, 100927.
<https://www.sciencedirect.com/science/article/pii/S2352914822000752>
- Sundar, B., & Bagyammal, T. (2022, December 31). American sign language recognition for alphabets using MediaPipe and LSTM *Procedia Computer Science*, 218, 1378–1385.
<https://www.sciencedirect.com/science/article/pii/S1877050922021378>
- Swe, B. M., Phy, M., Lun, M., & Kyaw, N. (2023). Myanmar sign language recognition using deep learning. In *Proceedings of ConsILR 2023: International Conference on Linguistic Resources* (pp. 80–90).
<https://conferences.info.uaic.ro/consilr/wp-content/uploads/2024/06/10.47743ConsILR2023.10.pdf>
- Thayiparampil, J. P., Kiran, K., Binu, T. B., & Viju, S. (2023). Real-time sign language translation using TensorFlow object detection. *International Journal of Innovative Science and Research Technology*, 8(5), 581–587.
https://www.ijisrt.com/assets/upload/files/IJI_SRT23MAY581.pdf
- Triwijoyo, B. K., Karnaen, L. Y. R., & Adil, A. (2023). Deep learning approach for sign language recognition. *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika*, 9(1), 12–21. <https://doi.org/10.26555/jiteki.v9i1.25051>
- Tulio, F. F., & Edaño, L. B. (2025). Bridging the silence: Exploring the unheard voices of hearing-impaired students in learning Physical Education. *International Journal of Research and Innovation in Social Science (IJRISS)*, 9(04), 1060–1093.
<https://doi.org/10.47772/IJRISS.2025.90400082>
- Tupal, I. J. L., Cabatuan, K. M., & Manguerra, M. (2022). Implementation of a Filipino Sign Language recognition model on a Raspberry Pi 3 using CNNs and TensorFlow Lite. *Proceedings of the De La Salle University Research Congress 2022 (HCT-10)*. De La Salle University.
<https://www.dlsu.edu.ph/wp-content/uploads/2022/07/HCT10.pdf>
- Velasco, J. M., & Labuguen, V. S. (2025). Silent voices, loud learning: Exploring English learning for the Deaf. *Asia Pacific Journal of Educators and Education*, 40(2), 45–62.
<https://doi.org/10.1007/s44163-025-00629-7>
- Velasco, L. U., Macatangga, R. S., Pepito, C. P., Chua, M. G., Galupo, E. G., Jr, Del Mundo, M. J. R., & Dilla, E. L. (2024). Sign for EveryJuan: an interactive Android application that teaches Filipino sign language using hand gesture recognition. In *Advances in Social Science, Education and Humanities Research* (pp. 504–513). Atlantis Press https://doi.org/10.2991/978-2-38476-273-6_55
- Violet, I. M., & Sri, R. L. (2025). A comprehensive survey on recent advances and challenges in sign language recognition systems. *Discover Artificial Intelligence*, 5(1), Article 629.
<https://doi.org/10.1007/s44163-025-00629-7>
- Yin, K., Moryossef, A., Hochgesang, J., Goldberg, Y., & Alikhani, M. (2021). Including signed languages in natural language processing. *arXiv preprint arXiv:2105.05222*.
<https://doi.org/10.48550/arXiv.2105.05222>



Ziadat, A. H., & Al Rahmeh, A. A. (2020). Facing the deaf and hearing-impaired. International Journal of Learning, Teaching and Educational Research, 19(11), 1–19.
<https://eric.ed.gov/?id=EJ1274036>