Online-Journals.org

iJIM International Journal of Interactive Mobile Technologies

iJIM | eISSN: 1865-7923 | Vol. 19 No. 16 (2025) | OPEN ACCESS

https://doi.org/10.3991/ijim.v19i16.54595

PAPER

Development and Evaluation of an Interactive AI-Powered Robotic Assistant for Language Learning

Salvador Joaquín Torres-Caballero, Esteban Joaquín Paz-Ortega, Cristian Castro-Vargas (✉)

Facultad de Ingeniería, Universidad Privada del Norte, Lima, Perú

cristian.castro@upn.pe

ABSTRACT

Accessibility and quality of educational resources remain significant challenges, particularly in resource-limited contexts. This study presents an AI-powered interactive robotic assistant designed to enhance language learning. The assistant integrates vocal, visual, and gestural interactions using a Raspberry Pi, Arduino, a TFT display, servo motors for motion, and a natural language processing system. In addition, the system incorporates mobile-based interaction for remote control and progress monitoring. The assistant dynamically adjusts content to student needs using educational personalization algorithms and provides automated assessments to adapt to the difficulty of activities. The solution is designed with low-cost hardware, such as cardboard enclosures and basic peripherals, and prioritizes sustainability and scalability. Preliminary results indicate improved student engagement and motivation by combining auditory, visual, and kinesthetic elements. The study concludes that the robotic assistant has significant potential to transform language learning through an accessible, mobile-enhanced interactive solution.

KEYWORDS

robotic assistant, artificial intelligence (AI), interactive learning, accessibility, raspberry pi, mobile technology

1 INTRODUCTION

In recent years, advances in educational technology have profoundly transformed teaching and learning processes, allowing the adoption of innovative tools such as educational robotics and virtual assistants. These technologies have proven effective in improving the accessibility and quality of educational materials, particularly in resource-limited contexts where traditional barriers hinder equitable access to quality education [1], [2].

Globally, inequalities in access to high-quality educational resources underscore the need for scalable and adaptable technological solutions. Limitations of traditional methods, such as a lack of personalization and hands-on interaction, have driven

Torres-Caballero, S. J., Paz-Ortega, E. J., Castro-Vargas, C. (2025). Development and Evaluation of an Interactive AI-Powered Robotic Assistant for Language Learning. *International Journal of Interactive Mobile Technologies (iJIM)*, 19(16), pp. 4–21. https://doi.org/10.3991/ijim.v19i16.54595

Article submitted 2025-01-23. Revision uploaded 2025-06-14. Final acceptance 2025-06-14.

the development of educational technologies based on artificial intelligence (AI) [3]. According to [4], AI offers personalized and dynamic approaches that improve both the learning experience and outcomes by adapting educational content to the specific needs of each student.

Implementing AI-based interactive robotic assistants has emerged as a promising solution to address these educational gaps. According to [5], robotic systems can offer inclusive educational experiences, even in resource-constrained environments. [6] They reinforce this perspective by pointing out that AI-based educational assistants are especially effective in challenging contexts, where the personalization and intelligent adaptation of educational content can compensate for the lack of resources.

In this context, the present project focuses on designing and developing an interactive robotic assistant that combines inexpensive hardware and advanced technologies, such as voice recognition, visual feedback through a TFT display, and basic movements using servo motors. Built on a platform such as Raspberry Pi, the system is designed to be accessible, scalable, and adaptable to individual students' needs, improving the quality of learning in challenging educational contexts [7].

Furthermore, this project adopts an iterative approach based on agile methodologies such as Scrum, allowing for continuous user feedback and optimizing system functionality. Ethical and safety considerations are also prioritized, especially when working with vulnerable populations, ensuring respect for students' rights and dignity [8], [9].

In sum, the main objective of this project is to develop an innovative and accessible educational solution that promotes equity and improves learning outcomes in resource-limited environments. This effort seeks to address the current challenges of global education and contribute to developing a more inclusive, equitable, and sustainable education.

2 LITERARY REVIEW

The study by [10] explores how humanoid robots can foster student engagement from functional and affective perspectives. A pilot test conducted with 64 primary school students in Hong Kong showed that the group using humanoid robots experienced significant improvements in their behavioral, emotional, and cognitive engagement and intrinsic motivation. Based on self-determination theory, the results provide valuable insights for developing more effective educational robots. These findings reinforce the importance of physical-digital interaction in academic environments to enhance student learning experience and motivation.

On the other hand, [11] proposes an AI-based robotic assistant to transform video-supported learning by automatically generating interactive questions, thus improving the educational experience. The system design includes a 3D-printed face and interactive devices that allow the collection of responses in real time, which shows great potential to optimize the learning process and reduce the teaching load. Automation in the generation of adaptive content contributes to a more efficient personalization in education, aligning with the purpose of innovative education.

Regarding language learning, the pilot study by [12] evaluates the impact of a conversational chatbot on 58 English language university students (B2 and C1 levels). The results reveal significant improvements in students' language skills and positive perceptions of using such tools, highlighting the potential of AI in second language acquisition. This study highlights the ability of AI to provide immediate feedback and real-time assistance, facilitating more significant interaction and a more engaging learning experience for students.

Furthermore, [13] presents a meta-analysis of 27 studies that evaluated robot-assisted language learning (RALL). The findings indicate significant improvements, particularly in primary school students (7–12 years old), guiding the design of educational robots more suitable for second language acquisition. These results underline the need for further research into the most effective methodologies for integrating robots into language teaching, prioritizing adaptive and contextual approaches.

In higher education, [14] evaluates the use of AI-powered applications in language learning. With 151 students at two Canadian universities, the results highlight the effectiveness of these applications as structured and convenient tools with great potential to personalize the learning experience through conversational simulations and voice recognition. These tools are aligned with the growing trend of interactive mobile learning, promoting accessibility and flexibility in digital education.

Finally, the study by [15] reviews 24 publications on using AI-powered chatbots to develop English-speaking skills. The results underline that, even though the use of chatbots in this field is in its early stages, they possess enormous potential to enhance English learning by alleviating speaking anxiety, improving pronunciation, and fostering students' motivation and confidence. Furthermore, the study provides recommendations for future teachers and researchers in this field, emphasizing the importance of human-machine interaction to reinforce students' autonomous learning and speaking practice.

3 METHODS

The Scrum methodology [16] was adopted to develop the AI-powered robotic assistant, as illustrated in Figure 1, for its agile and adaptable approach to managing complex projects, allowing essential tasks to be prioritized and ensuring iterative functional deliverables. Development is organized through a dynamic product backlog that includes key tasks such as voice recognition, hardware integration, and educational content creation, which are distributed in two-week Sprints with deliverables such as initial configuration, visual interaction with TFT screens, movements with servomotors, and auditory feedback. Daily meetings ensure continuous adjustments, while at the end of each Sprint, progress is presented to experts for feedback and improvements. Each increment adds functionalities such as voice recognition, visual interaction, auditory feedback, and basic synchronized movements. Table 1 shows key technical variables, including voice recognition to detect pronunciation, visual interaction via TFT display, auditory feedback through a speaker, and interactive movements with servomotors. These elements guarantee an enriched user experience:

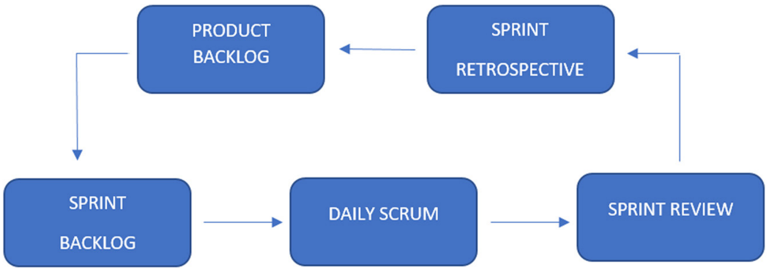


Fig. 1. Scrum system diagram

Table 1. Technical components and their functionality in robotic assistant interaction

Variable	Guy	Description
Voice recognition	Technique	Detects the user's pronunciation and words.
Visual interaction	Technique	Using the TFT screen to display expressions.
Auditory feedback	Technique	Speaker to provide real-time responses.
Arm movements	Technique	Servomotors that allow interactive gestures.

3.1 Robotic assistant architecture

Figure 2 presents the architecture of the robotic assistant designed to maximize functionality, scalability, and accessibility in educational environments. Its operating core is a Raspberry Pi, which is responsible for running the operating system, voice recognition, learning algorithms, and managing peripherals such as microphones and screens to ensure dynamic and scalable interaction [17]. Additionally, an SD card stores the operating system, libraries, and essential data, allowing updates and expansions according to the project's needs [18]. For visualization, a TFT screen is used, which improves the user experience by displaying adaptive visual expressions, encouraging more empathetic and inclusive learning [19]. This screen is activated by an Arduino, which acts as an intermediary by receiving signals from the Raspberry Pi for control, thus expanding the system's versatility and facilitating the integration of future modules thanks to its microprogramming capacity [20].

Regarding physical interaction, SG90 servomotors allow movements synchronized with educational activities, combining precision and ease of programming at a low cost [21]. Two-way communication is achieved through a speaker and microphone, where the microphone captures pronunciations for real-time correction, and the speaker provides auditory feedback, strengthening the development of language skills [22]. A portable charger ensures energy autonomy, allowing uninterrupted use in different environments [23]. Finally, a lightweight and sustainable cardboard casing optimizes the design's accessibility and functionality, favoring its integration into interactive and innovative educational contexts.

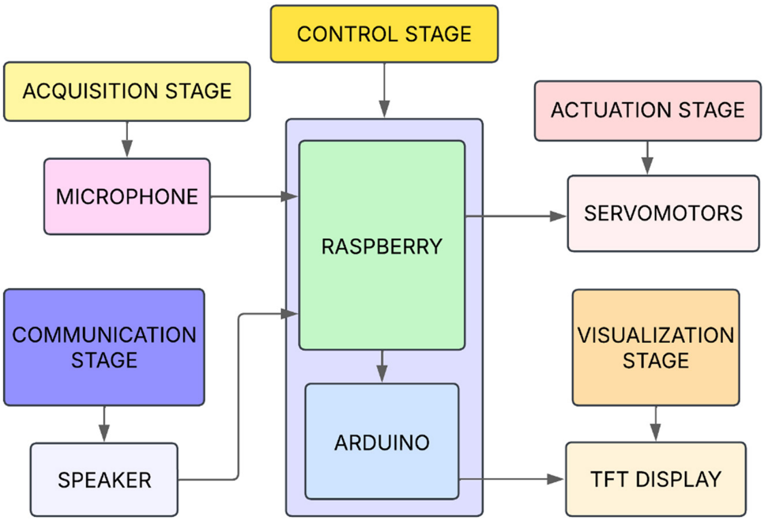


Fig. 2. Component architecture diagram

3.2 Fritzing circuit diagram (3d PCB Schematic)

Figure 3 presents the circuit design implemented in the robotic assistant, highlighting the interconnection of the leading hardware modules. The Raspberry Pi, as the central processing unit, manages communication with the peripherals through GPIO connections, facilitating system integration. The Arduino complements the control of the TFT display and servomotors, receiving signals from the Raspberry Pi to execute the programmed actions efficiently.

The diagram also illustrates the distribution of the power supply system, ensuring power supply through a portable charger and optimizing the device's autonomy in educational environments. Likewise, the connection of the microphone and speaker highlights the implementation of bidirectional communication, allowing real-time interaction between the user and the robotic assistant.

This circuit design provides a structured integration of the electronic components, ensuring functionality and scalability in interactive educational applications.

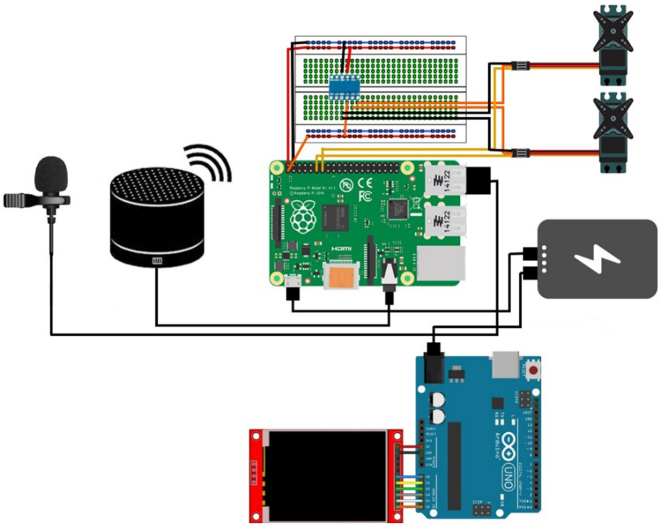


Fig. 3. Robotic assistant circuit design

3.3 Robotic assistant code flowchart

Figure 4 presents the flowchart describing the logical operation of the Raspberry Pi-based robotic assistant. This diagram details the key stages of the interaction cycle between the user and the system, highlighting the use of natural language processing (NLP) and auditory feedback.

The assistant's interaction flow starts in a standby state, ready to receive voice commands through the connected microphone, keeping an active listening to ensure that no important instructions are missed. Once the command is picked up, the system temporarily stores it for validation. Suppose the command is not recognized or does not meet the established criteria. In that case, the assistant prompts the user to repeat the instruction or displays an error message, ensuring clear and actionable instructions. If the command is valid, it is processed by NLP [24], [25] algorithms to understand the user's intent and generate an appropriate response, which can be provided through speech synthesis or by executing actions through the connected hardware. Finally, the system returns to standby to process new interactions, closing the loop if no further interaction is required. This logical flow highlights the modularity and robustness of the software design, allowing the assistant to handle complex interactions efficiently. At the same time, command validation and feedback contribute to an intuitive and effective user experience.

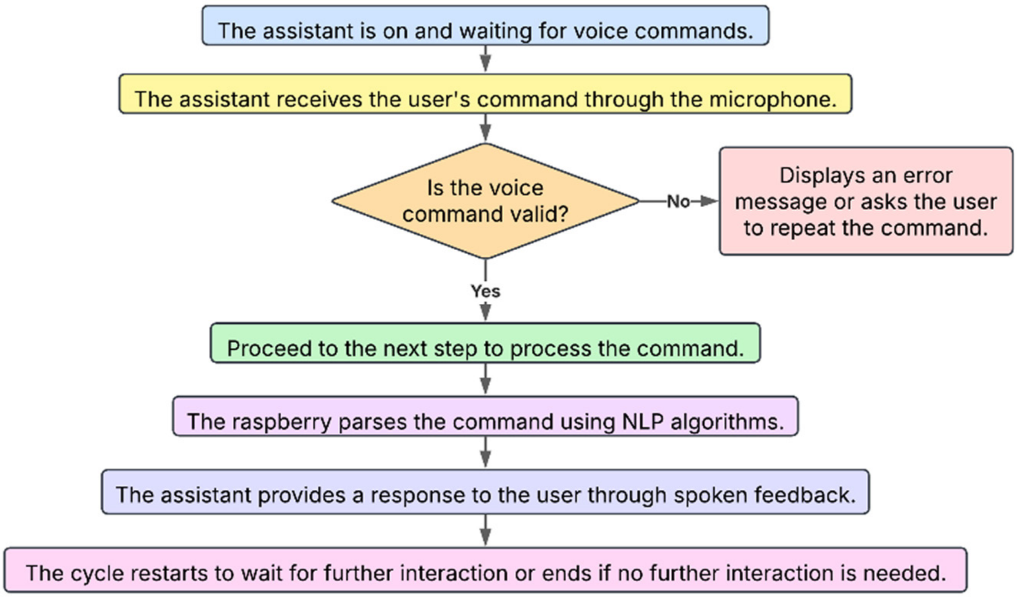


Fig. 4. Robotic assistant flowchart

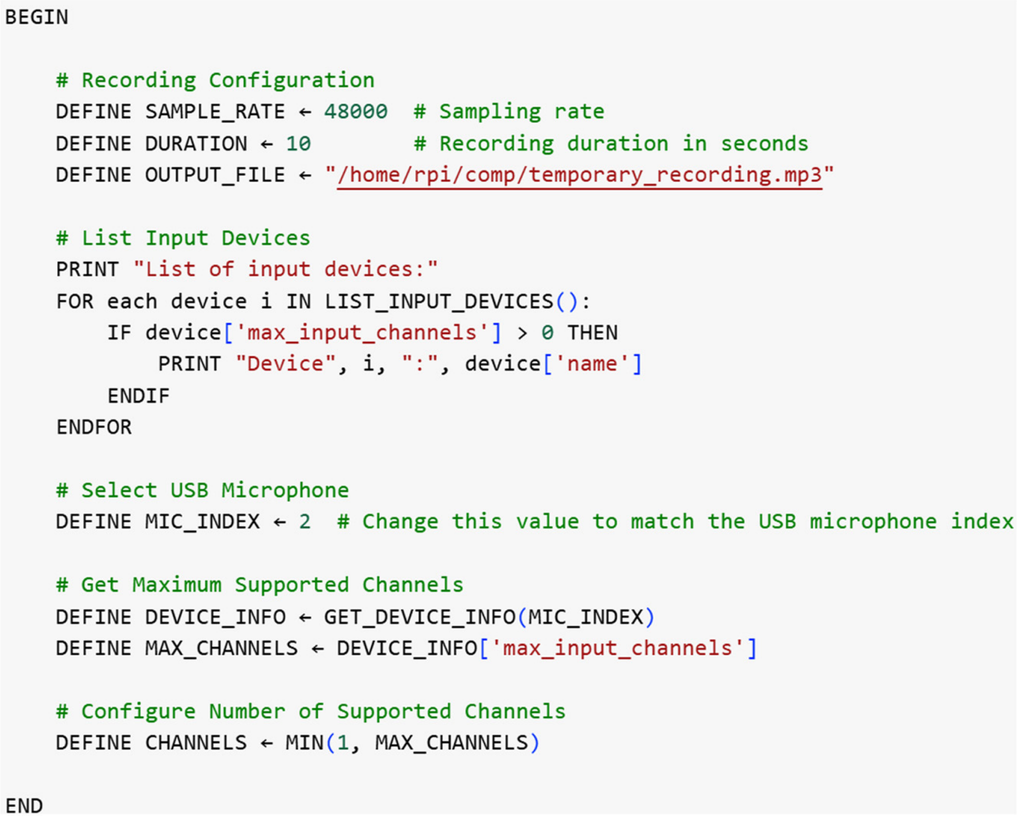
3.4 System implementation: Receiver, Process, and Player

The system comprises three main modules: 1) Receiver, 2) Process, and 3) Player, each designed to interact sequentially in the capture, processing, and playback of audio using artificial intelligence. The code implemented for each module and its functionality is described below.

Diagram of the "Receiver" module: This diagram describes the data flow from the USB microphone input to the converted audio in MP3 format storage in a shared folder. Key components include the USB microphone (audio source), device configuration (optimization for capture), audio recording, conversion to MP3 (to reduce size and optimize storage), and the shared folder (for centralized access by other modules). This module organizes the initial flow, ensuring efficient audio acquisition and storage compatible with real-time systems. (see Figure 5).

Fig. 5. Diagram of the receiver

Receiver: The receiver module aims to capture the user's voice and convert it into an audio file for further processing. The code uses the SoundDevice library for recording and Wave to store the audio file. (see Figure 6).



```
BEGIN

    # Recording Configuration
    DEFINE SAMPLE_RATE ← 48000  # Sampling rate
    DEFINE DURATION ← 10        # Recording duration in seconds
    DEFINE OUTPUT_FILE ← "/home/rpi/comp/temporary_recording.mp3"

    # List Input Devices
    PRINT "List of input devices:"
    FOR each device i IN LIST_INPUT_DEVICES():
        IF device['max_input_channels'] > 0 THEN
            PRINT "Device", i, ":", device['name']
        ENDIF
    ENDFOR

    # Select USB Microphone
    DEFINE MIC_INDEX ← 2  # Change this value to match the USB microphone index

    # Get Maximum Supported Channels
    DEFINE DEVICE_INFO ← GET_DEVICE_INFO(MIC_INDEX)
    DEFINE MAX_CHANNELS ← DEVICE_INFO['max_input_channels']

    # Configure Number of Supported Channels
    DEFINE CHANNELS ← MIN(1, MAX_CHANNELS)

END
```

Fig. 6. Pseudocode of the receiver

The system starts with audio capture using a USB microphone, which records the user's voice at a sampling rate of 48 kHz. The sampling rate, recording duration, and output format are configured to ensure file quality. Finally, the recorded audio is stored as a WAV file in a directory accessible by other modules, allowing further processing and analysis.

Diagram of the "Process" module: This module illustrates intelligent processing of captured audio. It starts from the shared folder, converting audio to text using the Whisper model, followed by advanced analysis using the Groq model. It ends with converting text to audio using Eleven Labs. The results are stored as MP3 files. This flow ensures that the system translates audio to text and generates responses in audio format, demonstrating an efficient approach for interactive applications. (see Figure 7).

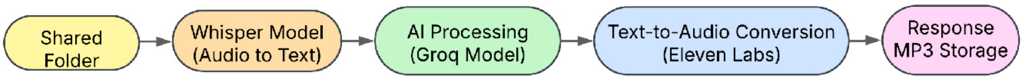
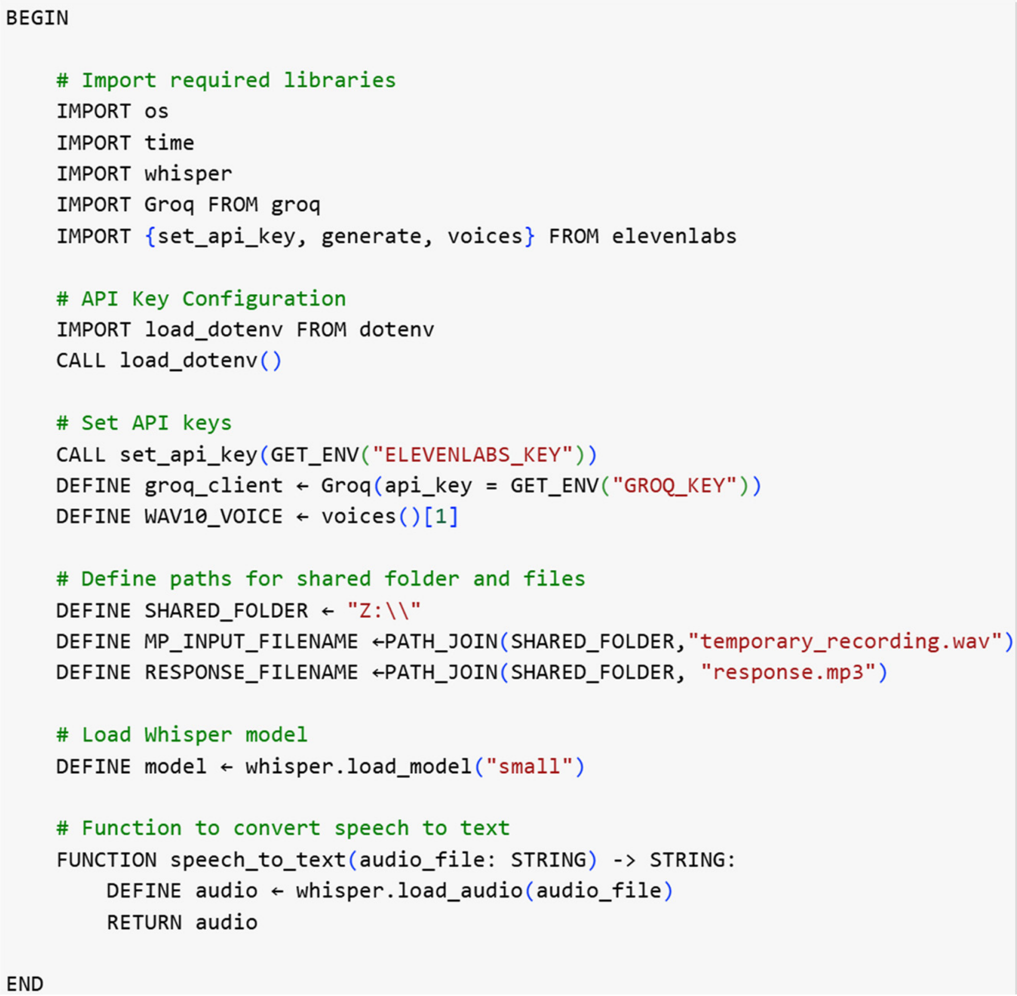


Fig. 7. Diagram of the process

Process: This module transforms audio into text, interprets the message using artificial intelligence, and generates a response in audio format. The Whisper, Groq, and ElevenLabs libraries are used for natural language processing and speech synthesis. (see Figure 8).



```
BEGIN

    # Import required libraries
    IMPORT os
    IMPORT time
    IMPORT whisper
    IMPORT Groq FROM groq
    IMPORT {set_api_key, generate, voices} FROM elevenlabs

    # API Key Configuration
    IMPORT load_dotenv FROM dotenv
    CALL load_dotenv()

    # Set API keys
    CALL set_api_key(GET_ENV("ELEVENLABS_KEY"))
    DEFINE groq_client ← Groq(api_key = GET_ENV("GROQ_KEY"))
    DEFINE WAV10_VOICE ← voices()[1]

    # Define paths for shared folder and files
    DEFINE SHARED_FOLDER ← "Z:\\"
    DEFINE MP_INPUT_FILENAME ←PATH_JOIN(SHARED_FOLDER,"temporary_recording.wav")
    DEFINE RESPONSE_FILENAME ←PATH_JOIN(SHARED_FOLDER, "response.mp3")

    # Load Whisper model
    DEFINE model ← whisper.load_model("small")

    # Function to convert speech to text
    FUNCTION speech_to_text(audio_file: STRING) -> STRING:
        DEFINE audio ← whisper.load_audio(audio_file)
        RETURN audio

END
```

Fig. 8. Pseudocode of the process

The system starts by converting audio to text using the Whisper model, which extracts the textual content from the audio file. Groq then interprets the text obtained and generates an appropriate response. Finally, ElevenLabs converts the response to MP3 audio format, thus completing the interaction cycle between the user and the system.

Diagram of the "Player" module: This diagram addresses the playback of processed audio. The shared folder allows you to select files, play them using Pygame, and synchronize physical movements with servos, creating an interactive experience. This module integrates essential components for physical and auditory output, maximizing user interaction. (see Figure 9).

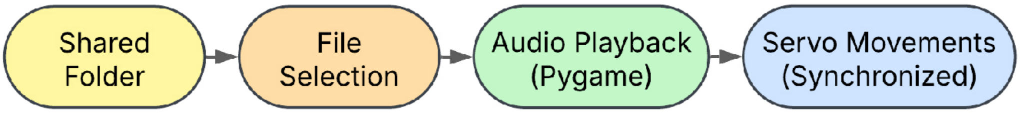
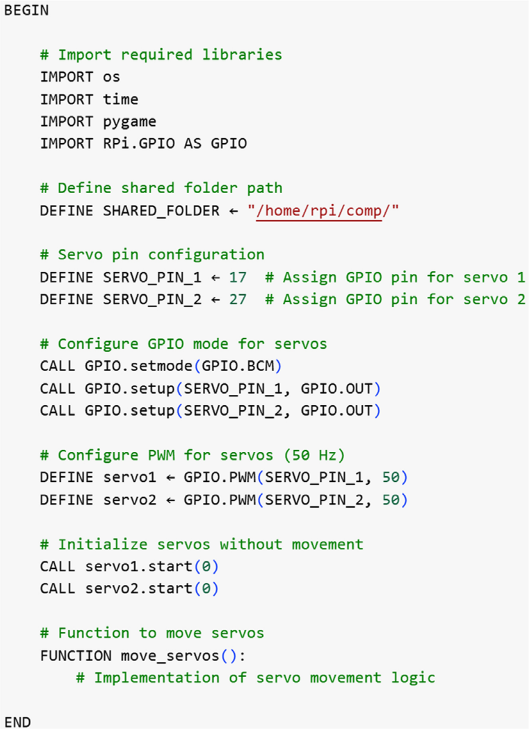


Fig. 9. Diagram of the player

Reproductive: The player accesses the file generated in the process module and plays it while simulating mechanical movements with servos. (see Figure 10).



```
BEGIN

    # Import required libraries
    IMPORT os
    IMPORT time
    IMPORT pygame
    IMPORT RPi.GPIO AS GPIO

    # Define shared folder path
    DEFINE SHARED_FOLDER ← "/home/rpi/comp/"

    # Servo pin configuration
    DEFINE SERVO_PIN_1 ← 17  # Assign GPIO pin for servo 1
    DEFINE SERVO_PIN_2 ← 27  # Assign GPIO pin for servo 2

    # Configure GPIO mode for servos
    CALL GPIO.setmode(GPIO.BCM)
    CALL GPIO.setup(SERVO_PIN_1, GPIO.OUT)
    CALL GPIO.setup(SERVO_PIN_2, GPIO.OUT)

    # Configure PWM for servos (50 Hz)
    DEFINE servo1 ← GPIO.PWM(SERVO_PIN_1, 50)
    DEFINE servo2 ← GPIO.PWM(SERVO_PIN_2, 50)

    # Initialize servos without movement
    CALL servo1.start(0)
    CALL servo2.start(0)

    # Function to move servos
    FUNCTION move_servos():
        # Implementation of servo movement logic

END
```

Fig. 10. Pseudocode of the Player

The system uses Pygame to play the generated audio, while gestures are simulated using servomotors to increase interactivity. Thus, the system combines auditory and mechanical elements that enrich the user experience.

4 RESULTS

4.1 Physical structure of the robotic assistant

Figure 11 presents the physical structure of the fully assembled and operational robotic assistant. This compact design integrates the essential components that enable its autonomous operation, highlighting the efficient integration of hardware and functional aesthetics.

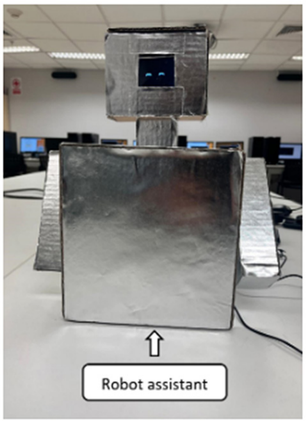


Fig. 11. Physical structure of the robotic assistant

4.2 Microphone interaction test

Figure 12 shows an interaction test where a team member uses the built-in microphone to ask the robot questions. This experiment highlights the assistant's ability to recognize voice commands and respond appropriately, demonstrating the effectiveness of the voice recognition system.

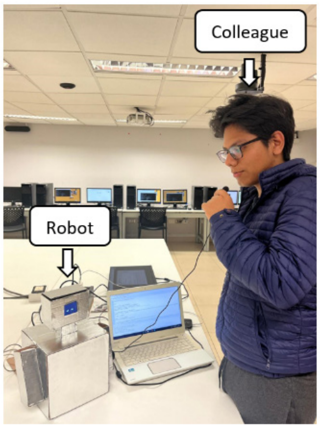


Fig. 12. Microphone interaction test

4.3 Components of the robotic assistant

Figure 13 details the components used in building the assistant, such as the Raspberry Pi, Arduino, breadboard, cables, servomotors, microphone, speaker, and portable charger. This configuration allows for autonomous operation, demonstrating a strategic distribution of the elements that ensures the system's functionality and optimization.

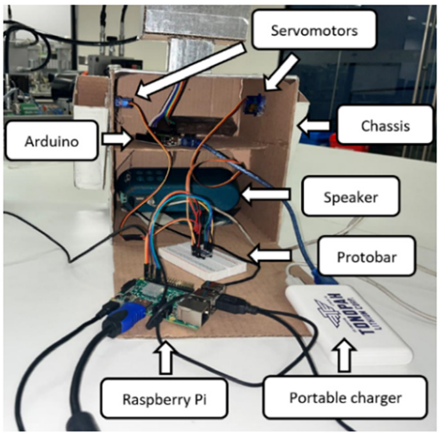


Fig. 13. Components of the robotic assistant

4.4 Implementation of servomotors

Figure 14 illustrates the internal arrangement of the servomotors in the assistant, responsible for the articulated movements of the arms. This design allows for the performance of basic gestures, improving user interaction and providing a more natural and engaging experience.

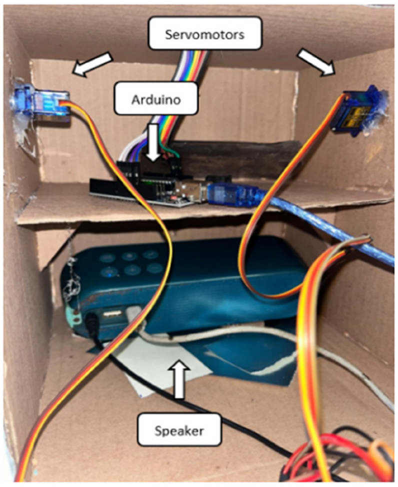


Fig. 14. Implementation of servo motors

4.5 Architecture of internal components

Figure 15 presents a detailed view of the robotic assistant's internal layout. Key components such as the breadboard, Raspberry Pi, microphone, SD card (containing the system programs and libraries), and portable charger are highlighted. This architecture ensures the assistant's smooth operation and maximizes the use of internal space.

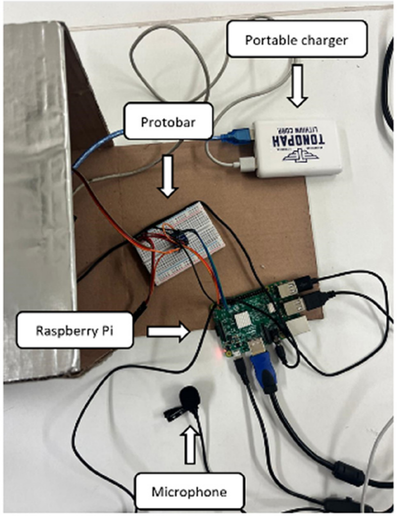


Fig. 15. The internal architecture of the robotic assistant

4.6 Visual interactions with the TFT screen

Figure 16 shows the visual interactions programmed on the TFT screen to represent facial expressions that simulate the robot's "eyes." These expressions add dynamism and visual appeal, enhancing the interactive experience and fostering a closer connection with users.

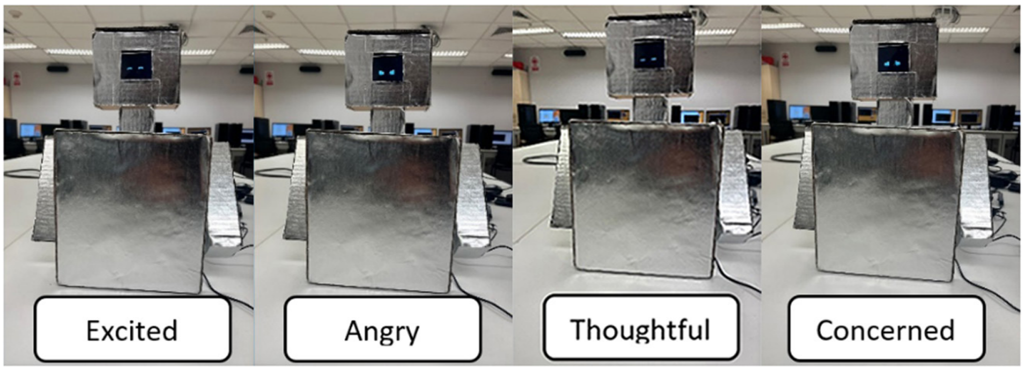


Fig. 16. Visual interactions with the TFT screen

4.7 Wiring interference test

Figure 17 verifies possible interferences in the wiring using a digital multimeter. This test was crucial to ensuring the system's stability and reliability and avoiding failures during real-time interactions.

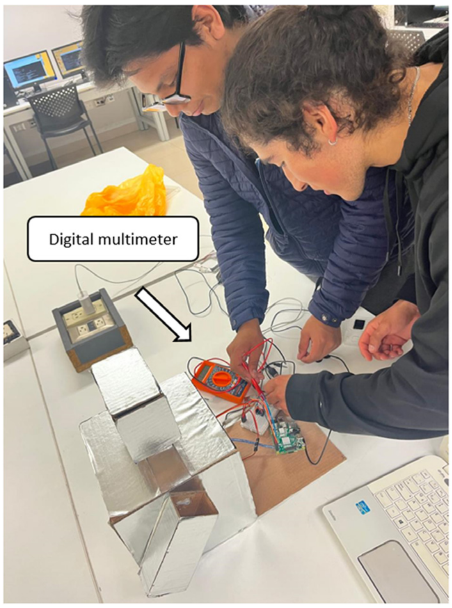


Fig. 17. Wiring interference test

4.8 Educational impact

Figure 18 shows the robotic assistant's educational impact, evidenced by significant improvements in three key metrics: vocabulary retention, pronunciation accuracy, and student motivation. Specifically, vocabulary retention increased by 25%, pronunciation accuracy improved by 30%, and student motivation grew by 25%, highlighting the system's effectiveness in enhancing learning.

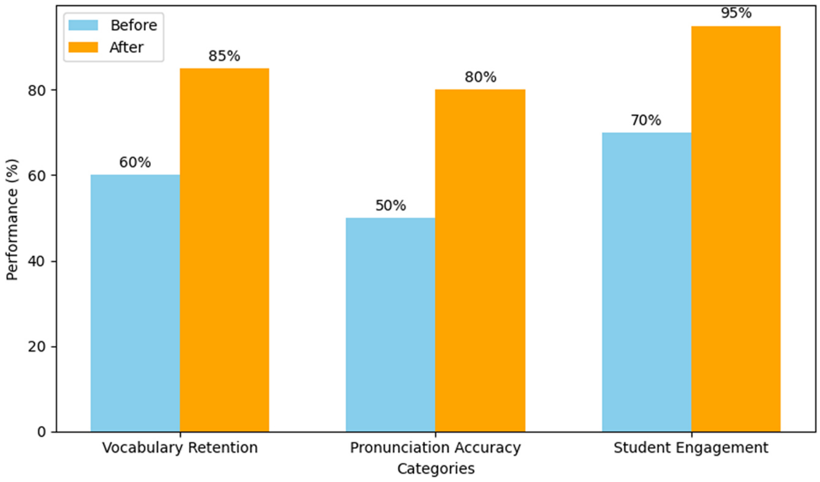


Fig. 18. Educational impact before vs. After using the robotic assistant

4.9 Progression of learning metrics

Figure 19 shows the progression of key learning metrics, including vocabulary retention, pronunciation accuracy, and student motivation, over six sessions. The results reveal a steady increase in vocabulary retention, reaching 85% in the last session; a significant improvement in pronunciation accuracy, from 50% to 80%; and continued growth in student motivation, reaching 95%, highlighting the effectiveness of the robotic assistant in enhancing learning.

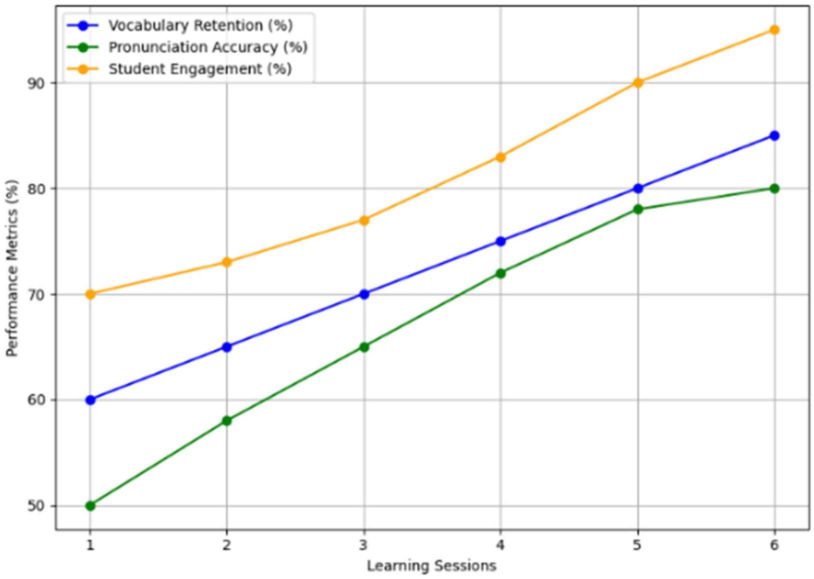


Fig. 19. Progression of learning metrics over sessions

4.10 Component specifications

Table 2 provides the technical specifications of the components used in the robotic assistant, highlighting affordability and functionality.

Table 2. Component specifications

Component	Specs
Raspberry Pi	Model 4B, 4GB RAM
TFT display	3.5 inches, 480×320
Microphone	USB Mic, 16-bit
Speaker	2 W output
Servomotors	SG90, 180° range

4.11 Comparison with traditional methods

Table 3 compares the robotic assistant with traditional methods, demonstrating superior personalization, real-time feedback, and scalability performance.

Table 3. Comparison with traditional methods

Metrics	Traditional Methods	Robotic Assistant
Personalization	Low	High
Real-Time Feedback	Limited	Comprehensive
Student Motivation	Moderate	High
Scalability	Low	Scalable

5 DISCUSSION

This study confirms the positive impact of the AI-based interactive robotic assistant on language learning, highlighting improvements in vocabulary retention (25%), pronunciation accuracy (30%), and learner motivation (25%). These findings align with previous research [10]–[12], which supports the effectiveness of technological tools in developing language skills through personalized and multisensory interactions.

The observed progress in vocabulary retention, which reached 85% in the last session, and the improvement in pronunciation accuracy, which increased from 50% to 80% over six sessions, confirm the assistant's effectiveness in reinforcing learning. Furthermore, its modular design with accessible components, such as the Raspberry Pi and SG90 servo motors, ensures replicability in resource-limited settings, while its optimized architecture facilitates autonomous user interaction.

As evidenced in Table 3, the robotic assistant demonstrates advantages in personalization, student motivation, and scalability compared with traditional methods. Its ability to provide real-time feedback and adapt content to individual needs reinforces its potential as an innovative tool in language teaching.

5.1 User experience evaluation

A user experience evaluation based on surveys and qualitative observations was incorporated to complement the analysis of the results. Thirty students who used the robotic assistant in language learning sessions were interviewed, addressing aspects such as ease of use, interactivity, and the motivation generated by the tool. The results are presented in Table 4.

Table 4. User experience evaluation results

Aspect Evaluated	Result (%)	Comments
Usability	87%	Most students found the interface intuitive and easy to use.
Interaction	90%	Real-time feedback improved pronunciation comprehension and correction.
Motivation	92%	Using the assistant made the sessions more dynamic and engaging.
Suggestions	–	It was recommended that more gestures and personalized responses be incorporated.

Teachers' perceptions of the assistant's integration in the classroom were also collected. Their usefulness was highlighted as a complementary tool for reinforcing phonetic aspects and improving students' confidence when practicing a new language.

These results demonstrate that implementing the assistant improves students' academic performance and optimizes their learning experience. Suggested improvements include incorporating greater personalization into the assistant's interaction and conducting a longitudinal evaluation to analyze its long-term impact.

5.2 Comparison with other AI-based assistants

To evaluate the proposed robotic assistant's impact, it is essential to compare it with other AI-based systems used in language learning. In previous studies, assistants such as Duolingo AI Chatbot and Google Read Along have demonstrated improvements in language practice by using NLP models and adaptive learning [13]–[15].

However, unlike these approaches, the assistant developed in this study integrates multisensory interaction through a physical robotic system, allowing for a more immersive experience. While traditional applications focus on graphical interfaces and text-to-speech, this assistant incorporates haptic and visual feedback, reinforcing learning through additional stimuli.

Table 5 presents a comparison of the main features between the proposed assistant and other AI-based solutions:

Table 5. Comparison of the proposed assistant with other AI systems

Feature	Proposed Assistant	Duolingo AI Chatbot	Google Read Along
Physical Interaction	Yes (physical robot)	No	No
Real-Time Feedback	Yeah	Yeah	Yeah
Adaptive Personalization	Yeah	Yeah	Yeah
Multisensory Support (visual, auditory, haptic)	Yeah	No	No
Focus on Pronunciation and Fluency	Yeah	Yeah	Yeah
Applicability in Educational Contexts	High	Average	Average

As can be seen, the proposed assistant offers advantages in terms of physical and multisensory interaction. These elements have improved vocabulary retention and students' confidence in speaking a new language. These results align with previous research highlighting the importance of tactile and visual interaction in foreign language teaching.

Despite these benefits, one of the system's limitations is its dependence on specific hardware, which may restrict its implementation in some educational environments. For future improvements, we recommend exploring hybrid assistant options, combining the robotic model with cloud-based platforms to expand its accessibility and adaptability to different users.

6 CONCLUSION

This study demonstrates that the AI-based interactive robotic assistant is a viable and effective solution to enhance language learning, overcoming the limitations of traditional methods. Its design and low-cost hardware, such as Raspberry Pi, Arduino, and SG90 servomotors, ensure accessibility and scalability in educational environments. The results show significant improvements in vocabulary retention (25%), pronunciation accuracy (30%), and student motivation (25%), highlighting the positive impact of multisensory interaction. Furthermore, the system excels in personalization and real-time feedback, increasing student engagement. However, it faces challenges adapting to linguistic variations and complex constructions, requiring more extensive databases and improved recognition algorithms. In future work, we propose to optimize adaptation to diverse linguistic contexts, integrate facial recognition to personalize interactions, and validate the system in educational environments with more participants. In conclusion, this robotic assistant not only improves educational accessibility and equity but also validates the potential of AI in teaching, laying the foundation for future adaptive and scalable educational technologies that promote inclusive and sustainable education globally.

7 REFERENCES

[1] M. Iqbal, N. U. Khan, and M. Imran, "The role of artificial intelligence (AI) in transforming educational practices: Opportunities, challenges, and implications," *Qlantic Journal Social Sciences*, vol. 5, no. 2, pp. 348–359, 2024. https://doi.org/10.55737/qjss.349319430

[2] R. Sajja, Y. Sermet, M. Cikmaz, D. Cwiertny, and I. Demir, "Artificial intelligence-enabled intelligent assistant for personalized and adaptive learning in higher education," *Information*, vol. 15, no. 10, p. 596, 2024. https://doi.org/10.3390/info15100596

[3] S. Mhlongo, K. Mbatha, B. Ramatsetse, and R. Dlamini, "Challenges, opportunities, and prospects of adopting and using smart digital technologies in learning environments: An iterative review," *Heliyon*, vol. 9, no. 6, p. e16348, 2023. https://doi.org/10.1016/j.heliyon.2023.e16348

[4] Z. Xu, "AI in education: Enhancing learning experiences and student outcomes," *Applied and Computational Engineering*, vol. 51, pp. 104–111, 2024. https://doi.org/10.54254/2755-2721/51/20241187

[5] A. Alam and A. Mohanty, "Integrated constructive robotics in education (ICRE) model: A paradigmatic framework for transformative learning in educational ecosystem," *Cogent Education*, vol. 11, no. 1, 2024. https://doi.org/10.1080/2331186X.2024.2324487

[6] K. N. Lam, L. H. Nguy, V. L. Le, and J. Kalita, "A transformer-based educational virtual assistant using diacriticized Latin script," *IEEE Access*, vol. 11, pp. 90094–90104, 2023. https://doi.org/10.1109/ACCESS.2023.3307635

[7] Z. Mamatnabiyev, C. Chronis, I. Varlamis, Y. Himeur, and M. Zhaparov, “A holistic approach to use educational robots for supporting computer science courses,” *Computers,* vol. 13, no. 4, p. 102, 2024. https://doi.org/10.3390/computers13040102

[8] E. C. Daraojimba, C. N. Nwasike, A. O. Adegbite, C. A. Ezeigweneme, and J. O. Gidiagba, “Comprehensive review of Agile methodologies in project management,” *Computer Science IT Research Journal*, vol. 5, no. 1, pp. 190–218, 2024. https://doi.org/10.51594/csitrj.v5i1.717

[9] P. Amajuoyi, L. B. Benjamin, K. B. Adeus, P. Amajuoyi, L. B. Benjamin, and K. B. Adeus, “Agile methodologies: Adapting product management to rapidly changing market conditions,” *GSC Advanced Research and Reviews*, vol. 19, no. 2, pp. 249–267, 2024. https://doi.org/10.30574/gscarr.2024.19.2.0181

[10] K. Y. Fung *et al.*, “Humanoid robot-empowered language learning based on self-determination theory,” *Education Information Technology*, vol. 29, pp. 18927–18957, 2024. https://doi.org/10.1007/s10639-024-12570-w

[11] C. Giladi, “Developing an AI-powered robotic assistant for interactive video-based learning: Engineering innovations and system design,” *The European Conference on Education 2024: Official Conference Proceedings*, 2024, pp. 843–852. https://doi.org/10.22492/issn.2188-1162.2024.65

[12] P. Polakova and B. Klimova, “Implementation of AI-driven technology into education – a pilot study on the use of chatbots in foreign language learning,” *Cogent Education*, vol. 11, no. 1, 2024. https://doi.org/10.1080/2331186X.2024.2355385

[13] A. Derakhshan, T. Teo, E. Saeedy Robat, M. Janebi Enayat, and A. A. Jahanbakhsh, “Robot-assisted language learning: A meta-analysis,” *Review of Educational Research,* vol. 95, no. 4, pp. 747–774, 2024. https://doi.org/10.3102/00346543241247227

[14] C. L. Yuen and N. Schlote, “Learner experiences of mobile apps and artificial intelligence to support additional language learning in education,” *Journal of Educational Technology Systems*, vol. 52, no. 4, pp. 507–525, 2024. https://doi.org/10.1177/00472395241238693

[15] J. Du and B. K. Daniel, “Transforming language education: A systematic review of AI-powered chatbots for English as a foreign language speaking practice,” *Computers and Education Artificial Intelligence*, vol. 6, p. 100230, 2024. https://doi.org/10.1016/j.caeai.2024.100230

[16] L. Bisikirskienė, L. Čeponienė, G. Vilutis, and A. Nečionytė, “Software project risk identification based on scrum artifacts,” *Appl. Sci.,* vol. 15, no. 2, p. 824, 2025. https://doi.org/10.3390/app15020824

[17] S. E. Mathe, H. K. Kondaveeti, S. Vappangi, S. D. Vanambathina, and N. K. Kumaravelu, “A comprehensive review on applications of Raspberry Pi,” *Computer Science Review*, vol. 52, p. 100636, 2024. https://doi.org/10.1016/j.cosrev.2024.100636

[18] D. Dunsin, M. C. Ghanem, K. Ouazzane, and V. Vassilev, “A comprehensive analysis of the role of artificial intelligence and machine learning in modern digital forensics and incident response,” *Forensic Science International Digital Investigation*, vol. 48, p. 301675, 2024. https://doi.org/10.1016/j.fsidi.2023.301675

[19] S. Sarkar, A. Jana, S. Maity, C. Maiti, and C. K. Bera, “IOT based performance display system using ESP32: Design and development,” *J. Phys. Conf. Ser.*, vol. 2576, p. 012002, 2023. https://doi.org/10.1088/1742-6596/2576/1/012002

[20] M. Cabanillas-Carbonell, V. Céspedes-Lluque, M. Yucra-Calla, and J. L. H. Salazar, “Mobile application with Arduino to improve the determination of the ripening status of the fruit,” *International Journal of Interactive Mobile Technologies (iJIM)*, vol. 16, no. 14, pp. 63–77, 2022. https://doi.org/10.3991/ijim.v16i14.29785

[21] S. Autsou, K. Kudelina, T. Vaimann, A. Rassõlkin, and A. Kallaste, “Principles and methods of servomotor control: Comparative analysis and applications,” *Appl. Sci.,* vol. 14, no. 6, p. 2579, 2024. https://doi.org/10.3390/app14062579

[22] R. Sharma, D. Govind, J. Mishra, A. K. Dubey, K. T. Deepak, and S. R. M. Prasanna, "Milestones in speaker recognition," *Artif. Intell. Rev.*, vol. 57, no. 58, 2024. https://doi.org/10.1007/s10462-023-10688-w

[23] S. Iqbal, N. F. Alshammari, M. Shouran, and J. Massoud, "Smart and sustainable wireless electric vehicle charging strategy with renewable energy and internet of things integration," *Sustainability,* vol. 16, no. 6, p. 2487, 2024. https://doi.org/10.3390/su16062487

[24] S. Vinoth Kumar, R. B. Saroo Raj, J. Praveenchandar, S. Vidhya, S. Karthick, and R. Madhubala, "Future prospects of large language models: Enabling natural language processing in educational robotics," *International Journal of Interaction Mobile Technology (iJIM),* vol. 18, no. 23, pp. 85–97, 2024. https://doi.org/10.3991/ijim.v18i23.51419

[25] T. A. Sandy, A. Ghufron, A. Muhtadi, and Pujiriyanto, "Text classification of Duolingo reviews on Google Play: Insights for enhancing m-learning applications," *International Journal of Interactive Mobile Technology (iJIM)*, vol. 19, no. 7, pp. 206–223, 2025. https://doi.org/10.3991/ijim.v19i07.52891

8 AUTHORS

Salvador Joaquín Torres-Caballero is a university student in the eighth semester of the Computer Systems Engineering program at Universidad Privada del Norte, Lima, Peru. He participated in the conceptualization of the solution and the definition of its core components. He contributed to the design of the system architecture, data flow models, and key functional modules. Additionally, he assisted in code development, documentation, and coordination of technical tasks within the team (E-mail: N00313919@upn.pe).

Esteban Joaquín Paz-Ortega is a university student in the eighth semester of the Computer Systems Engineering program at Universidad Privada del Norte, Lima, Peru. He focused on the implementation of backend logic and API integration, alongside frontend component development. He participated in modular programming, unit testing, and code optimization (E-mail: N00300617@upn.pe).

Cristian Castro-Vargas is an Electronic Engineer who graduated from the National University of Callao. He holds a doctorate in Education and a master's degree in University Teaching. He has extensive experience in research, electronic project development, networks, automation, and telecommunications and has publications indexed in Scopus (E-mail: cristian.castro@upn.pe).