Intelligent Human Systems Integration (IHSI), Vol. 200, 2026, 402–411
https://doi.org/10.54941/ahfe1007099

AI-Driven Text-to-Speech for Non-Playable Characters in XR Educational Environments

Sylker Silva[1], Isabelly Oliveira[1], and Rodrigo Costa[2]

[1]Federal University of Amazonas, Manaus, AM 69067-005, Brazil
[2]Fucapi University, Manaus, AM 69075-351, Brazil

ABSTRACT

This case study details integrating a text-to-speech (TTS) engine with conversational AI to create interactive, voice-acted non-playable characters (NPCs) for Virtual and Mixed Reality (VR/XR) educational environments. The project's goal was to design a course introducing students to AI, VR, and Extended Reality (XR) by having them develop these AI-driven NPCs as interactive learning agents. The study evaluated various state-of-the-art AI tools and game engines to find the best development environment. Conducted at the Ocean Center, an initiative of the Amazonas State University in Brazil, the project enhances existing AR/VR/XR courses with AI. This comprehensive framework empowers students to create engaging, intelligent virtual characters, fostering technical competence and creativity. The results suggest a promising model for integrating emerging, AI-enhanced technologies into education globally.

Keywords: Conversational AI, Text-to-speech, Virtual reality, Extended reality, Educational technology

INTRODUCTION

The convergence of immersive technologies and artificial intelligence is rapidly reshaping the landscape of education. Tools such as virtual reality (VR), augmented reality (AR), and extended reality (XR) offer unprecedented opportunities for creating interactive and engaging learning environments. When combined with conversational artificial intelligence (AI) and speech technologies, these environments become even more dynamic, enabling learners to interact with intelligent agents in natural, human-like ways. This integration holds significant promise for fostering deeper engagement, enhancing accessibility, and supporting personalized learning experiences.

Despite the increasing availability of these technologies, there remains a gap between their technical potential and their practical implementation in educational contexts—particularly in regions with limited access to specialized developers or institutional resources. Many educators and institutions are eager to explore the use of intelligent, voice-enabled non-playable characters (NPCs) in immersive environments, but face barriers related to tool complexity, integration challenges, and lack of clear methodological guidance.

Received December 2, 2025; Revised December 22, 2025; Accepted January 6, 2026; Available online February 1, 2026

This paper addresses that gap by presenting a prototype that combines conversational AI, text-to-speech (TTS), and speech-to-text (STT) services within a Unity-based VR/XR environment. The study was conducted in the context of the Metaverse Track of Samsung Ocean Manaus, a research and development initiative carried out in partnership with the Amazonas State University (UEA). Samsung Ocean is a corporate social responsibility project that offers free technology-focused courses, events, and training programs to the public, with a strong emphasis on democratizing access to innovation. Within this framework, the prototype was designed as part of an effort to provide students and developers with practical experience in emerging technologies, particularly those associated with immersive and conversational systems.

Rather than focusing on theoretical evaluation or formal classroom testing, this study emphasizes the practical steps and toolchains required to create such a system. The intention is to offer a roadmap for other educators, researchers, and developers interested in integrating similar technologies into their own immersive educational experiences. Future work will include deploying the prototype in classroom settings to evaluate its pedagogical impact and refine its design based on user feedback.

The resulting system successfully demonstrated real-time, voice-driven interaction between a user and a robot-styled NPC within an immersive virtual scene. Key features included natural language understanding via the OpenAI ChatGPT API, realistic speech synthesis with Google Cloud TTS, and automatic voice transcription through Google Speech-to-Text. A lightweight voice activity detection mechanism was also implemented to detect when the user had stopped speaking, triggering automatic transcription. The decision to use a robot avatar—with animated sound wave feedback instead of lip-syncing—proved to be an effective design alternative.

Overall, the prototype achieved its goal of integrating these technologies in a way that is functional, engaging, and accessible, laying the groundwork for future pedagogical applications and classroom deployment. Further tests with real students in an educational environment are needed and will be addressed in future studies.

BACKGROUND AND RELATED WORK

The convergence of Artificial Intelligence (AI), Text-to-Speech (TTS) technologies, and immersive environments such as Virtual Reality (VR) and Extended Reality (XR) has opened new horizons for educational innovation. These technologies, while individually transformative, offer even greater potential when integrated to create interactive, intelligent learning agents. This section provides an overview of the key technologies involved in the study, as well as relevant literature and projects that have informed and contextualized this work.

Conversational AI in Education

Conversational AI refers to systems that enable machines to interact with humans using natural language. In educational settings, these systems can serve as tutors, assistants, or peer learners, fostering dialogue-based learning

experiences. Research shows that conversational agents can enhance motivation, encourage self-regulated learning, and provide timely feedback to students (Winkler & Söllner, 2018). Moreover, AI-driven dialogue systems are now capable of adapting to learners' needs, offering personalized support and scaffolding (Graesser et al., 2018). Thus, Conversational AI is the very foundation of what this work proposes since we need a very human-line interaction between students and virtual characters.

Text-to-Speech and Voice-Enabled Systems

TTS technology has evolved significantly, producing natural-sounding speech that enhances accessibility and engagement in digital learning environments. When integrated with conversational AI, TTS allows virtual characters or agents to speak, making interactions more lifelike and immersive (Clark et al., 2020). TTS has also been recognized for supporting students with reading difficulties and improving the learning experience for auditory learners (Almalki et al., 2022).

Virtual and Extended Reality as Pedagogical Tools

Immersive technologies such as VR and XR provide learners with interactive, simulated environments that foster experiential learning. Studies have shown that VR environments can enhance spatial understanding, retention, and learner motivation (Radianti et al., 2020). XR, which encompasses both augmented and virtual reality, extends these benefits by allowing users to interact with digital content in more complex, context-aware scenarios (Bacca et al., 2014).

Non-Playable Characters (NPCs) in Learning Environments

NPCs, often used in video games, are increasingly being explored for educational applications. When designed with AI and voice capabilities, NPCs can simulate human-like interactions, guide learners, and support narrative-based learning. Such intelligent virtual agents can help build realistic social environments for practicing language, decision-making, or problem-solving skills (Lester et al., 1997; Johnson et al., 2000). Recent developments in game engines like Unity and Unreal have facilitated the creation of responsive, AI-driven characters with voice synthesis, furthering their applicability in education (Smith & DuBoulay, 2021).

Integration of Emerging Technologies

The convergence of AI, TTS, and VR/XR in education aligns with the movement toward more adaptive, interactive, and inclusive learning systems. Integrative frameworks that combine these technologies have been proposed as models for 21st-century skills development, particularly in underserved or remote contexts (Luckin et al., 2016). However, successful implementation requires not only technological resources but also pedagogical planning, institutional support, and user training.

METHODOLOGY

This study adopts an exploratory, practice-based methodology grounded in empirical evaluation of technological tools and their integration into immersive educational environments. The goal was to identify, test, and apply a set of digital resources—including a game engine, a text-to-speech (TTS) API, and a conversational AI API—to develop interactive, voice-enabled non-playable characters (NPCs) for use in Virtual and Extended Reality (VR/XR) educational contexts.

Research Design

The research was structured in two main phases:

1. Comparative Evaluation of Tools: A selection of game engines, TTS APIs, and AI-based conversational agents were evaluated to determine their technical compatibility, ease of use, pedagogical affordances, and suitability for deployment in educational contexts.
2. Prototype Development and Integration: Based on the results of the comparative analysis, the most appropriate tools were combined to develop an interactive NPC prototype embedded in a VR/XR educational scenario.

The first phase of the study involved identifying and comparing various available technologies across three main categories:

- Game Engines: platforms for building interactive 3D environments (e.g., Unity, Unreal Engine);
- TTS and APIs: Text to Speech tools for generating natural-sounding speech from text (e.g., Amazon Polly, Google Cloud Text-to-Speech, Microsoft Azure TTS);
- STT and APIs: Speech to Text tools for transcript voice to text (e.g., Amazon Polly, Google Cloud Speech-to-Text, Microsoft Azure STT);
- AI APIs: conversational agents capable of understanding and generating human-like responses (e.g., OpenAI GPT, Google Dialogflow, IBM Watson Assistant).

Each tool was evaluated according to a set of criteria relevant to the educational goals of the project, including:

- Ease of integration into a game engine environment;
- Licensing and cost structure, especially for academic and non-profit use;
- Multilingual support (with preference for English and Brazilian Portuguese);
- Speech naturalness and customization options (in the case of TTS);
- Dialog quality, adaptability, and training capabilities (for AI APIs);
- Support resources and developer community engagement.

This evaluation was conducted empirically, drawing on hands-on experimentation as well as secondary data, such as technical documentation, project case studies, and published reports from other researchers and developers. No formal benchmarking or quantitative scoring system was applied; rather, the selection process was informed by qualitative observations and experiential feedback aligned with the intended pedagogical application.

RESULTS

This section presents a comparative evaluation of tools tested for the development of interactive, voice-enabled NPCs within immersive educational environments. The evaluation followed an empirical, experience-based approach, prioritizing tools that offered a balance between ease of use, educational potential, and technical integration with VR/XR platforms.

Game Engines

Two industry-standard game engines were considered: Unity and Unreal Engine. Both platforms offer robust features for creating immersive VR/XR experiences, but key differences informed the final selection.

Table 1: Game engines comparison.

Criteria	Unity	Unreal Engine
Learning curve	Moderate (widely adopted in education)	Steeper, more suited to high-end graphics
Scripting language	C#	C++ / Blueprint (Visual)
XR integration	Strong support (Meta, SteamVR, etc.)	Strong support (native plugins)
Asset store/community	Extensive	Extensive
TTS/AI integration	Easier integration with REST APIs and SDKs	More complex SDK handling
Platform compatibility	Highly cross-platform	Strong, especially for high-end PC VR

Unity was selected for its widespread use in educational contexts, relative ease of scripting in C#, and seamless integration with third-party APIs. Its large online community and wealth of documentation made it especially appropriate for student-oriented prototyping (Unity Technologies, 2023).

Text-to-Speech (TTS) and Speech-to-Text APIs

Three TTS and STT services were compared: Google Cloud Text-to-Speech, Amazon Polly, and Microsoft Azure Speech. Evaluation focused on voice/transcription quality, latency, language support, and ease of integration.

Table 2: TTS comparison.

Criteria	Google Cloud TTS	Amazon Polly	Microsoft Azure TTS
Voice quality	High (WaveNet voices)	High	High (Neural TTS available)
Language support	Broad, incl. Brazilian Portuguese	Good	Broad
Customization	SSML, pitch/speed controls	SSML, lexicon control	SSML, styles, emotions
API integration	REST API with good documentation	REST API with AWS SDK	REST API with Azure SDK
Cost for education	$200 free/month (Cloud credit)	Free tier available	$200 free/month (Azure for Students)

Google Cloud Text-to-Speech and Speech-to-Text was selected due to its high-quality WaveNet voices, multilingual support, and ease of integration with Unity via REST APIs (Google Cloud, 2023). Also it offers $300,00 in API credits for new users, which is important for the development and testing stages.

Conversational AI

Three conversational AI platforms were assessed: OpenAI (ChatGPT API), Google Dialogflow, and IBM Watson Assistant. Each tool was evaluated on dialogue quality, context retention, ease of training, and integration potential.

Table 3: Conversational AI comparison.

Criteria	Open AI ChatGPT	Google Dialogflow	IBM Watson Assistant
Natural language output	Excellent (few-shot capable)	Good, rule-based + ML	Moderate
Context handling	Strong with memory support	Limited to session contexts	Moderate
Training complexity	Minimal prompt engineering	Requires dialog flows	Requires dialog structure
Integration flexibility	REST API; SDKs for Python/Node/Unity	Google Cloud ecosystem	REST API + IBM Cloud SDKs
Educational use license	Pay-as-you-go, free quota	Free tier available	Free Lite Plan

OpenAI's ChatGPT API was chosen for its advanced natural language generation, flexible prompt engineering, and pedagogical potential. Despite some limitations in latency or usage cost, its overall capabilities significantly enhanced the educational experience (OpenAI, 2023).

Prototype Development

Following the comparative evaluation of tools, a functional prototype was developed to demonstrate the feasibility and educational potential of integrating conversational AI and text-to-speech (TTS) within immersive reality (XR) environments. The prototype aimed to simulate a basic educational scenario where a virtual non-playable character (NPC) interacts vocally with the user in real time. In order to keep the prototype's modularity, the development phase was divided in two parts: Text to Speech and Speech to Text so they can be used together or separately depending on the project's needs.

Text to Speech

The development process involved connecting Unity to two external APIs: OpenAI's ChatGPT (for natural language understanding and response generation) and Google Cloud Text-to-Speech (for converting AI-generated text into audio). The interaction flow was structured as follows: the user inputs a question, which is sent via HTTP to the OpenAI API. Once the textual response is received, it is forwarded to the Google TTS API, which returns an MP3 audio stream. This audio is then played by Unity through an NPC's AudioSource component, simulating a spoken answer.

One of the key challenges encountered was managing the integration of these APIs within Unity's environment. Unity does not natively support dynamic JSON parsing via the dynamic keyword, which requires the creation of strongly typed data classes to deserialize API responses properly. Another challenge involved ensuring compatibility and proper handling of asynchronous web requests and audio streaming within Unity's coroutine system.

Additionally, there were API-related issues such as authentication errors (e.g., incorrect API keys), access limitations (e.g., lack of billing setup on the OpenAI account), and rate limiting—particularly HTTP 429 errors caused by exceeding the allowed number of requests per minute. These issues required careful debugging and the implementation of basic request throttling mechanisms to ensure reliable interaction without triggering API rate limits.

Voice synchronization with character animations was initially considered a potential challenge for creating realistic interactions. To avoid the complexity of implementing precise lip-syncing—particularly the need for phoneme-matching systems—a stylized approach was adopted. Instead of using a human-like avatar, a robot character model was employed, which eliminated the expectation of natural mouth movement. The character is the Ocean Robot, a mascot mostly used in Ocean's events. This 3D version was modeled in Blender 4.2. As a form of visual feedback, a simple sound wave sprite animation was displayed in the robot's mouth area to represent speech activity. This solution proved effective in conveying that the NPC was "speaking" without requiring complex facial animation, and aligned well with the technological and aesthetic context of the prototype.

Figure 1: Ocean robot (The author, 2025).

Speech to Text

To enable fully voice-driven interaction with the NPC, a speech-to-text (STT) system was integrated into the prototype using Unity's built-in microphone capture and the Google Cloud Speech-to-Text API. This allowed users to speak naturally, with their questions transcribed automatically and submitted to the conversational AI system without the need for manual input.

The implementation leveraged Unity's Microphone class to record short audio clips (up to 10 seconds) at a sampling rate of 16 kHz. The audio was converted to a linear PCM WAV format compatible with the Google STT API, then sent via an HTTP POST request in base 64-encoded form. The API returned a text transcript of the user's speech, which was subsequently forwarded to the OpenAI ChatGPT API for response generation. The resulting answer was then synthesized into voice using the Google TTS API, completing the spoken interaction loop.

To improve usability, a simple voice activity detection (VAD) mechanism was developed to automatically detect when the user had finished speaking. This was accomplished by monitoring the real-time amplitude of the microphone input. If the input volume dropped below a defined threshold for a specified duration (e.g., 1.5 seconds), and after a minimum speech time had elapsed (e.g., 0.5 seconds), the system would automatically stop recording and initiate transcription. This approach significantly streamlined the interaction process and allowed for hands-free conversational experiences.

A microphone button was added to the user interface to control recording manually when desired, along with a visual feedback indicator that displayed the recording status. These additions contributed to a more intuitive and accessible user experience, particularly in immersive VR/XR settings where text input is often impractical.

The resulting system enabled real-time, natural voice interaction with an intelligent NPC, reinforcing the project's central goal of creating engaging and pedagogically effective conversational agents in virtual environments.

CONCLUSION

This study presented the design and implementation of a functional prototype that integrates conversational artificial intelligence and text-to-speech technologies within immersive AR/VR educational environments. The project demonstrated how widely available tools—such as Unity, OpenAI's ChatGPT API, and Google Cloud services—can be orchestrated to create interactive, voice-enabled non-playable characters (NPCs) for educational purposes. The result is a flexible, replicable framework that can serve as the foundation for developing engaging and accessible immersive learning experiences.

To support replication and adaptation, the development process followed a clear sequence of steps: (1) tool selection based on educational and technical criteria; (2) implementation of a prototype using Unity as the game engine; (3) integration of OpenAI's ChatGPT API for natural language generation; (4) use of Google Cloud Text-to-Speech for voice synthesis; (5) addition of voice recognition via the Google Speech-to-Text API; and (6) implementation of a robot-based NPC with animated visual feedback to represent speech. These components were orchestrated using Unity's coroutine-based web request system and standard audio handling features. The code was structured to prioritize modularity, accessibility, and cross-platform potential, enabling others to reuse or adapt it to different educational scenarios.

Throughout the prototyping process, practical challenges were addressed, including API integration, real-time voice feedback, and user interface considerations. Alternative design choices—such as the use of a robot character to bypass lip-sync limitations—highlight the importance of creative adaptation in balancing technical constraints with pedagogical goals. The implementation of automatic voice activity detection further enhanced the naturalness and usability of the system, contributing to a more seamless interaction experience in both AR and VR contexts.

The primary objective of this paper was to provide a methodological roadmap and a curated set of tools for other educators, researchers, and developers interested in building similar applications. By documenting the technical architecture, development process, and implementation choices, this work aims to lower the entry barrier for those seeking to experiment with conversational agents in immersive educational scenarios.

Future studies will focus on testing the prototype in real classroom settings with students to evaluate its pedagogical effectiveness, user engagement, and potential for content retention. These upcoming trials will offer valuable insights into the practical application of the system in formal education and will inform further refinements and enhancements. Ultimately, this work contributes to the growing body of research exploring the integration of emerging technologies in education and supports a vision of more interactive, personalized, and inclusive learning experiences.

ACKNOWLEDGMENT

The authors would like to thank the Ocean Center team for their technical support and the opportunity to integrate applied research into the free educational programs offered to the local community. Special thanks are extended to the students and instructors who participated in early prototype tests and provided valuable feedback. The authors also acknowledge Samsung and Amazonas State University for their commitment to democratizing access to technology and innovation through open, inclusive educational initiatives.

REFERENCES

Almalki, A., Phung, D., & Tran, T. (2022). Text-to-speech technology in education: A review of uses and benefits. Computers & Education: Artificial Intelligence, 3, 100053. https://doi.org/10.1016/j.caeai.2022.100053

Bacca, J., Baldiris, S., Fabregat, R., Graf, S., & Kinshuk. (2014). Augmented reality trends in education: A systematic review of research and applications. Educational Technology & Society, 17(4), 133–149.

Clark, R. C., Nguyen, F., & Sweller, J. (2020). Efficiency in learning: Evidence-based guidelines to manage cognitive load. John Wiley & Sons.

Graesser, A. C., McNamara, D. S., & VanLehn, K. (2018). Scaffolding deep learning in computer-based environments. In D. H. Jonassen & S. M. Land (Eds.), Theoretical foundations of learning environments (2nd ed., pp. 109–136). Routledge.

Johnson, W. L., Rickel, J. W., & Lester, J. C. (2000). Animated pedagogical agents: Face-to-face interaction in interactive learning environments. International Journal of Artificial Intelligence in Education, 11, 47–78.

Lester, J. C., Converse, S. A., Kahler, S. E., Barlow, S. T., Stone, B. A., & Bhogal, R. S. (1997). The persona effect: Affective impact of animated pedagogical agents. In Proceedings of the SIGCHI conference on Human factors in computing systems (pp. 359–366). https://doi.org/10.1145/258549.258797

Luckin, R., Holmes, W., Griffiths, M., & Forcier, L. B. (2016). Intelligence unleashed: An argument for AI in education. Pearson Education.

Radianti, J., Majchrzak, T. A., Fromm, J., & Wohlgenannt, I. (2020). A systematic review of immersive virtual reality applications for higher education: Design elements, lessons learned, and research agenda. Computers & Education, 147, 103778. https://doi.org/10.1016/j.compedu.2019.103778

Smith, T., & DuBoulay, B. (2021). Creating smart NPCs: The role of AI in educational games. Journal of Educational Computing Research, 59(5), 866–890. https://doi.org/10.1177/07356331211008767

Winkler, R., & Söllner, M. (2018). Unleashing the potential of chatbots in education: A state-of-the-art analysis. Proceedings of the Academy of Management Annual Meeting, 2018(1), 16052. https://doi.org/10.5465/AMBPP.2018.16052abstract