Alqalam Journal of Medical and Applied Sciences. 2025;8(4):2499-2510
https://doi.org/10.54361/ajmas.258458

Original article

Speech-Driven Hybrid Assistive Application for Motor-Impaired Students: Embedding Agile-DBR Framework

Mokhtar Alkhattali*, Forqan Alsarwi, Kareem Abusrafa

Department of Computer Science, Higher Institute of Science and Technology, Qaser Bin Ghashir, Libya
Corresponding Email. alkhtale@hinstitute-bcv.edu.ly

Abstract
Despite the rapid evolution of digital pedagogy, Students with Upper Limb Motor Impairments (USULMI) continue to encounter difficulties accessing Educational Support Tools (ESTs), particularly WhatsApp and Google Meet. This study introduces Ewoxa Meet, a Windows-based, speech-controlled hybrid application designed to enable independent, multilingual interaction with these tools under variable connectivity conditions. In this study, we adopted an enhancement-driven methodological paradigm that embeds a Design-Based Research (DBR) approach with Agile methodology under a consolidated framework. We conducted performance tests by executing 120 speech commands across three internet scenarios, alongside usability testing with 17 university students via the System Usability Scale (SUS). The developers' assessment confirmed technical stability across diverse usage conditions, with no crashes or forced terminations observed, achieving a near-optimal successful execution rate of 98% for speech commands in target ESTs. The results indicated an overall mean reaction time of 1.92 ± 0.69 s, with stable internet achieving an average of 1.39 ± 0.27 s, weak connectivity extending to 2.74 ± 0.39 s, and offline mode yielding 1.64 ± 0.41s. Field usability testing produced a mean SUS score of 79.41 with a Cronbach's Alpha (α = 0.821), indicating high reliability and strong user satisfaction. Ewoxa Meet consequently illustrates resilient accessibility through multilingual speech control, adaptive connectivity, and a cognitively transparent interface. Moreover, these integrated methodologies scale the development of inclusive Assistive Technologies (ATs). Future work will endeavor to integrate more ESTs and investigate long-term implementation in diverse educational contexts.
Keywords. Digital Pedagogy, Educational Support Tools, DBR, Agile Framework, Motor-Impaired Students, Adaptive Speech-Controlled Applications.

Introduction

The convergence of Artificial Intelligence (AI) systems and Digital Technologies (DTs) engenders a techno entanglement, fundamentally reconfiguring the notion of inclusivity in Higher Education (HE) and necessitating the development of a framework to ensure equitable student access. Although Digital Pedagogical Transformation (DPT) provides a conceptual framework to broaden educational equity and participation, empirical evidence reveals persistent disparities, particularly among Students with Disabilities (SWDs) [1],[2],[3]. The majority of these students face significant obstacles when using digital platforms that are not designed to meet their specific accessibility needs [4],[5]. This exclusion primarily stems from the pervasive reliance on Graphical User Interfaces (GUIs) demanding complete manual dexterity, such as keyboards, mice, and touchscreens, which remain dominant in mainstream ESTs like Google Classroom, Google Meet, WhatsApp, and Telegram.

Despite the growing reliance on ESTs and AI in academic collaboration, Distance Learning (DL) and course content access do not eliminate the persistent exclusion of Students with Motor Impairments (SMIs) from meaningful participation [6],[7],[8],[9], thus consigning these learners to passive positions and limiting their academic autonomy. Although different ATs, such as Computer Vision (CV) [10],[11], brain–computer interfaces [12], gesture control [13],[14] eye-tracking systems [15], and Automatic Speech Recognition (ASR) [16], have been developed to address accessibility challenges, their real-world adoption remains limited. Significant operational constraints are represented by high costs, limited cross-linguistic accommodation, internet connectivity dependencies, and insufficient harmonization with mainstream educational environments [17],[18],[19]. These barriers are even more critical for SMIs because they directly impact engagement with ESTs and contextual adaptability across diverse educational enticements.

Consequently, establishing ATs that are intelligent and readily adaptable to real-world educational settings is essential [20]. This imperative is driven by the increasing recognition that equitable access to technology is a fundamental right that must be safeguarded even under the restrictive educational conditions [21],[22],[23],[24]. Accordingly, an exigent imperative materializes for assistive systems (ASs) characterized by both universal design inclusivity and dynamic user-requirements adaptability [25],[26]. The present study seeks to advance equitable access to technology and digital inclusion for USULMI within HE institutions. To achieve this overarching goal, the study proceeds through four interrelated objectives: enabling independent interaction with ESTs through adaptive and context-aware speech commands; developing a robust hybrid system architecture that ensures consistent functionality in both online and offline modes; incorporating multilingual speech recognition to accommodate USULMI from diverse linguistic backgrounds, enabling them to interact equally with DL environments; and designing simplified and adaptive GUIs that improve usability and accessibility. These objectives converge toward realizing an inclusive and accessible DL across

***Received**: 26-08-2025 - **Accepted**: 25-10-2025 - **Published**: 03-11-2025*

varied educational settings. Guided by the identified research problem, this study explores how adaptive and multilingual speech-command technologies can enable USULMI to interact autonomously with EST. This exploration theorizes a resilient architecture designed to ensure systemic integrity under conditions of internet volatility, with simplified GUIs to promote equitable access. This investigation gains its scholarly gravity from interrogating the inequities in access to DL environments among USULMI. Such a design substantially reduces reliance on physical input devices and enhances user autonomy, thereby fostering sustained engagement and full digital inclusion for USULMI. This application frames the technical innovation as the core enabler for the practical framework that HE institutions can adopt to ensure sustainable and inclusive learning for SWDs.

Methodology

This study adopted a hybrid methodological framework, fundamentally grounded in Design-Based Research with Agile (DBRA). This integrated framework was systematically applied to develop a hybrid intelligent system designed for USULMI, aiming to enhance digital accessibility and universal design for learning. DBR was strategically chosen for its capacity to address complex, real-world educational challenges through iterative cycles of analysis, design, evaluation, and refinement. Its iterative nature facilitates both practical solutions and the generation of generalizable design principles for ATs and accessibility [27]. The DBR framework guided the overall research phases, while the Agile methodology [28], with its characteristic sprints, was employed within the design phase of DBR for the rapid building, testing, and refinement of the prototype.

The specific placement and function of these Agile sprints within the broader DBR iterations are detailed in (Figure 1). The research commenced with a foundational analysis phase to define the core requirements. This involved a comprehensive literature review on ATs and the challenges that SMIs face in keeping pace with the digital transformation in HE. This initial phase directly informed the study's four primary objectives.

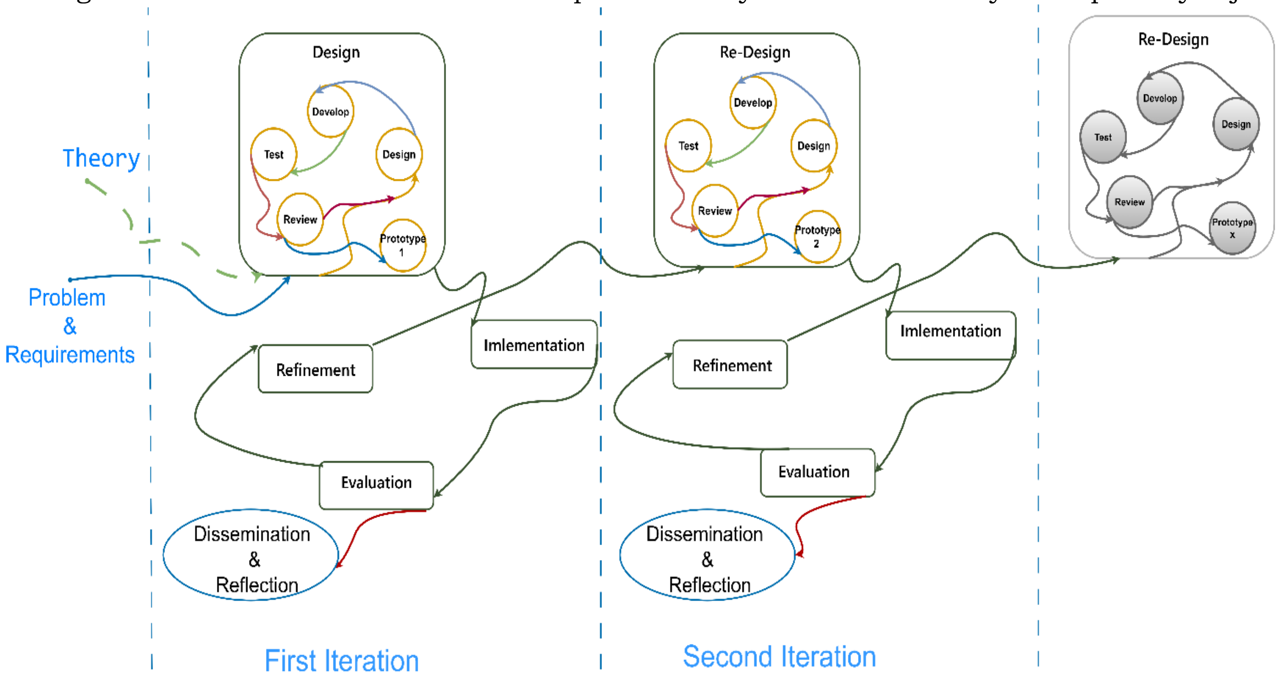

Figure 1. The Integrated Methodological Framework Illustrating DBRA in This Research

The research was then executed through two sequential DBR iterations as follows:

Iteration 1: This initial phase focused on translating the identified user requirements and foundational theoretical principles to build Prototype-1. The primary objective was to identify core usability challenges and initial design flaws through preliminary testing and feedback. This phase involved internal testing and iterative refinements conducted by the development team within Agile sprints, which enabled incremental task delivery and ensured technical robustness aligned with the original design objectives. The Prototype-1 underwent preliminary testing and evaluation by a group of specialized colleagues in ATs and software development. This formative assessment provided critical feedback on the design of GUIs and the need to implement a comprehensive set of speech commands to manage core functionalities within target ESTs—namely, WhatsApp and Google Meet—thereby providing a hands-free operational layer for these essential applications. This feedback provided the empirical basis for the subsequent refinement cycle.

Iteration 2: Building on the empirical findings from Iteration 1, this phase was dedicated to refining Prototype-1 to create Prototype-2. The resulting Prototype-2 was meticulously designed and implemented as a Windows application called "Ewoxa Meet". This phase culminated in a field evaluation to rigorously validate the prototype's real-world effectiveness and user acceptance. The Ewoxa Meet was installed on the personal

Received*: 26-08-2025 -* ***Accepted****: 25-10-2025 -* ***Published****: 03-11-2025*

devices of the participants for a one-week period, allowing for evaluation in their natural environment. The field evaluation was conducted with a cohort of 17 university students already familiar with WhatsApp and Google Meet. This group included able-bodied students; these able-bodied participants were tasked to emulate hands-free operation to rigorously evaluate the application's French speech commands.

Materials and Methods

Architectural And Technical Design

To clarify how speech signals are converted into executable commands in the developed application, (Figure 2) illustrates the structural architecture of the Ewoxa Meet application. This architecture is built upon the three pathways that analyze and execute speech commands depending on Internet access and the user–application interaction language. Across all pathways, the Microphone (Mic) remains active to continuously capture speech signals, while the listening module (listen_in_background) remains operational to ensure uninterrupted audio streaming. These pathways are detailed as follows:

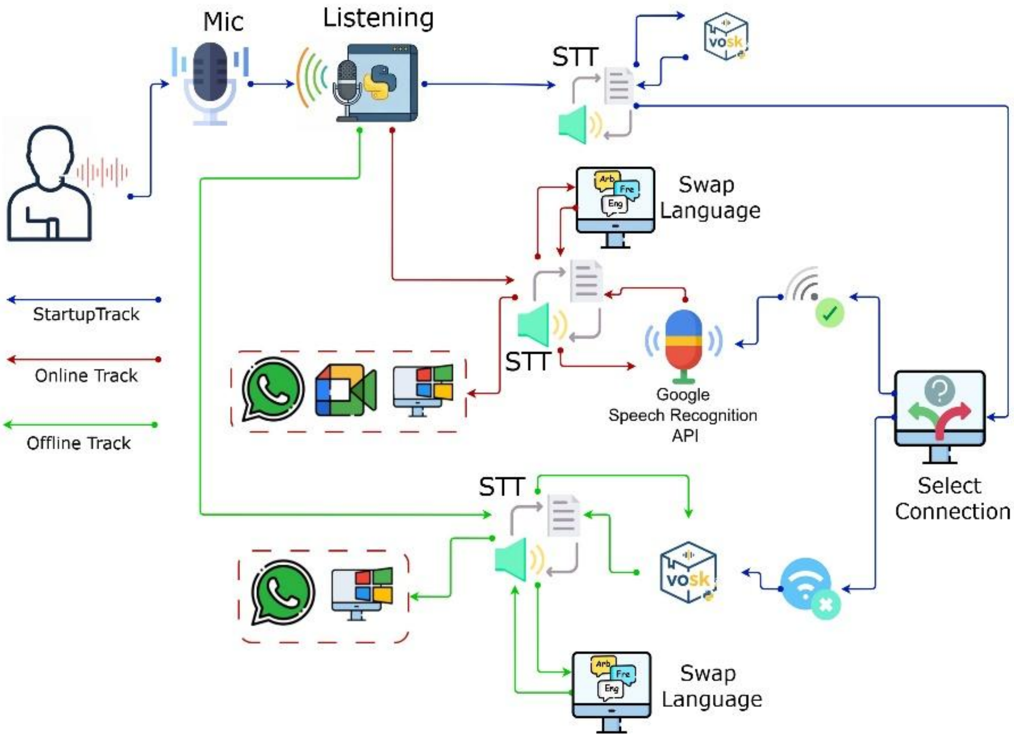

Figure 2. The architecture of Ewoxa Meet

The first pathway (blue) is automatically activated at startup and is configured to operate entirely offline. Captured speech is immediately processed by the Vosk library for text conversion. The resulting text is then programmatically translated into executable commands. The second pathway (Red) is activated when the user selects that online mode is available, in which the application utilizes the Google Speech Recognition API to provide highly accurate recognition based on the user's selected language. Subsequently, a continuous interactive loop is established, which responds to the user's speech commands for control over WhatsApp and Google Meet, as well as the execution of desktop commands within the Windows operating system. In case the internet is unavailable or unstable, a user may choose the Third Pathway (Green). This path relies on local Vosk processing to support the user's language preferences and convert STT in offline mode. Execution is limited to commands in offline mode, such as navigating stored WhatsApp chats, opening downloaded files, and playing multimedia messages.

Tools and programming language used

- ***Programming Language:*** Python version 8.13 was adopted as the strong language for developing the application due to its open-source libraries and ability to support the development of complex functionalities, including ASR, image processing, and lightweight GUI design.
- ***Development Environment:*** Visual Studio Code version (1.103.1) was utilized as a free IDE code editor within a Windows 11 environment, due to its robust support for Python, advanced debugging capabilities, and capacity for seamless project and library management. These features collectively improved workflow efficiency.
- ***GUIs Libraries:*** The GUIs of this application were built using the Tkinter (Version 8.6), a library serving as a wrapper for the Tcl/Tk GUIs toolkit. Its selection was predicated on its lightweight architecture. Complementing the core graphical interface, the Winotify library (Version 1.1.0) was integrated to

Received*: 26-08-2025 -* ***Accepted****: 25-10-2025 -* ***Published****: 03-11-2025*

https://doi.org/10.54361/ajmas.258458

manage application notifications. It was installed as a discrete dependency via (pip install winotify) to render explanatory alerts directly within the Windows environment. Supporting a title, text, an icon, and audio alerts, its primary function is to provide immediate and unambiguous feedback for critical events, such as a loss of connectivity or a language swap.

- ***Voice Libraries:*** The Vosk library processes voice locally through pre-trained neural language models, thereby eliminating any dependence on internet connectivity. For Ewoxa Meet, specific language models were selected to balance accuracy with a minimal resource footprint: English (small-en-us-0.15), Arabic (ar-mgb2-0.4), and French (small-fr-0.22). To provision a cloud-based adjunct, the SpeechRecognition library (Version 3.14.1) was employed as a wrapper to access the Google Speech Recognition API, leveraging Google's pre-trained Deep Learning Models (DLM). This cloud-based dependency ensures high-accuracy recognition of speech commands for use in Ewoxa Meet.
- ***Command Execution Libraries:*** The application integrates a synergy of three key libraries: The PyAutoGUI library (version 0.9.54) was installed via the (pip install pyautogui) command to enable programmatic control of the mouse and keyboard, thereby facilitating the automation of GUIs. During the development of Ewoxa Meet, it was specifically utilized to simulate keyboard keys and mouse clicks, which ensured the precise and reliable execution of commands. To complement this high-level control with granular event detection, the Keyboard library (version 1.13.5) was employed to gain low-level access to global key events. Its primary function within Ewoxa Meet is to capture specific key presses in real time, enabling the precise detection of both key-press and key-release events to trigger actions. As a dependency installed via (pip install keyboard), it operates independently but can be used in conjunction with PyAutoGUI to create complex automation routines. To orchestrate application-level interactions, the built-in Subprocess library (version 4.11.0) was utilized to programmatically launch and manage external processes within the Ewoxa Meet application to open executable files and launch URLs in default browsers. This process was essential for ensuring reliable coordination with GUI automation tasks.
- ***CV and Image Analysis Libraries:*** Real-time image processing and visual analytics were implemented through the OpenCV library (V4.11.0), an open-source CV library that provides a comprehensive suite of optimized algorithms for multi-scale image analysis. However, the efficacy of these functions, particularly the template matching technique harnessed for GUI element localization, is fundamentally underpinned by the NumPy library (v2.2.4). Within this synergistic architecture, a clear division of labor emerges: OpenCV furnishes the high-level algorithmic logic for comparing a predefined icon to a screen capture, while NumPy provides the requisite multidimensional array objects that enable efficient, pixel-level numerical manipulation. This integrated capability was critical, allowing the Ewoxa Meet application to direct the cursor over identified on-screen elements with high precision.

UML Sequence Diagram

To illustrate the dynamic behavior and operational flow of the application, a UML Sequence Diagram is presented in (Figure 3). This diagram meticulously maps the command-passing interactions between the participating lifelines, commencing with the user and cascading through the internal components of the Ewoxa Meet application and target ESTs.

Received*: 26-08-2025 -* ***Accepted****: 25-10-2025 -* ***Published****: 03-11-2025*

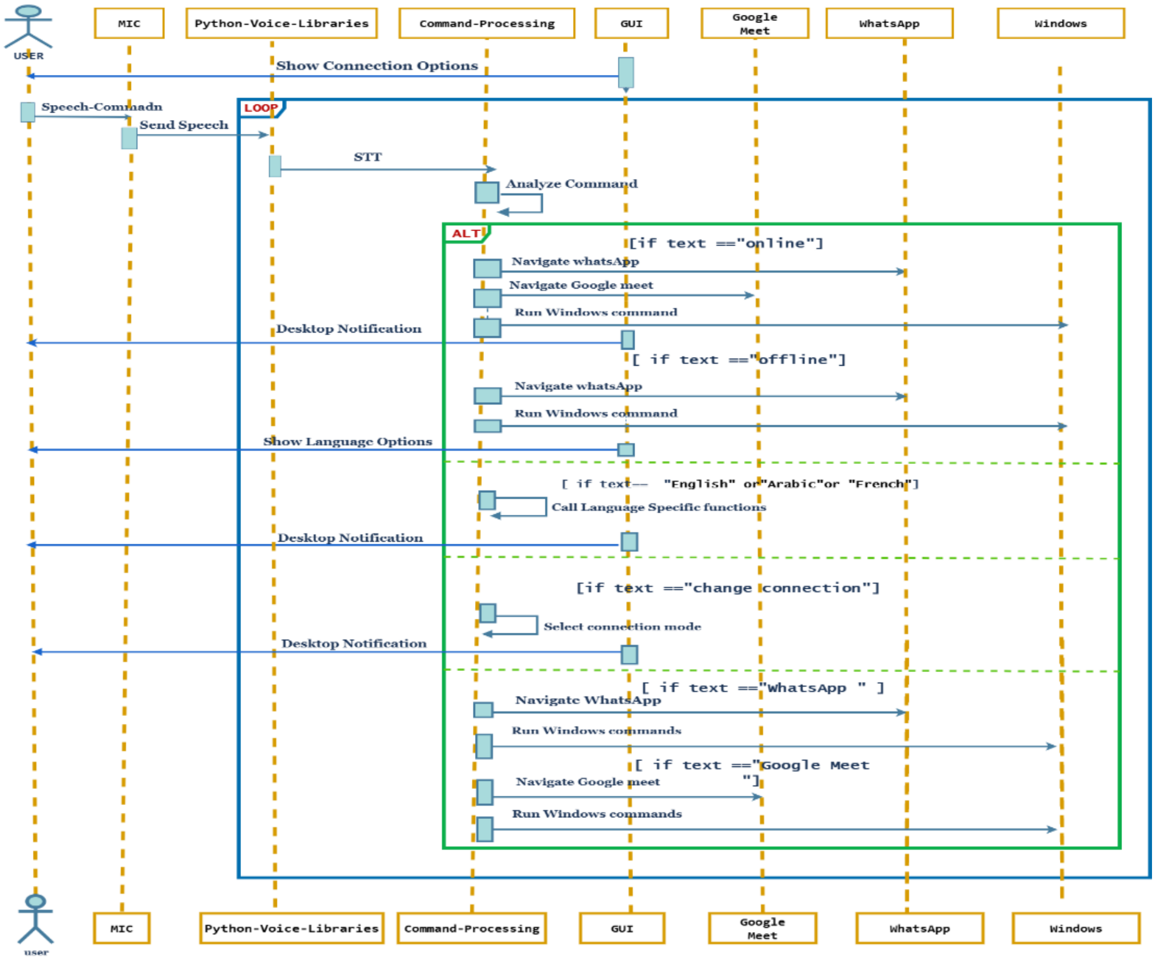

Figure 3. UML Sequence Diagram of the System's Operational Flow

The interaction cycle is initiated when the *USER* issues a speech command, which is captured by the *MIC* lifeline and forwarded to the *Python-Voice-Libraries* lifeline, which is responsible for the STT conversion. The entire process is encapsulated within a *LOOP* fragment to signify the system's design for continuous speech listening. The extracted text is then passed to the *Command-Processing* lifeline, which executes command analysis to identify the intended event. The core logic is structured as a sequence of logical conditions that direct the system to the next processing step. For instance, the condition [if text == "online"] triggers a series of concurrent messages to launch *WhatsApp* and *Google Meet* online. Conversely, other conditions, such as [if text == "change connection"] to toggle between internet modes, and [if text == "down"] or [if text == "close"], invoke specialized functions for operating-system-level operations.

The application triggers a Windows notification upon the completion of critical operations to ensure instant on-screen feedback. Additionally, the UML diagram illustrates that the integration with the *GUI* allows for further configuration, such as language selection, thereby highlighting the synergy across the system's entire linguistic ecosystem. This includes the user's spoken language, Ewoxa Meet interface language, and the WhatsApp and Google Meet language interfaces. This model's idea adopts a decoupled system into distinct lifelines, each operating in a message-driven manner, thereby ensuring new commands or application integrations can be incorporated by extending the logic within the *Command-Processing* component without altering the application's core architecture.

Results

Installation of Ewoxa Meet

A key development decision was to bundle all required setup and operational components in one installer file compatible with Windows 10 and later versions. The installation process of Ewoxa Meet was configured to ensure accessibility and to eliminate the need for technical expertise through a step-by-step wizard, as illustrated in (Figure 4). Upon completion of the installation, Ewoxa Meet is automatically added to the Windows Startup list, ensuring a seamless launch at system boot without requiring user interaction.

Received: *26-08-2025* - ***Accepted***: *25-10-2025* - ***Published***: *03-11-2025* https://doi.org/10.54361/ajmas.258458

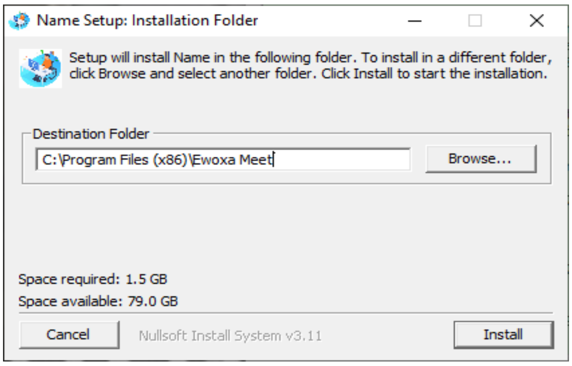

Figure 4. Ewoxa Meet installation wizard

Ewoxa Meet Operation

- ***Launch and Connectivity Configuration:*** The user is presented with a minimalist introductory interface, displayed in English only, which prompts the configuration of the connectivity mode according to current internet availability. This selection process is conducted entirely via user speech interaction, thereby eliminating the need for manual input, as shown in (Figure 5).
-

Figure 5. Minimalist Interface for Connectivity Configuration via Speech commands

- ***Language Selection:*** The application seamlessly transitions to the language selection stage once the selected internet mode has been chosen, with English designated as the default command language. This interface also relies entirely on speech interaction. Subsequently, the system executes user commands instantaneously, as shown in (Figure 6).

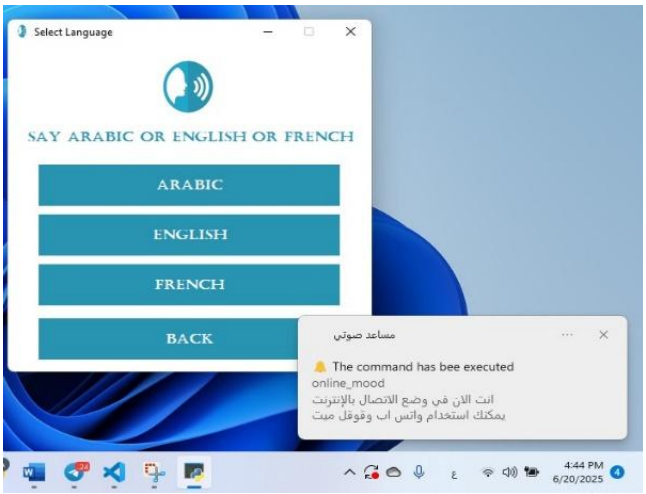

Figure 6. Voice-Controlled Interface for Selecting the Language of Speech Commands

- ***Internet Disconnection Handling:*** The system issues a real-time desktop notification, as illustrated in figures 7a and 7b, to alert the user of the disconnection during online operation. then autonomously it redirects the workflow to (Select Mode) interface, enabling the user to re-establish the appropriate operational mode without interrupting ongoing work.

***Received**: 26-08-2025 - **Accepted**: 25-10-2025 - **Published**: 03-11-2025*

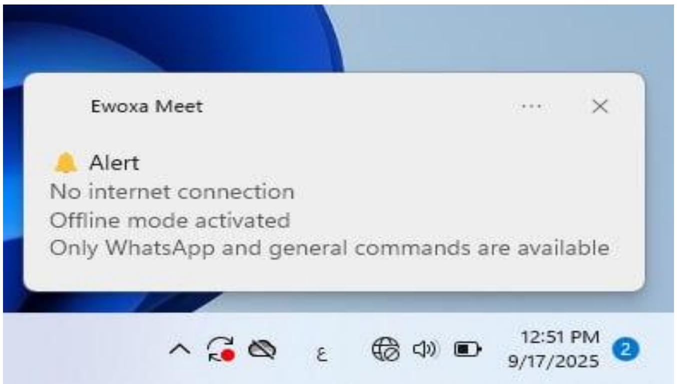

Figure 7a. Real-Time Desktop Notification for Internet Disconnection

Figure 7b. Real-Time Desktop Notification for Internet Connection

Ewoxa Meet Commands

The application provides a structured set of direct speech commands, carefully designed to correspond closely with the icons and events in the target ESTs, thereby reducing cognitive load and facilitating user interaction. The speech commands cover all core functionalities to use WhatsApp and Google meet, including selecting the operational internet mode "online" or "offline", choosing the preferred language using commands "عربي" or "English" or "Français", navigating between applications features, and executing Windows-specific actions "up" or "Enter" or "نسخ" or "اتصال" or" copier". Each command is linked to its corresponding function through keyword recognition algorithms, which interpret user spoken input and trigger the appropriate action through keyboard emulation or image icon recognition. Synchronization spoken commands with visual interface elements helps reduce user effort and improves efficiency in operations. (Table 1) presents a representative sample of available speech commands in the three languages, along with the specific functions each command executes within the application.

Table 1. Representative Sample of Multilingual Speech Commands in Ewoxa Meet

Command	Language	Application	Function
Online	English	Ewoxa Meet	To switch the application to work online
Offline	English	Ewoxa Meet	To switch the application to work offline
إنشاء مجموعة	Arabic	WhatsApp	To create a new group
Chercher contact	French	WhatsApp	To search for contact
تسجيل صوتي	Arabic	WhatsApp	To start a voice recording
أسفل	Arabic	Windows	To simulate the key cursor downward
حذف	Arabic	Windows	To simulate the key Backspace
اقفل الكمبيوتر	Arabic	Windows	To shut down the PC
Open Google Meet	English	Google Meet	To open the Google Meet
Start Meeting	English	Google Meet	To initiate a new meeting on Google Meet
Copier	French	Google Meet WhatsApp Windows	To copy text or a link
Activer caméra	French	Google Meet WhatsApp	To enable the camera
Appel	French	WhatsApp	To make a voice call
مكالمة صوتية	Arabic	WhatsApp	To make a voice call
Créer un groupe	French	WhatsApp	To create a new group
Démarrer l'enregistrement	French	WhatsApp	To start a voice recording
Changer la langue en français	French	Ewoxa Meet	To change the speech control language to French

Performance Evaluation

Evaluating the operational efficacy of the Ewoxa Meet application is essential to validate its reliability and usability in real-world assistive scenarios. As summarized in (Table 2), the developers' assessment indicated that the application demonstrates technical stability across various usage conditions, maintaining uninterrupted functionality without crashes or forced terminations. The end-to-end response time across all 120 commands yielded a mean of 1.92 seconds (s) with a standard deviation was 0.69 s. Performance

***Received**: 26-08-2025 - **Accepted**: 25-10-2025 - **Published**: 03-11-2025*

varied based on network conditions, with stable internet averaging 1.39 ± 0.27 s, weak internet 2.74 ± 0.39 s, and offline mode 1.64 ± 0.41 s. Notably, the application exhibited strong resilience to connectivity disruptions by autonomously issuing visual pop-up notifications and seamlessly transitioning into offline mode, thereby preserving continuous user engagement and control even in constrained environments. Recognition accuracy exceeded 91% across inputs in three languages, reflecting the reliability of the underlying acoustic models. Performance was programmatically enhanced through the use of advanced codes and leveraging integrated internal dictionaries, thereby ensuring execution of speech command rates exceeding 98% in the target usage environments. Finally, the GUIs reinforced usability by providing accessible interactive elements across diverse scenarios. Collectively, these findings indicate that the Ewoxa Meet application successfully meets the core technical requirements necessary to support USULMI.

Table 2. Summary of Observed System Performance Under Varied Usage Conditions

Performance Metric	Observation Summary	Notes
System Stability	No crashes or forced terminations during the testing period	Validated under multiple usage contexts
Response Time	• All 120 commands: 1.92 ± 0.69 s. • Stable internet: 1.39 ± 0.27 s. • Weak internet: 2.74 ± 0.39 s. • Offline: 1.64 ± 0.41 s.	• Latency was measured manually from speech onset to command execution on screen, reflecting the user-perceived response time. • A total of 120 commands were equally distributed across the three internet conditions.
Internet Failure Handling	• Immediate Windows notifications. • Mode switching upon disconnection.	Offline fallback triggered seamlessly
Speech Command Recognition Accuracy	• Recognition accuracy exceeded 91% across language models. • Performance was enhanced to near-optimal rates, exceeding 98%.	Based on 50 separate command samples per language.
User Interface Adaptability	• Clear fonts with excellent resolution contrast consistency. • Visual and short voice alerts that do not interfere with or distract users.	No accessibility issues were recorded during testing.

Complementing the technical performance metrics, a comprehensive user-centered evaluation was carried out using the SUS, as presented in (Table 3). Participants successfully completed tasks such as sending voice messages, initiating WhatsApp conversations, initiating Google Meet meetings, disactivating the Google Meet camera, and sharing links, demonstrating the system's effectiveness in supporting hands-free operation.

Table 3. Descriptive Statistics of SUS Questionnaire Responses (n = 17)

Item No.	Questionnaire Item	Mean Score	SD
Q1	I think I would use this application frequently.	4.12	0.70
Q2	I found the application unnecessarily complex.	1.88	0.78
Q3	I thought the application was easy to use.	4.18	0.73
Q4	I would need technical support to use this application.	1.88	0.78
Q5	The functions were well integrated.	4.24	0.66
Q6	There was too much inconsistency in the application.	1.76	0.75
Q7	I imagine most students would learn to use it quickly.	4.24	0.75
Q8	The application was cumbersome to use.	1.82	0.81
Q9	I felt very confident using the application.	4.12	0.86
Q10	I needed to learn a lot before I could use the application.	1.76	0.75

Data analysis was conducted by use of a Python-based pipeline, which generated various quantitative indicators such as means and Standard Deviations (SD). In addition, a stacked bar chart was generated to visualize the distribution of responses across the Likert scale for each SUS item, providing a concise overview of user perception patterns, as illustrated in (Figure 8).

***Received**: 26-08-2025 - **Accepted**: 25-10-2025 - **Published**: 03-11-2025*

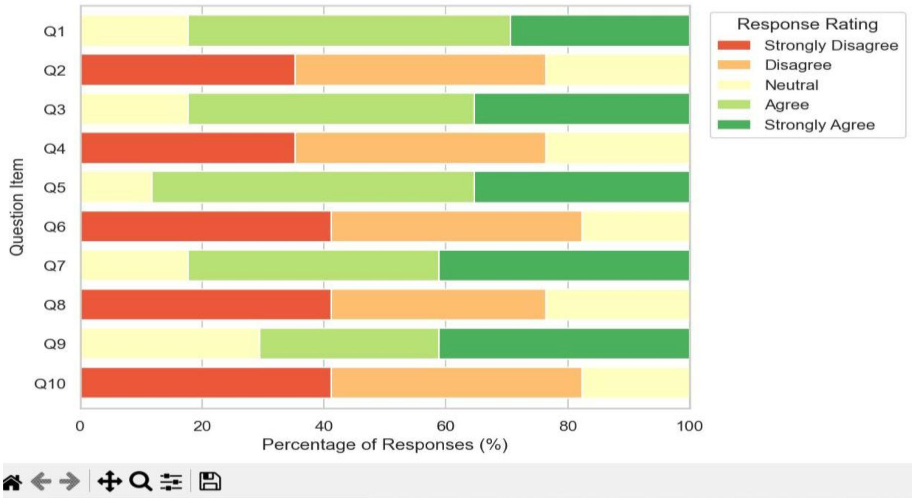

Figure 8. Distribution of Responses Across the Likert Scale for Each SUS Item

The results summarized in (Tables 4 and 5) demonstrated strong internal reliability, with an average SUS score of 79.41 exceeded the widely recognized benchmark of 68 points, situating the system within usability range (Good). Moreover, Cronbach's Alpha is classified as Good (α=0.821). These statistical findings support the operational performance results reported in (Table 2). They confirm that Ewoxa Meet demonstrates high usability and effectively promotes accessibility and user interaction in DL environments.

Table 4. Descriptive Statistics for SUS Scores

Mean SUS Score	SD	Interpretation
79.41	11.74	Good

Table 5. Reliability Analysis of the SUS

Scale	Number of Items	Cronbach's Alpha	Interpretation
SUS	10	0.821	Good

Discussion

This study presents and evaluates the Ewoxa Meet application, a novel hybrid voice assistant designed to address critical gaps in ASs for USULMI. The findings are demonstrating high technical stability, robust performance, and excellent usability, not only confirming the application's efficacy but also providing strong validation for our central hypothesis: that the integration of multimodal interaction, flexible internet connectivity, and multilingual support is essential for achieving effective digital accessibility. The ensuing discussion contextualizes this study's findings by comparing them with modern scholarship to explore their consonance and unique contribution.

Overcoming Language and Connectivity Limitations in Existing Systems

The scientific literature reveals that the majority of ASs are heavily built around the English language. This orientation limits accessibility and restricts its global applicability. While notable achievements have been made, such as H_P_G application [29] and developer-centric utilities [30], alongside practical frameworks for desktop control that leverage recent advances in DLM [31],[32], these successes remain circumscribed by monolingual contexts. Ewoxa Meet directly addresses this limitation by supporting voice commands in English, Arabic, and French. This is not merely an incremental enhancement but a functional necessity, considering the complex morphology and wide dialectal diversity that make Arabic one of the most challenging languages for ASR models [33],[34]. The systematic integration of Arabic—a language conspicuously absent from the corpus of most ASs—is a substantial step toward inclusivity, expanding accessibility to user groups previously marginalized in DL environments. This advancement in linguistic inclusivity, however, starkly contrasts with the persistent architectural rigidity revealed in the scholarly literature, where a critical deficit persists in the development of adaptive hybrid frameworks endowed with the capacity for operating seamlessly across both online and offline modes. High-performance architectures such as Google's Universal Speech Model (USM) [35] and OpenAI's Whisper [36] remain entirely dependent on cloud infrastructure, thereby raising significant privacy concerns due to the transmission of user data to third-party servers and the continuous need for network connectivity [37],[38]. Conversely, offline solutions such as Vosk and Sopare provide pre-trained frameworks that achieve real-time accuracy rates exceeding 80% [39], yet their functionality remains limited compared to cloud-based models. Ewoxa Meet transcends this binary divide through its automatic connectivity-switching mechanism. Its ability to issue real-time notifications and autonomously redirect to offline mode upon network disruption effectively resolves a

***Received**: 26-08-2025 - **Accepted**: 25-10-2025 - **Published**: 03-11-2025*

practical and critical challenge. This adaptive behavior leverages internet modes as a contextual parameter to directly address the absence of unified frameworks capable of seamless operation across connectivity modes, as highlighted in the reviewed literature.

A Hybrid Multimodal Approach to Seamless On-Screen Interaction
The fundamental innovation of Ewoxa Meet is its successful, real-time integration of speech and vision inputs. Existing ASs typically handle these modalities in isolation, performing either ASR [29],[40] or CV-based analysis [41] without enabling a speech command to directly trigger visual scrutiny of the screen and act upon the results. For instance, while InsightNav utilizes CV to organize screen layouts, it does so independently of the speech command flow [14]. Similarly, Waver fuses an AI-driven hand-gesture mouse to deliver hands-free computer control [42], but its operation is not initiated by speech commands Our developed application bridges this functional dissociation by seamlessly linking spoken commands with icon-based image recognition and keyboard emulation. In doing so, it concretizes the vision emphasized in recent studies for a context-aware and adaptive interactive system [43],[44]. In the same context, our system's speech-recognition accuracy of 98 % across three languages is highly competitive with state-of-the-art models. For instance, a Whisper variant optimized for users with dysarthria, which reached 95% [45], and also another study demonstrated a CNN-based system achieving 91.8% accuracy [46]. More importantly, the mean response time of 1.92s, even under weak internet conditions (2.74 ± 0.39s), demonstrates that the hybrid integration does not compromise speed. This efficiency, achieved through the use of advanced codes and leveraging integrated internal dictionaries, directly challenges the concern regarding computational demands and the need for complex DLM infrastructures [47],[48].

User-Centric Validation and Perceived Usability
While technical metrics are essential, the acceptance of ATs ultimately hinges on the user experience. The mean SUS score of 79.41, which places Ewoxa Meet in the Good category, is compelling evidence of successful user-centered design. This result stands in stark contrast to findings indicating that Arabic voice user interfaces consistently fail to meet user expectations due to high error rates and unintuitive command structures [49],[50],[51], and a lack of voice interaction in UI/UX frameworks, which ultimately compels users to abandon their tasks [52]. The consistently high scores on positively-worded SUS items Q3 and Q9 align to lower cognitive load; participants perceived the application as effortless to operate, a perception commonly associated with minimalist interfaces and a restricted set of direct-speech commands [38],[26]. The high reliability of the questionnaire (Cronbach's Alpha = 0.821) further confirms the consistency of these positive perceptions among participants, providing robust validation that the Ewoxa Meet application is not only functional but is also perceived as effective and reliable by its target users.

Limitations and Avenues for Future Research
Notwithstanding the study's manifold contributions, its scope inherently introduces certain delimitations that, in turn, illuminate promising trajectories for subsequent inquiry. The small sample of this study—limited to 17 participants—curtails the generalizability of its findings, despite the depth of insight it may afford. This is particularly salient for individuals exhibiting more profound motor deficits that may impede speech intelligibility, a demographic not adequately represented in this work. Secondly, the empirical validation was circumscribed to two applications: Google Meet and WhatsApp. Despite their ubiquity in digital educational milieus, this constrained purview limits a comprehensive assessment of the system's performance across a more expansive DL ecosystem of diverse tools. Accordingly, these limitations mandate a more strategically delineated trajectory for future scholarly inquiry. More expansive trials are imperative to ascertain the system's resilience across a more heterogeneous user cohort, particularly those with more severe motor impairments that might affect speech clarity. Furthermore, extending the system's integration to encompass complementary educational platforms, such as Google Classroom, and emergent tools like generative-AI models, represents a critical next phase. Finally, probing the longitudinal adoption patterns and the system's tangible impact within authentic educational contexts over extended periods would yield invaluable insights into its pragmatic utility and the nuanced vicissitudes of the user experience.

Conclusion
This study advances equitable technology access and digital inclusion for USULMI. Utilizing the DBRA methodology, we have developed and rigorously evaluated Ewoxa Meet, an innovative speech-enabled application that embodies the research's methodological and practical innovation. Ewoxa Meet enables independent interaction with Google Meet and WhatsApp through multilingual speech commands (Arabic, English, French), supports adaptive operation in both online and offline modes, and provides a simplified, adaptable GUI to ensure an inclusive and usability user experience. Experimental evaluations validated the application's reliability and stability, with the key findings summarized as follows:

- Mean response time for 120 selected commands: 1.92 ± 0.69 s.
- Internet-specific response times: offline 1.64 ± 0.41 s; stable 1.39 ± 0.27 s; weak 2.74 ± 0.39 s.
- Speech-recognition accuracy: 98% across the three languages.

***Received**: 26-08-2025 - **Accepted**: 25-10-2025 - **Published**: 03-11-2025*

- Usability metrics: α = 0.821; mean SUS = 79.41.

These findings demonstrate that the enhanced methodological framework employed in this study—namely DBRA, which embeds DBR with Agile—provides a scientifically rigorous approach for designing applications that are both effective and inclusive. The results also revealed that Ewoxa Meet delivers stable, reliable performance, enhances digital independence, and reduces reliance on manual input or physical devices, thereby promoting inclusive participation in DL environments. However, future research should further validate the model with larger cohorts, extend integration to complementary assistive applications, including Google Classroom, generative-AI models, and investigate long-term adoption in real-world educational contexts. Overall, this study contributes a distinctive and transferable model that provides HE institutions with a clear pathway for designing and implementing inclusive digital educational environments for SWDs.

Conflict of interest. Nil

References

1. Makarova EA, Makarova EL. Blending pedagogy and digital technology to transform educational environment. Int J Cogn Res Sci Eng Educ. 2018;6(2):57–66.
2. Al-Motrif A. Enhancing learning experiences for students with learning disabilities through digital pedagogies: insights from Saudi schools. Interact Learn Environ. 2025;1–28.
3. Zhadlenko I, Morenko O, Nikolenko L, Lastochkina O, Mykytenko M. Enhancing Educational Access: Digital Solutions for Students with Special Needs. Am J Psychiatr Rehabil. 2025;28(1):465–75.
4. Bercaru V, Popescu N. A systematic review of accessibility techniques for online platforms: Current trends and challenges. Appl Sci. 2024;14(22):10337.
5. McNicholl A, Casey H, Desmond D, Gallagher P. The impact of assistive technology use for students with disabilities in higher education: a systematic review. Disabil Rehabil Assist Technol. 2021;16(2):130–43.
6. Pagliara SM, et al. The integration of artificial intelligence in inclusive education: A scoping review. Information. 2024;15(12):774.
7. Chen L, Chen P, Lin Z. Artificial intelligence in education: A review. IEEE Access. 2020;8:75264–78.
8. Pierrès O, Christen M, Schmitt-Koopmann FM, Darvishy A. Could the use of AI in higher education hinder students with disabilities? A scoping review. IEEE Access. 2024;12:27810–28.
9. Batanero-Ochaíta C, Fernández-Sanz L, Rivera-Galicia LF, Rueda-Bernao MJ, López-Baldominos I. Estimation of interaction time for students with vision and motor problems when using computers and e-learning technology. Appl Sci. 2023;13(19):10978.
10. Xuanfeng D. Gesture recognition and response system for special education using computer vision and human–computer interaction technology. Disabil Rehabil Assist Technol. 2025;1–18.
11. Bhat S, Bhat P, Kolekar S. From Vision to Voice: A Multi-Modal Assistive Framework for the Physically Impaired. IEEE Access. 2025.
12. Rajarajan S, Kowsalya T, Gupta NS, Suresh PM, Ilampiray P, Murugan S. IoT in Brain-Computer Interfaces for Enabling Communication and Control for the Disabled. In: 2024 10th Int Conf Commun Signal Process (ICCSP). IEEE; 2024. p. 502–7.
13. Qi J, Ma L, Cui Z, Yu Y. Computer vision-based hand gesture recognition for human-robot interaction: a review. Complex Intell Syst. 2024;10(1):1581–1606.
14. Sharma AK, Wakchaure SB, Twinkle T. InsightNav-Empowering Desktop Navigation with Computer Vision. In: 2025 IEEE Int Students' Conf Electr Electron Comput Sci (SCEECS). IEEE; 2025. p. 1–7.
15. Chhimpa GR, Kumar A, Garhwal S, Dhiraj. Development of a real-time eye movement-based computer interface for communication with improved accuracy for disabled people under natural head movements. J Real Time Image Process. 2023;20(4):81.
16. Alkhattali M, Dow M. Improve Navigation of the Webmail Interface using Arabic Voice Commands for Elderly and Disabled Employees: A Computer Application. Acad Ind Res Collab Cent (AIRCC). 2023;13:223–35.
17. Ariza JÁ, Pearce JM. Low-cost assistive technologies for disabled people using open-source hardware and software: a systematic literature review. IEEE Access. 2022;10:124894–927.
18. Merugu S, Kumar A, Ghinea G. A review of some assistive tools and their limitations. Track Trace Manag Syst Dementia Intellect Disabil. 2022;7–17.
19. Lourenço JW, de Jesus PAC, Schaefer JL, Canciglieri Junior O. Challenges and strategies for the development and diffusion of assistive technologies. Disabil Rehabil Assist Technol. 2025;1–14.
20. Restianty A, Sumartias S, Hadisiwi P, Hafiar H. Digital applications as assistive technology for students with disabilities. ASEAN J Sci Eng. 2024;4(3):445–70.
21. Hendricks P, Wahl L, Stull J, Duffield J. From policy to practice: Achieving equitable access to educational technology. Inf Technol Disabil. 2003;9(1).
22. Lee H, Templeton R. Ensuring equal access to technology: Providing assistive technology for students with disabilities. Theory Pract. 2008;47(3):212–9.
23. Simeoni R, Pirrera A, Meli P, Giansanti D. Promoting universal equitable accessibility: an overview on the impact of assistive technology in the UN, UNICEF, and WHO Web Portals. Healthcare (Basel). 2023;2904.
24. Hushin H. Increasing Global Access to Education with Digital Technology. Int J Educ Digit Learn. 2025;3(4):167–76.
25. Stefanidi Z, Margetis G, Ntoa S, Papagiannakis G. Real-time adaptation of context-aware intelligent user interfaces, for enhanced situational awareness. IEEE Access. 2022;10:23367–93.
26. Mosquera-DeLaCruz JH, Loaiza-Correa H, Nope-Rodríguez SE, Restrepo-Girón AD. Human-computer multimodal interface to internet navigation. Disabil Rehabil Assist Technol. 2021;16(8):807–20.

***Received**: 26-08-2025 - **Accepted**: 25-10-2025 - **Published**: 03-11-2025*

27. Obczovsky M, Bernsteiner A, Haagen-Schützenhöfer C, Schubatzky T. Systematizing Decisions in Design-Based Research: From Theory to Design. Sci Educ. 2025;109(2):523–36.
28. Theobald S, Schmitt A, Diebold P. Comparing scaling agile frameworks based on underlying practices. In: Int Conf Agile Softw Dev. Springer; 2019. p. 88–96.
29. Alkhattali M. Development of a Computer Application for Handicapped People to Use Gmail. Int J Emerg Technol Innov Eng. 2019;5(11). Available from: https://ssrn.com/abstract=3492254
30. Rahul S, Santhosh P, Rajvignesh BS, RS P. Design and Development of Smart AI-Powered Desktop Assistant for Productivity Enhancement. In: 2025 Int Conf Adv Power Signal Inf Technol (APSIT). IEEE; 2025. p. 1–7.
31. Juneja S, Bakshi P, Kaur G, Chandra R, Yadav RK. Leo: Personal Desktop Assistant Using Python. In: 2025 3rd Int Conf Disruptive Technol (ICDT). IEEE; 2025. p. 1508–12.
32. Abraham A, Mathew BS, Mathew DL, Mohammad FS, Krishnan G. Eva: Python-based desktop virtual assistant for visually impaired. In: 2024 7th Int Conf Circuit Power Comput Technol (ICCPCT). IEEE; 2024. p. 582–6.
33. Aboelela E, Mansour O. A Review of Speech Recognition and Application to Arabic Speech Recognition. In: Future Inf Commun Conf. Springer; 2025. p. 13–31.
34. Rahman A, Kabir MM, Mridha MF, Alatiyyah M, Alhasson HF, Alharbi SS. Arabic speech recognition: Advancement and challenges. IEEE Access. 2024;12:39689–716.
35. Zhang Y, et al. Google USM: Scaling automatic speech recognition beyond 100 languages. arXiv Preprint. 2023;arXiv:2303.01037.
36. Radford A, Kim JW, Xu T, Brockman G, McLeavey C, Sutskever I. Robust speech recognition via large-scale weak supervision. In: Int Conf Mach Learn. PMLR; 2023. p. 28492–518.
37. Oumard C, Kreimeier J, Götzelmann T. Pardon? An overview of the current state and requirements of voice user interfaces for blind and visually impaired users. In: Int Conf Comput Helping People Spec Needs. Springer; 2022. p. 388–98.
38. Esquivel P, Gill K, Goldberg M, Sundaram SA, Morris L, Ding D. Voice assistant utilization among the disability community for independent living: A rapid review of recent evidence. Hum Behav Emerg Technol. 2024;1:6494944.
39. Setiawan P, Yusuf R. IoT device control with offline automatic speech recognition on edge device. In: 2022 12th Int Conf Syst Eng Technol (ICSET). IEEE; 2022. p. 111–5.
40. Dudhapachare R, Awatade M, Kakde P, Vaidya N, Kapgate M, Nakhate R. Voice guided, gesture controlled virtual mouse. In: 2023 4th Int Conf Emerg Technol (INCET). IEEE; 2023. p. 1–6.
41. Avilala AR, Dagada SR, Oli JM, Neelima N. Virtual Voice Assistant (Jenny). In: 2025 Int Conf Intell Innov Technol Comput Electr Electron (IITCEE). IEEE; 2025. p. 1–6.
42. Wazir Y, Shah S, Kaur T, Tripathi S, Toradmalle D. Waver: Hands-Free Computing—A Fusion of AI-driven Gesture based Mouse and Voice Companion. In: 2023 6th Int Conf Adv Sci Technol (ICAST). IEEE; 2023. p. 245–9.
43. Lu Y, Zhou L, Zhang A, Wang M, Zhang S, Wang M. Research on designing context-aware interactive experiences for sustainable aging-friendly smart homes. Electronics (Basel). 2024;13(17):3507.
44. Paplu S, Ahmed H, Ashok A, Akkus S, Berns K. Multimodal perceptual cues for context-aware human-robot interaction. In: IFToMM Int Symp Sci Mech Mach (SYROM). Springer; 2022. p. 283–94.
45. Mulfari D, Villari M. A voice user interface on the edge for people with speech impairments. Electronics (Basel). 2024;13(7):1389.
46. Alzubi TM, Alzubi JA, Singh A, Alzubi OA, Subramanian M. A multimodal human-computer interaction for smart learning system. Int J Hum Comput Interact. 2025;41(3):1718–28.
47. Boumehdi A, Yousfi A. Arabic speech recognition independent of vocabulary for isolated words. In: Proc Sixth Int Congr Inf Commun Technol (ICICT 2021), London, Vol 3. Springer; 2021. p. 585–95.
48. Kheddar H, Hemis M, Himeur Y. Automatic speech recognition using advanced deep learning approaches: A survey. Inf Fusion. 2024;109:102422.
49. Abed S, Alshayeji M, Sultan S. Diacritics effect on Arabic speech recognition. Arab J Sci Eng. 2019;44(11):9043–56.
50. Majrashi K. Voice versus keyboard and mouse for text creation on Arabic user interfaces. Int Arab J Inf Technol. 2022;19(1):132–42.
51. Izzuddin A. Software Quality Testing In Mobile Application (ArabEasy) Based on the PACMAD Model. Int J Interact Mob Technol (iJIM).
52. Hsu C, Tung H-C, Shuai H-H, Chang Y-J. Predicting and exploring abandonment signals in a banking task-oriented chatbot service. Int J Hum Comput Interact. 2024;40(24):8497–512.

Received*: 26-08-2025 -* ***Accepted****: 25-10-2025 -* ***Published****: 03-11-2025*