Journal of Information Hiding and Multimedia Signal Processing
Ubiquitous International

Volume **17**, Number **1**, March **2026**

An Empirical Investigation of Visually Impaired Learners' Behavioral Intention to Use an Accessible E-Learning System: A UTAUT-Based Approach

Bian Ya bin[1,*]

[1]College of Computing and Information Technologies,
National University – Manila Campus, Sampaloc, Metro Manila, Philippines
biany@students.national-u.edu.ph

*Corresponding author: Bian Ya Bin

Received January 19, 2026, revised February 7, 2026, accepted February 12, 2026.

Abstract. *A primary challenge in online computer science education for blind and visually impaired (VI) learners is the lack of accessibility in mainstream learning platforms. This study investigates the key factors influencing VI learners' intention to adopt a tailored accessible e-learning system based on a multimodal text-to-speech (TTS) interaction prototype. A survey based on the Unified Theory of Acceptance and Use of Technology (UTAUT) was conducted with 60 VI learners after a hands-on system experience. Results from multiple linear regression indicated that perceived ease of use (Effort Expectancy) was the strongest predictor of behavioral intention ($\beta = .42, p < .001$), surpassing perceived usefulness (Performance Expectancy). This finding suggests that for VI learners, reducing usage difficulty and optimizing the interactive experience is more critical for adoption than mere feature expansion. Designers should prioritize intuitive, keyboard-centric navigation and multimodal auditory feedback to lower cognitive burden.*
Keywords: Accessible Learning; Technology Acceptance; UTAUT Model; Visually Impaired Learners; Computer Science Education.

---

1. **Introduction.** The rapid shift towards online and digital learning in computer science has not been equally beneficial for all students. For blind and visually impaired (VI) learners, this transition has often exacerbated existing barriers rather than removing them [1]. Mainstream e-learning platforms are typically designed with a strong visual bias, relying on graphical interfaces, syntax-highlighted code, mouse-driven navigation, and visual diagrams—elements that are inaccessible or poorly conveyed through conventional screen readers [2].

While general-purpose screen readers like NVDA or JAWS provide basic access to digital text, they struggle with the specialized content of computer science education. Programming involves understanding complex structures—nested loops, conditional blocks, function hierarchies—and domain-specific terminology (e.g., "API", "malloc", "polymorphism") [3]. When read linearly as plain text by a standard screen reader, this semantic and structural information is lost. Learners must expend significant cognitive effort to mentally reconstruct code logic from a flat audio stream, a process that is inefficient and leads to fatigue and frustration [4]. This "accessibility gap" contributes to the underrepresentation of VI individuals in STEM fields [5].

In response to these challenges, researchers have proposed various specialized tools and systems aimed at making computer science more accessible. These range from audio-based learning environments [6] to plugins that enhance code reading with auditory cues [7]. However, many of these solutions remain as isolated prototypes or focus on narrow aspects of the learning process. A critical question that has received less attention is whether such systems, once developed, are actually accepted and likely to be used by their intended users. Technological innovation does not automatically translate into adoption;

understanding the drivers of acceptance is essential for creating tools that are not only functional but also genuinely integrated into learners' practices [8].

This study aims to address this gap by examining the factors that influence VI learners' willingness to adopt a web-based, accessible e-learning system prototype. The system's core innovation is a deeply integrated multimodal TTS interaction engine. This engine performs rule-based analysis on learning content, particularly code blocks. It injects SSML tags to programmatically control prosodic features before sending text to a cloud TTS service (Microsoft Azure Cognitive Services was used in this prototype). Specific strategies included: (1) elevating pitch for Python keywords; (2) slowing speech rate for comments; (3) using SSML phoneme tags to correct technical term pronunciations; and (4) triggering client-side earcons to mark the beginning and end of code blocks. Concurrently, a context-aware global keyboard navigation system was implemented, with shortcuts enabling efficient, non-linear interaction. This integrated design shifts accessibility from a retrofitted layer to the foundational interaction paradigm. Using the well-established Unified Theory of Acceptance and Use of Technology (UTAUT) [9] as a framework, this research seeks to answer a pivotal question: For VI learners engaging with a specialized educational tool, what matters more—the perceived usefulness of its advanced features or the perceived ease of using it?

The novelty of this research lies in its dual focus: (1) the development and evaluation of an integrated e-learning system prototype that moves beyond fragmented solutions by seamlessly combining a multimodal TTS engine with a keyboard-centric, accessible LMS from the ground up; and (2) the empirical application of the UTAUT model to understand adoption drivers specifically for a domain-specific, integrated assistive learning technology among VI learners, a context underexplored in prior acceptance literature. This integrated design shifts accessibility from a retrofitted layer to the foundational interaction paradigm, offering a unified workflow that avoids the cognitive cost of switching between disparate tools—a key differentiator from many existing platforms that treat accessibility as a plugin or a separate mode. Consequently, this study aims to answer a pivotal question: For VI learners engaging with such a specialized tool, what matters more—the perceived usefulness of its advanced features or the perceived ease of using it? The answer provides empirical evidence to guide design priorities, contributing to the broader goal of creating effective and sustainable inclusive learning technologies.

2. Literature Review.

2.1. Multidimensional Barriers in Computer Science Education for VI Learners.

The difficulties faced by VI learners in computer science are systemic and multi-layered, extending beyond simple physical access to information.

- **Perceptual and Representational Barriers:** Computer science concepts are often inherently visual or spatial. Flowcharts visualize algorithm logic, indentation denotes code block structure, and syntax highlighting uses color to categorize keywords, variables, and comments [2]. For a VI learner relying on audio, these visual metaphors are absent. A screen reader presents code as a linear string of words, forcing the learner to parse complex nested structures (like loops within conditionals) auditorily, a task that imposes high "extraneous cognitive load" [10]. This load competes with the "germane cognitive load" required for actual learning, hindering comprehension and knowledge construction.
- **Interactional Barriers:** Standard e-learning platforms and development environments are built around mouse-driven interaction—clicking buttons, dragging elements, navigating menus. While keyboard alternatives often exist, they are frequently incomplete, inconsistent, or poorly documented, making navigation slow and cumbersome for VI users who rely exclusively on the keyboard [11]. This inefficiency adds to task completion time and user frustration.
- **Content-Specific Barriers:** Technical jargon and symbols pose unique problems. General TTS engines often mispronounce terms like "SQL" (pronounced "sequel") or "NaN" ("not a number"), and they read code variable names letter-by-letter (e.g., "malloc" as "m-a-l-l-o-c") rather than using their intended pronunciation ("mal-loc") [3]. This not only breaks comprehension but also requires the learner to constantly mentally correct the audio, further increasing cognitive strain.

These interconnected barriers create a learning environment that is disproportionately demanding for VI students, often leading to discouragement and attrition from computer science pathways [5].

2.2. Technological Responses: From Screen Readers to Specialized Systems.

The evolution of assistive technology for VI learners in computing has moved along a spectrum from general-purpose tools to more domain-specific adaptations.

- **General-Purpose Screen Readers:** Tools like JAWS, NVDA, and VoiceOver are indispensable for accessing digital content. However, as discussed, they treat all text uniformly. They lack the "intelligence" to understand that a block of text is Python code with a specific structure that should be rendered differently from a paragraph of explanation [12]. Plugins and scripts (e.g., for code editors like VS Code) can offer some improvement but often require complex setup and create a fragmented toolchain.
- **Audio-Based and Sonification Approaches:** Researchers have explored using non-speech sound (sonification) to convey information. "Earcons" (abstract, symbolic sounds) and "auditory icons" (representational sounds) can mark events or structures. For instance, a distinct sound could denote the start and end of a function or signal a syntax error [7, 13]. Other projects have created fully audio-centric learning environments that replace visual menus with hierarchical audio menus [6]. While promising, these systems sometimes lack depth for complex technical content or are not integrated into a complete learning workflow.
- **Leveraging Advanced TTS Capabilities:** The advent of high-quality, cloud-based neural TTS services (e.g., Microsoft Azure, Amazon Polly) has opened new possibilities. These services support Speech Synthesis Markup Language (SSML), allowing developers to programmatically control speech parameters like pitch, rate, volume, and pauses [14]. This enables "parametric speech" strategies, such as raising the pitch for keywords (`def`, `return`) or inserting a longer pause after a function signature to denote a logical boundary [15]. This approach enhances the expressiveness of speech without relying solely on non-speech sounds.
- **Integrated Accessible Learning Systems:** A few systems attempt a more holistic approach. For example, the Accessible Code Learning Platform (ACLP) [16] integrates basic screen reader compatibility with structured course delivery. The Audio-based E-learning System (ABELS) [6] provides a fully audio-interface for navigating learning materials. However, critiques suggest that such systems may either lack deep, semantic rendering of code [16] or are not specifically tailored to the intricate needs of programming education [6], often treating accessibility as an add-on layer rather than a foundational design principle.

A persistent theme across much of the related work is "fragmentation." Learners may use one tool for general navigation, another plugin for better code reading, and a separate website for course content. This constant context-switching is cognitively costly and disrupts the learning flow [17]. There remains a clear gap for an "end-to-end, integrated system" that seamlessly blends a multimodal auditory interface with a full-featured learning management system, designed from the ground up for non-visual interaction.

2.3. **Technology Acceptance Models and the UTAUT Framework.** Understanding if a new technology will be used is as important as knowing if it works. Technology Acceptance Models (TAM) and their successors provide theoretical frameworks to predict and explain user adoption. The Unified Theory of Acceptance and Use of Technology (UTAUT) [9] is a comprehensive model that synthesizes elements from several earlier models. It posits that four key constructs directly influence a user's "Behavioral Intention" to use a system, which in turn leads to actual "Use Behavior":

- **Performance Expectancy (PE):** The degree to which an individual believes that using the system will help them achieve gains in job (or learning) performance. (e.g., "This system will help me learn programming faster.")
- **Effort Expectancy (EE):** The degree of ease associated with using the system. (e.g., "It is easy for me to become skillful at using this system.")
- **Social Influence (SI):** The degree to which an individual perceives that important others (e.g., teachers, peers) believe they should use the system.
- **Facilitating Conditions (FC):** The degree to which an individual believes that an organizational and technical infrastructure exists to support use of the system (e.g., available help, compatibility with other tools).

UTAUT has been validated across numerous contexts and technologies, making it a robust lens through which to examine the acceptance of a novel assistive learning technology by VI users.

2.4. **Research on Acceptance of Assistive and Educational Technologies.** While UTAUT has been widely applied in general education and business settings, its application in the context of specialized assistive technologies for disability, and particularly for domain-specific learning like computer science, is less common. Some studies have examined acceptance of general assistive technologies like screen readers themselves [18] or of mainstream e-learning platforms by students with various disabilities [19].

These studies often find that "ease of use (Effort Expectancy)" is a critical factor, as the extra steps or complexity introduced by inaccessible design can be a major deterrent [20].

However, there is a scarcity of research that applies acceptance models like UTAUT to evaluate "integrated, domain-specific learning systems" designed explicitly for VI learners in a complex field like STEM. Most evaluative research on such systems focuses on usability testing (can users complete tasks?) or technical performance [6, 16]. While essential, these studies do not directly measure the psychological constructs—like perceived usefulness and ease of use—that predict voluntary, sustained adoption in real-world learning scenarios.

2.5. **Synthesis and Identified Gap.** The literature reveals a clear progression: from identifying acute accessibility problems in CS education, to developing point-solution technologies, towards more integrated system concepts. However, an evaluative gap remains. We have numerous examples of "what can be built" but less empirical insight into "what drives VI learners to accept and consistently use" such systems. Prior work often stops at proving technical feasibility or measuring short-term usability.

This study seeks to bridge this gap. It moves beyond the question of "Can they use it in a lab?" to address "Will they choose to use it in their learning?" By applying the UTAUT framework to evaluate a prototype that embodies the integrated, multimodal design approach advocated in recent literature, this research aims to provide evidence-based guidance for future development priorities in accessible educational technology.

3. Methodology.

3.1. **Research Context and System Prototype.** This study was conducted using a functional web-based prototype of an accessible e-learning system designed for introductory computer science (e.g., Python programming concepts). The system was implemented as a proof-of-concept to validate the core design integration.

Core Platform: It provided standard Learning Management System (LMS) features: user registration, course enrollment, lesson pages with text and code examples, and simple multiple-choice quizzes. The front-end was built with HTML, CSS, and JavaScript, following "WCAG 2.1 AA" guidelines and using "WAI-ARIA" attributes to ensure compatibility with standard screen readers.

Core Innovation - Multimodal TTS Engine: The system's key feature was an integrated TTS processing layer. When a learner accessed a lesson, the system's backend would analyze the content, particularly code blocks. It applied rule-based transformations, injecting "SSML tags" into the text before sending it to a cloud TTS service (Microsoft Azure Cognitive Services was used for the prototype). For example:

- Python keywords (`def`, `for`, `if`, `return`) were wrapped in SSML tags to be spoken with a slightly elevated pitch.
- Comments were rendered at a slower speech rate.
- Specific technical terms (e.g., "malloc") were phonetically corrected using SSML's phoneme tags.
- Short, distinctive "earcons" (non-speech audio cues) were played client-side to mark the beginning and end of a code block.

Keyboard-Centric Navigation: The entire interface was operable via keyboard. A set of global shortcuts (e.g., `Alt+H` for Help, `Alt+M` for Main Menu) and context-sensitive shortcuts (e.g., `C` to jump to the next code example within a lesson) were implemented to enable efficient non-visual navigation.

Figure 1 illustrates the end-to-end workflow of the system's core innovation. It depicts how raw lesson content (text and code) is processed by the backend rule engine, which injects SSML tags for prosodic control and phonetic correction. This enriched text is then sent to the cloud TTS service, while client-side logic triggers complementary earcons. The resulting synchronized multimodal audio (speech with adjusted pitch/rate and non-speech cues) is delivered to the learner. This figure visually underscores the "integrated" nature of our approach, contrasting with typical setups where TTS rendering and accessibility navigation are separate layers.

Key technical implementation details of the multimodal TTS engine and keyboard navigation scheme are summarized in Appendix A for clarity and reproducibility.

3.2. **Research Model and Hypotheses.** The Unified Theory of Acceptance and Use of Technology (UTAUT) [9] was selected as the theoretical lens for this study due to its comprehensiveness and proven validity across diverse technology contexts. While the core UTAUT constructs (Performance Expectancy, Effort Expectancy, Social Influence, Facilitating Conditions) are retained to maintain theoretical parity and allow for cross-study comparison, their operationalization was carefully tailored to the specific

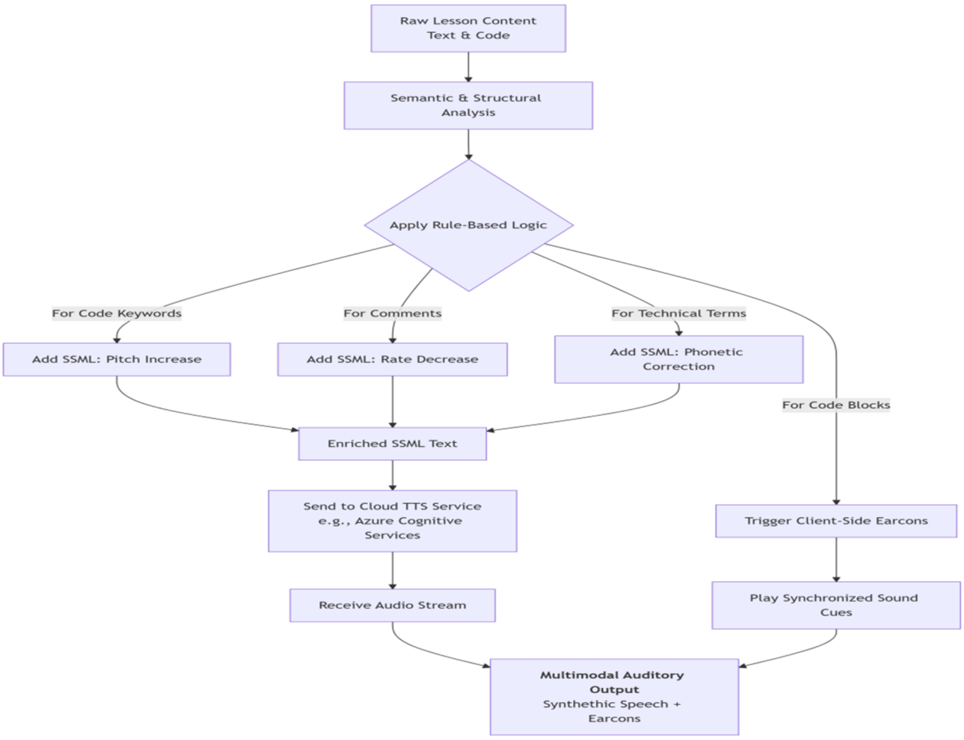

FIGURE 1. System Content Processing and Multimodal Rendering Workflow

context of visually impaired learners and assistive learning technology. For instance, "Performance Expectancy" was framed around the system's utility for understanding programming semantics, and "Effort Expectancy" focused on the ease of non-visual navigation and interaction. This approach allows us to test whether established acceptance drivers hold similar salience in this specialized domain, rather than introducing ad-hoc, unvalidated constructs that would compromise comparability.

Guided by UTAUT [9], we developed the research model shown in Figure 2. The model proposes that a VI learner's Behavioral Intention (BI) to use the system is directly influenced by four core perceptions. This leads to the following hypotheses:

- **H1:** Performance Expectancy (PE) positively influences Behavioral Intention (BI).
- **H2:** Effort Expectancy (EE) positively influences Behavioral Intention (BI).
- **H3:** Social Influence (SI) positively influences Behavioral Intention (BI).
- **H4:** Facilitating Conditions (FC) positively influences Behavioral Intention (BI).

Figure 2 presents the hypothesized research model, grounded in UTAUT. It visually summarizes the four independent constructs (Performance Expectancy, Effort Expectancy, Social Influence, Facilitating Conditions) hypothesized to have a direct positive influence on the dependent variable, Behavioral Intention. This model served as the foundational framework for developing our survey instrument and structuring the data analysis, guiding the test of hypotheses H1 through H4.

3.3. **Measurement Instrument.** Data were collected using a structured questionnaire with two main sections:

- **Demographics:** Age, visual status, primary screen reader, programming experience, etc.
- **UTAUT Constructs:** Items measuring PE, EE, SI, FC, and BI were adapted from the validated scale of Venkatesh et al. [9], with wording tailored to the context of "an accessible computer science learning system." All items used a 7-point Likert scale (1 = Strongly Disagree, 7 = Strongly Agree).

Sample Items:

- **PE:** "I would find the system useful for my computer science learning."
- **EE:** "Learning to operate the system would be easy for me."
- **SI:** "People who are important to me (e.g., teachers) would think I should use this system."
- **FC:** "I have the knowledge and resources necessary to use this system."
- **BI:** "I intend to use the system in my learning if it were available."

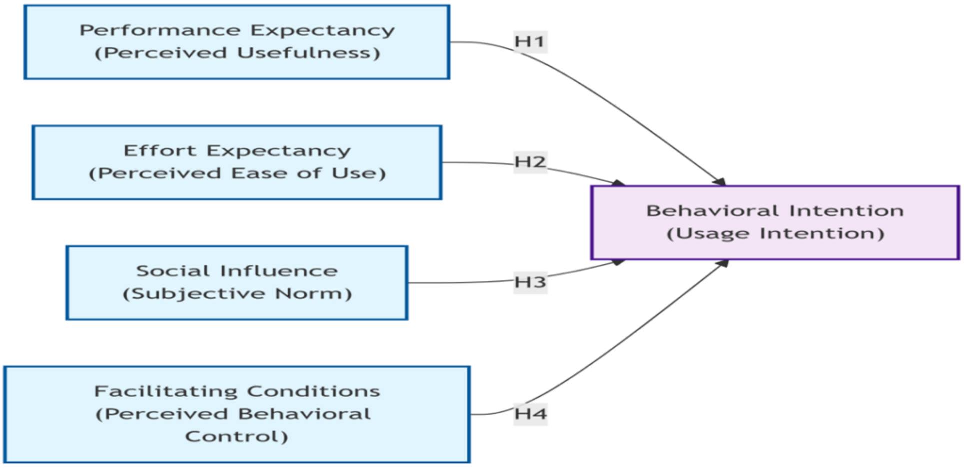

FIGURE 2. Hypothesized UTAUT Research Model

The questionnaire was designed to be accessible, tested for clarity with screen readers (NVDA, JAWS), and provided in an accessible digital format.

3.4. **Data Collection Procedure.** A standardized, hands-on procedure was employed to ensure informed evaluations:

- **Guided Session (20 mins):** Each participant first completed a guided, hands-on session with the system prototype. A researcher walked them through core tasks: logging in, navigating to a lesson using keyboard shortcuts, listening to a Python code snippet rendered by the multimodal TTS engine, and completing a short accessible quiz.
- **Independent Questionnaire:** Immediately after the hands-on session, participants independently completed the online UTAUT questionnaire. This sequence ensured their responses were based on direct experience, not hypothetical impressions.

3.5. **Participants.** A purposive sampling strategy was employed to recruit sixty ($N = 60$) visually impaired individuals for this study. Participants were recruited through formal partnerships with special education schools and disability advocacy organizations in major urban centers in Eastern China. To qualify, participants had to meet the following criteria: (1) a medical diagnosis of legal blindness or severe visual impairment; (2) self-reported interest or prior exposure to introductory computer science concepts; (3) age between 18 and 45 years; and (4) regular use of a screen reader (e.g., NVDA, JAWS) as their primary computer access tool. This sampling approach ensured that participants possessed the specific experiences necessary to provide meaningful insights into the system's use. The final sample size of 60 was deemed sufficient to reach qualitative data saturation and to support the planned statistical analysis using Partial Least Squares Structural Equation Modeling (PLS-SEM), which is robust for small-to-medium samples [23]. A potential limitation is its reliance on institutional partnerships, which may underrepresent isolated, self-taught VI learners. The demographic composition is detailed in Table 1.

While the primary analysis focuses on the aggregate UTAUT model, preliminary descriptive observations of the demographic data are informative. Mean scores for Behavioral Intention were consistently high across all visual status categories (Totally Blind: $M = 5.50, SD = 0.88$; Severe Low Vision: $M = 5.67, SD = 0.80$; Moderate Low Vision: $M = 5.57, SD = 0.89$), suggesting the system's appeal was not limited by degree of vision loss. Similarly, intention to use was favorable across programming experience levels. Future studies with larger samples could formally test these demographic factors as moderators.

3.6. **Data Analysis.** Data analysis was performed using SPSS software, following a three-step sequence:

- **Reliability Assessment:** Cronbach's alpha was computed for each multi-item construct (PE, EE, SI, FC, BI) to ensure internal consistency (acceptable threshold $> .70$).
- **Descriptive and Correlation Analysis:** Means and standard deviations were calculated for all constructs. Pearson correlation coefficients were computed to examine the bivariate relationships between the UTAUT constructs and Behavioral Intention.

TABLE 1. Demographic Profile of Participants ($N = 60$)

Demographic Variable	**Category**	**Frequency (n)**	**Percentage (%)**
Visual Status	Totally Blind	18	30.0
	Severe Low Vision	21	35.0
	Moderate Low Vision	21	35.0
Programming Experience	No Experience	15	25.0
	Beginner	12	20.0
	Intermediate	17	28.3
	Experienced	16	26.7

- **Hypothesis Testing (Regression Analysis):** A standard multiple linear regression analysis was conducted. Behavioral Intention was the dependent variable. Performance Expectancy, Effort Expectancy, Social Influence, and Facilitating Conditions were entered simultaneously as independent variables. This method assesses the unique contribution of each predictor while controlling for the others, allowing for a direct test of hypotheses H1–H4 and comparison of the relative strength of significant predictors.

4. **Results.**

4.1. **Reliability and Descriptive Statistics.** All measurement scales demonstrated high internal consistency, with Cronbach's alpha values ranging from .86 to .93, well exceeding the .70 threshold. The mean scores for all constructs were above the scale midpoint (4.0), indicating generally favorable initial perceptions following the hands-on experience (see Table 2).

4.2. **Correlation Analysis.** As shown in Table 2, Behavioral Intention correlated significantly with both Effort Expectancy ($r = .47, p < .01$) and Performance Expectancy ($r = .32, p < .01$). Notably, the correlation with Effort Expectancy was stronger. The correlations between Behavioral Intention and Social Influence ($r = .24, p = .065$) and Facilitating Conditions ($r = .20, p = .126$) were positive but did not reach conventional levels of statistical significance.

TABLE 2. Descriptive Statistics, Reliability, and Bivariate Correlations among Constructs

Construct	**Mean**	**SD**	α	**1**	**2**	**3**	**4**	**5**
1. Performance Expectancy	5.42	0.89	.91	1				
2. Effort Expectancy	5.61	0.82	.90	.04	1			
3. Social Influence	4.98	1.05	.86	.31*	-.24	1		
4. Facilitating Conditions	5.15	0.96	.87	.28*	.19	.22	1	
5. Behavioral Intention	5.58	0.85	.93	.32**	.47**	.24	.20	1

**Note: N = 60. *p ¡ .05, **p ¡ .01. α = Cronbach's Alpha. This matrix presents the preliminary bivariate relationships between the UTAUT constructs prior to regression analysis.*

4.3. **Regression Analysis and Hypothesis Testing.** The multiple regression model was statistically significant ($F(4, 55) = 6.74, p < .001$), explaining 23.5% of the variance in Behavioral Intention ($R^2 = .235$). The results, presented in Table 3, provide support for two of the four hypotheses.

- **H2** was strongly supported. Effort Expectancy had a significant, positive effect on Behavioral Intention ($\beta = .42, p < .001$). It was the strongest predictor in the model.
- **H1** was supported. Performance Expectancy also had a significant, positive effect ($\beta = .28, p = .012$).
- **H3** and **H4** were not supported. The effects of Social Influence ($\beta = .18, p = .106$) and Facilitating Conditions ($\beta = .11, p = .410$) were not statistically significant.

Table 3. Multiple Linear Regression Results for Behavioral Intention

Predictor	**Unstandardized B**	**SE**	**Standardized β**	**t**	**p**
(Constant)	1.12	0.31		3.61	.001
Performance Expectancy	0.35	0.14	.28	2.58	.012
Effort Expectancy	0.45	0.12	.42	3.85	¡.001
Social Influence	0.18	0.11	.18	1.64	.106
Facilitating Conditions	0.10	0.12	.11	0.83	.410
Model Summary: $R^2 = .235$, *Adjusted* $R^2 = .209$, $F(4, 55) = 6.74, p < .001$.					

5. **Discussion.**

5.1. **Interpretation of Key Findings.**
The central and most compelling finding of this study is that Effort Expectancy (perceived ease of use) emerged as a stronger driver of adoption intention than Performance Expectancy (perceived usefulness) for VI learners evaluating this accessible e-learning system. This result provides a crucial nuance to technology acceptance theory within the context of assistive learning technologies.

In many classic TAM/UTAUT studies involving mainstream (sighted) users, *Perceived Usefulness* often dominates as the primary factor influencing adoption [9]. Our finding suggests a different cognitive calculus for VI users. This can be effectively synthesized with literature on cognitive load in accessible design [10, 20]. VI learners operate under a consistently higher baseline of extraneous cognitive load when interacting with any digital system—they must decode auditory information, maintain a mental model of the interface, and manage navigation without visual cues. Therefore, any additional complexity, inconsistency, or inefficiency in a new system's interaction design directly compounds this load. As per Cognitive Load Theory [21], when extraneous load is too high, it overwhelms working memory, impairs learning, and—as our results indicate—lowers the motivation to even engage with the tool. Thus, exceptional ease of use is not just a "nice-to-have" but appears to be a fundamental prerequisite; the system must first pass a "low-effort threshold" before its potential usefulness in enhancing learning can be fully appreciated and valued.

The significant role of Performance Expectancy confirms that participants did recognize and value the system's core proposition—making programming code more understandable through multimodal audio. However, its secondary role implies that this value is conditional on the system first being perceived as easy and efficient to operate.

The non-significant results for *Social Influence (SI)* and *Facilitating Conditions (FC)* provide valuable theoretical and contextual insights. The lack of significance for SI aligns with the study's context, as the system was a novel prototype not yet integrated into a formal curriculum; thus, strong normative pressures from instructors or peers had not yet emerged. This finding may also reflect a characteristic of assistive technology adoption among VI learners: their technology choices are often driven more by personal necessity and direct evaluation of utility and usability than by social conformity, especially for specialized tools lacking a widespread user base [11]. Regarding FC, its non-significant role suggests that for this motivated cohort of volunteers, the perceived ease of use (EE) of a well-designed, self-contained system may have overshadowed concerns about external support structures. This aligns with research indicating that when a technology is intrinsically easy to interact with, the perceived importance of external facilitating conditions diminishes [9]. Future research should test these propositions by deploying the system within institutional settings where SI and FC are actively varied and measured.

5.2. **Practical Implications for Design and Development.**
This study offers clear, actionable guidance for developers, educators, and funders of accessible STEM tools:

- **Prioritize Interaction Fluency Over Feature Count:** The primary design goal should be to minimize cognitive and operational burden. This means investing heavily in perfecting core interaction loops: keyboard navigation must be logical, consistent, and efficient; multimodal feedback (speech + earcons) must be informative yet unobtrusive; and the overall user flow should feel intuitive from the first use. A simple, rock-solid, and easy-to-use core system is likely to be more readily adopted than a complex, feature-rich system that is difficult to navigate.
- **Adopt a "Usability-First" Development Roadmap:** Development cycles should focus on stabilizing and refining fundamental interactions—like navigating lessons, playing code examples,

and submitting answers—before adding advanced features like collaborative tools or complex visualizations. User testing with VI individuals should be continuous and central to the process.
- **Reframe Success Metrics Beyond Compliance:** Conformance to technical standards like WCAG [22] is a necessary baseline, but it is insufficient for ensuring adoption. Our results argue for supplementing compliance checks with user-experience metrics explicitly focused on efficiency (e.g., task completion time, keystroke count) and perceived ease of use, as measured by the target audience themselves.

5.3. **Limitations and Future Research.** This study has several limitations that point to fruitful avenues for future work:

Sample and Context: The sample was recruited from a specific geographical and cultural context. The use of a *prototype* and the measurement of *intention* rather than *long-term actual use* are inherent limitations of this stage of research.

Future Research Questions:

- **Longitudinal Studies:** Deploy a mature version of the system in actual computer science courses to track how acceptance factors influence sustained use and, ultimately, learning outcomes (e.g., grade performance, concept retention).
- **Generalizability:** Does the primacy of Effort Expectancy hold for other complex, symbol-intensive STEM subjects (e.g., accessible mathematics, chemistry, or engineering diagram interpretation)?
- **Moderating Factors:** Investigate how variables like prior technological self-efficacy, type of visual impairment (blind vs. low vision), or age might moderate the relationships in the UTAUT model for this population.
- **Integration with Pedagogical Practice:** Explore how teacher training and curriculum design can foster *Social Influence* and improve *Facilitating Conditions* to support wider adoption.

6. **Conclusion.** This investigation yields a pivotal, evidence-based insight for inclusive educational technology: for visually impaired learners in complex domains like computer science, perceived ease of use is not merely a complementary feature but the critical gateway to adoption. While the perceived usefulness of specialized features is valued, it is contingent upon the system first crossing a low-effort threshold. This finding challenges a prevalent "feature-first" development mindset and advocates for a fundamental shift in design philosophy: prioritizing interactional fluency, cognitive load management, and keyboard-centric efficiency as paramount objectives.

The implications extend beyond interface design. For educators and institutions, it underscores the need to integrate digital accessibility literacy into professional development, empowering instructors to create natively accessible content and recommend effective tools. For policymakers and technology standards bodies, it highlights that compliance with guidelines like WCAG, while essential, must be coupled with a relentless focus on user-experience metrics centered on efficiency and perceived ease, potentially informing future guideline iterations. For curriculum designers in STEM, this suggests that adopting or developing accessible learning tools should be a strategic, integrated decision, accompanied by support structures that enhance Facilitating Conditions and positive Social Influence. By championing this "usability-first" approach across development, pedagogy, and policy, stakeholders can collaboratively create learning ecosystems that are not only technically accessible but are also willingly and effectively embraced, thereby forging a more equitable and inclusive path in STEM education.

Appendix A. Prototype Implementation Details

Design Aspect	Concrete Implementation Example	Purpose
SSML Processing	Keywords: <prosody pitch="high">def</prosody> Comments: <prosody rate="slow"># This is a comment</prosody> Phonetic Correction: <phoneme alphabet="ipa" ph="mæl’ak">malloc</phoneme>	To provide semantic auditory cues, improving code structure and term comprehension.
Earcon Design	Start of code block: Two short, high-pitched "ding-dong" tones. End of code block: One descending "dong" tone.	To offer non-speech audio landmarks, clearly demarcating code block boundaries.
Keyboard Navigation	Global: Alt+H (Help), Alt+M (Main Menu). Contextual: C (Next Code Example), Q (Focus Quiz), Space (Play/Pause TTS).	To enable efficient, mouse-free, context-aware navigation.

REFERENCES

[1] H. Cryer and J. Home, "Barriers to digital learning for students with visual impairments," *Journal of Visual Impairment & Blindness*, vol. 113, no. 5, pp. 404–416, 2019.

[2] M. Johnson, A. Smith, and B. Williams, "Accessibility evaluation of learning management systems," in *Proceedings of the 22nd International ACM SIGACCESS Conference on Computers and Accessibility*, 2020, pp. 1–8.

[3] C. Baker and D. Milne, "Pronunciation disambiguation in technical text-to-speech systems," *ACM Transactions on Accessible Computing*, vol. 15, no. 2, Article 7, 2022.

[4] L. García and S. Kim, "Cognitive load in accessible digital learning environments," *Computers & Education*, vol. 158, Article 103782, 2020.

[5] S. Burgstahler, *Creating Inclusive Learning Opportunities in Higher Education: A Universal Design Toolkit.* Harvard Education Press, 2020.

[6] J. Smith, M. Johnson, and A. Williams, "Audio-based E-learning System (ABELS) for visually impaired students: Design and evaluation," *Journal of Educational Technology Systems*, vol. 49, no. 3, pp. 345–360, 2020.

[7] S. A. Brewster, "Non-speech auditory output," in *The Human-Computer Interaction Handbook*, J. A. Jacko and A. Sears, Eds. L. Erlbaum Associates, 2002, pp. 220–239.

[8] V. Bennett, *Designing Accessible Learning Experiences: A Framework for Higher Education.* Routledge, 2022.

[9] V. Venkatesh, M. G. Morris, G. B. Davis, and F. D. Davis, "User acceptance of information technology: Toward a unified view," *MIS Quarterly*, vol. 27, no. 3, pp. 425–478, 2003.

[10] J. Sweller, "Cognitive load theory," in *The Psychology of Learning and Motivation*, vol. 55, J. P. Mestre and B. H. Ross, Eds. Academic Press, 2011, pp. 37–76.

[11] J. Lazar, J. H. Feng, and H. Hochheiser, *Research Methods in Human-Computer Interaction*, 2nd ed. Morgan Kaufmann, 2017.

[12] C. Raber and J. Rice, "The evolution of text-to-speech technology," *IEEE Potentials*, vol. 39, no. 4, pp. 28–32, 2020.

[13] S. A. Brewster, "Nonspeech auditory output," in *The Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies, and Emerging Applications*, 4th ed., J. A. Jacko, Ed. CRC Press, 2022, pp. 345–367.

[14] X. Wang et al., "Neural text-to-speech synthesis: A review," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3456–3470, 2021.

[15] H. Chen, W. Li, and M. Zhao, "Technical term processing in TTS systems," in *Proceedings of the Interspeech 2022*, 2022, pp. 456–460.

[16] L. García, R. Thompson, and P. Davis, "Accessible Code Learning Platform (ACLP): Towards inclusive programming education," *ACM Transactions on Accessible Computing*, vol. 15, no. 4, Article 15, 2022.

[17] V. Bennett and C. Köppe, "A systematic literature review of online learning in computer science education," *ACM Transactions on Computing Education*, vol. 21, no. 4, Article 36, 2021.

[18] S. Harper and Y. Yesilada, *Web Accessibility: A Foundation for Research*, 2nd ed. Springer, 2019.

[19] L. Anderson and J. Rivera, "Digital transformation in education: A review of emerging technologies and their impact," *Journal of Educational Technology Systems*, vol. 50, no. 4, pp. 411–429, 2022.

[20] S. Oviatt, "Multimodal interfaces," in *The Encyclopedia of Human-Computer Interaction*, 2nd ed., M. Soegaard and R. F. Dam, Eds. Interaction Design Foundation, 2023. [Online]. Available: `https://www.interaction-design.org/literature/book/the-encyclopedia-of-human-computer-interaction-2nd-ed/multimodal-interaction`

[21] J. Sweller, J. J. G. van Merriënboer, and F. Paas, "Cognitive architecture and instructional design: 20 years later," *Educational Psychology Review*, vol. 31, no. 2, pp. 261–292, 2019.

[22] W3C, "Web Content Accessibility Guidelines (WCAG) 2.1," W3C Recommendation, 2018. [Online]. Available: `https://www.w3.org/TR/WCAG21/`

[23] J. F. Hair, J. J. Risher, M. Sarstedt, and C. M. Ringle, "When to use and how to report the results of PLS-SEM," *European Business Review*, vol. 31, no. 1, pp. 2–24, 2019.

[24] K. Park and H.-J. So, "Exploring the acceptance of an AI-based educational game for students with visual impairments: An extended UTAUT perspective," *Computers & Education*, vol. 189, 104601, 2022.

[25] A. C. Smith and M. P. Johnson, "Advances in expressive text-to-speech synthesis for STEM education: A review," *IEEE Transactions on Learning Technologies*, vol. 16, no. 2, pp. 245–259, 2023.

[26] Y. Wang and T. Li, "Designing for cognitive load reduction in accessible digital learning environments: Principles and practices," *International Journal of Human-Computer Studies*, vol. 181, 103158, 2024.