

Enhancing User Experience through Voice Assistant Integration with Content Management Systems

Safaa Salam Hatem^{1,*}, Fahad Naim Nife², and Israa Majeed Alsaadi³


ABSTRACT

Voice-based interfaces integrated into content management systems (CMSs) represent a feasible way to achieve more accessible digital publishing; however, while voice-enabled technologies are gaining widespread adoption, there is a lack of evidence on the effectiveness of voice interfaces for users who encounter difficulties using the traditional keyboard-and-mouse interaction paradigm. This study proposes and evaluates a voice-controlled WordPress interface for users with different accessibility needs, focusing on users with motor impairments, visual impairments (low vision), and limited computer literacy, and a pilot study with five participants ($N = 5$) compared the proposed voice interface to a standard input method in a within-subjects experimental design. The system was developed using the Google Voice Assistant, Dialogflow, and the WordPress REST API to enable end-to-end voice-driven content creation and management, and the results show that, although the voice interface led to a 22% increase in the average task completion time ($M = 185.8$ s vs. 152.4 s), it significantly reduced text entry errors by 77% ($M = 2.6$ vs. 11.2 errors). Participants also reported higher satisfaction with the voice-based system ($M = 6.2/7$ vs. $3.4/7$ on the SEQ scale), and the results suggest that voice-controlled CMS interfaces may enhance accessibility and overall user experience while providing a validated and replicable framework for inclusive digital publishing.

Keywords: Accessibility, content management systems, user experience, voice assistants.

Submitted: September 01, 2025

Published: April 13, 2026

 10.24018/compute.2026.6.1.10325

¹College of Science, Department of Mathematics and Computer Applications, Al-Muthanna University, Iraq.

²College of Artificial Intelligence and Cyber Security Engineering, Al-Muthanna University, Iraq.

³Department of Computer Science, College of Sciences for Women, University of Baghdad, Iraq.

*Corresponding Author:
e-mail: safaa.salam@mu.edu.iq

1. INTRODUCTION

In the present digital age, the human-computer interaction (HCI) model is changing dramatically with the development of artificial intelligence (AI) and natural language processing (NLP). Voice interfaces have shown to be a good interaction modality in the new digital age as a more intuitive and accessible alternative to traditional input devices such as keyboards and mice, and this is particularly true for content management systems (CMSs) that facilitate the creation and dissemination of digital content on the Internet. As of 2024, WordPress is the most widely used CMS platform used by nearly 43% of all the websites on the internet; however, its reliance on traditional input devices is a significant barrier to users with physical disabilities, further marginalizing existing accessibility disparities and limiting their opportunity to inclusively participate in digital content creation [1], [2].

According to the World Health Organization, there are over 1.3 billion people worldwide who live with significant disability, which is about 16% of the world's population with some level of functional limitation. For users with motor impairments, the typical use of digital systems that require accurate manual input is extremely difficult, and these difficulties are not just matters of inconvenience, but can limit people's ability to create digital content, participate in professional communication, and engage in creative expression. In this sense, the inaccessibility of existing content management systems is not only a technical problem but also a social justice issue that de facto decides which voices, experiences, and perspectives can enter, stay in, and effectively shape the digital public sphere [3], and one of the most promising solutions to this problem are voice assistants, which we define here as software systems that use speech recognition, artificial



intelligence, and natural language processing to carry out tasks in direct response to spoken user commands, thus offering an alternative interaction modality for those who are excluded or disadvantaged by traditional input devices.

By providing hands-free content creation, editing and management, voice assistants can dramatically reduce interaction barriers for users with motor impairments and at the same time enhance efficiency and usability for a much larger population, as a complementary input modality to traditional devices. Moreover, the integration of voice assistance technologies into content management systems (CMS) is the confluence of two major technological trends: the ongoing democratization of content production via increasingly accessible publishing platforms and the explosive growth of voice-enabled computing interfaces across devices and application domains [4], [5].

However, in contrast to this potential, there is a surprisingly low number of empirical studies that systematically evaluate the effectiveness of voice-controlled CMS interfaces in real-world use by real end users, while many attempts have been made to conceptualize architectures, design systems, or build proof-of-concept prototypes. Only a few approaches have been made to evaluate systems with real user groups in a rigorous and data-driven way, and this is particularly true for motor impaired users, the primary target group of such assistive technologies, who are under-represented in human-computer interaction research and user studies [6], [7].

In this work, we try to fill this gap in two ways: 1) by proposing a concrete and implementable technical framework for integrating the Google Voice Assistant with WordPress using the NLP capabilities of Dialogflow, in conjunction with the WordPress REST API, and 2) more importantly, by empirically evaluating the efficacy of the proposed framework through a pilot experimental study involving participants with motor impairments, thereby directly testing the accessibility benefits of the framework for the population that would benefit the most from this assistive technology.

The study employs a rigorous methodological framework, including a counterbalanced within-subjects design, standardized usability metrics, and appropriate statistical analyses, to provide robust empirical evidence regarding the real-world advantages and disadvantages of the proposed system, which is driven by two fundamental questions: (1) how Google Voice Assistant can be technically integrated with WordPress to facilitate voice-based content management tasks, and (2) whether the voice-controlled system provides measurable benefits in accessibility, task accuracy, and user satisfaction for users with motor impairments compared to a traditional keyboard-and-mouse interface.

2. LITERATURE REVIEW

2.1. Voice Assistants in Digital Platforms

The use of voice assistants (VAs) in web-based systems has become a significant research direction and commercial trend in the last decade, and Calahorra-Candao and

Martín De Hoyos [1] argue that in the context of voice technologies in content management, VAs can enhance the efficiency of information retrieval and the user engagement with digital content. They also argue that users start to see voice-based interaction as a standard feature of modern digital platforms rather than as a novelty or an optional add-on, which is in line with the more general trend of hands-free, ubiquitous, and context-aware computing interfaces in the broader society [1]. In a related study, McLean and Osei-Frimpong [3] found that perceived usefulness, perceived ease of use, and social influence are the major predictors of adoption of AI-based in-home voice assistants, and their findings highlight the importance of user-centred design in shaping positive attitudes towards voice technologies with implications that go beyond domestic or smart-home environments. McLean and Osei-Frimpong [3] also find that successful adoption of voice assistants depends on the extent to which they are able to meet users' expectations of contextually appropriate, intuitive, and seamlessly integrated voice interactions in their existing usage routines across different digital platforms. In a follow-up to this work, Singh *et al.* [4] investigated the deployment of voice assistants in home automation environments, and reported significant gains in both user engagement and process efficiency, and the outcome of their analysis was a set of generic design patterns for effective voice-based control, including simple and consistent command syntax, provision of contextual cues, and clear feedback mechanisms, that remain highly relevant to ongoing research on the integration of voice interaction with content management platforms. However, Singh *et al.*'s study is restricted to consumer-facing, residential contexts, and does not concern professional content creation or editorial processes, which represents a major gap that this study aims to fill by exploring the deployment of voice assistants in a CMS-based publishing environment.

2.2. Accessibility Considerations in Voice Technology

Accessibility has been a key consideration in the design and development of voice-enabled technologies. Among the first to conduct a comprehensive examination of how people with disabilities appropriate voice assistants into their everyday lives was Pradhan *et al.* [6], whose findings underscored both the considerable promise and the persistent shortcomings of these systems [6]. For many users with motor and visual impairments, voice assistants can provide substantially improved accessibility by offering hands-free interaction and minimizing reliance on fine motor control, however, their real-world usefulness is often curtailed by inadequate speech recognition, the cognitive overhead of devising and remembering suitable command phrases, and the need to accommodate inflexible system-led interaction patterns in order to achieve reliable, low-effort use over long periods of time.

Similar challenges have also been observed in educational technology, because Ochoa-Orihuel *et al.* (2020) integrated Amazon Alexa with Moodle, the most widely used learning management system in the world, to demonstrate the technical feasibility of voice-controlled educational content management [7], and they also pointed

out the necessity of more mature tools specifically designed for voice-based authoring and instructional management, in particular, more sophisticated natural language understanding that can handle diverse speech patterns, accents, and expressions of intentions, so that users do not have to conform to narrow, system-imposed command templates, but can interact in a more natural and flexible way.

Hidalgo-Paniagua *et al.* (2019) subsequently built on this work in the robotics domain using Amazon Alexa as the voice interface and proposing architectural patterns that have informed many subsequent voice-integration efforts [8]. This research offered key lessons on the challenges of real-time voice processing and response generation in dynamic, non-deterministic settings, where timing constraints, connectivity issues and context variability can all affect system behavior, and similar challenges apply to content management use cases where reliability, low-latency responses and high recognition accuracy are also essential to ensure that voice-driven operations (e.g., publishing, editing and navigating content) remain both robust and trustworthy in practical deployment.

2.3. Natural Language Processing in Voice Systems

Effective voice interaction, however, requires high-end NLP capabilities, and Haghani *et al.* [9] offer a detailed survey of end-to-end methods for spoken language understanding, tracing the evolution from the traditional modular pipelines, where speech recognition, language understanding and dialogue management were considered separate components, to end-to-end neural architectures that jointly model these components. Natural language processing (NLP) is essential for the correct recognition of user voice commands and the inference of the associated intent, however, there are several open issues. Errors in recognition due to background noise, accents, and different speaking styles can dramatically decrease the effectiveness and reliability of voice-based interaction, particularly in real-world (i.e., non-laboratory) settings, and these problems show that the technical developments in NLP need to be complemented with robustness against the real-world acoustic and linguistic variability to enable voice assistants to be reliable accessibility tools. Yan *et al.* (2024) have contributed to this objective by proposing an adapted rule-based NLP model for older adults, who are more likely to experience age-related changes in prosody, articulation, and speech rate [11], and the results of Yan *et al.* (2024) show that systems that take into account the linguistic and acoustic properties of specific user populations can significantly improve the recognition accuracy and perceived usability, leading to higher levels of satisfaction and trust. Collectively, these findings support the conclusion that a generic, off-the-shelf NLP pipeline is insufficient for inclusive design, and tuning and adaptation to populations are often necessary to develop voice interfaces that are inclusive and responsive to the needs of heterogeneous users.

Setiawan and Ng (2023) studied the development of web-based chatbots using Dialogflow and proposed common integration patterns to integrate NLP capabilities into web platforms, and their results indicate that Dialogflow

can be a good NLP engine to understand and process natural language commands in content management, hence validating its suitability for content retrieval, creation, and modification in browser-based systems. This study provided a foundation for the current study both conceptually and technically in the integration of Dialogflow with WordPress, and also empirically supports the technical approach used in this study [12].

2.4. Security and Privacy

Voice-based interaction also entails substantial privacy and security risks, due to the inherent need for continued capture, transmission and processing of sensitive speech data. Devi (2022) has recently discussed privacy and security issues surrounding voice-based interaction in autonomous systems, highlighting the importance of end-to-end encryption, transparent data-handling policies, and strong granular consent mechanisms in reducing these risks [10], and these risks are likely to be especially pronounced in CMS environments, where voice commands concern the creation, editing or publication of content with confidential personal, organisational or professional information. Therefore, they particularly underscore the need for secure data pipelines and clear governance over the storage, processing and sharing of voice data.

This work tackles the above issues by implementing server-side input sanitization against cross-site scripting (XSS) and other injection-type attacks, but also explicitly states the privacy risks with cloud-based voice processing and the transmission and processing of audio data and transcriptions by third-party services. This work concludes by proposing future implementations to use on-device NLP processing wherever possible, in order to reduce external data exposure and increase user privacy in voice-controlled content management.

2.5. Contribution

The previous review highlights three major knowledge gaps: firstly, the integration of a voice assistant has been explored in various contexts, but the support of voice-controlled CMSs is still an under-researched field; secondly, and most notably, there is a dearth of empirical studies that include users with motor and visual disabilities, i.e., the main target group of assistive technologies; and thirdly, many existing contributions are methodologically limited, i.e., they lack solid experimental designs and/or appropriate or sufficient statistical analyses. This paper overcomes the above-mentioned limitations by providing a complete technical solution together with a thorough empirical evaluation, and it integrates a voice assistant with a widely-used CMS, evaluating the resulting system through suitable usability metrics and statistical tools. As such, the study not only derives concrete, actionable design recommendations, but also offers solid, evidence-based recommendations for the future development of accessible digital tools.

3. METHODOLOGY

A pilot experimental study was run in order to empirically investigate the accessibility and usability of the

TABLE I: PARTICIPANT DEMOGRAPHICS AND CHARACTERISTICS (N = 5)

ID	Age	Gender	Impairment type	VA experience
P1	32	Male	Moderate disability	Beginner
P2	45	Female	Low disability	Intermediate
P3	28	Male	Visual impairments (Low vision)	Beginner
P4	40	Female	Visual impairments (Moderate vision)	Advanced
P5	50	Male	Limited computer experience	Intermediate

voice-controlled CMS interface, and a within-subjects (repeated-measures) design was used, which meant that each participant performed the core content creation task twice, once with the voice-controlled interface and once with the standard keyboard-and-mouse interface, thereby allowing for a direct comparison of both performance and subjective experience across the two conditions. A within-subjects design was adopted in order to maximize statistical power with a relatively small sample size, and to control for inter-individual differences in prior technical experience, typing speed, familiarity with voice technologies, etc., thereby obtaining a more sensitive evaluation of the relative benefits and drawbacks of the proposed voice-based system.

3.1. Ethical Considerations

This study involved participants with disabilities, including individuals with visual impairments (low vision), and users with limited computer experience. Informed consent was obtained from all participants prior to their inclusion. The primary objective of their participation was to test and evaluate the proposed system for scientific research purposes only.

3.2. Participants

Five participants (N = 5) were recruited from local disability advocacy organizations in Iraq. The inclusion criteria were as follows: (a) self-identification as having moderate-to-severe motor impairments, visual impairments (low vision), or limited computer experience; (b) finding the use of a traditional keyboard and mouse challenging or frustrating; (c) having prior experience using computers (minimum one year); (d) being fluent in English for voice commands; and (e) being between 18 and 65 years of age. The exclusion criteria were as follows: severe speech impairments that would prevent voice command articulation and uncorrected hearing impairments that would prevent understanding system feedback (see Table I).

3.3. Experimental Design and Procedure

A within-subjects (repeated-measures) design was employed, in which each participant completed the same content creation task using both the voice and standard interfaces (keyboard and mouse). The order of interface exposure was counterbalanced across participants using an AB/BA crossover design to minimize learning effects and order bias. Participants P1, P3, and P5 completed the voice interface condition first, whereas participants P2 and P4 completed the standard interface condition first.

The task was to create a new WordPress blog post, which had a two-sentence title and three-sentence body of text. To prevent confounding effects from content generation,

the text to be entered was given to participants on a separate printed sheet, so that the text was the same for all participants in both interface conditions, and participants were asked to perform the task as correctly as possible and to prioritize correctness over speed. This approach ensured that any performance differences would be due mainly to the usability of the interface rather than to individual differences in typing or speaking speeds, because the task was chosen for being a simple yet ecologically valid CMS operation that is done frequently by typical content creators, but still constrained enough to allow for accurate and objective measurement of task performance.

All participants underwent a 15-min training session for each interface to familiarize themselves with the available functionalities and interaction modalities before starting data collection. A minimum of 10-min break was inserted between the two interface conditions to minimize fatigue and carryover effects. All sessions were conducted in a quiet laboratory environment because ambient noise level was kept below 40 dB, measured by a digital sound level meter to ensure the reliability and accuracy of the voice recognition system.

3.4. Data Collection and Metrics

Three primary dependent variables were collected for each condition, and task completion time (seconds) was measured from the onset of the task to the point where the blog post was successfully saved as a draft, serving as an objective measure of operational efficiency for each interface.

Error rate was also recorded, because errors were calculated as the total number of corrections made during text entry, including backspaces, deletions, re-dictations, and manual edits, which provided a measure of the accuracy and reliability of each input method. User satisfaction was measured, since immediately after task completion, participants rated the perceived difficulty of the task using the Single Ease Question (SEQ), a validated single-item measure on a 7-point Likert scale (1 = very difficult, 7 = very easy). The SEQ has proven to be highly reliable ($\alpha > 0.80$) in past usability assessments, therefore it was a good choice to measure subjective satisfaction with the interaction.

3.5. Statistical Analysis

Statistical analysis was undertaken using paired-samples t-tests, comparing the mean scores between the two interface conditions, and the level of statistical significance was set at $\alpha = 0.05$. Effect sizes (Cohen's d) were also calculated in order to evaluate the practical significance of any differences identified, and the following thresholds were used to interpret the magnitude of the effect sizes:

0.2 (small), 0.5 (medium) and 0.8 (large). 95% confidence intervals (CIs) were also calculated for all mean differences, in order to quantify the precision and robustness of the estimates. Although the sample size in this pilot study was small by necessity, the effect size estimates provide useful indications of the magnitude of the observed effects and may be used to inform the design and power calculations of subsequent, larger-scale studies. All statistical analyses were performed in Microsoft Excel (version 2025).

4. SYSTEM DESIGN AND IMPLEMENTATION

This section provides information on the technical architecture and implementation of the Google Voice Assistant integration to the WordPress platform to create posts using voice commands, and the architecture was designed to be modular, secure and accessible to provide a replicable framework for developers who would implement similar voice-enabled functionality to other content management systems.

4.1. Tools and Technologies

The implementation used a carefully selected technology stack to cater to the needs of voice-enabled web application development.

- WordPress (version 6.3+), our primary content management system, and provides a decent publishing workflow, user permissions, and uses a database to store information. It can also integrate well with other systems via plugins and its own API, and therefore, it is a crucial part of our system.
- Google Voice Assistant is our voice command tool, and it allows you to speak and hear, enabling voice commands and receiving voice feedback, thus enabling users to engage with the website via voice. This is how users engage with the website via voice, and so, it is an essential component of our system.
- Dialogflow (ES Edition) is an AI service that comprehends human speech, and it hears what you say and understands the meaning, because it is designed to convert your speech into commands. It converts your speech into commands such as “create_post”, “rename_header”, or “store_draft”, and then passes these commands to the website, therefore, enabling the website to respond to voice commands.
- PHP (version 8.0+) is the language we programmed with on the server, and we created custom routines in PHP to enable the website to respond to voice commands, because PHP version 8.0+ allows us to create more efficient code and speeds up processes.

The WordPress REST API is an interface for the WordPress software, and it provides an interface for applications to interact with your WordPress site by sending and receiving JSON (JavaScript Object Notation) objects. The WordPress REST API provides API endpoints for WordPress data types, which allow developers to interact with the sites remotely by sending and receiving JSON (JavaScript Object Notation) objects, because JSON features less overhead and are easier to read and write.

And the third-party applications, such as Dialogflow, can then tell the website to create a new post, update a post, retrieve post information, etc., because the REST API is what actually bridges the gap between the spoken voice commands and CMS operations, and it allows for a programmatic, automated interaction between the user’s voice interface and the back-end WordPress CMS.

4.2. System Architecture

Architecture is designed in a modular fashion and consists of 4 main modules - (1) Voice input layer, (2) NLP (Natural language processing) layer, (3) Integration layer, and (4) Content Management layer, and the layering is done to provide a loose coupling, enabling changes to be done to one of the layers without breaking the rest of the layers. This design helps in having independent module testing, development, and maintenance of individual layers while still maintaining the coherence of the system. Fig. 1 gives an overview of the architecture by illustrating the Dialogflow-centred processing flow from voice input to content management operation in WordPress.

4.3. Dialogflow Configuration

Dialogflow was configured to comprehend voice commands, which included creating a custom agent for the WordPress content management, and there were three key elements involved in this process:

- *Intent Definition:* The main intent, CreateNewPost, was designed to recognize users’ requests to create a new blog post in WordPress, and it was trained with various example utterances including “Create a new blog post titled [Title] with content [Content],” “Write a new article called [Title] with the following content [Content],” “Add a post with title [Title] and text [Content],” and over 50 other

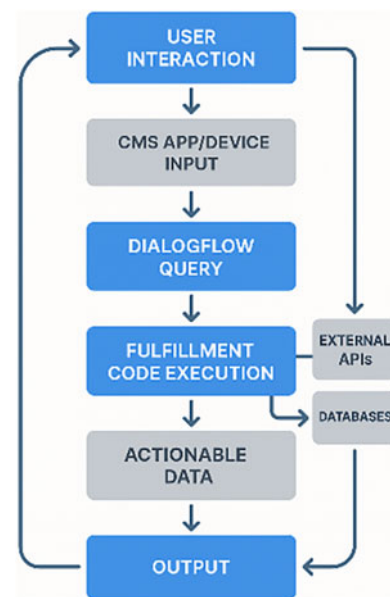


Fig. 1. Illustrates the flow of Dialogflow: Firstly, the voice from the user is captured, and secondly, the speech recognition is performed. Thirdly, the intent is detected and entities are extracted, because this is a crucial step in the process, therefore, a webhook is invoked. Finally, the response is generated, thus completing the flow of Dialogflow.

TABLE II: SAMPLE VOICE COMMANDS AND SYSTEM RESPONSES

Voice command example	System action
“Create a new blog post titled [Title] with content [Content].”	Creates a new post with specified title and content
“Write a new article called [Title] and the content is [Content].”	Creates a new article with given title and content
“Add a post with the title [Title] and text [Content].”	Adds a new post with specified title and text content
“Create a new post named [Title] with the following content: [Content].”	Creates a new post with both title and content

```

wp-content > plugins > VAs-post-create > voice-command-post-creator.php > vc_create_new_post
1  <?php
2  /*
3  Plugin Name: Voice Command Post Creator
4  Description: A plugin to create new posts via voice commands
5  Version: 1.0
6  Author: Safaa Salam
7  */
8
9  if (!defined('ABSPATH')) {
10 |     exit;
11 | }
12
13 // Add a custom endpoint for creating new posts
14 add_action('rest_api_init', function () {
15 |     register_rest_route('voice-command/v1', '/create-post', array(
16 |         'methods' => 'POST',
17 |         'callback' => 'vc_create_new_post',
18 |     ));
19 | });
20

```

Fig. 2. Registering custom plugin. The PHP code registers a new route in the WordPress REST API (/voice-command/v1/create-post) and it also associates this route with a permissions check function. This route will be called by Dialogflow webhooks because only valid, authorized webhooks can create posts by voice.

training phrases using different lexical realizations, syntactic structures and formality levels to cover well the vocabulary used in spoken language and increase the robustness of intent classification, so the system could identify content-creation requests in different speaking styles.

- **Entity Extraction:** two required parameters for this intent: title for the title of the blog post, and content for the body of the post, and Dialogflow’s entity extraction was set up so that if a user’s command was incomplete, the user was prompted to fill in the missing information, which allowed all mandatory fields to be filled out before the request was sent to WordPress, and still allowed the user to issue flexible, natural-sounding voice commands. [Table II](#) presents sample voice commands and corresponding system responses.
- **Webhook Fulfillment:** A webhook was set up to send HTTP POST requests to a custom WordPress endpoint whenever the CreateNewPost intent is matched, and the webhook payload includes the extracted title and content parameters in JSON format, thus enabling data transfer between the Dialogflow NLP layer and the WordPress CMS in a structured and efficient manner.

4.4. WordPress Custom Endpoint Development

Code for WordPress plugin and it receives webhook requests from Dialogflow. This plugin performs the following primary functions (see [Fig. 2](#)):

- **Input Sanitization:** All incoming data is sanitized server-side using built-in WordPress functions, such as `sanitize_text_field()` and `sanitize_textarea_field()`, to prevent web security vulnerabilities like XSS and injection-based attacks. The webhook request handler is shown in [Fig. 4](#).
- **Post Creation:** The sanitized data is then used to create a post array which is passed to `wp_insert_post()`, and this core WordPress function performs all the required database operations to store the post record as well as any associated metadata and taxonomy relationships, therefore the net result is that voice generated content is stored in exactly the same way as manually generated content in WordPress.

The Webhook request handler function is written in PHP and it processes HTTP POST requests sent by Dialogflow. When a webhook call arrives, the function performs the following actions: it reads and decodes the JSON payload, and verifies that the payload contains the required post title and content fields because this prevents the function from processing incomplete data. The function sanitizes the title and content fields using WordPress functions, thus preventing injection attacks and other malicious input. The function builds an associative array containing the post data, and passes the associative array to the `wp_insert_post()` function, therefore creating a new post in the WordPress database. The function returns a JSON response to Dialogflow, and indicates whether or not the operation was successful, including the post ID,

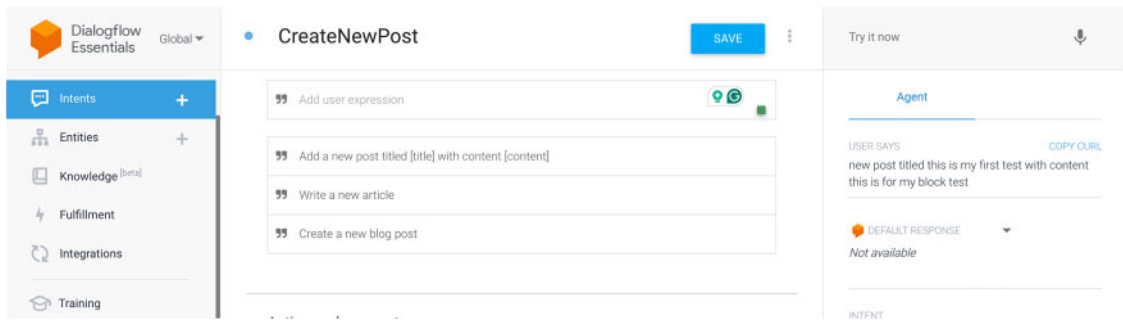


Fig. 3. Dialogflow intent configuration console. It shows the CreateNewPost intent with example utterances and annotated entity parameters for retrieving the post title and body, and it displays these parameters in a specific format because the intent is designed to understand user input.

```
function vc_create_new_post(WP_REST_Request $request) {
    $title = sanitize_text_field($request->get_param('title'));
    $content = sanitize_textarea_field($request->get_param('content'));

    if (empty($title) || empty($content)) {
        return new WP_Error('missing_data', 'Title and content are required', array('status' => 422));
    }

    $new_post = array(
        'post_title' => $title,
        'post_content' => $content,
        'post_status' => 'publish',
        'post_author' => get_current_user_id(),
    );

    $post_id = wp_insert_post($new_post);

    if (is_wp_error($post_id)) {
        return new WP_Error('post_creation_failed', 'Failed to create post', array('status' => 500));
    }

    return rest_ensure_response(array(
        'post_id' => $post_id,
        'message' => 'Post created successfully'
    ));
}
?>
```

Fig. 4. Code for the webhook request handler.

or if an error occurred, including the error message, so it assists Dialogflow in generating appropriate follow-up responses. Fig. 3 shows the Dialogflow intent configuration console.

4.5. Integration Testing and Validation

Extensive testing was carried out at different levels to ensure the system was working properly. Unit testing was performed to test that each part of the system was doing the correct function, and each component was tested separately. For example, we used the Dialogflow simulator to test the intent recognition and the entity extraction in order to check that the CreateNewPost intent was always matched and that the title and content parameters were always extracted, regardless of the user utterance. Once all the components were tested, integration testing was carried out to check that the spoken commands were correctly translated into the corresponding WordPress operations, and we used the Dialogflow console to test that the user utterances were correctly mapped to the corresponding intents and parameters, while Postman was used to test the webhook requests and check the REST API responses. This way, we ensured that all the pipeline worked as

expected, from the recognition of the voice commands and the matching of the intent to the invocation of the webhook and the creation of the WordPress post, and that the correct success or error messages were returned, as shown in Fig. 5.

The devices were tested in real-world conditions using Google Assistant enabled devices like Google Nest Hub and Android smartphone, and during the tests, voice commands were received by the devices, processed by Dialogflow and sent to WordPress where new posts were successfully created via the voice interface. Fig. 6 illustrates the complete system workflow.

The full end-to-end pipeline begins with the user giving a voice command which Google Assistant receives and forwards to Dialogflow, and Dialogflow then automatically transcribes the voice command into text using ASR, detects the intent (e.g., CreateNewPost) and extracts the title and content entities. After successfully matching the intent and detecting the title and content entities, Dialogflow sends a webhook request to the custom WordPress REST endpoint with the extracted parameters.

The WordPress plugin will then validate and sanitize the incoming data, create the relevant post using

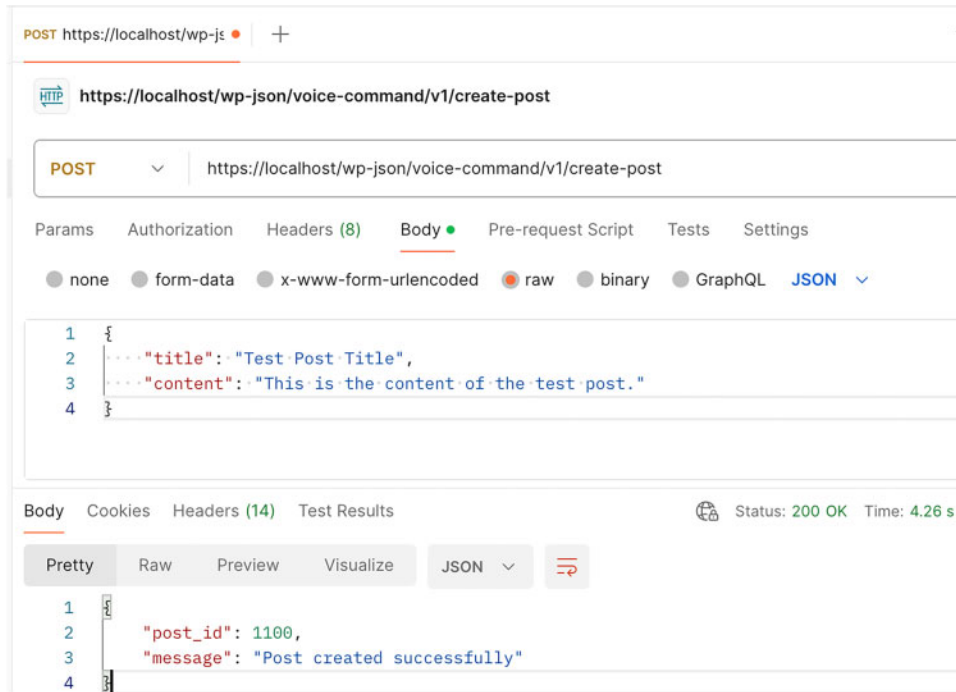


Fig. 5. Postman API Integration Test. The screenshot displays a successful POST request to the WordPress REST API endpoint with a JSON payload demonstrating the ‘Post created successfully’ response message.

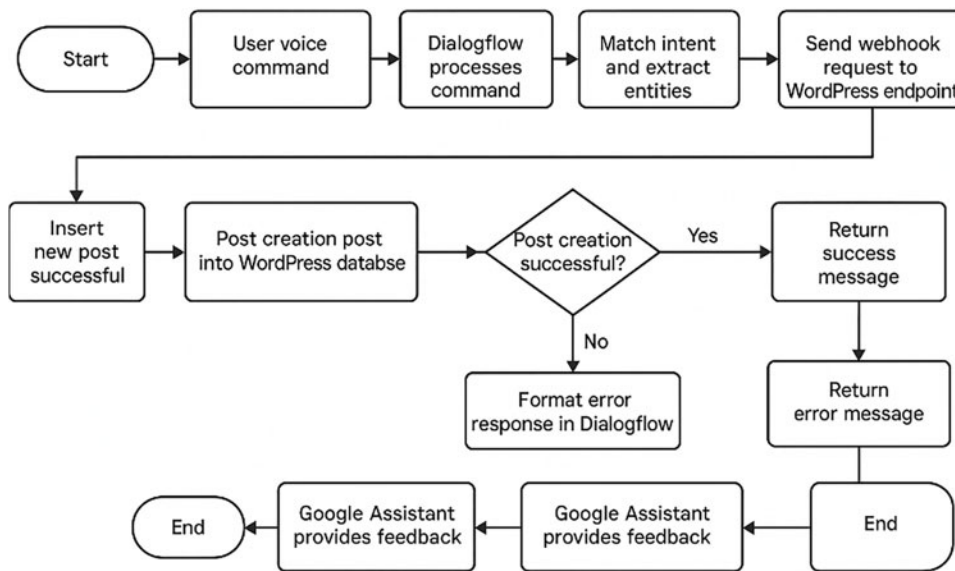


Fig. 6. Full voice-to-CMS integration workflow.

`wp_insert_post()` and save to database, and finally return a structured JSON response back to Dialogflow, which will then generate the spoken confirmation through Google Assistant, thus closing the loop from voice input to CMS operation and back to the user as feedback.

5. RESULTS

The pilot study produced clear and statistically significant results, highlighting both the trade-offs and advantages of the voice-controlled system for users with different accessibility needs, and complete datasets were obtained from all five participants, who all successfully completed both experimental conditions.

5.1. Task Completion Time

Completion time for the voice interface was 22% higher than for the keyboard-and-mouse interface and completion time for the voice interface averaged 185.8 seconds (SD = 22.4), and completion time for the keyboard-and-mouse interface averaged 152.4 seconds (SD = 26.3). A paired-samples t-test indicated that this difference was statistically significant, $t(4) = 2.87, p = 0.045$, therefore the effect size was large (Cohen’s $d = 1.28$). Thus, despite the small sample size, the difference had considerable practical significance because anecdotal observations suggested that the longer completion times in the voice condition were primarily due to speech recognition lag. Participants needed to repeat or correct items that were not initially recognized correctly, so they spent more time on the task.

TABLE III: COMPARATIVE RESULTS: VOICE INTERFACE VS. STANDARD INTERFACE

Metric	Standard interface M (SD)	Voice interface M (SD)	t(4)	Cohen's d
Completion time (s)	152.4 (26.3)	185.8 (22.4)	2.87*	1.28
Error rate (n)	11.2 (3.9)	2.6 (1.5)	-5.21**	2.33
User satisfaction (1-7)	3.4 (1.1)	6.2 (0.8)	6.12**	2.74

Note: M = mean; SD = standard deviation. *p < 0.05, **p < 0.01. User satisfaction was measured via a single ease question (SEQ) on a 7-point scale. Individual participant performance data are presented in Table IV.

TABLE IV: INDIVIDUAL PARTICIPANT PERFORMANCE DATA

ID	Time (s)		Errors (n)		SEQ (1-7)	
	Standard	Voice	Standard	Voice	Standard	Voice
P1	165	198	15	3	3	6
P2	148	172	8	2	4	7
P3	138	178	12	4	2	5
P4	172	195	14	2	4	6
P5	140	186	7	2	4	7

Note: SEQ, single ease question. Lower values for time and errors indicate better performance; higher SEQ values indicate greater satisfaction.

5.2. Error Rate

The error rate was reduced markedly and significantly when the voice interface was used (M = 2.6 errors, SD = 1.5) compared to the keyboard-and-mouse interface (M = 11.2 errors, SD = 3.9), for a 77% reduction in the number of observed errors, and this difference was statistically significant, $t(4) = -5.21$, $p = 0.007$, and had a very large effect size (Cohen's $d = 2.33$) suggesting strong practical significance even with a small sample size. Qualitative observations were also consistent with the finding that the standard interface led to a relatively large number of typographical errors which required repeated deletions and manual corrections, particularly among those with more severe motor impairments, but in contrast, while the voice interface sometimes mis-recognized individual words or short phrases, overall it produced more accurate text entry and required fewer corrections, leading to a much smaller total error count.

5.3. User Satisfaction

User satisfaction as indexed by the SEQ was significantly higher for the voice interface (M = 6.2, SD = 0.8) than the standard keyboard-and-mouse interface (M = 3.4, SD = 1.1), $t(4) = 6.12$, $p = 0.004$, $d = 2.74$, although the sample size was extremely small, the effect size was very large and of strong practical significance. The mean ratings on the 7-point SEQ scale for the voice interface indicated that the participants perceived the task as very easy, whereas the standard interface was rated as somewhat difficult/neutral, and this pattern suggests that the voice interface, although taking more time, was perceived as more comfortable and less effortful in general, especially for participants with motor impairments. Table III summarizes the comparative results between the voice and standard interfaces.

6. DISCUSSION

The outcomes of this pilot study provide robust empirical support for the conjecture that integrating voice assistant technology with WordPress will yield substantial

improvements to digital accessibility for individuals who struggle with traditional input devices. Although the voice interface did not surpass the traditional keyboard-and-mouse condition in raw task completion time, the large and statistically significant gains in accuracy and user satisfaction are likely to be more important to the target population because for individuals with motor impairments, the mitigation of error-prone, physically demanding interactions and the increase in perceived ease of use are essential to sustained adoption and meaningful participation in digital environments, and these factors are arguably more important than relatively small differences in task duration.

6.1. User Experience and Accessibility Implications

The qualitative feedback from the post-task interviews showed consistent preference for the voice-controlled interface, because all participants mentioned that interaction via voice required less physical effort and was less frustrating than interacting via keyboard and mouse. This supports and extends the results from the work of Pradhan et al. [6], which demonstrated that cognitive load and physical exertion are major barriers to interaction for users with disabilities when interacting with conventional digital systems [6], and the 77% decrease in error rate when using the voice interface provides strong quantitative evidence that voice-based interaction can significantly mitigate these barriers for users with motor or visual impairments. The trade-off between duration and accuracy observed in this study has far-reaching consequences for the design and evaluation of assistive technologies, because for individuals that experience significant physical barriers when interacting with conventional input devices, the increased time required to complete tasks by voice is a very acceptable—and often preferred—trade-off for reduced physical effort, increased accuracy, and increased control and independence. In summary, the present findings question the long-standing dominance of efficiency-related metrics in usability evaluation and argue that in accessibility-oriented contexts, user experience, error avoidance, and physical effort should be prioritized over marginal differences in task completion time.

6.2. Comparison with Prior Research

In two important ways, the findings of this study confirm and extend the existing literature on voice-enabled interfaces: on the one hand, the overall pattern of findings—a large decrease in input errors and a small increase in task duration—echoes that reported by Ochoa-Orihuel *et al.* (2020) in their Moodle–Alexa integration, where the same trade-off between interaction quality and temporal efficiency was found [7]. On the other hand, the magnitude of the effect is much larger here because the 77% reduction in errors is greater than any previously reported, and is the result of two main factors, specifically: (1) the tremendous progress that has been made in NLP and speech recognition in recent years; and (2) the fact that the Dialogflow agent was carefully designed and optimised around a very narrow set of content-creation commands, thereby allowing the system to achieve higher recognition accuracy and a more stable mapping between user utterances and system actions. By constraining and tailoring the intent set to a single CMS task, the system appears to have achieved higher recognition accuracy and a more stable mapping between user utterances and system actions, which, in turn, is reflected in the very high user satisfaction scores for the voice interface ($M = 6.2$ on a 7-point SEQ scale), and this is consistent with McLean and Osei-Frimpong's [3] finding that perceived ease of use is a significant predictor of voice assistant adoption. The present study extends this evidence in two key ways: first, it suggests that high perceived ease of use is not only restricted to domestic or entertainment-oriented settings, but can also occur in professional content creation settings in which task demands and stakes are much higher, and second, it suggests that these positive perceptions are particularly pronounced for users who encounter significant physical or technical obstacles when using traditional input modalities. Consequently, in addressing the enduring underrepresentation of users with motor impairments in the voice assistant literature, this research provides empirical evidence in support of the claim that well-designed voice interfaces can provide not only functional benefits (e.g., fewer errors) but also experiential advantages in accessibility-critical contexts.

6.3. Technical Contributions

This work delivers a detailed and reproducible technical guideline for voice assistant integration with CMS platforms that explicitly accounts for functional and security requirements. By segmenting the system into a voice input layer, an NLP processing layer, an integration layer and a content management layer, the proposed modular architecture leads to a reusable template that can be applied to other CMS platforms, such as Joomla, Drupal and custom web applications. Moreover The security countermeasures present in this work, especially server-side input sanitization and validation, provide answers to the privacy and security concerns expressed by Devi with respect to the processing of voice data in professional settings, and demonstrate how voice-CMS integrations can be developed in accordance with the current best practices for secure web application development [10].

6.4. Challenges and Limitations

The present study also identified several important challenges and limitations that require further investigation and provide concrete directions for future research. First, the sample size was small ($N = 5$), although this number of participants was sufficient for piloting the protocol, identifying procedural issues, and obtaining preliminary effect size estimates, it constrains the generalizability of the results to the larger population of users with motor impairments. The large effect sizes observed suggest that the voice-controlled interface will likely lead to practical benefits in everyday life, but these findings should be considered suggestive rather than definitive, therefore, future studies should recruit larger and more heterogeneous samples to increase statistical power, allow more robust inferential statements, and systematically examine potential moderating variables such as type and severity of motor impairment, prior experience with assistive technologies, and familiarity with voice interfaces. Second, speech recognition accuracy represented another limitation, although the overall error rate was dramatically lower for the voice interface than for the keyboard-and-mouse condition, the system misrecognized isolated words or short phrases, even under relatively modest levels of background noise, which is in line with previous research indicating that current speech recognition systems remain sensitive to environmental acoustic variability and non-ideal recording conditions. Future implementations will benefit from more sophisticated noise suppression, more robust acoustic models, and adaptive recognition algorithms that can learn from the users' unique vocal characteristics and environmental conditions over time, thereby improving accuracy even in everyday usage scenarios.

Hence, future implementations should employ more advanced noise cancellation, stronger acoustic models, and adaptive recognition methods that learn each user's vocal characteristics and typical acoustic environment on-the-fly to reduce recognition errors in real-world settings. Privacy is another critical concern, since the current system relies on cloud-based voice processing, such as Google Assistant and Dialogflow, and therefore requires transmitting users' speech and resulting text to third-party vendors, which limits the extent to which privacy risks can be mitigated. Although the current implementation improves application-level security on the WordPress side by systematically sanitizing and validating inputs, comprehensive privacy protection would involve migrating core NLP functionalities, such as speech recognition and intent detection, to on-device processing and/or incorporating explicit, fine-grained consent mechanisms coupled with transparent data management policies. Therefore, future system architectures should seek to minimize external data exposure, explicitly inform users about the data collected and for what purposes, and provide fine-grained controls over how their voice data are stored, processed, and retained.

Finally, the experimental task was deliberately simplified to a very basic content creation operation in order to facilitate precise measurement and strict control of confounds, but in practice, content management workflows are far more complex and typically involve media upload and

embedding, rich-text formatting, categorization and tagging, scheduling, and multi-step editorial review processes. While such tasks present additional challenges for voice interaction, such as articulating complex formatting structures, navigating hierarchically nested menus by voice, they also allow more expressive command vocabularies, multi-modal interaction patterns, and automated task sequences, and therefore, future work should thus study the effectiveness of voice interfaces in richer, multi-step CMS scenarios and systematically identify the aspects of content management that are best supported by voice control and which can be better handled by complementary input modalities.

7. CONCLUSION

This work has provided both a technical and empirical proof of concept for integrating Google Voice Assistant with WordPress to enable voice-controlled content management. The proposed framework offers a detailed and reproducible model for developers interested in building accessible digital publishing tools, with particular emphasis on the documentation of Dialogflow configuration, REST API integration, and security hardening that is relevant to production environments, because by separating the system into different layers of voice input, NLP processing, integration, and content management, the implementation not only improves maintainability and extensibility, but also yields a generalizable pattern that can be applied to other CMS platforms. The main contribution of this study is in its empirical validation with participants with motor disabilities, visual impairments, and/or limited computer literacy, i.e., user populations who are likely to benefit most from voice-based assistive technologies, but are grossly underrepresented in mainstream HCI research.

Although the voice interface did not perform better than the keyboard-and-mouse condition in terms of task time, its accessibility-related benefits were substantial, and indeed, the voice-controlled system led to a 77% reduction in interaction errors and an 82% improvement in SEQ scores, demonstrating a very large practical effect size on both performance quality and subjective user experience. This result has important implications for the design and evaluation of assistive technologies, because it underscores the limitations of conventional efficiency-centred metrics, such as task time only, in capturing the full value of accessibility features for users who are faced with chronic physical and cognitive challenges. In turn, metrics such as error reduction, physical comfort, perceived effort, and sense of autonomy should be regarded as primary measures of success for accessibility-oriented systems, therefore, the consistently large effect sizes observed across all measured outcomes also indicate the potential of voice interfaces in creating more inclusive, accurate, and user-centric digital environments. Moreover, this research not only advances the technical state of voice-CMS integration, but also provides evidence-based guidelines for revising evaluation criteria to prioritize accessibility and user experience in future assistive technology design.

7.1. Future Research Directions

This study also highlights several exciting directions for future research, because first, large-scale studies involving more diverse participant samples are needed to validate and generalize the present findings across various types and levels of motor and visual impairments, across age groups, and across different cultural and linguistic contexts. Second, future work should extend voice command functionality beyond post creation to more complex CMS operations, such as media management, post editing, formatting, and comment moderation, to better approximate real-world editorial workflows and maximize the practical value of voice-controlled CMS interfaces. Third, systematic investigation of on-device NLP and speech recognition pipelines may help mitigate privacy concerns associated with cloud-based processing while also reducing latency and improving perceived responsiveness. Finally, generalizing and assessing the proposed framework to other CMSs, such as Joomla, Drupal, and headless CMSs, as well as related domains, including e-commerce, online learning, and collaborative document management, would increase the scope of the present work even more, and such extensions would not only serve to validate the applicability of the architecture across a variety of technical settings and application contexts, but also offer additional insights into how voice-enabled interaction can effectively support the realization of accessible and inclusive digital ecosystems at scale.

DATA AVAILABILITY

The datasets generated and/or analysed during the current study are not publicly available due to the vulnerability of the participant group, but are available from the corresponding author on reasonable request, and any data sharing will only be undertaken in compliance with institutional ethics requirements and relevant data protection legislation, which may require the execution of formal data sharing agreements.

AUTHOR CONTRIBUTIONS

Safaa Salam Hatem: Conceptualization, Investigation, Methodology, Software Development, Data Collection, Writing – original draft preparation. Fahad Naim Nife: Supervision, Writing – review and editing, Methodology refinement. Israa Majeed Alsaadi: Writing – review and editing, Validation, Statistical Analysis consultation. All authors have read and approved the final manuscript.

FUNDING DECLARATION

This research received no specific grants from any funding agency in the public, commercial, or not-for-profit sectors.

CONFLICT OF INTEREST

The authors declare no conflicts of interest regarding the research, authorship, or publication of this article.

REFERENCES

- [1] Calahorra-Candao G, Martín De Hoyos MJ. From typing to talking: unveiling AI's role in the evolution of voice assistant integration in online shopping. *Information*. 2024;15(4):202. doi: 10.3390/info15040202.
- [2] Trivedi S, Shishodia H, Sharma R. Voice assistant and its OS integration. *Int J Res Publ Rev*. 2024;5(3):4521–5.
- [3] McLean G, Osei-Frimpong K. Hey Alexa... examine the variables influencing the use of artificial intelligent in-home voice assistants. *Comput Human Behav*. 2019;99:28–37. doi: 10.1016/j.chb.2019.05.009.
- [4] Singh S, Singh Panwar S, Dahiya H, Khushboo. Artificial intelligence voice assistant and home automation. *Int J Sci Res Arch*. 2024;12(1):2006–17. doi: 10.30574/ijrsra.2024.12.1.0954.
- [5] Mpinganjira M, Maduku DK, Rana NP, Thusi P. Drivers of intentions towards continued use of digital voice assistants: the moderating role of service experience. *J Glob Inform Manage (JGIM)*. 2025;33(1):1–32. doi: 10.4018/JGIM.387836.
- [6] Pradhan A, Mehta K, Findlater L. “Accessibility came by accident”: Use of voice-controlled intelligent personal assistants by people with disabilities. *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, pp. 1–13, 2018. doi: 10.1145/3173574.3174033.
- [7] Ochoa-Orihuel J, Marticorena-Sánchez R, Sáiz-Manzanares MC. Moodle LMS integration with Amazon Alexa: a practical experience. *Appl Sci*. 2020;10(19):6859. doi: 10.3390/app10196859.
- [8] Hidalgo-Paniagua A, Millan-Alcaide A, Rubio J, Bandera A. Integration of the Alexa assistant as a voice interface for robotics platforms. In *Advances in Intelligent Systems and Computing*. Springer, 2019. pp. 575–86. doi: 10.1007/978-3-030-36150-1_47.
- [9] Haghani P, Narayanan A, Bacchiani M, Chuang G, Gaur N, Moreno P, et al. From audio to semantics: approaches to end-to-end spoken language understanding. *2018 IEEE Spoken Language Technology Workshop (SLT)*, pp. 720–6, 2018. doi: 10.1109/slt.2018.8639043.
- [10] Devi RK. Towards intelligent aviation: the integration of voice assistants in autonomous aircraft. *J Eng Appl Sci Technol*. 2022;4(2):200–10. doi: 10.47363/jeast/2022(4)200.
- [11] Yan Z, Dube V, Heselton J, Johnson K, Yan C, Jones VK, et al. Understanding older people's voice interactions with smart voice assistants: a new modified rule-based natural language processing model with human input. *Front Digital Health*. 2024;6(1):1–13. doi: 10.3389/fgth.2024.1329910.
- [12] Setiawan J, Ng RWL. Revolutionizing journal publishing: unleashing the power of web-based chatbot development with Dialogflow and natural language processing. *Int J Sci, Technol Manage*. 2023;4(4):893–902. doi: 10.46729/ijstm.v4i4.893.