

RESEARCH ARTICLE

A Framework to Improve Web Form Accessibility for the Visually Impaired

WAFAA ABU SHAMLAH HAKAMI^{1,2} AND ARWA YOUSUF AL-AAMA¹¹Department of Computer Science, King Abdulaziz University, Jeddah 22254, Saudi Arabia²Department of Computer Science, Jazan University, Jazan 45142, Saudi Arabia

Corresponding author: Wafaa Abu Shamlah Hakami (wmohammedhakami@stu.kau.edu.sa)

ABSTRACT Due to the growth in the number of Internet users and the services provided by websites, making websites accessible to everyone has become necessary. Web forms play a crucial role in collecting information from users to provide services. Despite the increasing number of visually impaired users, with the World Health Organization reporting at least two billion people globally with vision impairments, website developers still fail to adhere to web accessibility guidelines, making it difficult, if not impossible, for visually impaired individuals to easily fill out web forms. This study introduces the Web Form Accessibility Framework for the Visually Impaired (WAFI). By using WAFI, web form developers can create more accessible web forms for visually impaired users, enabling them to fill out forms faster, easier, and with fewer errors. WAFI is designed with components that encompass all necessary aspects for building an accessible web form. The framework was thoroughly evaluated by experts and tested with a sample of visually impaired users. They were asked to fill out two web forms: AWF, which followed WAFI, and IWF, which did not. The results clearly demonstrate that AWF was submitted faster and with fewer errors than IWF. Based on the evaluation and testing outcomes, the framework underwent further refinement, leading to the production of the final framework. It is hoped that WAFI will significantly improve web form development and enhance the overall experience of visually impaired users.

INDEX TERMS Accessibility, framework, screen reader, visually impaired, web form.

I. INTRODUCTION

Visual impairments represent significant health challenges due to their increasing prevalence, making them one of the most common causes of disability worldwide [1], [2]. However, the indispensability of the World Wide Web in various aspects of life is undeniable, serving as a crucial source of information and services in education, economics, government, entertainment, and more. Ensuring equitable access to the Internet for everyone is of utmost importance. Unfortunately, accessing the Web is particularly challenging for visually impaired users, given its primarily visual nature, despite advancements in assistive technologies (ATs) like screen readers [3]. Moreover, with the growing prevalence of web forms used by governments and the private sector, people with visual disabilities face difficulties in dealing with web

forms compared to those without visual impairments. In light of these challenges, our study aimed to develop the Web form Accessibility Framework for the Visually Impaired (WAFI) to aid web form developers in creating more accessible web forms, enabling visually impaired users to fill them out faster, easier, and with fewer errors.

A. MOTIVATION

Web form filling is essential in numerous tasks, such as booking airline tickets, reserving hotel rooms, registering on different websites, shopping online, applying for jobs, and providing delivery information in online food delivery apps. However, visually impaired users encounter difficulties in filling out web forms [4]. To promote equal web access for all, regardless of disabilities, some researchers have addressed this topic to motivate developers to enhance web accessibility for visually impaired users [5] and [6]. Nevertheless, little progress has been made in web form

The associate editor coordinating the review of this manuscript and approving it for publication was Giuseppe Desolda¹.

accessibility, leaving these issues unresolved. This motivated our work in proposing WAFI to assist web form developers in improving web form accessibility for visually impaired users, empowering them to complete web forms independently without relying on other people's assistance.

B. OBJECTIVES

Our objective was to develop a framework that empowers web form developers to build web forms that enable visually impaired individuals to:

- Fill out forms with improved speed.
- Fill out forms with improved accuracy.
- Fill out forms with improved ease.

II. LITERATURE REVIEW

A. CONCEPT OF WEB ACCESSIBILITY

In today's digital world, accessibility is a crucial concept that should be understood by everyone. Accessibility refers to the ability of people with disabilities to use computers and other digital devices in a meaningful and safe manner [7]. By ensuring that all websites are accessible, we can guarantee that all users have equal access to information and resources on the Web. The World Wide Web Consortium (W3C) emphasizes that web accessibility is "essential for developers and organizations that want to create high-quality websites and web tools and not exclude people from using their products and services" [8]. The Web Accessibility Initiative considers accessibility as "something we all want" [9]. It is important to note that accessibility goes beyond just visual appearance; it is about recognizing and respecting the diverse needs of every individual who uses a website, regardless of their ability levels or disabilities. Different types of disabilities exist, and each person has their unique requirements when it comes to accessing and using online content [10].

B. WEB CONTENT ACCESSIBILITY GUIDELINES (WCAG)

The Web Content Accessibility Guidelines (WCAG) constitute a set of standards that web developers must adhere to in order to make their websites accessible to people with disabilities. These guidelines were developed by the (W3C), an international non-profit organization dedicated to web standards [11]. Currently, there are more than 20 different versions of WCAG in use. The primary objective of WCAG is to ensure that users with disabilities can effectively utilize web-based information and applications, regardless of their level of computer expertise or the time available to them [3]. WCAG also plays a crucial role in enhancing overall usability by providing developers with guidelines to follow when creating websites or digital content to meet the specific requirements set forth by the W3C. The guidelines are categorized into three levels: A, AA, and AAA, each level encompassing different accessibility requirements.

- Level A: This is the most basic level of website accessibility, which requires no special accommodations

for users with disabilities beyond those needed for general web accessibility standards.

- Level AA: This level includes all the requirements of Level A accessibility and adds unique features like keyboard navigation or screen readers (helpers that read text aloud while browsing) to enhance the user experience for people with disabilities.
- Level AAA: This represents the highest standard for web accessibility. It is the only standard that provides a comprehensive approach to making websites accessible to all users, including those with disabilities.

C. FOUR PRINCIPLES OF WEB ACCESSIBILITY

Web Content Accessibility Guidelines are organized under four principles: perceivable, operable, understandable, and robust [12]. The first principle of web accessibility is perceivable, which means that the information on a website needs to be easily expressed in a language and format that people with disabilities can understand. The second principle is operable, ensuring that the website is usable by everyone, including those with disabilities. The third principle is understandability, which means the website should be designed in a way that allows all users, regardless of their characteristics (sighted or partially sighted, young or old, male or female), to comprehend its content and use it correctly [13]. Finally, the fourth principle is robustness, meaning the rules of web accessibility require that all users can access a site without any problems and obtain the desired information.

D. MAKING WEB FORMS ACCESSIBLE

As we prioritize accessibility for forms, the following section discusses efforts related to simplifying the task of filling out forms.

1) PAPER-BASED FORMS

To streamline form-filling, save time, and reduce paper usage, researchers have implemented voice recognition technology [14]. They developed an application that allows visually impaired users to complete forms using their voice. Once the form is filled out, it can be printed directly from a mobile device connected to a printer. In a different approach, researchers [15] utilized haptic feedback to help visually impaired users locate specific fields on a printed form. They used smartwatches with built-in audio-haptic feedback empowering visually impaired individuals to independently fill out the form. Similarly, [16] proposed an alternative method for filling out printed forms using a custom 3D-printed smartphone attachment. This attachment, in combination with well-established computer vision algorithms, generates dynamic audio instructions that direct blind users to different form fields. Additionally, [17] designed a form filling assistant prototype specifically for visually impaired individuals, utilizing affordable off-the-shelf products. The prototype consisted of a smartphone with a cradle, a clipboard with a sliding ruler, and a black ballpoint pen. Testing their

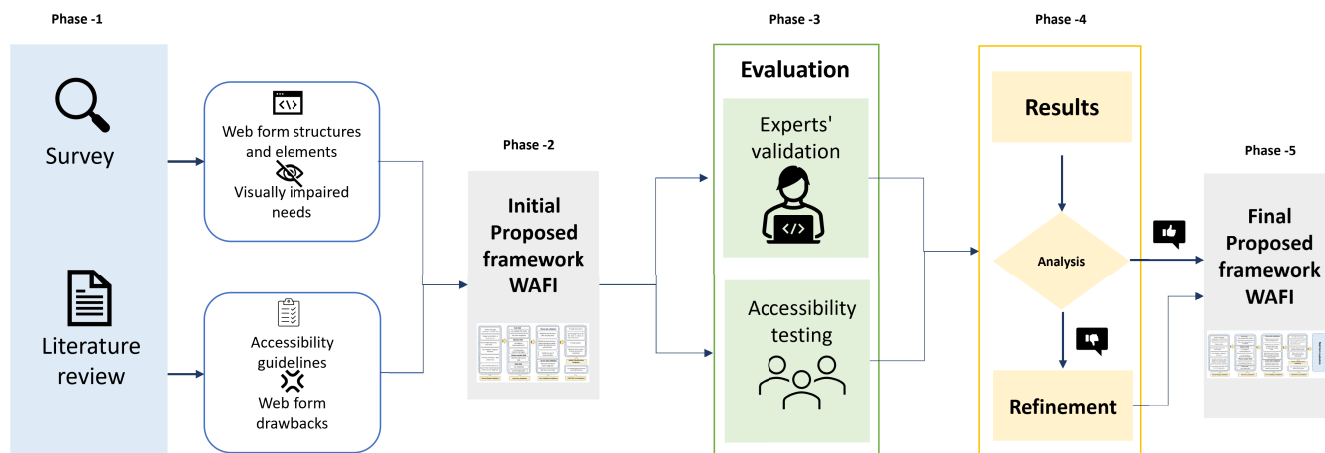


FIGURE 1. Research methodology.

prototype with a sample of blind participants yielded positive feedback.

2) WEB-BASED FORMS

In addition to the previously discussed solutions for filling out printed forms, [18] presented a solution to expedite the process of filling web forms by automatically transmitting user profile data over a network. This is achieved through a mobile application that scans a QR code displayed on a webpage, identifies the user profile data, and sends it to a campaign server. Furthermore, [19] introduced a framework designed as an AT to aid blind users in completing web forms. The framework utilizes the Verify Instead of Filling method, text-to-speech technology, and XForms to enable automated web form filling. However, it focuses solely on specific personal information and does not address additional fields or elements in web forms. Additionally, the user interface of the framework appears to be available only in English, which limits its usability for non-English speaking users. To emphasize the significance of alerting users to error messages in forms, [20] conducted a study on the impact of screen size on the usability of Arabic forms for smartphones. They found that the likelihood of errors occurring on a larger screen is lower than on a smaller screen. The study also recommended placing the error message text box below for smaller screens. In the context of automated web form filling, [21] developed a web application that utilizes data mining and machine learning algorithms to update users' data, streamlining the process. Moreover, another solution developed by [22], which aimed to assist visually impaired and uneducated individuals in filling Challans (a common method of crediting money to bank accounts through a form used in India and Pakistan) used face detection and voice recognition.

III. RESEARCH METHODOLOGY

The research methodology consisted of five main phases, as shown in Fig. 1. The first phase aimed to gain a

comprehensive understanding of accessibility concepts, web form components, and how visually impaired users interact with them. This phase also involved identifying web form issues and gaps in existing work to define the problem and objectives of the study. Based on the findings from the first phase, the second phase focused on identifying and producing the basic components for building the initial proposal of the framework. The proposed framework was then evaluated in the third phase through feedback from experts and accessibility testing involving visually impaired users. In the fourth phase, the results from both evaluations were recorded and analyzed to determine what aspects were successful and what areas needed further refinement. Finally, the final version of the proposed framework was produced, incorporating the insights and improvements from the previous phases.

IV. DESIGNING AND ANALYZING THE INITIALLY PROPOSED FRAMEWORK

The initially proposed WAFI framework consisted of the basic components required for overall form design. Additionally, it included proposed solutions to facilitate data entry and error correction, a proposed method to help visually impaired individuals solve the CAPTCHA test, and helpful considerations to improve the web form filling process. Fig. 2 illustrates the framework which we initially proposed as a reference to enable web developers to create more accessible web forms that are easier and faster for visually impaired users to fill and that enable them to fill them out with less errors. The following sections provide more details about each component of WAFI:

- **Part- 1: Overall Web Form Design Guidelines:** This component addressed the general design of the web form. It is based on the WCAG guidelines related to components that may appear in web forms, including the following:
 - 1) The main language used for the web content must be defined as the primary language and specified

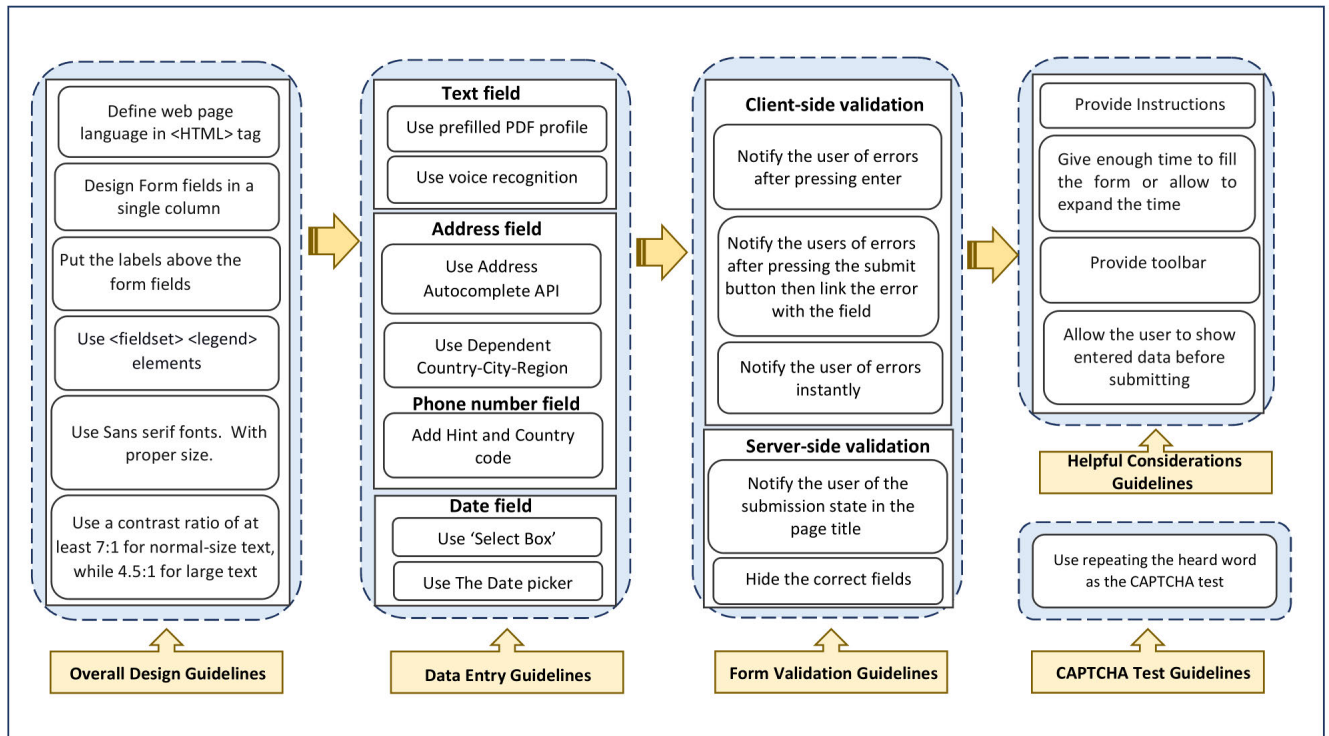


FIGURE 2. The initial proposed framework (WAFI).

using the <HTML> element at the beginning of the page, along with the direction of content (left to right or right to left) to ensure proper reading by screen readers.

- Fields should be designed in a single column to prevent visually impaired users from misunderstanding or misinterpreting fields due to the linear nature of the screen reader.
- Labels must be positioned above the fields. According to WCAG, placing labels above a form's textboxes and selection boxes helps reduce horizontal scrolling for people with low vision and mobile device users [23].
- A web form must be designed with a proper <fieldset> and <legend> to organize the input fields. The <fieldset> element groups related fields, while the <legend> element provides a header to identify the group, making it easier to understand due to the association of related fields [24].
- The font type should belong to the Sans Serif font family with the appropriate font size, for instance, 14 points for Arabic text and 12 points for English.
- The visual presentation of text and background for normal font size must have a contrast ratio of at least 7:1, and for large-scale text, it should be 4.5:1 [25].

- **Part- 2: Data Entry Guidelines:** This component pertained to data entry and methods to facilitate it for

visually impaired users when filling out different types of web form fields, which include:

- One important solution proposed in WAFI to improve the form filling process in a shorter time with higher accuracy is the automation of the input process from an uploaded file instead of manual entry. The uploaded file may contain most of the required information in a web form, such as the user's CV or any prefilled file that the user can create with the assistance of a sighted person. This file can serve as a profile to be uploaded whenever needed. Fig. 3 illustrates the mechanism for extracting and filling out forms from the CV uploaded by the user.
- Speech processing is also employed to help visually impaired individuals interact with a web form, especially for those who prefer speaking over writing.
- To facilitate the process of entering data that requires a specific format with minimal possibility of errors or unexpected formats, the following considerations are made:
 - The user can enter the date and time using multiple selected boxes in the required format instead of using a text box for the full date. An optional date picker can also be added, which the user can enable if preferred.
 - For entering addresses, WAFI utilizes IP-based geolocation to obtain a user's geographical

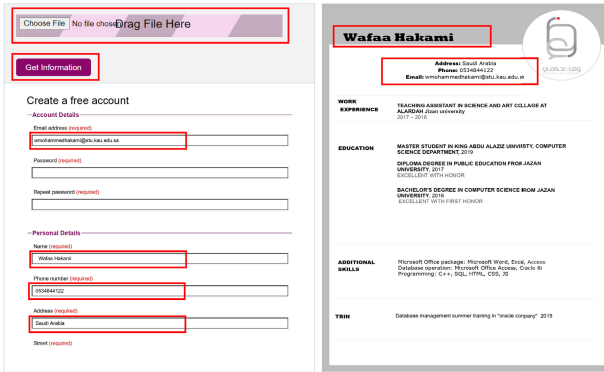


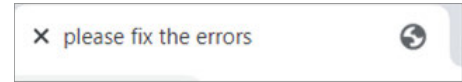
FIGURE 3. Extracting personal information that is required in most web forms like name, email, phone number, and address from the users' CVs.

location. If the user's browser supports geolocation and they agree to provide permission, their location will be used to fill in the address details. Additionally, a dependent country-city-region approach can be employed if geolocation is not enabled, allowing the user to select the country, after which only dependent cities and regions will appear in the next drop-down. This method helps eliminate typos and is more efficient than selecting from a long drop-down list that includes all countries and cities worldwide.

- To ensure that phone numbers or credit card numbers are entered in the required format without any confusion, input masking is used to autoformat the number. Furthermore, if needed, an inline description of the field is added to inform the user about the requirement. For the phone number field, the attribute "type=tel" is included in the input element, prompting the telephone input keyboard to appear on mobile devices.

• **Part- 3: Form Validation Guidelines:** This component addressed methods to notify visually impaired users of errors that may occur during form filling and how to enable them to correct these errors easily. The guidelines included the following:

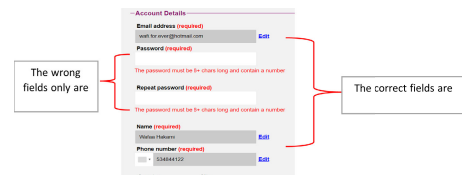
- 1) The user must be promptly notified of the success or failure of the form submission using the page title, ensuring that screen reader users receive feedback as soon as the web page loads.
- 2) To strike a balance between providing helpful and accessible information about errors and avoiding overwhelming the user with numerous error messages, a combination of client-side and server-side validation should be employed.
- 3) Feedback about errors and how to rectify them should be clear and sensible.
- 4) Error messages must be expressed in plain language without using codes, precisely indicating the



(a)



(b)



(c)

FIGURE 4. Design of the error message: (a) Notifying the user about the submission state in the page title; (b) Error explanation with an example of the correct format; (c) Hiding the correct fields after server validation.

problem, and offering constructive suggestions for a solution.

- 5) When notifying users of errors, provide clear descriptions about the nature of the error to help the user identify where and why the error occurred, whether due to an unfilled required field, an invalid value, or other reasons.
- 6) Using the color red to inform users about errors is beneficial, as it is commonly associated with errors [26]. This also aids people with low vision in understanding the meaning of the message.
- 7) Error messages should be positioned where the user expects to see them and can correct them quickly. For desktops, this means placing them next to the fields (right of the field for English forms and left for Arabic). For mobile devices, they should be placed below the field or below the field for both [27], [28], and [29].
- 8) If validation occurs on the server side, the form should be returned with the user's data still in the fields instead of being returned empty.
- 9) To enhance the user experience, only erroneous fields should be visible on the error page, and all validated data should be summarized below with an "Edit" link rather than reloading the full page and showing all form fields. Fig. 4 (a-c) demonstrate the design of error messages.

• **Part-4: CAPTCHA Test Considerations:** This component proposed an alternative method to help visually impaired users solve CAPTCHA tests. The method is based on speech and hearing instead of visual perception, making it accessible and easy to perform for visually impaired users and screen reader users. The following is a description of this method:

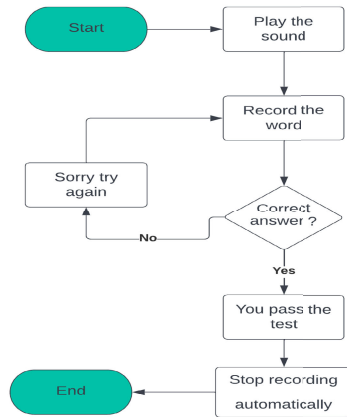


FIGURE 5. Flowchart of the proposed CAPTCHA.

Name:

Repeat the heard word

Description:

This approach for human verification utilizes voice processing with the following steps:

- 1) The user will listen to a randomly selected word from a dictionary.
- 2) The user will be asked to repeat the pronunciation of that word.
- 3) If the pronunciations of the two words match, the user will pass the CAPTCHA test.

Features:

- Playback feature to relisten to the word.
- Dictionary containing short and easy words.
- Support for at least English and Arabic languages.

Recommendations:

Use an HTML version that supports audio, such as HTML5 and above.

Flowchart: Refer to Fig. 5.

- **Part- 5: Helpful Considerations:** This component included additional features that can enhance the accessibility of web forms, as follows:

- 1) General instructions should be provided at the top of the form and inline instructions throughout the form to guide users. These instructions should indicate required and optional inputs, data formats, and other information to assist users in using form controls and completing the filling process.
- 2) If a session timeout is necessary for security or other reasons, it should be adjustable and accommodating to users. To ensure that the time spent to be filled by people with no impairment and suitable with the form length, and provide a warning dialog that allows users to extend the time limit by a simple keypress, such as by pressing the space bar.
- 3) For low-vision and color-blind users, a customizable toolbar is recommended. This toolbar enables users to change the color theme, font type,

and font size according to their preferences and accessibility needs.

- 4) To provide users with confidence in the information they entered, an option is included at the end of the web form (before the submission button) for users to review their entries.

V. EVALUATING THE PROPOSED FRAMEWORK

In this phase, we conducted a preliminary validation of our proposed framework following the methodology of Quiñones et al. [30], which provides various options for validation. For our research, we utilized validation through expert judgment and user testing. The primary goal of this validation was to review and assess the WAFI framework through various experiments.

A. VALIDATION THROUGH EXPERT JUDGMENT

At this stage, we sought input from experts regarding the usefulness, completeness, and effectiveness of the WAFI framework. The panel of experts included two specialists in Human-Computer Interaction and two web developers. To evaluate the framework, we utilized a questionnaire comprising five dimensions:

- D1 - Utility: How useful is the framework?
- D2 - Clarity: How clear is the framework?
- D3 - Ease of use: How easy it is to understand framework components?
- D4 - Necessity of an additional checklist: How necessary is it to complement the framework with a checklist?
- D5 - Applicability: How applicable is the framework to designing a web form?

Each expert was asked to rate each framework component on each of the five dimensions, using a 5-point Likert scale.

B. VALIDATION THROUGH USER TESTING

As the second stage of validation, we conducted accessibility testing on two samples of web forms. The first form was an inaccessible web form that we created. We added to it input fields that are similar to the ones found in a job application that existed on the official website of one of the Saudi ministers. We also followed their same design. We referred to this form as IWF. The second form was an accessible web form that we also created. We added to it input fields that are similar to the ones found in an official registration form. We followed the WAFI framework during the design stage to make it accessible to the best of our knowledge. We referred to this form as AWF. To avoid the learnability effect on users, AWF and IWF were designed with similar types and amounts of form elements but with some different requirements. The testing was performed with the assistance of a sample of visually impaired individuals. Our target audience comprised people with any type of visual impairment who were reasonably familiar with using the Internet and comfortable using AT. We recruited participants based on this criterion through Twitter and blind and visually impaired foundations, and 13 volunteers agreed to participate

TABLE 1. Experts' evaluation results.

	Ease of use	Clarity	Utility	Necessity of checklist	Applicability
Expert -4	4	4	5	5	5
Expert -3	3	3	5	5	5
Expert -2	4	3	5	5	5
Expert -1	4	4	5	5	5

in the study. Each participant was asked to complete both AWF and IWF to ensure that AWF, created using WAFI, was accessible to individuals with visual disabilities. We evaluated AWF using three metrics: accuracy, speed, and ease of use. The test commenced with a pre-test interview to gather participants' demographic information and facts about their usage of web forms. Next, we observed the participants interacting with AWF, designed based on WAFI, while they performed specific tasks they were asked to complete (refer to Appendix A). During the final step, we conducted a post-test interview to determine the participants' satisfaction with AWF. The following section presents and discusses the results of the testing.

VI. RESULTS AND ANALYSIS

A. EXPERTS VALIDATION RESULTS

The experts evaluated the WAFI framework based on the five dimensions discussed in Section V-A, using a 5-point Likert scale where 5 indicated full compliance and 1 indicated non-compliance. The following is a detailed explanation of their findings.

- Utilization of the framework: The evaluators unanimously agreed that the framework's components are sufficient to build an accessible web form, as it comprehensively covers all the accessibility aspects needed for visually impaired users.
- Clarity of the framework: Some components of the framework were not entirely clear to some of the evaluators.
- Ease of use and understanding of the framework: Some evaluators had questions about the usage of certain components.
- Necessity of an additional checklist: All the evaluators considered it important to add a checklist as an evaluation tool for web forms designed based on the WAFI framework.
- Applicability of the framework: The evaluators expressed their comfort with the applicability of the framework.

Table 1 summarizes the results of the experts' evaluation of the WAFI framework.

B. ACCESSIBILITY TESTING RESULTS

1) PARTICIPANTS PRE-INTERVIEW RESULTS

The study participants included four males and nine females. Three participants were high school graduates, nine had a

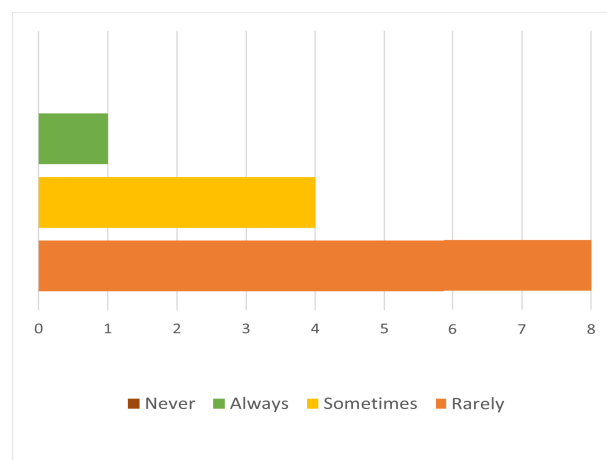


FIGURE 6. Participants needed help filling out web forms.

bachelor's degree, and one had a master's degree. The mother language of all the participants was Arabic. Nine participants (69.23%) were completely blind. Two participants were partially blind, representing 15.38% of the sample. One participant had low vision, representing 7.69% of the sample. One participant was colorblind, representing 7.69% of the sample. In terms of the ATs used to fill out web forms, 12 participants relied on screen reader technology to browse the Internet, representing 92.31% of the sample. Among them, three participants used a reader that is compatible with a Braille keyboard, and one participant used a reader that is compatible with speech recognition software. Additionally, one participant reported not using any AT except for some tools for zooming. With respect to the screen readers used to browse the Internet, 8 participants (61.54%) relied on Apple's iOS Voice Over, and 4 participants (30.77%) used Microsoft NVDA. Regarding the web browsers they normally used, 8 participants reported using Safari, representing 61.54% of the sample. Four participants used Chrome, representing 30.77% of the sample, and 1 participant used Firefox, representing 7.69% of the sample. Concerning the devices they used, 9 participants reported using smartphones while browsing the Internet, representing 69.23% of the sample, and 4 participants used personal computers (desktop, laptop, notebook, etc.), representing 30.77% of the sample. In terms of the help needed to fill out web forms, 8 participants reported that they seldom needed help, representing 61.54% of the sample. Four participants sometimes needed help to fill out web forms, representing 30.77% of the sample. One participant reported that they always needed help to fill out web forms, representing 7.69% of the sample. As for the usage of web forms, Fig. 6, Fig. 7, and Fig. 8 illustrate their need for help to fill out web forms, the common uses of web forms, and the common difficulties preventing completion of the web forms.

2) TASKS COMPLETION RESULTS

After the pre-interview, the participants started the accessibility testing. The results were recorded for each task for

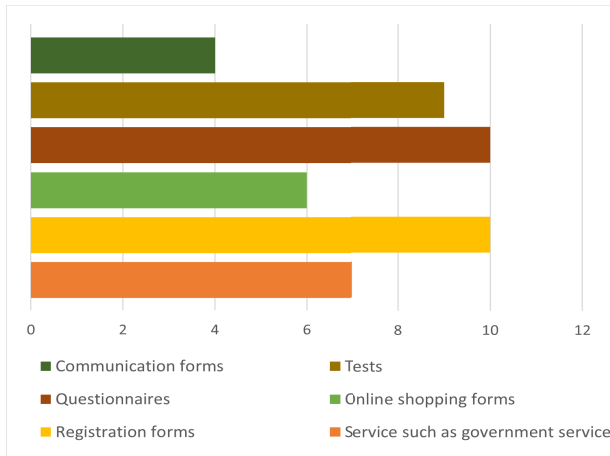


FIGURE 7. Common uses of web forms by the participants.

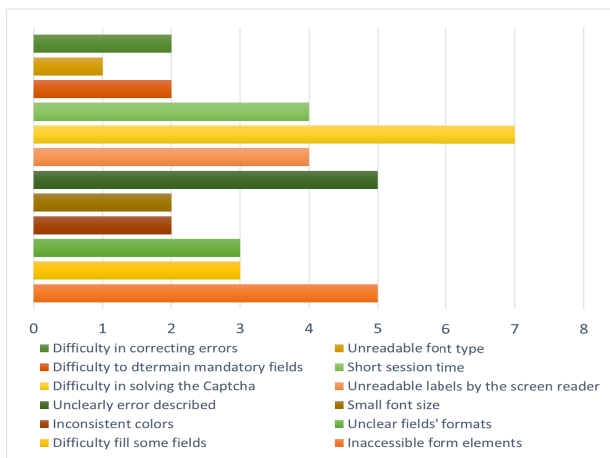


FIGURE 8. Common difficulties that prevent participants to complete the web form.

the five test cases. The first test case, TCI (1), pertained to tasks related to how the user enters data into the form, encompassing seven tasks for different items in the web form. The second test case, TCI (2), involved tasks related to how users were notified of errors and how they resolved them. The third test case, TCI (3), covert tasks related to passing the CAPTCHA test. The fourth test case, TCI (04), addressed tasks related to form submission. The fifth test case, TCI (05), dealt with tasks related to the helpful considerations we added to assist users while filling out the web form. Tables 2 - 6 provide a description of each test case and the results of its tasks. In the tables, (TCI) refers to the test case ID, (Completed) refers to the number of participants who completed the task, (Incorrect) refers to the number of incorrect actions, (Correct) refers to the number of participants who completed the task with the correct input, and (Avg) refers to the average time taken to complete each task -in seconds-.

3) COMPARISON BETWEEN AWF AND IWF RESULTS

In order to evaluate WAFI’s effectiveness, we conducted a comparison between AWF and IWF. Both forms contained an equal amount and type of form elements. To measure their

TABLE 2. TCI (1) testing results.

TCI	Groups	Tasks	Completed	Incorrect	Correct	Avg
TCI (1)	Data entry	Filling text fields using prefilled PDF	13	4	13	52 s,
		Filling text fields using voice recognition	1	0	1	29 s,
		Entering the date using the date picker	3	6	1	79 s,
		Entering the date using selecting box	13	0	13	38 s,
		Entering the address using the API address	13	0	13	NA
		Entering the address using the selection box	13	2	13	38 s,
		Entering the phone number with help of a hint and country code	13	2	13	27 s,

performance, we timed the duration it took to complete each task and checked the accuracy of the data entered. Accuracy was calculated by dividing the number of correct entries by the total number of form elements. Additionally, we recorded if the participant was able to successfully submit the form in the submission status column. Table 7 presents the results of the AWF and IWF comparison.

4) POST-INTERVIEW RESULT

After completing the accessibility test, a post-interview was conducted to evaluate participants’ satisfaction with the AWF components. They were asked 13 questions using a 4-point Likert scale, where the value 1 indicated that they did not get any benefit from the component, and value 4 indicated that the component was very useful for them. For each option, the Relative Importance Index (RII), Mean, and Standard Deviation (Std) were calculated. Table 8 shows the results of the post-interview.

VII. DISCUSSING THE FINDINGS

A. DEMOGRAPHIC ANALYSIS

The results indicate that neither gender, education, nor educational level had an effect on performance, as no differences were found in the results between males and females in our sample. One possible reason for this is that all participants were familiar with using the Internet and comfortable using AT. While participants with different screen readers and browsers could complete the test, there was a slight difference in the completion time for some participants who sped up their screen reader reading, depending on their ability to understand rapid speech.

B. TASKS COMPLETION RESULTS ANALYSIS

The results obtained from the users’ tasks given in AWF testing showed that all participants passed the task “Filling the form using a prefilled PDF file.” We expect that the reason

TABLE 3. TCI (2) testing results.

TCI	Groups	Tasks				Avg
			Completed	Incorrect	Correct	
TCI (02)	Form validation	Entering all required fields	13	0	13	233 s,
		Entering the expected format	13	0	13	NA
	Error notification	Accessing and understanding instant validation (client side)	13	1	13	NA
		Accessing and understanding validation after pressing enter (client side)	13	2	13	NA
		Accessing and understanding validation after pressing submit and linking the error with the field (client side)	13	3	13	NA
		Accessing and understanding validation after submitting the form (server side)	12	1	12	NA
	Error recovery	Fixing errors with instant validation. (client side)	13	0	13	7 s,
		Fixing errors that were validated after pressing enter (client side)	13	0	13	12 s,
		Fixing errors that were validated after pressing submit and linking the error with the field (client side)	13	2	13	50 s,
		Fixing errors that were validated after submitting the form (server side)	12	0	12	87 s,

TABLE 4. TCI (3) testing results.

TCI	Groups	Tasks				Avg
			Completed	Incorrect	Correct	
TCI (03)	Pass CAPTCHA test	Understanding the idea of the test	13	NA	13	11 s,
		Accessing the button CAPTCHA	13	NA	13	23 s,
		Passing the test	12	1	12	35 s,

they prefer to use this method to fill text fields is that they can fill more than one field using one action. The file was prefilled with the help of sighted people, ensuring that the correct data

TABLE 5. TCI (4) testing results.

TCI	Groups	Tasks				Avg
			Completed	Incorrect	Correct	
TCI (4)	Submit the form	Accessing the button	13	0	13	8 s,
		Submitting form	12	1	12	NA

TABLE 6. TCI (5) testing results.

TCI	Groups	Tasks				Avg
			Completed	Incorrect	Correct	
TCI (05)	helpful considerations	Using toolbar	2	6	2	19 s,
		Reading instruction	NA	NA	NA	31 s,
		Expanding time	NA	NA	NA	NA
		Showing data before submitting	3	0	3	15 s,

TABLE 7. Comparison result between AWF and IWF.

Participants	AWF			IWF		
	Speed	Submission state	Accuracy	Speed	Submission state	Accuracy
UID 01	09:40	✓	100.00%	13:23	✓	100.00%
UID 02	10:24	X	92.86%	08:09	X	81.25%
UID 03	10:22	✓	100.00%	07:23	X	81.25%
UID 04	09:23	✓	100.00%	11:17	✓	100.00%
UID 05	08:57	✓	100.00%	07:37	X	86.67%
UID 06	06:54	✓	100.00%	07:02	✓	100.00%
UID 07	10:30	✓	100.00%	07:57	X	81.25%
UID 08	09:22	✓	100.00%	08:21	X	86.67%
UID 09	10:29	✓	100.00%	08:53	X	92.86%
UID 10	09:18	✓	100.00%	07:39	X	86.67%
UID 11	11:02	✓	100.00%	08:23	X	92.86%
UID 12	10:33	✓	100.00%	08:42	X	92.86%
UID 13	10:11	✓	100.00%	09:19	X	86.67%

were entered into the fields. It is worth mentioning that users who used mobile devices took slightly more time to search for the file compared to computer users, possibly because files in some mobile devices are saved in unfamiliar destinations, unlike computers where the user can save the file in a location they know and can easily access. In the second task, “Filling

TABLE 8. Post-interview results.

No	Form element and feature	Mean	RII	Std	Rank
1.	Error messages that could have been read by a screen reader	3.92	98%	0.27	1
2.	Error messages that were easy to understand and fix	3.77	94%	0.42	2
3.	CAPTCHA gave in the web form (repeating the heard word)	3.62	90%	0.62	3
4.	Example of the required format is given for date and phone entry	3.62	90%	0.49	4
5.	Clear and legible field labels from a screen reader	3.62	90%	0.84	5
6.	Field sequencing and design	3.54	88%	0.63	6
7.	Giving enough time to fill out the web form and check to make sure I'm ready to submit it	3.54	88%	0.93	7
8.	Important information and instructions mentioned before filling out the form	3.46	87%	0.75	8
9.	Simple words and short sentences are used in the form	3.38	85%	0.84	9

text fields using voice”, we found that most participants chose the keyboard to fill the data, as the voice entry was an additional option, allowing users to fill the data in the way they preferred. However, the results of the second task, “Filling text fields using voice”, go against our expectations, as we initially thought that most visually impaired people would prefer using a microphone over a keyboard. Only one participant tried to use the microphone as an input method, and that too was for one field only. One possible explanation for this could be that they were unsure about the microphone settings on their browsers or devices. Another reason may be that they felt more confident using the keyboard and listening to the screen reader when reading what has been written letter by letter. In contrast to using a microphone, the keyboard allows users to record a long phrase and convert it into text. However, this text may be affected by background noise or mispronunciations, which aligns with what was mentioned in the study [31]. They found that editing voice recognition errors could be frustrating and that the average time to fix such errors was high. As our test sample consisted of Arabic speakers, the possibility of errors may increase due to the relatively immature state of speech recognition technologies for the Arabic language [32]. Consistent with another study [33], which found that correcting error rates using both the keyboard and speech were higher compared to the keyboard-only condition. Therefore, we allowed participants to use both methods to enter data into the web form. Regarding input time for both methods, it is natural for speech input to be faster than writing using the keyboard. This was demonstrated by the only person who chose speech input,

and it was supported by a study [33], stating that speech input was nearly five times faster than keyboard input. However, the time required to fix the entered text may be twice as long. In summary, entering text by voice may be suitable for general correspondence that does not require high accuracy or for messages circulated in audio format without converting them to text. However, it may not be the most suitable method for web forms that necessitate precise data input. Nonetheless, it can be added as an optional tool and may be more fitting for simple inputs or numerical data. Furthermore, the participants’ failure in “Entering the date using the date picker” might be due to some screen readers not being capable of enabling consistent interaction with dynamic content like a date picker. As a result, entering the date using select boxes was preferred by all participants. The results also show that “Entering the address using API” is a good choice, as it can save time and effort if the user wants to enter the same address where they are using the computer. Additionally, “Entering the address using the selection box” serves as an alternative way if the user wants to change the address that was entered using the API or if their browsers do not allow API address input. The positive outcomes of “Entering phone numbers with the help of country code and hint,” “Entering all required fields,” and “Entering the expected format” without confusion or inquiring about the required format confirms what the study [34] indicates: that unclear identification of the required fields and data format challenges low vision individuals, leading to errors in their data input. Regarding the users’ performance results for error notifications and fixing them, we believe that the preference for client-side validation is due to the fact that with server-side validation, participants may mistakenly believe they have completed the form smoothly without any errors, only to discover the opposite after pressing the submit button. This experience led to disappointment for some users; one participant did not even notice that errors were present and mistakenly thought the process was successful simply by pressing the submit button. However, the feature of hiding the correct fields had a positive effect on the speed of participants’ awareness and correction of errors. In contrast, client-side validation provided immediate feedback either after pressing enter to navigate to the next field, which was the preferred choice for participants as it allowed them to complete writing without distractions, with an error message if necessary. Instantaneous error messages that appeared while writing were not preferred by some participants, as they found them distracting when the message appeared before they finished filling the field. The last preference was for validation after pressing submit and linking the error with the field, as this was somewhat similar to server-side validation but without sending the form. Regarding the CAPTCHA test results, repeating the heard word CAPTCHA was a positive experience for most participants. This was expected, as visually impaired individuals cannot see and solve visual CAPTCHAs, which aligns with findings from a previous study [35]. However, it should be noted that some participants

TABLE 9. Refinements of some guidelines based on experts' evaluation and user testing.

Part	Guideline in the initial framework	Guideline in the final framework	Refined based on
Overall Design Guidelines	Use sans serif fonts with a proper size	Use sans serif fonts, with a proper size. E.g.: 14 pt for Arabic, 12 pt for English	UT
Data Entry Guidelines	Use voice recognition	Use voice recognition as an option	EE
	Use the date picker	Removed	UT
	Add hint and country code	Add country code and hint about the format	EE
Form Validation Guidelines	Client-side validation: 1) Notify the user of errors after pressing enter 2) Notify the users of errors after pressing the submit button then link the error with the field 3) Notify the user of errors instantly	Client-side validation: 1) Notify the user of errors after pressing enter 2) Notify the user of errors instantly after stopping writing at a suitable time 3) Notify the user of errors after pressing the submit button then link the error with the field	UT & EE
Helpful Considerations Guidelines	Provide toolbar	Provide a toolbar for font style and color theme	EE

UT: User testing; EE: Expert evaluation

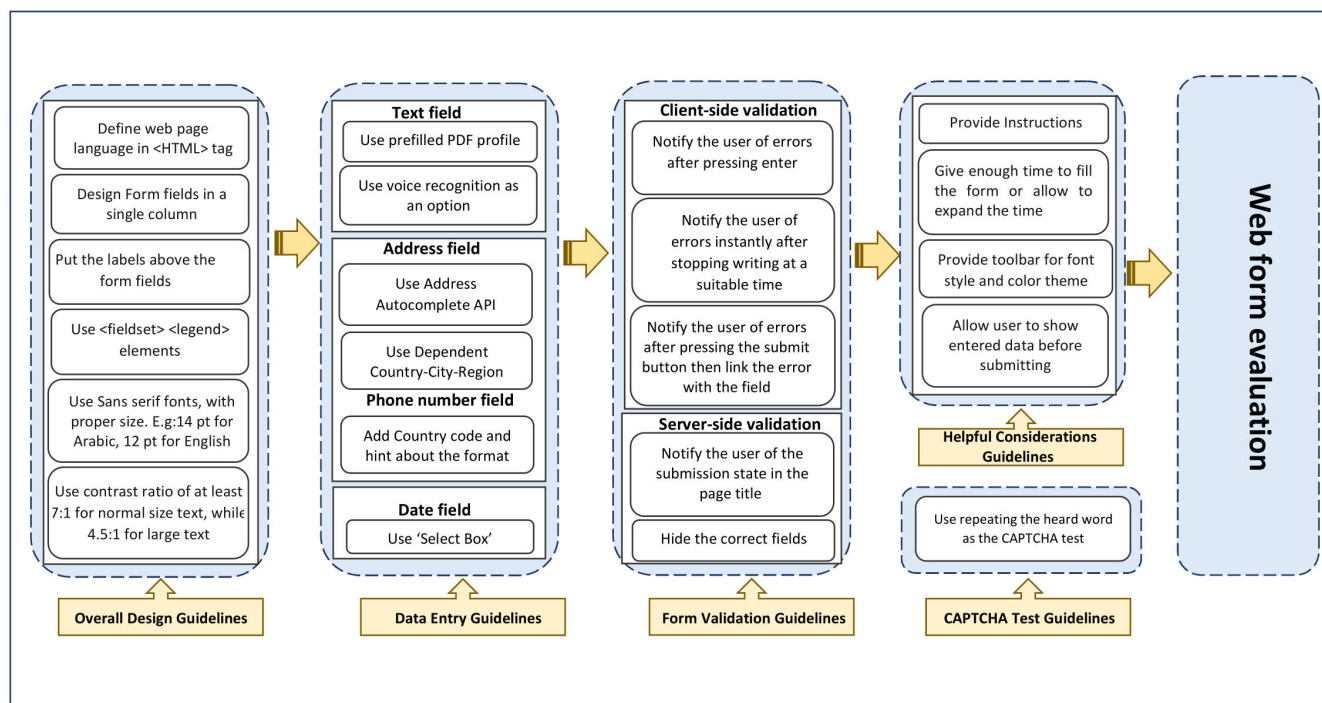


FIGURE 9. The final proposed framework (WAFI).

required more than two attempts to solve the CAPTCHA, and one participant could not repeat the word due to a problem with her microphone. This issue could be addressed by allowing the user to write the heard word or choose it from multiple options. The clear submission button proved helpful in enabling users to submit the form, and the given instructions had a positive impact, as participants were able to fill out the form without needing to inquire about required fields or data format. Unfortunately, we could not analyze their performance in the “Expanding the time” task, as all participants completed the filling process before the need to expand the time. Finally, we think the reason that a few users

only reviewed their entered data before submitting it is that they knew that this form was only for testing purposes only. In actual web forms, users may choose to review their entered data before submitting it.

C. COMPARISON BETWEEN AWF AND IWF RESULTS ANALYSIS

The comparison results of AWF and IWF demonstrate that the majority of participants were able to submit AWF successfully, while only three participants could submit IWF. The completion time for IWF was shorter than that for AWF,

TABLE 10. Test cases and tasks.

Accessibility testing tasks

Test case Id			Accessed	Used	Number of Clicks		Number of attempts	Completed	Entered correct	Time (sec)
					Correct	Wrong				
TCI (1)	Data entry	Filling text fields using prefilled pdf								
		Filling text fields using voice recognition								
		Entering the date using the date picker								
		Entering the date using selecting box								
		Entering the address using the API address								
		Entering the address using the selection box								
		Entering the phone number with help of a hint and country code								
TCI (02)	Form Validation	Entering all required fields								
		Entering the expected format.								
	Error notification	Accessing and understanding instant validation (client side)								
		Accessing and understanding validation after pressing enter (client side)								
		Accessing and understanding validation after pressing submit and linking the error with the field (client side)								
		Accessing and understanding validation after submitting the form (server side)								
	Error recovery	Fixing errors with instant validation. (client side)								
		Fixing errors that were validated after pressing enter (client side)								
		Fixing errors that were validated after pressing submit and linking the error with the field (client side)								
		Fixing errors that were validated after submitting the form (server side)								
TCI (03)	Pass CAPTCHA test	Understanding the idea of the test								
		Accessing the button CAPTCHA								
		Passing the test								
TCI (04)	Submit the form	Accessing the button								
		Submitting form								
TCI (05)	Helpful considerations	Using toolbar								
		Reading instruction								
		Expanding time								
		Showing Data before submitting								

primarily because the participants did not fully fill out the form, indicating a tradeoff between speed and accuracy.

VIII. PRODUCING THE FINAL FRAMEWORK

After the initial framework was produced and evaluated by the experts and visually impaired users, certain components were refined, eliminated, and/or created based on the

validation results. Table 9 provides details of the changes made based on experts' evaluation and user testing.

A. WAFI FRAMEWORK EVALUATION TOOL

In addition to the refinements made in Table 9, the sixth part was added, as recommended by all experts, which consists of an evaluation tool comprising a set of 24 verification items

TABLE 11. Wafi evaluation tool.

WAFI Checklist						
	Item	Guideline	Occurrence	Compliant	Not-Compliant	Accessibility Score
Overall Design Guidelines	1	Page language	A web page language and direction is defined in <HTML> tag			
	2	Form field	The form fields are designed in a single column			
	3	Lable	The <lable> tag is used for all fields			
		Lable	All labels are located above the form fields			
	5	Form field	<fieldset> <legend> elements are used for the related fields			
	6	Font	Sans serif font is used as font family			
		Font	A proper font size is used			
	8	Color	A contrast ratio of at least 7:1 is used for normal size text, and 4.5:1 is used for large text			
Data Entry Guidelines	9	Form field	The user can fill fields using prefilled PDF profile			
	10	Form field	The user has the option to fill fields using voice recognition			
	11	Adrees field	The address field is filled using autocomplete API			
	12	Adrees field	Dependent Country-City-Region is used fill address field			
	11	Phone number field	Country code and hint about the required format are given in the phone number field			
	13	Date Field	'Select Box' is used for the date field			

Form Validation Guidelines	14	Field notification	The user is notified of an error using: After pressing enter OR Instantly after stopping writing at a suitable time OR After pressing the submit button then link the error with the field			
	15					
	16					
	17	Form submission	The user is notified of the submission state through the page title			
18	Wrong fields	The correct fields are hidden while server validation				
Helpful Considerations	19	Form instruction	Instructions about the web form are given before filling			
	20	Form time	Enough time to fill out the form is given			
		Form time	The user can expand the time if he needs			
	22	Toolbar	Toolbar for font style and color theme is provided			
23	Form inputs	The users can show entered data before submitting				
CAPTCHA test Guidelines	24	CAPTCHA test	The idea of repeating the heard word is used as a CAPTCHA test			
						Average

organized in the form of a checklist. The complete tool can be found in (Appendix B). It was assumed that all items have the same weight in terms of accessibility, based on the research of [36]. Each item is evaluated to determine whether it is compliant with the guidelines for each occurrence of the item. Based on the answers, the accessibility score (AccSc) will be calculated as follows:

Accessibility Score (AccSc) = (Number of times the item is compliant with the guideline) / (Number of times the item occurs in the web form)

Then, the average of all the scores will be computed and converted to a percentage. A 100% score implies that the web form is fully accessible, whereas a 0% score implies that the web form is inaccessible. A higher score implies that fewer accessibility problems exist.

B. THE FINAL PROPOSED FRAMEWORK

Finally, the final version of the framework is produced and presented again to the experts as the final step of validation. The feedback provided by the experts was generally positive. Fig. 9 shows the final form of WAFI.

IX. CONCLUSION

This study aimed to develop a Web form Accessibility Framework for the Visually Impaired (WAFI), which, when

used correctly by web form developers, will enhance access to web forms for visually impaired users. It enables them to fill out web forms faster, easier, and with fewer errors through its comprehensive components that cover all the necessary aspects to build an accessible web form. The framework comprises six components. The first component addresses the overall design of the web form. The second component focuses on data entry methods to facilitate filling out different types of web form fields for visually impaired users. The third component provides methods to notify users of errors during form filling and enables easy error correction. The fourth component proposes an alternative method to help visually impaired users solve CAPTCHA tests. The fifth component includes additional enhancements to improve web form accessibility. Lastly, the sixth component is an evaluation tool that developers can use to evaluate the accessibility of their web forms. The WAFI framework underwent validation by experts and testing by a sample of visually impaired users. The results demonstrate that web forms designed using the WAFI framework improved speed, accuracy, and ease of use for visually impaired users.

A. IMPLICATIONS OF THE STUDY

This study may contribute to solving the suffering of many people with different types of vision disabilities while

filling out web forms that have become a part of almost every existing website or application. In addition, it can be universally used as it is applicable to any language. The use of this framework to build an accessible web form can increase the confidence of visually impaired users with vision disabilities. Also, this study can help web form developers build accessible web forms enabling their clients to use them more efficiently.

B. CHALLENGES AND LIMITATIONS

This study was not completed without challenges. In fact, several challenges were faced. First, we encountered several difficulties in the testing phase, where the sample of visually impaired users was not easy to collect, and most of them were not available to conduct the test offline. As a result, our sample size was very small, and the test was conducted via a live-streaming meeting. Additionally, since we had never dealt with this segment of users before and were not familiar with ATs, it took a long time to establish the test with some participants. During expert validation, the developers did not use WAFI to build a web form; instead, the validation was conducted through a survey. Finally, this study was limited to visual impairment only. We aim to address these issues in the future.

C. FUTURE WORK

In the future, we plan to expand the participant sample size to gather more comprehensive data. Additionally, we aim to develop similar frameworks to cover other disabilities, like hearing impairments. As part of our framework, we will explore the implementation of Natural Language Processing to improve speech-to-text functionality. Furthermore, we intend to propose enhanced solutions for CAPTCHA tests, making them more accessible and user-friendly for visually impaired individuals. Finally, we will enhance the evaluation tool and transition it into an automated system, moving away from a manual checklist approach.

In conclusion, despite its size, we believe that our work can significantly contribute to the improvement of web form development and enhance the experience of the growing number of people with vision disabilities. By addressing the challenges they face while filling out web forms, we hope to make web accessibility more inclusive and seamless for all users.

APPENDIX A

See Table 10.

APPENDIX B

See Table 11.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviews for their invaluable comments.

REFERENCES

- [1] C.-Y. Tsai, L.-C. Woung, P. Chou, C.-S. Yang, M.-M. Sheu, J.-R. Wu, T.-L. Chuang, and T.-H. Tung, "The current status of visual disability in the elderly population of Taiwan," *Jpn. J. Ophthalmol.*, vol. 49, no. 2, pp. 166–172, Mar. 2005.
- [2] S. Vishnu Prasad, M. Kumar, M. Ramakrishnan, and D. Ravikumar, "Report on oral health status and treatment needs of 5–15 years old children with sensory deficits in Chennai, India," *Special Care Dentistry*, vol. 38, no. 1, pp. 58–59, Jan. 2018.
- [3] A. Aizpurua, S. Harper, and M. Vigo, "Exploring the relationship between web accessibility and user experience," *Int. J. Hum.-Comput. Stud.*, vol. 91, pp. 13–23, Jul. 2016.
- [4] U. Uckun, A. S. Aydin, V. Ashok, and I. Ramakrishnan, "Breaking the accessibility barrier in non-visual interaction with PDF forms," *Proc. ACM Hum.-Comput. Interact.*, vol. 4, pp. 1–16, Jun. 2020.
- [5] M. Ferati, B. Raufi, A. Kurti, and B. Vogel, "Accessibility requirements for blind and visually impaired in a regional context: An exploratory study," in *Proc. IEEE 2nd Int. Workshop Usability Accessibility Focused Requirements Eng. (UsARE)*, Aug. 2014, pp. 13–16.
- [6] B. Raufi, M. Ferati, X. Zenuni, J. Ajdari, and F. Ismaili, "Methods and techniques of adaptive web accessibility for the blind and visually impaired," *Proc.-Social Behav. Sci.*, vol. 195, pp. 1999–2007, Jul. 2015.
- [7] H. Petrie, A. Savva, and C. Power, "Towards a unified definition of web accessibility," in *Proc. 12th Int. Web All Conf.*, May 2015, pp. 1–13.
- [8] A. Alsaedi, "Comparing web accessibility evaluation tools and evaluating the accessibility of webpages: Proposed frameworks," *Information*, vol. 11, no. 1, p. 40, Jan. 2020.
- [9] J. M. Fernández-Batanero, M. Montenegro-Rueda, J. Fernández-Cerero, and I. García-Martínez, "Assistive technology for the inclusion of students with disabilities: A systematic review," *Educ. Technol. Res. Develop.*, vol. 70, no. 5, pp. 1911–1930, 2022.
- [10] S. Brinsmead, "Towards an accessible iPad for children and young people with cerebral palsy," *J. Enabling Technol.*, vol. 13, no. 4, pp. 228–239, Dec. 2019.
- [11] S. Paul and S. Das, "Accessibility and usability analysis of Indian e-government websites," *Universal Access Inf. Soc.*, vol. 19, no. 4, pp. 949–957, Nov. 2020.
- [12] S. Alim, "Web accessibility of the top research-intensive universities in the UK," *SAGE Open*, vol. 11, no. 4, 2021, Art. no. 21582440211056614.
- [13] P. Acosta-Vargas, M. González, and S. Luján-Mora, "Dataset for evaluating the accessibility of the websites of selected Latin American universities," *Data Brief*, vol. 28, Feb. 2020, Art. no. 105013.
- [14] S. Usharani, P. M. Bala, and R. Balamurugan, "Voice based form filling system for visually challenged people," in *Proc. Int. Conf. Syst., Comput., Autom. Neww. (ICSCAN)*, Jul. 2020, pp. 1–5.
- [15] S. M. Billah, V. Ashok, and I. Ramakrishnan, "Write-it-Yourself with the aid of smartwatches: A wizard-of-oz experiment with blind people," in *Proc. 23rd Int. Conf. Intell. User Interfaces*, Mar. 2018, pp. 427–431.
- [16] S. Feiz, S. M. Billah, V. Ashok, R. Shilkrot, and I. Ramakrishnan, "Towards enabling blind people to independently write on printed forms," in *Proc. CHI Conf. Human Factors Comput. Syst.*, May 2019, pp. 1–12.
- [17] E. Peng, P. Peursum, L. Li, and S. Venkatesh, "Portable form filling assistant for the visually impaired," *Proc. HCI*, 2010, pp. 269–277.
- [18] S. Morel and N. Genty, "Automatically uploading user profile information," U.S. Patent 14 704 786, Sep. 3, 2015.
- [19] K. K. Cheung, E. Mit, and C. E. Tan, "Framework for new generation web form and form filling for blind user," in *Proc. Int. Conf. Comput. Appl. Ind. Electron.*, Dec. 2010, pp. 276–281.
- [20] R. Alsalamen and G. Shahin, "Screen size effect on usability of Arabic forms for smartphones," in *Proc. Int. Conf. New Trends Comput. Sci. (ICTCS)*, Oct. 2017, pp. 264–269.
- [21] M. Pomurugesu, M. Rajashobika, K. Usha, and C. S. Kumar, "Automatic form filler," *Int. Res. J. Eng. Technol.*, vol. 6, no. 1, 2019.
- [22] K. Karuppasamy, P. Dharshini, N. Lavanya, and S. Santharoban, "Touch free challan form filling system using face detection," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 8, no. 9, pp. 485–488, 2020.
- [23] W3C. (Jul. 19, 2022). *Labeling Controls*. [Online]. Available: <https://www.w3.org/WAI/tutorials/forms/labels/>
- [24] W3C. (Aug. 2, 2022). *Grouping Controls*. [Online]. Available: <https://www.w3.org/WAI/tutorials/forms/grouping/>
- [25] W3C. (Aug. 17, 2022). *Contrast(Enhanced): Understanding SC 1.4.6*. [Online]. Available: <https://www.w3.org/TR/2016/NOTE-UNDERSTANDING-WCAG20-20161007/visual-audio-contrast7.html>

- [26] J. Jovanovic. (Jul. 22, 2022). *Web Form Validation: Best Practices and Tutorials*. [Online]. Available: <https://www.smashingmagazine.com/2009/07/web/form/validation/best/practices/and/tutorials/>
- [27] UX Movement. (May 2019). *The Best Place for Error Messages on Forms*. [Online]. Available: <https://uxmovement.medium.com/the-best-place-for-error-messages-on-forms-6e9e84740be8>
- [28] J. Bargas-Avila and G. Oberholzer, "Online form validation: Don't show errors right away," in *Human-Computer Interaction Interact*, vol. 3. Webernstrasse, Switzerland, 2003, pp. 848–851.
- [29] M. Penzo. (Jul. 2006). *Label Placement in Forms*. [Online]. Available: <https://www.uxmatters.com/mt/archives/2006/07/label-placement-in-forms.php>
- [30] D. Quiñones, C. Rusu, and V. Rusu, "A methodology to develop usability/user experience heuristics," *Comput. Standards Interfaces*, vol. 59, pp. 109–129, Aug. 2018.
- [31] S. Azenkot and N. B. Lee, "Exploring the use of speech input by blind people on mobile devices," in *Proc. 15th Int. ACM SIGACCESS Conf. Comput. Accessibility*, Oct. 2013, pp. 1–8.
- [32] W. Algihab, N. Alawwad, A. Aldawish, and S. AlHumoud, "Arabic speech recognition with deep learning: A review," in *Social Computing and Social Media, Design, Human Behavior and Analytics*. Orlando, FL, USA: Springer, 2019, pp. 15–31.
- [33] B. Bhikne, A. Joshi, M. Joshi, C. Jadhav, and P. Sakhardande, "Faster and less error-prone: Supplementing an accessible keyboard with speech input," in *Human-Computer Interaction—INTERACT*. Paphos, Cyprus: Springer, 2019, pp. 288–304.
- [34] J. Lazar, A. Olalere, and B. Wentz, "Investigating the accessibility and usability of job application web sites for blind users," *J. Usability Stud.*, vol. 7, no. 2, pp. 68–87, 2012.
- [35] M. Alnfai and F. Alassery, "Tapcaptcha: Non-visual captcha on touchscreens for visually impaired people," *J. Multimodal User Interfaces*, vol. 16, no. 4, pp. 385–398, 2022.
- [36] A. T. Blog. (Apr. 2, 2022). *Accessibility Score Methodology Deep Dive*. [Online]. Available: <https://audioeye.medium.com/accessibility-score-methodology-deep-dive-4e405e0f923c>

WAFAA ABU SHAMLAH HAKAMI received the bachelor's degree from the Department of Computer Science, Jazan University. She is currently pursuing the master's degree with the Department of Computer Science, FCIT, King Abdulaziz University, Jeddah. She was a Teaching Assistant with Jazan University. Her research interests include web development and human-computer interaction.

ARWA YOUSUF AL-AAMA received the B.Sc. degree in computer science from King Abdulaziz University, Saudi Arabia, in 1992, and the M.Sc. and D.Sc. degrees in computer science from George Washington University, Washington, DC, USA, in 1996 and 2003, respectively. She was the Vice Mayor of IT Affairs, Jeddah, Saudi Arabia, for over ten years and held several academic posts. She is currently a Professor of computer science with King Abdulaziz University. She is also a Digitization Consultant and a Senior Consultant with the Saudi Digital Economy Center. Her current research interests include accessibility, cybersecurity, and machine learning.

• • •