BellChat: An Inclusive Web Application for Messaging

Gleiston Guerrero-Ulloa [1,2][0000-0001-5990-2357] and Víctor Romero-Castro [1][0000-0003-4920-1009] and Janer Torrales-Peralta [1][0000-0001-6587-9587] and Tyrone Tocta-Bonilla [1][0000-0002-5812-0617] and Orlando Erazo [1][0000-0001-5642-9920]

[1] Universidad Técnica Estatal de Quevedo, Quevedo, 120301 Los Ríos, Ecuador
{gguerrero,victor.romero2016,janer.torrales2016,tyrone.tocta2016, oerazo\}@uteq.edu.ec}
[2] University of Granada, Granada, 18071 Granada, Spain
gleiston@correo.ugr.es

Abstract. The goal of the Web is universal access, but minority groups of web users, such as people with disabilities and elderly people, are limited in their ability to communicate through chat applications. These types of applications include social networking, which are especially used in communication among their users. However, social networks such as Facebook (Messenger), Twitter and WhatsApp are not adapted for these groups, and therefore, they do not fully meet the needs for which they were created. Moreover, for people with disabilities to use web-based messaging applications, they are required to use third-party tools, such as screen readers. For these reasons, this paper presents BellChat, a responsive web application for communication between everybody, adapted for people with disabilities, especially for those ones with visual or hearing disabilities. BellChat is the only application adapted to convert text messages into speech, or speech messages into text, depending on the disability of the user who is connected. BellChat was developed following the Scrum development framework with adaptations to very small groups, where the roles were modified to be according to the project team. In addition, developers worked remotely, and meetings were held in person twice a week. The evaluation results confirm the acceptance of the application by users.

Keywords: Visual impairment, Hearing impairment, web accessibility, Text-to-speech, Speech-to-text.

1 Introducción

Although inclusivity goes beyond accessibility, its basis is accessibility. An inclusive web application must first be an accessible web application. This means that an application should be thought of from the design to the building and deployment as an application that is accessible to all people without exception [1]. To achieve this goal, the Web Accessibility Initiative (WAI) of the World Wide Web Consortium (W3C) has been established to raise awareness of universal accessibility [2]. The WAI provides developers with guidelines that can help make web pages widely accessible [3]. Considering that websites are designed to serve different purposes such as infor-

mation, entertainment, advertising, to name a few, and that to achieve this goal they present a wide range of information to meet user needs, they must be accessible to everybody. However, most of these websites do not comply with the web accessibility standards designed by the W3C, making it difficult for part of the population to access to web content [4].

The Web was created with the purpose of giving access to all people without exception. Therefore, the Web is the place where everybody should feel their full right to equality, regardless of their conditions or disabilities [5]. However, what happen with people with disabilities? In general, disability encompasses impairments, limitations, and other factors that affect certain bodily functions or prevent from performing common activities, among other problems. There are several types of disabilities, such as physical, sensory (visual, hearing), intellectual and multiple disabilities [6]. It is very common to find websites that are not adapted for people with disabilities; for example, they do not have their own content reader, or they do not have a voice command interface. Likewise, not all websites take care of people with hearing disabilities, i.e., the voice files generally do not have the transcription of the content into text to allow for people with hearing disabilities understanding the content [7].

Given these disadvantages, people with visual or hearing impairments face the difficult task of using communication resources. Visually impaired people lessen this difficulty by using screen readers provided by third parties [8], while tools to reduce the difficulty faced by the hearing-impaired people are poorly suited for instant transcription of voice messages by downloading the message and uploading it to the applications for transcription. Some of these tools and the most popular ones can be found at González [9]. Two authors of this work share with hearing impaired people, who have shared with them that. When they receive voice messages (by mistake or due to ignorance of the senders), they must ask for help from third parties to know the content. This situation is also affirmed by Pereira [10]. In addition, these two groups of people with disabilities are joined by older adults who, due to their condition, are diminished in their abilities, and therefore do not enjoy all the benefits offered by the evolution of the functionalities of web applications for communication [6].

Communication between human beings is what makes people live in society. Currently around 3.4 billion users actively use social media platforms daily for an average of 2.5 hours. Social media has become an effective means of communication between people, including for companies with their customers. However, the lack of inclusive platforms may make people with disabilities feel excluded. As a contribution to achieve the inclusion of this group of people, this paper presents BellChat, a responsive web application that ensures accessibility to the greatest number of people, regardless of their conditions (visual, hearing, physical). Its virtual assistant ensures interaction between the user and the application. It enables the actions allowed on each page to be performed by voice commands; it also allows any user to send messages in any format to people with or without disabilities. In addition, to reach as many people as possible, keyboard commands have been implemented for those who have problems with the mouse and for people who may have problems with the pronunciation of voice commands [11, 12].

BellChat implements the W3C standards and complies with the success criteria to make web pages accessible to as many people as possible [1]. Based on these standards, the communication between people with disabilities and without disabilities is facilitated and communication accessibility problems are reduced. Other communication websites comply with certain W3C standards, but they do not achieve the objective of maximizing the number of people because they do not provide ease of use for people with disabilities. Thus, this document details the development and evaluation of BellChat.

2 Related Work

In digital libraries, such as WOS (Web of Science), ACM, IEEE, Scopus, and Elsevier, few papers have been found on the development of accessible web applications. However, there are some articles that deal with the evaluation of web accessibility in public interest websites.

Among the works that present the development of accessible web applications, we can find Web-ALAP, a web-based application for writing mathematical documents in LaTeX developed by Arooj et al. [13]. Web-ALAP supports users with low vision through assistive functions and the manifestation of error indications by voice. To support users with the same disability, Lee et al [14], introduce TableView, which focuses on solving the problem of users with low vision who have difficulty using the on-screen magnifier. TableView extracts content and information from the page and presents it in a more compact form to make the most of the expanded space. Despite their contribution, these works are focused on a specific problem: people with low vision.

Among the works that have been concerned with people with profound deafness problems, Lyall et al. [15], developed a smartphone application that recognised six sentences dictated by doctors for post-operative patients. These messages, delivered by voice by doctors, were converted to text to be read by the patients. The results shown in this work., with a clear extension, can be used for telephone conversations and even in face-to-face conversations between people without disabilities and with hearing loss.

Along the same lines, we find the work of Shadiev et al. [16], in which they apply speech-to-text technology in a virtual classroom. Their results were a motivation to propose BellChat. Although Shadiev et al. [16], apply it directly, BellChat does it in a deferred way, in text messages and speech-to-text over voice files. Moreover, a mobile application that enables communication between hearing impaired and non-disabled people (or those who do not understand sign language) was presented by Ali et al. [17] which, via Bluetooth, allows for exchanging information over short distances.

Alsaif et al. [18], present a very useful application for people with speech problems. It predicts through statistics the words that could be pronounced in a conversation. In addition to converting text to speech, the system provides different categories of frequently used phrases that are labelled with a representative image for ease of

use. The user can also add images to the system and record the human voice or use the automatic text-to-speech synthesiser. The limitation of this application is that it is smartphone-only and converts speech to text and text-to-speech using only Arabic language.

On the other hand, one of the activities that help make websites accessible is accessibility evaluation. Academic websites and government websites have been targeted by evaluators. For example, Sala et al. [19], conducted a user study to evaluate five Spanish governmental websites. This study involved five people with low vision out of a total of ten participants. The authors concluded that all sites presented barriers that claimed accessibility for people with disabilities.

Among the articles on the evaluation of educational websites, we find the work of Al Ghurair et al. [3]. The study consisted of evaluating a sample of websites of selected universities in Jordan, the Arab region and England. It determined that all websites have drawbacks with compliance with accessibility guidelines. A year later, Whabi et al. [20] exposed the shortcomings in most Jordanian universities. On the other hand, Sala et al. [19] present a user study of five public Spanish e-services with ten participants, five of them with low vision. In another paper, Lorca et al. [21] evaluated the websites of universities in different countries, such as Germanic, Nordic, Anglo-Saxon and Latin countries, and they concluded that Anglo-Saxon countries pay more attention to the accessibility of web pages. From the results of these investigations and the conclusions of their authors, it can be affirmed that there is no fully accessible site for people with hearing or vision disabilities.

All the papers read before and while BellChat was being designed, present solutions that help people with specific disabilities. This is the reason why the authors of this paper present BellChat as an inclusive application as a means of communication between the greatest number of people, supporting the right of equality for which the web was created.

3 Proposal

Inclusivity is an issue addressed from the design of web-based tools to new smart technologies for improving lifestyles of the people [22]. Also, in a web development project, the information manager must be aware of the problem of accessibility and the strategies and techniques available to overcome it. This fact means prioritising content over aesthetic aspects, if necessary [23]. Moreover, web accessibility is postulated as an important activity for the information professional and even as a possible professional niche. Developing accessible websites is becoming an increasingly important goal for people daily lives, as it results in services that can be perceived, understood, and used by all users, including those ones with disabilities. This goal is becoming a concern for governments. Several countries are making various efforts to promote best practices in accessibility [4]. It can be said that the Web is one of the ways to facilitate communication between users and access to services. In this way, it alleviates the disadvantages people face when accessing services in the real world, especially people with visual or hearing impairments [23].

Given this landscape, BellChat is a smart web application capable of converting text to speech and speech to text depending on the disability registered in the user's profile. Its responsiveness gives it the feature of being easily accessed from a smartphone. This quality allows for reaching a much larger group of users, thus integrating into society those users who have been relegated by their situation. Moreover, the text-to-speech and the speech-to-text conversion help not only people with any of the disabilities; they also help people with full capacities to perform other tasks while learning about the messages they receive or while sending messages to their contacts [24]. In addition, BellChat has an innovative design to ensure its goal of being accessible to everyone [25].

BellChat eliminates the most impactful problems for people with hearing impairment such as those mentioned by Pascual et al. [26]. In addition, it includes visually impaired people, allowing them to interact seamlessly with other users regardless of their abilities [27]. Also, BellChat can adapt to any device (smart phones, laptops, desktop computers) with Internet access in a friendly way.

3.1 Requirements

In the development of BellChat, functional requirements have been considered to try to include as many people as possible and non-functional requirements to provide all users with the security necessary in today's web applications.

Functional Requirements. In the development of BellChat, the following requirements were considered:

- *User identification:* To become a user of the application, a person must register as it. Among the information that the user must register is whether s/he has a disability and which one. BellChat will allow the user to enter his/her username and password, by voice or typing on a keyboard.
- *Self-programmable system:* Once the user has logged into the application, the respective functions will be self-programmed in its configuration, for example: the inbox will be prepared to present the user with the textual content of the messages (speech-to-text), or play the audio content of the messages (text-to-speech), the user's contact search format, voice management control, among others, or simply maintain appropriate options according to the user's profile.

Requirements for visually impaired persons:

- *Voice assistant:* The voice assistant must allow interacting with a browser, being always ready to solve user requests.
- *For each page (screens)*, a set of commands must be defined to help the user to perform the respective actions (in this case, the executor of the voice commands starts its work when Lili is invoked). Please note that the first paragraph of a section or subsection is not indented. The first paragraphs that follows a table, figure, equation etc. does not have an indent, either.

Requirements for hearing impaired persons:

- *Voice message to text converter:* If a hearing-impaired user receives an audio message, the application identifies the message format and converts it to text, letting the recipient read the message so that he/she is aware of its content.
- *Writing messages:* The user will be able to write in text format the message he/she wants to send no matter to whom (recipient capabilities) he/she is going to send it.

Requirements for people with motor disabilities:

- *Voice assistant (Lili):* The assistant should allow users to interact with a browser, always waiting for user requests. Each time a page is loaded in the browser, the respective commands can be invoked by voice to perform the permitted actions.
- *Sending messages:* The user may decide to send a message in text or in audio format, according to his capabilities, without considering to whom (recipient capabilities) he is going to send the message.

Non-Functional Requirements. BellChat eliminates the most impactful problems for people with hearing impairment such as those mentioned by Pascual et al. [26]. In addition, it includes visually impaired people, allowing them to interact seamlessly with other users regardless of their abilities [27]. Also, BellChat can adapt to any device (smart phones, laptops, desktop computers) with Internet access in a friendly way.

4 Methodology

BellChat was developed following the Scrum development framework, which is considered the most widely used in agile software development. The first task after defining the objectives of this project was the assignment of roles [28]. The first author played the role of product owner, the third author played the role of scrum master, and the second and fourth authors played the role of developers (development team). As the team was very small, the roles were expanded and interchanged, with the product owner also playing the role of scrum master who also was part of the development team.

Another adjustment made to Scrum was the frequency of face-to-face meetings. Scrum specifies that they should be daily, the team of this project considered that the meetings should be less frequent, always considering the needs of the development team to responsibly fulfil the assigned sprint in the estimated time. In addition, the sprints had a maximum duration of two weeks while Scrum estimates that two weeks is the minimum time they should take (two to eight weeks) [28–30].

The advantage of working with small teams is the level of control in both compliance and attention to prioritization of activities by the developers, and attention to the developers by the product owner. Also, in small teams, the distances between team members can be short, improving communication [31]. The daily dedication for the

development of BellChat was on average 2 hours per day for each team member. In addition, developers worked remotely, and meetings were held in person twice a week. The list of the most important Sprints (Sprint Backlog) that had to be executed to achieve the successful development of BellChat is shown in the **¡Error! No se encuentra el origen de la referencia.**. The time spent on the project by the product owner is considered only when the activities are to be executed by the project team.

Table 1. Sprint Backlog of the BellChat development.

Description of the Sprint	Duration (hours)	Implemented by	Status*	Week	1		… 9
				Days	1	2	… 44
				P.H.+	326	318	… 8
Gathering application requirements	16	Project team					
Client/server application architecture	16	Project team					
Interface design specifications and definition of development standards	16	Project team					
Creation of the layouts (templates) for the interfaces of the application	32	Development team					
Register user	4	Scrum master					
Login and control of the session	8	Scrum master					
Sending and receiving messages via WebSocket	40	Scrum master					
Function to automatically configure the system depending on the registered disability	60	Development team					
Text-to-speech and speech-to-text message conversion	20	Scrum master					
Commands for contact search	12	Development team					
Virtual assistant	10	Development team					
Chat data persistence	10	Scrum master					
Testing of the application by the project team	16	Project team					
Deployment of the application on a test server	6	Development team and Scrum master					
Application testing with end users	32	Project team					

* The status of the Sprint can be: Not started, in progress, completed.

+ Pending hours.

5 Results

5.1 Requirements elicitation and analysis

The requirements were taken as the needs of real users (who are related to and friends with some of the authors of this paper) with their own visual and hearing impairments (al Ghurair et al., 2021). In addition, we analysed messaging applications currently in use, despite they have poor accessibility and do not comply with W3C standards. Some of the use cases as coarse-grained requirements for BellChat are shown in the use case diagram in the **Fig. 1** y **Fig. 2**. The security-related use case diagram is shown in **Fig. 1** and the BellChat functional requirements use case diagram is shown in **Fig. 2**.

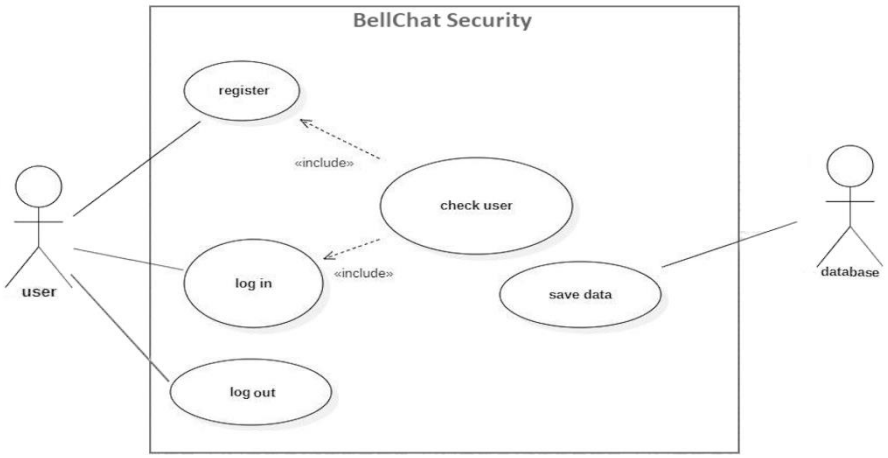

Fig. 1. Diagram of security-related use cases.

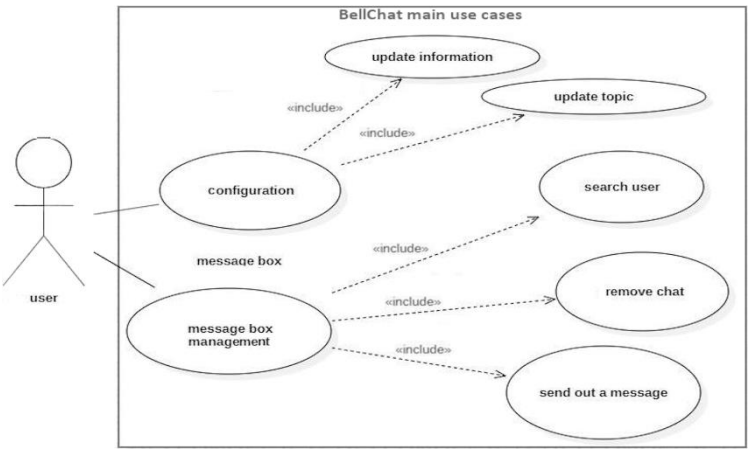

Fig. 2. Diagram of BellChat functional requirements use cases.

User stories were used as a tool for eliciting the system requirements. **Table 2**, **Table 3** y **Table 4** show user stories among those that were implemented.

Table 2. User Story: Log in to BellChat.

History number:	UH06
Story name: Login and control of the session.	*Type of activity:* New
Priority in business logic: High	*Technical Priority:* Medium
Estimated time: 8 hours	*Iteration assigned:* 1
Person responsible: Scrum master	
Narrative: As a user, I want to log in to the application.	

Table 3. User Story: Conversion of audio messages to text.

History number:	UH09
Story name: Converting speech-to-text messages.	*Type of activity:* New
Priority in business logic: High	*Technical Priority:* High
Estimated time: 10 hours	*Iteration assigned:* 1
Person responsible: Scrum master	
Narrative: As a user, I want a message received in audio format to be displayed in text format automatically.	

Table 4. User Story: Interaction through voice commands.

History number:	UH11
Story name: Virtual assistant	*Type of activity:* New
Priority in business logic: High	*Technical Priority:* Medium
Estimated time: **1**0 hours	*Iteration assigned:* 1
Person responsible: Development team	
Narrative: As a visually impaired person, I wish to interact with the system by voice commands and receive responses to my requests from the application.	

User stories are provided by the user (client) from the point of view of his interest in using the system, while the use cases were used to focus more on the analysis of the actions that needed to be executed to achieve the user story objective. User stories complement use cases very well. While user stories denote the importance from the point of view of how the user will use the application, use cases specify the sequence of interaction between actors (feeding or receiving information from the application) and the application [32–35].

5.2 Bell Chat architecture

BellChat has been developed following a 3-tier architecture: Client, Application Server, and Database Server. The **Fig. 3** shows the BellChat architecture from the deployment point of view.

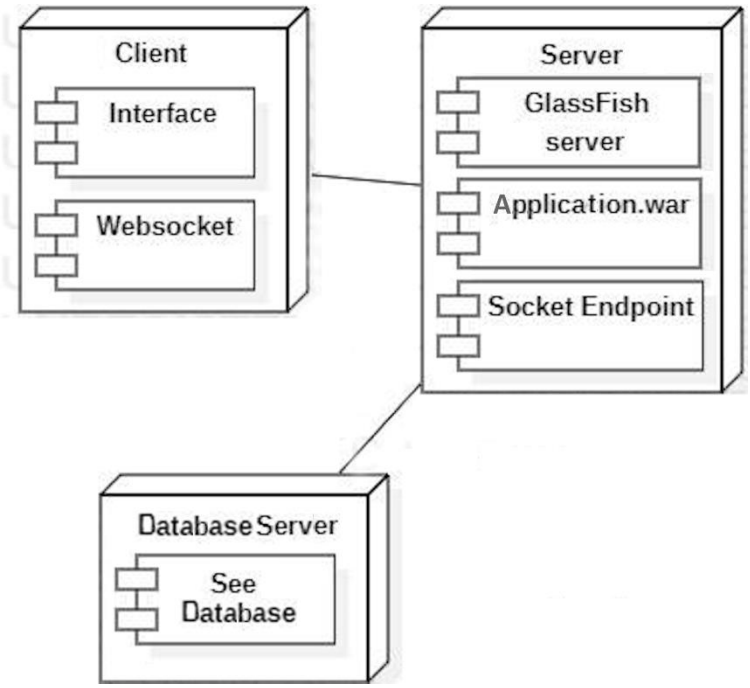

Fig. 3. BellChat architecture.

Client layer. The client-side layer is given by the view or interaction with users. Users can interact from any device that has internet access and a browser (the browser preferred by them). Browsers must be enabled to execute JavaScript and CSS. The browsers that have been tested and have performed very well are Chrome, Opera Gx, Microsoft Edge and Brave (current versions as of October 2022).

BellChat client is a responsive web application, for which CSS was used. Responsiveness gives the application the ability to adapt to any device regardless of screen size. For voice interaction between a user and BellChat, the SpeechRecognition Application Programming Interface (API) was used. In addition, it employs WebSockets for the exchange of messages between users. WebSockets allow the client in client/server applications to receive unsolicited information, which is not normally the case in a web application [36, 37]

The BellChat login interface is shown in **Fig. 4** A user can log in with his own account of the application, for which the user had to register beforehand, or with a third-party account (Facebook or Google).

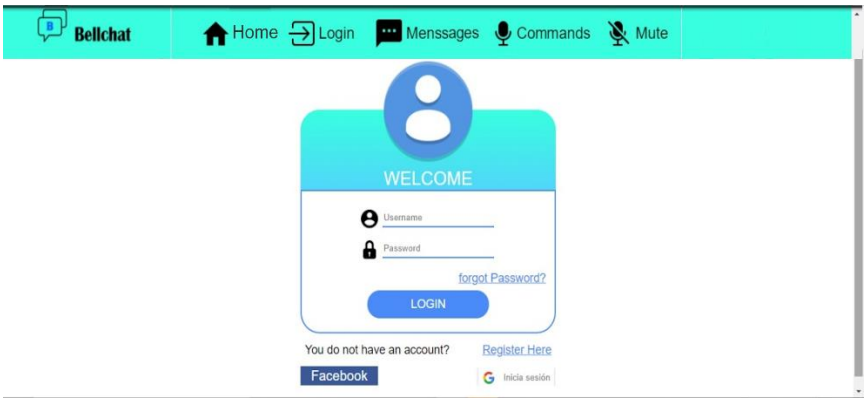

Fig. 4. Screenshot for logging in to BellChat.

Fig. 5 shows the screenshot of the chat screen. The logged in user is a hearing-impaired person. The user has sent a text message to his contact *johannapea*. Likewise, from this user he has received an audio message as a reply. BellChat has automatically converted the audio content to text so that the user is aware of its content. BellChat shows both the original message (audio) and the text of its content.

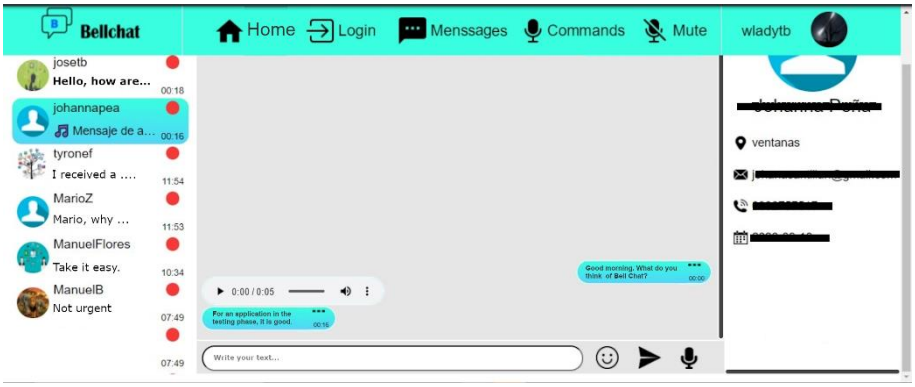

Fig. 5. Screenshot of the message receiving and sending tray in BellChat.

Server layer. In web applications it is usual to have at least a 3-tier architecture, which is the case of BellChat. The second layer, i.e., the layer with which a client interacts, is the application server (also the web server) where BellChat is hosted. The structure of the classes that needed to be implemented in BellChat is shown in the diagram of the Figure 5.

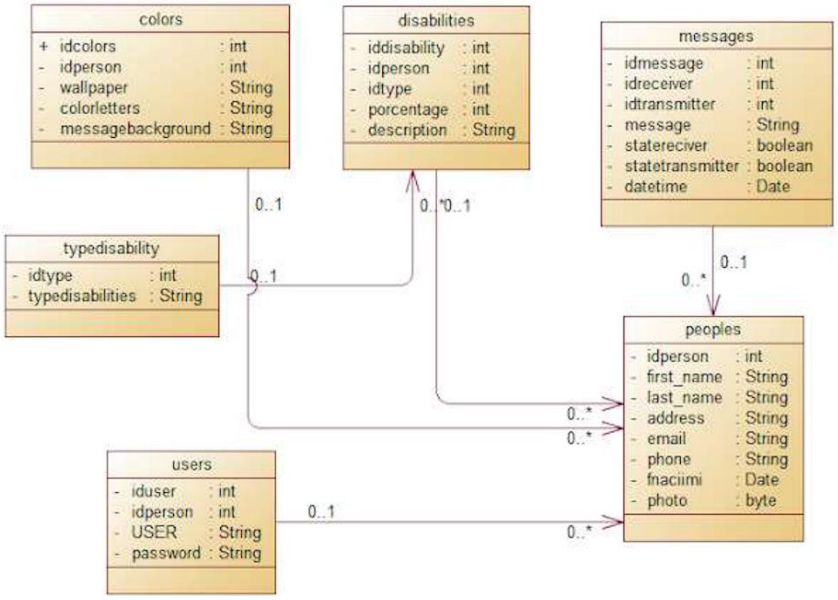

Fig. 6. Class diagram of BellChat.

The web application was developed in Java with its JSF 2.3 framework. Moreover, it was developed using the Model-View-Controller (MVC) model, which makes the web application has all the advantages of this model [38]. In addition, in this layer we find the Sockets Endpoint, which makes it possible to send and receive data in real time [36, 37]. All this runs on the Glassfish Server 4.1 application server.

Data layer (database server). The third layer of a three-tier architecture is the data persistence layer [39]. PostgreSQL 131 was the database server on which the Bell-Chat data persistence was implemented.

The tools and technologies used in the development of BellChat are free tools for education and research, so that the cost of producing the application would be the purchase of the domain, server rental and maintenance. In addition, as it is for vulnerable groups of people such as the visually impaired, the hearing impaired, the mobility impaired and the elderly, it is possible to try to obtain all services free of charge.

6 Evaluation of BellChat

Both the elicitation of requirements and the evaluation were carried out through an indirect survey. The answers expressed by the participants were taken directly by the researchers. Among the questions asked to elicit requirements for the implementation of BellChat were demographic questions such as age, disability or not, type of disabil-

[1] https://www.postgresql.org/

ity, percentage of disability, experience in the use of computer applications, among others. Questions to identify user needs included the difficulties of using third party tools for screen reading (in the case of people with visual impairment), tools for transcribing audio to text, and captioning a video for people with hearing impairment.

On the other hand, the questions to obtain the acceptance criteria of BellChat and the possible improvements that can be implemented, sought to know from the respondents the importance for them of the functionalities implemented in BellChat, the ease of use of these functionalities. In addition, we wanted to know what features BellChat lacks, what features could be improved, and a general opinion of BellChat as a whole.

Six people accepted to part in the evaluation. Three of them have a diagnosed visual disability. An elderly person had reduced vision and tremor in his hands. Two people have a hearing disability. There was no collaboration from people with motor disabilities.

The opinion of 100% of the respondents on the implemented functionalities is they are very useful, both the transcription of text-to-speech messages for visually impaired people and voice-to-text messages for hearing impaired people. In addition to considering the auto-configuration of the application according to user profile to make communication between users with different disabilities more transparent. One suggestion everyone made was Bellchat should be implemented as a mobile application. The assessment of the user-friendliness of the functionalities of BellChat is related to the experience in using computer applications, and to the percentage of disability of the respondent. People with little experience in using computer applications in general and with a high percentage of visual impairment were a bit self-conscious about using the application, but in the end, they found it easy to use. While people with a lot of experience in using computer applications needed a brief guide in the use of the application; in the end they stated that it was very easy to use.

7 Conclusions and future work

This paper has presented the development and a first evaluation of BellChat, an application for interpersonal communication aimed specifically at people with visual, hearing and motor disabilities, and older people with some pathologies. It is very useful for people considered by society to have normal abilities, who can be doing other tasks while listening to or sending messages to their contacts. Without a doubt, the social group with disabilities is only recently being considered when developing web applications open to the public. The use of web applications for interpersonal communication makes people feel that they belong to a group or society.

BellChat was developed following the Scrum development and project management framework, with some adaptations. Scrum proved to be a framework capable of adapting to a project with a very small team. For the development of BellChat the team consisted of 4 people: product owner, Scrum master and 2 developers. The experience of developing the same between the Scrum master and the developers makes the developers more collaborative to the project goals.

The evaluation of BellChat has motivated the project team to propose soon an application that uses WhatsApp services and has an inclusive interface especially for people with visual, hearing and motor disabilities, as well as the elderly, without leaving out people without physical disabilities, and to try to include people with intellectual disabilities. Also, the authors plan to develop a native Android and iOS application as a future work for the convenience of users. Although the current web application is responsive, the user must use a browser to access it. As BellChat was developed using the MVC development model, its business logic can be reused in other types of applications.

References

1. WCAG 2.0: Web Content Accessibility Guidelines (WCAG) Overview. Web Accessibility Initiative. 1–29 (2012)
2. W3C: Web Accessibility Initiative (WAI). https://www.w3.org/WAI/
3. Shawar, B.A.: Evaluating web accessibility of educational websites. International Journal of Emerging Technologies in Learning. 10, 4–10 (2015). doi:10.3991/ijet.v10i4.4518
4. Broccia, G., Manca, M., Paternò, F., Pulina, F.: Flexible Automatic Support for Web Accessibility Validation. Proc ACM Hum Comput Interact. 4, 1–24 (2020). doi:10.1145/3397871
5. Pelzetter, J.: A declarative model for accessibility requirements. In: Proceedings of the 17th International Web for All Conference, W4A 2020. pp. 1–10. Association for Computing Machinery, Inc, New York, NY, USA (2020)
6. Lister, K., Coughlan, T., Iniesto, F., Freear, N., Devine, P.: Accessible conversational user interfaces: Considerations for design. In: Proceedings of the 17th International Web for All Conference, W4A 2020. pp. 1–11. Association for Computing Machinery, Inc, New York, NY, USA (2020)
7. Domínguez Vila, T., Alén González, E., Darcy, S.: Website accessibility in the tourism industry: an analysis of official national tourism organization websites around the world. Disabil Rehabil. 40, 2895–2906 (2018). doi:10.1080/09638288.2017.1362709
8. Morris, M.R., Johnson, J., Bennett, C.L., Cutrell, E.: Rich representations of visual content for Screen reader users. In: Conference on Human Factors in Computing Systems - Proceedings. pp. 1–11. Association for Computing Machinery, New York, NY, USA (2018)
9. González, S.: Best Apps for Deaf People [Ranking 2022] (In Spanish). https://blog.audifono.es/aplicaciones-perdida-auditiva/
10. Pereira, J.: Handbook of research on personal autonomy technologies and disability informatics. IGI Global (2010)
11. Wang, K., Iwai, D., Sato, K.: Supporting Trembling Hand Typing Using Optical See-Through Mixed Reality. IEEE Access. 5, 10700–10708 (2017). doi:10.1109/access.2017.2711058
12. Onay Durdu, P., Soydemir, Ö.N.: A Systematic Review of Web Accessibility Metrics. IGI Global. 77–108 (1). doi:10.4018/978-1-7998-7848-3.ch004
13. Arooj, S., Zulfiqar, S., Qasim Hunain, M., Shahid, S., Karim, A.: Web-ALAP: A Web-based LaTeX Editor for Blind Individuals. In: ASSETS 2020 - 22nd International ACM SIGACCESS Conference on Computers and Accessibility. pp. 1–6. Association for Computing Machinery (ACM), New York, NY, USA (2020)

14. Lee, H.N., Uddin, S., Ashok, V.: TableView: Enabling Efficient Access to Web Data Records for Screen-Magnifier Users. In: ASSETS 2020 - 22nd International ACM SIGACCESS Conference on Computers and Accessibility. pp. 1–12. Association for Computing Machinery (ACM), New York, NY, USA (2020)
15. Lyall, F.C., Clamp, P.J., Hajioff, D.: Smartphone speech-to-text applications for communication with profoundly deaf patients. Journal of Laryngology and Otology. 130, 104–106 (2016). doi:10.1017/s0022215115003151
16. Shadiev, R., Hwang, W.-Y., Huang, Y.-M.: Investigating Applications of Speech-to-Text Recognition to Assist Learning in Online and Traditional Classrooms. International Journal of Humanities and Arts Computing. 8, 179–189 (2014). doi:10.3366/ijhac.2014.0106
17. Ali, F.A.B.H., Aydah, S.M.: Development of Prototype Chat System Using Mobile Platform for Disable People. Procedia Soc Behav Sci. 57, 33–39 (2012). doi:10.1016/j.sbspro.2012.09.1154
18. Alsaif, A., Albadrani, N., Alamro, A., Alsaif, R.: Towards intelligent Arabic text-to-speech application for disabled people. In: 2017 International Conference on Informatics, Health and Technology, ICIHT 2017. Institute of Electrical and Electronics Engineers Inc. (2017)
19. Sala, A., Arrue, M., Pérez, J.E., Espín-Tello, S.M.: Towards less complex e-government services for people with low vision. ACM SIGACCESS Accessibility and Computing. 1–10 (2020). doi:10.1145/3441497.3441499
20. Wahbi, I., A. Heider, M. Izzat, N. Mohammed: Evaluating Web Accessibility Metrics for Jordanian Universities. International Journal of Advanced Computer Science and Applications. 7, 113–122 (2016). doi:10.14569/ijacsa.2016.070716
21. Lorca, P., Andrés, J. de, Martínez, A.B.: The Relationship Between Web Content and Web Accessibility at Universities: The Influence of Social and Cultural Factors. Soc Sci Comput Rev. 36, 311–330 (2018). doi:10.1177/0894439317710435
22. Meadows, M., Kouw, M.: Future-making: Inclusive design and smart cities. Interactions. 24, 52–56 (2017). doi:10.1145/3046429
23. Greco, A.: Social Sustainability: From Accessibility to Inclusive Design. EGE-Expresión Gráfica en la Edificación. 18 (2020). doi:10.4995/ege.2020.14072
24. Dwivedi, A.K., Virmani, D., Ramasamy, A., Acharjee, P.B., Tiwari, M.: Modelling and analysis of artificial intelligence approaches in enhancing the speech recognition for effective multi-functional machine learning platform – A multi regression modelling approach. Journal of Engineering Research. 10, (2022). doi:10.36909/jer.icmet.17161
25. Magdalinou, A., Mantas, J.: Greek hospitals web accesibility. In: Studies in Health Technology and Informatics. pp. 1299–1300. IOS Press (2020)
26. Pascual, A., Ribera, M., Granollers, T.: Impact of web accessibility barriers on users with a hearing impairment. DYNA (Colombia). 82, 233–240 (2015). doi:10.15446/dyna.v82n193.53499
27. al Ghurair, N., Alnaqi, G., Doush, I.A.: An AAC tablet application for children with language impairment. Journal of Engineering Research. 9, 106–123 (2021). doi:10.36909/jer.v9i2.9201
28. Schwaber, K., Sutherland, J.: The scrum guide. Acrum Allience. 21, (2011)
29. Scrum.org: Scrum Master Trends 2019. , Burlington, USA; Berlin, Germany (2019)
30. Rising, L., Janoff, N.S.: Scrum software development process for small teams. IEEE Softw. 17, 26–32 (2000). doi:10.1109/52.854065
31. Obeidat, B., Muhammad, B.B.: A Study of Communication Ties Among Team Members in Work Environments. Journal of Engineering Research. 9, 1–13 (2021). doi:10.36909/jer.v9i3a.10529

32. Spurrier, G., Topi, H.: Synergistically Employing User Stories and Use Cases in the Practice and Teaching of Systems Analysis and Design. Communications of the Association for Information Systems. 51, 15 (2022). doi:10.17705/1cais.05108
33. Amaral, L.M., Siqueira, F.L., Brandão, A.A.F.: A survey on requirements notations in software engineering research. Proceedings of the ACM Symposium on Applied Computing. 1291–1298 (2022). doi:10.1145/3477314.3507253
34. Elallaoui, M., Nafil, K., Touahni, R.: Automatic Transformation of User Stories into UML Use Case Diagrams using NLP Techniques. Procedia Comput Sci. 130, 42–49 (2018). doi:10.1016/j.procs.2018.04.010
35. Lozano, A.: Requirements vs Use Cases vs User Stories (In Spanish). http://www.angellozano.com/requisitos-del-sistema-vs-casos-uso-vs-historias-usuario/
36. Gupta, Y., Dewan, H., Leekha, A.: Real-time monitoring using AJAX and WebSockets. Journal of Statistics and Management Systems. 23, 125–134 (2020). doi:10.1080/09720510.2020.1714154
37. Ciuffoletti, A.: Design of an Open Remote Electrocardiogram (ECG) Service. Future Internet. 11, 101-119 (2019). doi:10.3390/fi11040101
38. Zhou, K.: Research and Implementation of MVC Design Pattern on J2EE Platform. 1122–1127 (2022). doi:10.1007/978-981-19-4132-0_153
39. Marco, P., Marcu, M.: Impact of the persistence layer on performance and architecture for a Java web application. SAMI 2022 - IEEE 20th Jubilee World Symposium on Applied Machine Intelligence and Informatics, Proceedings. 261–266 (2022). doi:10.1109/sami54271.2022.9780780