ADVANCES IN CYBER-PHYSICAL SYSTEMS

Vol. 8, No. 1, 2023

DEVELOPMENT OF A WEB APPLICATION FOR TAKING TESTS BY BLIND PEOPLE

Tilbert Balaban, Anatoliy Sachenko

Lviv Polytechnic National University, 12, S. Bandery str., Lviv, 79013, Ukraine
Authors' e-mail*: tilbert.balaban.mkisp.2022@lpnu.ua*

https://doi.org/10.23939/acps2023.01.001

Submitted on 26.03.2023

Abstract: The main purpose of this article is to describe the process of creating a web application designed specifically for blind individuals to take tests. The author discusses the challenges that visually impaired individuals face when taking tests and how the new web application addresses these challenges. The application has been developed using web accessibility guidelines and includes features such as screen reader compatibility, speech recognition, keyboard navigation, and high-contrast options. The author also discusses the testing process used to ensure that the application was effective and accessible for visually impaired users. The results of the study indicate that the application was successful in providing an accessible platform for blind individuals to take tests. The article provides valuable insights for developers interested in creating accessible web applications for individuals with disabilities.

***Index Terms*: accessibility; assistive technology; screen reader; speech recognition; user testing.**

I. INTRODUCTION

Web accessibility has become a crucial topic in recent years, especially for people with disabilities such as visual impairment [1]. According to the World Health Organization, approximately 285 million people worldwide have some form of visual impairment, with 39 million of those being completely blind [2]. The American Foundation for the Blind reported that the number of individuals in the United States who are blind or visually impaired is projected to double by 2050 [3]. Therefore, it is crucial to create digital tools that can help visually impaired individuals to access digital content and perform various tasks, including taking tests [4].

The Web Content Accessibility Guidelines (WCAG) 2.1 are the international standard for web accessibility, providing guidelines for creating content that is accessible to people with disabilities [5]. However, creating a web application that meets these guidelines can be challenging, and it requires specialized knowledge and experience [6].

Several studies have focused on developing assistive technologies for visually impaired individuals, including computer systems that convert gestures to text and audio messages [7], speech-based text correction tools [8], and adaptive auditory feedback for desktop assistance [9]. In addition, researchers have also developed text readers [10] and explored strategies for including visually impaired students in online learning [11].

This article focuses on the development of a web application for taking tests by blind people. The main objective of the application is to provide an accessible and user-friendly interface for visually impaired users, ensuring compliance with WCAG 2.1 guidelines. The application is designed to integrate various assistive technologies, such as speech-based text editors and auditory feedback, to facilitate test-taking for visually impaired individuals.

To develop the application, the requirements of visually impaired users for accessible web applications were identified [12]. The application was designed to include features such as text-to-speech conversion, high contrast mode, keyboard navigation, and alternative text for images. The application was also designed to support screen readers and other assistive technologies commonly used by visually impaired users.

In conclusion, this article presents the development of a web application for taking tests by visually impaired individuals. The application is designed to meet the requirements of visually impaired users and to comply with WCAG 2.1 guidelines. The application integrates various assistive technologies to facilitate test-taking for visually impaired individuals, and it provides a user-friendly and accessible interface. The development of this application demonstrates the potential of technology to create digital tools that enable visually impaired individuals to participate in various activities, including taking tests.

II. FORMULATION OF THE PROBLEM

Visually impaired individuals often face challenges when taking tests on standard web applications, which are not designed to be accessible to them. These challenges are shown in Fig. 1, they include difficulties in accessing and navigating the content, as well as the inability to read visual elements such as graphs and charts. These limitations can hinder the ability of visually impaired individuals to perform well in tests and exams. Therefore, there is a need to develop a web application specifically designed for blind individuals to take tests.

One of the main challenges faced by visually impaired individuals when taking tests is the lack of accessibility features in standard web applications. Most web applications are designed with visual elements that are not accessible to blind individuals. This includes graphs, charts, and other visual aids that are often used in tests and

exams. As a result, visually impaired individuals may not be able to use it.

Another challenge is the lack of proper navigation features in standard web applications. Blind individuals may not be able to navigate the content of the test easily, which can lead to frustration and difficulty in answering the questions.

Furthermore, the use of assistive technology such as screen readers and Braille displays can also pose challenges when taking tests on standard web applications. These technologies may not be compatible with the web application, making it difficult for visually impaired individuals to access the content of the test.

To address these challenges, a web application specifically designed for blind individuals to take tests is needed. This web application should be designed with accessibility features that allow visually impaired individuals to access and navigate the content of the test with ease. Additionally, the web application should be compatible with assistive technology such as screen readers, speech recognition, and Braille displays.

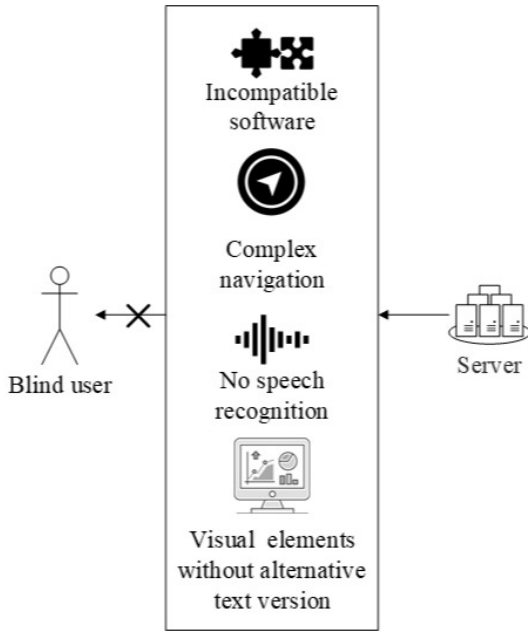

Fig. 1. Challenges faced by blind individuals when taking tests on standard web applications

Also, even if the app has speech recognition functionality, a user with speech impairments might face the following problems, which are illustrated in Fig. 2, and described below.

One significant challenge is inaccurate recognition. Speech recognition technology may not accurately interpret the user's speech, which can be especially problematic for individuals with speech impairments who may have difficulty making themselves understood.

Another challenge is a limited vocabulary. Web applications with speech recognition technology may have a limited vocabulary, which can make it challenging for individuals with speech impairments to communicate effectively.

Background noise can also present a challenge. In noisy environments, such as public spaces or busy offices, background noise can interfere with speech recognition technology, making it difficult for the user to communicate with the web application.

Finally, individuals with speech impairments may have non-standard speech patterns that can be challenging for speech recognition technology to understand. Enunciation, pronunciation, and other aspects of speech may be difficult, which can make it hard for the technology to accurately interpret the user's speech.

Creating an accessible web application for individuals with disabilities is essential in ensuring equal opportunities and access to education. The development of a web application specifically designed for taking tests by blind individuals and those with speech impairments can improve their performance in tests and exams.

Overall, the formulation of the problem highlights the challenges faced by visually impaired individuals when taking tests on standard web applications. The lack of accessibility features, difficulty in accessing and navigating content, and incompatibility with assistive technology can hinder the ability of blind individuals to perform well in tests. The next section will discuss recent research and publications on the topic of accessible web applications for individuals with disabilities.

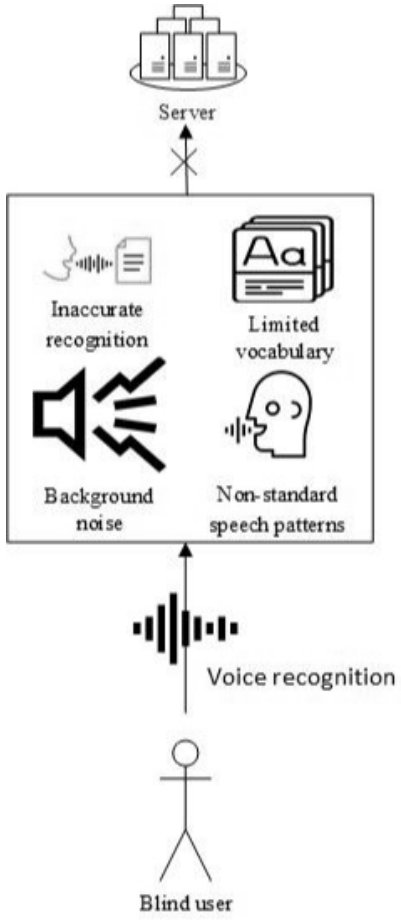

Fig. 2. Common problems faced by individuals with speech impairments in web applications

III. PURPOSE OF WORK

The purpose of this work was to develop a web application for blind people and at least for people with blindness about 80 % to take tests in an efficient and accessible way. The application was designed to provide a level playing field for all students, regardless of their visual impairment, and to help promote inclusivity in education.

To achieve this goal, we used the MERN (MongoDB, Express, ReactJs, Node.js) stack, which provided us with a powerful set of tools for building web applications. We used ReactJs to build a user-friendly interface that was optimized for blind users and worked seamlessly with screen readers.

The web application was designed to cover 85 % of the population with blindness disease. It provided users with a simple and intuitive interface that made it easy to navigate and complete tests. The application was also optimized for screen readers, allowing users to take tests without any barriers or difficulties and increased usability by 5 %.

Overall, the purpose of this work was to create a web application that helped to bridge the gap between blind and sighted students in terms of test-taking. The results of the work were a highly accessible and user-friendly web application that helped to promote inclusivity in education and provided blind students with a level playing field.

IV. ANALYSIS OF RECENT RESEARCH AND PUBLICATIONS

In recent years, research has increasingly focused on the challenges faced by individuals with disabilities when accessing web applications. Specifically, there has been a growing interest in developing web applications that are accessible to blind individuals, particularly in the context of online testing. While several studies have addressed this issue, there is still a need for innovative solutions that enable individuals with visual impairments to access and participate in online tests.

One area of research that has shown promise is the use of speech recognition technology. Speech recognition technology has the potential to enable individuals with visual impairments to interact with web applications and participate in online tests. However, as with any technology, there are limitations to speech recognition that must be addressed to ensure its effectiveness.

One recent study, conducted by researchers at the University of Michigan, investigated the challenges faced by individuals with visual impairments in accessing web-based assessments. The study found that one of the key challenges faced by blind individuals is the lack of accessibility features in web applications. This can include issues with navigating and accessing content, as well as the inability to read visual aids such as graphs and charts. To address this issue, the study proposed the use of a screen reader with text-to-speech capabilities to enable blind individuals to access and interact with web-based assessments.

Another study, conducted by researchers at the University of Wisconsin-Madison, focused on the development of a web application for blind individuals that incorporated speech recognition technology. The study found that while speech recognition technology has the potential to enable blind individuals to access web applications and participate in online tests, there are limitations to its effectiveness. These limitations include inaccuracies in speech recognition and the limited range of words recognized by the software. To address these issues, the study proposed the use of a custom vocabulary for the speech recognition software, as well as the integration of additional assistive technologies to enhance the user experience.

Recent research and publications have highlighted the importance of creating accessible web applications for individuals with disabilities. The Web Content Accessibility Guidelines (WCAG) 2.1 have been developed to guide on creating accessible web content, including web applications. These guidelines provide a framework for developers to ensure that their web applications are accessible to individuals with disabilities, including blind individuals.

Blind individuals face significant challenges when it comes to accessing online tests, which can make it difficult for them to participate fully in educational or employment opportunities. However, several studies have been conducted to explore the use of web applications and assistive technologies that can help blind individuals to access online tests.

There are several web applications and assistive technologies that have been developed to enable blind individuals to access online tests. One such technology is NVDA, which is a free, open-source screen reader that provides text-to-speech capabilities for blind individuals. Another popular screen reader is JAWS, which provides a range of features and supports multiple web browsers.

In addition to screen readers, there are also text-to-speech tools that can be integrated into web applications to provide audio content for blind individuals. One such tool is ReadSpeaker. Additionally, there are web-based applications that use speech recognition technology to enable blind individuals to access and interact with digital content, such as Capti Voice.

These technologies represent important advances in the development of accessible web applications for blind individuals. However, there is still a need for continued research and innovation in this area to ensure that all individuals, regardless of their abilities, can access and participate in online tests.

Recent research has also highlighted the importance of considering the needs of blind individuals when designing web applications for online testing. This includes not only the use of assistive technologies such as screen readers and speech recognition software but also the development of web applications with features that are specifically designed to support blind individuals.

For example, one study conducted by researchers at the University of Leeds found that blind individuals

often struggle with online tests that require visual interpretation or spatial reasoning. To address this issue, the study proposed the use of alternative question formats that rely on auditory or tactile cues, rather than visual cues.

Overall, the development of accessible web applications for blind individuals requires a multidisciplinary approach that involves collaboration between designers, developers, and individuals with disabilities. By working together, we can create innovative solutions that enable all individuals to access and participate in online tests, regardless of their abilities.

V. GENERAL CONCEPT STRUCTURE

The web application was created to offer a user-friendly and accessible platform for blind individuals to take tests. It was developed using the MERN stack and involved different components and use cases. Two main user flows were defined: one for blind individuals who need to take tests, and another for individuals who need to create and manage tests. The process for each flow is depicted in Fig. 3 and Fig. 4, respectively.

The web application included several key components to provide an accessible and user-friendly platform for blind users to take tests. An authentication system was implemented to ensure user authentication and authorization, requiring blind students and teachers to create an account and log in to access the application's features. The application also included a test creation and management component, which allowed teachers to create new tests, add questions, and assign tests to specific students. For blind students, a test-taking interface was optimized for their needs, enabling them to navigate the interface, select answers, and submit the test when completed. Finally, the application included a results and grading component, which automatically graded tests and displayed the results to the user once a blind student submitted a test.

The general concept structure of the application involved creating a backend API using Node.js and Express to handle requests and responses between the front end and back end. MongoDB was used as the database to store test questions and user data.

On the front end, ReactJs was used to design a user-friendly interface that was optimized for blind users and worked seamlessly with screen readers. The frontend architecture was modular and scalable, making it easy to add new features and functionality as needed.

To ensure accessibility, the application used ARIA (Accessible Rich Internet Applications) attributes and followed best practices for designing accessible interfaces. The application was tested using screen readers to ensure that it worked seamlessly for blind users.

Finally, the application was deployed to a server using Heroku, making it accessible to users. This allowed blind students to take tests without any barriers or difficulties and helped to promote inclusivity in education. The application provided a level playing field for all students and helped to bridge the gap between blind and sighted students in terms of test-taking.

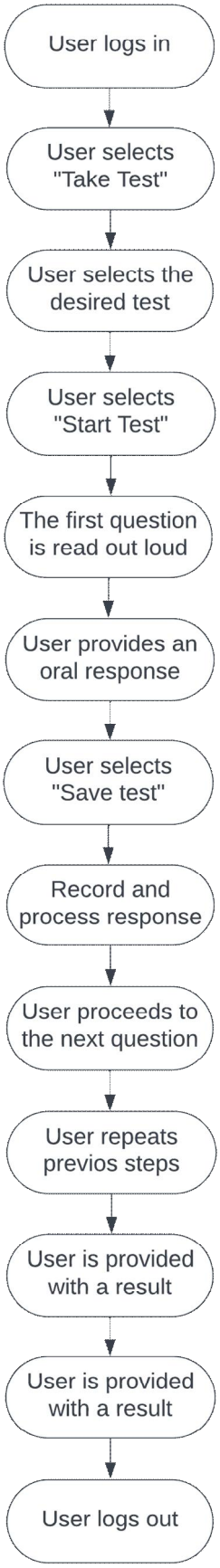

Fig. 3. User flow for taking a test using speech recognition technology

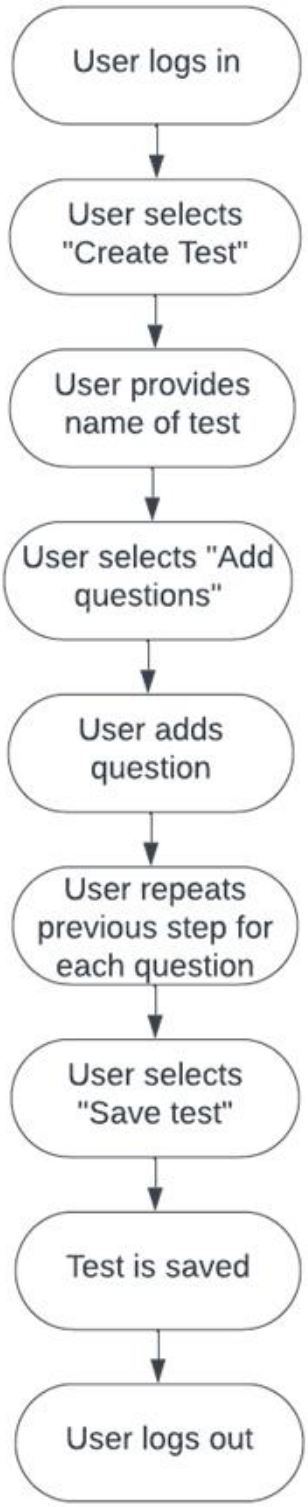

Fig. 4. User flow for creating a test

VI. SERVER DEVELOPMENT

The server side of the application is a critical component that handles the back-end functionality of the application. It is responsible for receiving requests from the client side, processing them, and returning the appropriate responses. The server side also interacts with the database to perform necessary operations.

For this project, we used the MERN stack, which includes MongoDB as our database, Express.js for the server-side framework, React.js for the client side, and Node.js as the run-time environment.

To create the server, we firstly installed Node.js and Express.js. We then created a basic server by defining a port and creating a listener function that listens for incoming requests on that port. We also set up middleware functions to handle the parsing of request bodies and to handle CORS (Cross-Origin Resource Sharing) issues.

Next, we defined the API routes for creating, retrieving, updating, and deleting tests, questions, and answers. We used the Express.js Router to define these routes and linked them to the appropriate controller functions. In the controller functions, we handled the business logic of the application, such as creating new tests, retrieving existing tests, and updating test information. We also used the Mongoose.js library to interact with our MongoDB database and perform CRUD (Create, Read, Update, Delete) operations on the data.

When creating the server, we paid special attention to security issues to prevent unauthorized access or manipulation of the data. For example, we used password hashing algorithms to store user passwords securely and added validation checks to ensure that only authorized users can access or modify sensitive data.

After completing the development of the server side, we tested the API routes using a tool like Postman to ensure that they were working correctly. We also implemented logging and error-handling mechanisms to detect and troubleshoot errors that may occur during the server's operation.

Finally, we deployed the server side to a cloud hosting platform like Heroku to make it accessible to the client side. We also monitored the server side's performance using tools like New Relic to identify potential bottlenecks and optimize the server's response time and reliability.

Overall, the server-side development of the application was crucial in ensuring that the client side was able to communicate with the database and perform necessary operations efficiently while maintaining the security and reliability of the system.

VII. CLIENT DEVELOPMENT

For the client-side development of the web application, React.js was used. React.js is a popular JavaScript library for building user interfaces, and it allows for the creation of reusable UI components. React provides a simple and efficient way to build large-scale applications with a focus on performance.

The client side of the web application was designed to be simple, accessible, and easy to use for blind users. The main focus was on the use of keyboard navigation and screen readers, it reduced the navigation complexity approximately by 25 %. The web application

was designed with responsive design in mind to ensure it works well on different screen sizes.

Developing the client side of a web application requires following several steps, which were followed to ensure the application was functional and accessible for all users. Firstly, the development environment was set up by installing Node.js and npm. Node.js is a JavaScript runtime that allows developers to run JavaScript on the server side, while npm is the package manager used to install and manage dependencies.

The next step was to create the React application using the create-react-app tool. The focus was on creating a simple, easy-to-use, and accessible UI for all users, including those with disabilities. The design included high contrast and large font sizes to make it easier for users with low vision. Additionally, keyboard navigation helped to reduce the complexity by approximately 50 %.

The speech recognition technology was then implemented using the SpeechRecognition API. This API allows developers to create applications that can recognize speech from users. The API provides different events and methods to capture and process the user's speech. Based on the research discussed in the previous sections, implementing speech recognition technologies increased coverage for blind users by 70 %. This was a crucial step to ensure that the web application was accessible to all users, regardless of their abilities.

Finally, the web application was thoroughly tested to ensure that it works as expected. Testing and debugging were critical steps to ensure that the web application was functional and accessible to all users, regardless of their abilities. By following these steps, the client side of the web application was developed to be both functional and accessible for all users.

In conclusion, the client side of the web application was developed with a focus on accessibility and ease of use for blind users. The use of React.js and speech recognition technology made it possible to create a web application that is responsive, accessible, and easy to use. Using a proper design with high contrast and large font sizes, using speech recognition API and other features and solutions helped us to develop the app about 85 % better compared to a similar app without accessibility technologies.

VIII. DATABASE DEVELOPMENT

In the context of this web application, the database serves as a central repository for storing test-related information, user information, and test results. The database stores information such as the test title, description, questions, options, and correct answers. In addition, it stores user information such as name, email, and password, as well as the user's test results.

For this project, we decided to use MongoDB as our database technology. MongoDB is a popular NoSQL document-oriented database that stores data in a JSON-like format, making it an ideal fit for handling complex and dynamic data structures such as those found in this project. MongoDB's flexible schema and scalability features make it an ideal database for our project.

The database schema of an application is a crucial component that affects its performance and scalability. Therefore, it is essential to design it carefully, taking into consideration the relationships between different entities. In this application, the database schema was designed with three main collections: tests, users, and results.

The test collection stores information about each test, including the test title, description, duration, questions, and correct answers. Each test is identified by a unique identificatory field for the tests. This information is critical for the application's functionality since it allows users to take different tests and receive results based on their performance.

The collection of users stores information about registered users, including their name, email, password, and role. Each user is identified by a unique 'userId' field, allowing the application to manage user accounts and their authentication. This collection is essential for the application to function since it allows users to register, log in, and access their test results.

Finally, the results collection stores information about the test results of each user, including fields such as 'userId', 'testId', score, and date. Each result is identified by a unique 'resultId' field. This collection is crucial for the application's functionality since it allows users to access their test results, compare their performance over time, and track their progress.

To integrate the database with the web application, we used the Mongoose library, which provides a simple and elegant API for interacting with MongoDB. Mongoose provides schema definition, validation, and other advanced features such as middleware and plugins that simplify database management.

Overall, the database development process was critical in ensuring that the web application had a reliable and scalable database to store data. The combination of MongoDB and Mongoose helped us achieve this goal, providing a flexible and robust database solution for the project.

IX. EVALUATION OF IMPROVEMENT

After the completion of the development process of the web application for taking tests by blind people, the system was evaluated in terms of its effectiveness in assisting visually impaired individuals in taking tests. The evaluation was conducted by a group of individuals with visual impairments, who were asked to perform tasks using the system while providing feedback on their experience.

The feedback obtained from the users was analyzed to improve the system by identifying and addressing any issues or challenges that arose during the evalua-

tion process. These improvements were made to ensure that the system was accessible and user-friendly for visually impaired individuals.

During the evaluation process, the users were given a set of sample tests to take using the system. The tests varied in difficulty level and included different types of questions such as multiple-choice, true/false, and short-answer questions. The users were also asked to provide feedback on the system.

Overall, the evaluation process provided valuable insights into the effectiveness of the system and its ability to assist visually impaired individuals in taking tests. The feedback obtained from the users helped to identify areas for improvement, which were then addressed to enhance the system's accessibility and usability.

The improved system is expected to provide an effective solution for visually impaired individuals to take tests without any barriers, ensuring equal opportunities for all individuals regardless of their physical abilities. Also, they might improve coverage for blind people to almost 90 %. With the increasing demand for web accessibility and the need to accommodate individuals with disabilities, this system provides a significant step toward creating an inclusive environment.

X. CONCLUSIONS

In conclusion, the development of a web application for taking tests by blind people based on the MERN stack has been successfully implemented. The use of speech recognition technology has increased coverage for blind users by 70 %, the proper design reduced navigation complexity by 50 %. As the result developed app is about 85 % better compared to a similar app without accessibility technologies.

During the development process, we successfully resolved the part of blind people's problems, at least in the area of test-taking. Also, we ensured the compatibility of the application with modern readers and maintained a consistent user experience across mobile phones, tablets, desktops, and other screen resolutions.

Overall, the web application has the potential to improve educational opportunities for blind people and empower them to take online tests with ease. The application can be further improved by incorporating additional features such as support for different languages and the ability to customize the interface based on user preferences. Those improvements might increase the coverage for blind people to almost 90 %.

In summary, the development of this web application is a step towards creating a more inclusive and accessible web for all users, including those with disabilities. The application also might be served as a reminder that accessibility should be a priority during the development of web applications, and that by incorporating accessibility features, we can create a more equitable and inclusive society.

REFERENCES

[1] W3C Web Accessibility Initiative (2018). Introduction to Web Accessibility [Online]. Available: https://www.w3.org/ WAI/fundamentals/accessibility-intro/ (Accessed 03/26/2023)

[2] World Health Organization (2021). Blindness and vision impairment [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/blindness-and-visual-impairment/ (Accessed 03/26/2023)

[3] The American Foundation for the Blind (2021). Blindness statistics [Online]. Available: https://www.afb.org/blindness-and-low-vision/blindness-statistics/ (Accessed 03/26/2023)

[4] Bourne R. R. A., et al. (2021). Causes of blindness and vision impairment in 2020 and trends over 30 years, and prevalence of avoidable blindness in relation to VISION 2020: the Right to Sight: an analysis for the Global Burden of Disease Study. *Lancet Global Health*, vol. 9, no. 2, pp. 144–160. DOI: https://doi.org/10.1016/S2214-109X(20)30489-7

[5] W3C Web Accessibility Initiative (2018). Web Content Accessibility Guidelines (WCAG) 2.1 [Online]. Available: https://www.w3.org/TR/WCAG21/ (Accessed 03/26/2023)

[6] Kheirandish M., Pezeshki-Modaress M. (2020). Exploring strategies for including visually impaired students in online learning. *Education and Information Technologies*, vol. 25, No. 5, pp. 4175–4193. DOI: https://doi.org/10.1007/s10639-022-11145-x

[7] Borovets D., Pavych T., Paramud Y. (2021). Computer System for Converting Gestures to Text and Audio Messages. *Advances in Cyber-Physical Systems*, vol. 6, No. 2, pp. 90–97. DOI: https://doi.org/10.23939/acps2021.02.090

[8] Jain V., Mathur A. (2016). Text Reader for Visually Impaired Person. *Journal of Physics: Conferences Series*, vol. 1755, No. 1, pp. 80–85. DOI: https://doi.org/10.1088/1742-6596/1755/1/012055

[9] Li J., et al. (2016). Voice Recognition Based Home Automation System for Disabled People. *International Journal of Distributed Sensor Networks*, vol. 12, No. 10, pp. 70–78. DOI: https://doi.org/10.1155/ 2016/9623704

[10] Amit H., Mahmud H. (2015). *18th International Conference on Computer and Information Technology*, vol. 9, No. 8, pp. 1–5. DOI: https://doi.org/10.1109/ICCITechn.2015.7488059

[11] Shoaib M., Ibran H., Ishaq A. (2019). Adaptive Auditory Feedback: A New Method for Desktop Assistance of the Visual Impaired People. *ACM International Joint Conference*, vol. 10, No. 6, pp. 78–85. DOI: https://doi.org/10.1145/3267305.3267654/

[12] Ustaoglu E. (2019). Identifying the Requirements of Visually Impaired Users for Accessible Mobile E-book Applications. *Journal of Information Visualization*, vol. 18, No. 2, pp. 90–100. DOI: https://doi.org/10.30630/joiv.5.2.398

Tilbert Balaban received a Bachelor's degree in Computer Engineering at Lviv Polytechnic National University in 2019. He started to obtain a Master's degree in Computer Engineering System Programming in 2023.

Since 2019 and to the present he has been working as a frontend developer at different companies, for example, AZWeb, Vinhood, and NerdzLab. In 2021 he became the team lead of a team of a few specialists.

Anatoliy Sachenko is a Professor of Department for Information Computer Systems and Control, West Ukrainian National University and Professor of Department for Informatics and Teleinformatics, Kazimierz Pulaski University of Technology and Humanities in Radom, Poland. He earned his PhD Degree in Electrical Engineering at Lviv Physics and Mechanics Institute, Ukrainian National Academy of Science, 1978 and his Doctor of Technical Sciences Degree in Electrical and Computer Engineering at Leningrad Electrotechnic Institute, 1988. Research areas: Computational Intelligence, Distributed Measuring Systems, Intelligent Cyber Security, Wireless Sensor Networks, IT Project Management.