Serhiienko G. Y., Shevchenko K. M., Roienko L. V.
Kyiv National University of Technologies and Design

ACCESSIBILITY-FIRST APPROACH: HOW DESIGN AND CODE ENSURE INCLUSIVITY

Abstract. *This article examines the principles of digital accessibility, explores ways of ensuring access to digital tools for users with disabilities. The study analyzes the characteristics of certain health problems and their impact on the use of digital technologies. The article offers advice for application developers to ensure accessibility for users with disabilities, examines types of assistive technologies, and important aspects of digital accessibility testing.*

Keywords*: digital inclusion, digital accessibility, accessibility-first approach, integrated development solutions, digital environment.*

Сергієнко Г. Є., здобувач бакалаврату, Шевченко К. М., здобувач бакалаврат, Роєнко Л. В., старший викладач
Київський національний університет технологій та дизайну

ПІДХІД, ОРІЄНТОВАНИЙ НА ДОСТУПНІСТЬ: ЯК ДИЗАЙН ТА КОД ЗАБЕЗПЕЧУЮТЬ ІНКЛЮЗИВНІСТЬ

Анотація *У статті проаналізовано принципи цифрової доступності, досліджуються способи забезпечення доступу до цифрових інструментів для користувачів з обмеженими можливостями. У дослідженні проаналізовано особливості певних проблем зі здоров'ям та їх вплив на користування цифровими технологіями. У статті запропоновано поради для розробників програм з метою забезпечення доступності для користувачів з обмеженими можливостями, розглянуто типи допоміжних технологій та важливі аспекти тестування цифрової доступності.*

Ключові слова: *цифрова інклюзія, цифрова доступність, підхід, орієнтований на доступність, інтегровані рішення для розвитку, цифрове середовище.*

Introduction. Digital inclusion or digital accessibility is the concept of the ability or capacity to use digital tools and technologies by users that belong to vulnerable and socially sensitive groups, such as people with hearing and vision disabilities, reduced mobility and other physiological or social problems. The goal of inclusion is to establish equal accessibility for all users, which implies the readiness of systems and services to meet the needs of the community. The concept of digital accessibility was formed in response to the need to ensure that all people, regardless of their abilities, can fully participate in the digital world [1, p. 25]. The accessibility-first approach means that accessibility issues are taken into account at all stages of creating a digital product – from idea and design to development and testing. This is not an "additional check before release", but an integrated principle that determines the quality of the project.

In order to achieve the implementation of inclusion and its dissemination developers must rely on a wide range of rules, recommendations, methodologies, and assistive technologies that regulate the content of the application to existing standards or modify the presentation and interaction with applications to ensure their use despite the existing obstacles of human mobility. The concept of inclusion should be introduced at the very beginning of the digital project modelling and developed in parallel with all other non-functional needs of the application, since its implementation affects a larger number of development processes, starting from mockups and ending with software implementation and testing.

Problem statement. Based on the above, the purpose of this paper is to highlight the principles of digital accessibility, analyse the development solutions for digital accessibility approach, outline the main aspects of digital accessibility testing and standards.

Research results. When we talk about existing digital resources we also visualise them in front of our eyes, which makes it possible for us to imagine a route of our actions and subsequent interaction with the application. Most of the users use such a way to find out for themselves how to interact with source or analyze the opportunities that the program provides in order to plan further actions. This makes product design an important and integral part of defining the range of functionality that is and is not available to the user. Further software development is based on visual solutions to connect components which are visible to the user with those that are hidden in the system. Therefore, design is an important part of digital development and must also follow the principles of inclusion. The solution to this problem is not for people with disabilities to adapt to existing conditions. On the contrary, it is designers and engineers who must create environments, products and services that are accessible to all users regardless of their abilities.

Speaking about visual components of the platform and its content, developers must heavily rely on specifics of visual impairments and neurological disorders. Globally, at least 2.2 billion people have a near or distance vision impairment. In at least 1 billion of these, vision impairment could have been prevented or is yet to be addressed [2].

High pressure in the retina usually is caused by glaucoma, damaging the optic nerve, which transmits images to the brain. This can cause peripheral vision problems, which can limit the field of vision. Patients often complain of difficulty seeing in low light and in motion. So it is important to note that people with peripheral vision problems can only see central elements.

Diabetic retinopathy is a complication of diabetes that affects the eyes. This can lead to spots or dark strings floating in sight, blurred vision, dark or empty areas in vision or even vision loss itself.

Color blindness is defined as the inability to perceive and distinguish colors. The most common type is the red-green type, followed by the blue-yellow type and complete absence of color perception. This disorder affects about 4.5% of the world's population, or one in 12 men and one in 200 women. Color blindness should be taken into account when choosing a color palette. It is also better to avoid designating elements through colors, instead giving priority to symbols and additional titles or labels, for example, in indicating the state of elements, such as errors or successes. For users with color blindness, it is important to add other visual cues, such as icons or text, that help them clearly understand the state of elements on the page or their meanings.

For people with visual impairments, such as color blindness, it is important to provide sufficient contrast between text and background. According to WCAG 2.0 standards, the contrast ratio for standard-sized text should be at least 4.5:1, and for large text, at least 3:1. Using simple sans-serif fonts (e.g., Arial, Verdana) also improves readability for all users.

"Neurodiversity" broadly refers to naturally occurring diversity in human cognition. It has been used as an umbrella term for a range of neurocognitive developmental disorders such as attention deficit hyperactivity disorder (ADHD), autism, dyslexia, dyspraxia/developmental coordination disorder (DCD), Tourette syndrome (TS), dyscalculia and dysgraphia [3, p. 354]. Users with the listed syndromes and disorders may experience difficulties interacting with digital applications, as described below.

Dyslexia is a neurological difference and can have a significant impact during education, in the workplace and in everyday life. People with dyslexia may have literacy, memory and information processing skills difficulty [4]. It affects a person's ability to perceive language and associate letters with sounds, which requires special attention to the design of text and visual content. People with dyslexia often face difficulties in reading text, so font choice is one of the key aspects of accessibility. For example, sans serif fonts might be better solutions, such as Arial, Calibri, Century Gothic, Trebuchet, Tahoma, Verdana. Additionally, there are fonts specifically designed for people with dyslexia, including Open Dyslexic, Read Regular, Dyslexie, Sylexiad, and Lexia. To avoid visual overload, it is recommended to use a font size of 12–14 pt and a line spacing of 150% of the font size. And to reduce the amount of text and make it simpler, it is advisable to use bulleted and numbered lists, divide information into blocks, and use punctuation marks.

Most of the recommendations above can also be applied to users with an autistic spectrum disorder. Autism Spectrum Disorders (ASD) is a neuro developmental disorder that occurs in the early stage and is characterized by social interaction, communication disorders, narrow range of interests, and stereotyped behaviors. The text should be simply structured, avoiding idioms and metaphors. Humor and ambiguous expressions should be avoided, as they may be misinterpreted by users with ASD. Also, to avoid overload, as individuals with ASD can be quite sensitive to bright colors and sounds, it is necessary to choose soft and muted palettes.

It is also very important for users with epilepsy. Epilepsy is a neurological disease characterized by the brain's tendency to have repeated, spontaneous seizures. Seizures can range in type from brief "blackouts" (absence seizures) to generalized seizures. In addition to physical symptoms, epilepsy affects cognitive function, concentration, and information processing. Some users with epilepsy are sensitive to flickering, stroboscopic effects, high-contrast colors, and fast animations, which can trigger a seizure. Overly dynamic content, such as pop-up messages, fast transitions, and noise effects, can cause fatigue and increase the risk of seizures. People after seizures may have temporary cognitive difficulties with concentration, memory, and reaction, which affects the use of websites and applications with complex and detailed interfaces. It is recommended to give users a choice and the ability to customize the visual appearance. For example, allow them to disable animations, video autoplay, flashes, or dynamic content. Also provide options to "reduce motion" or "dark mode". In overall it is important to minimize cognitive load and implement simple and predictable navigation.

At the design stage, this is manifested in the choice of solutions that are convenient for the widest possible range of users. This includes color palettes with sufficient contrast, clear typography without excessive decorative elements, logical page structure, and predictable navigation. In large companies, designers use ready-made systems (Material Design, Fluent UI), which have already tested accessible components, which reduces the risk of errors.

During development, the main task becomes compliance with standards – semantic HTML markup, correct use of ARIA attributes, ensuring keyboard navigation, support for scaling and adaptability. The developer does not just "make it work", but immediately lays down the possibility of correct interaction with screen readers, alternative input devices, and assistive technologies.

Next comes testing, which in the case of accessibility-first is moved "to the left" in the development process. This means that accessibility is checked not after the work is completed, but in each sprint. For this, automated tests (axe, Pa11y, Lighthouse [5]) and manual testing using emulators or screen readers are used. Ideally, teams involve people with disabilities in usability tests, since it is their experience that reveals problems that are not visible to the tools.

An important component is the culture in the team: developers and designers undergo training in digital accessibility, create internal guides and checklists that become the “definition of readiness” for tasks. When each team member understands the meaning of inclusion, accessibility ceases to be an additional requirement and becomes part of the professional standard.

Thus, accessibility-first is not a set of formal rules, but an approach that makes the product convenient, predictable and useful for everyone, regardless of perception or physical capabilities. This is both an ethical obligation and a strategic advantage, as inclusive products reach a much wider audience.

Types of assistive technology:

1. Screen Readers. Screen readers translate text and interactive elements on the screen into speech, allowing people with visual impairments to use websites. They can be standalone programs (e.g., NVDA), browser extensions (e.g., Chrome Vox), or built into operating systems and similar software (e.g., Apple iOS Voiceover) [6].

What is important in the code:

- Use semantic HTML markup (<h1>, <nav>, <main>, <button>, etc.).
- Give clear names to buttons and links (aria-label, title).
- Alternative text (alt) for images.

2. Screen Magnifiers. Screen magnification tools provide varying levels of magnification for on-screen content and pointers on a computer, making it easier for people with low or no vision to interact with web pages (such as Windows Magnifier)

What is important in the code:

- Fonts should scale without breaking the layout (responsive design).
- Text/background contrast ≥ 4.5:1 (WCAG 2.1).
- Avoid fixed sizes in px – it is better to use em/rem/%.

3. Eye Trackers. Like alternatives to keyboards, there are also alternatives to mice and pointers. Some people with limited mobility use eye trackers to determine where a person is looking and what they want to click.

What is important in the code/design:

- Provide large interactive areas for eye clicks (minimize small elements).
- Use clear visual focus (highlight the selected element).
- Implement the dwell time option (select an element after fixing your gaze on it for a certain time).
- Minimize dynamic elements that distract or interfere with tracking accuracy.
- Provide a simple navigation hierarchy (no nested menus in several levels).

4. Speech Recognition Software. Speech recognition software allows users to control websites and computers using their voice, converting spoken words into text and commands. Example – Windows Speech Recognition

What is important in the code:

- Use clear button names and form labels so that they can be controlled by voice.
- Ensure that the names are unambiguous ("Submit", not 5 different buttons with the text "OK").

5. Alternative Keyboards or Input Devices. People with mobility or cognitive impairments may rely on alternative keyboards–such as larger or one-handed models, Bluetooth options, or on-screen keyboards–to facilitate interaction with web content.

What is important in the code:

- Full support for keyboard navigation (Tab, Enter, Space).
- Visible focus state.
- Ability to perform all actions without a mouse.

Users with disabilities interact with digital products using special technologies – screen readers, speech recognition programs, screen magnifiers, etc. But for developers and designers, it’s important not only to know about these tools, but also to be able to check whether their product actually works with them. This is where inclusivity auditing tools come in handy.

Digital accessibility testing is not only based on compliance with standards, but also on the use of practical tools that allow you to detect problems before they become critical for users. They can be divided into three broad categories: automated tools, user experience emulators, and integrated development solutions.

Automated tools (such as axe DevTools, WAVE, or Lighthouse) allow you to quickly analyze a web page for compliance with WCAG standards. They detect common errors such as incorrect use of headings, lack of alternative text for images, low contrast, or problems with keyboard navigation. Such solutions are convenient for initial testing, but do not cover all the nuances, as they are able to find only about 30–40% of potential problems.

User experience emulators help you understand how people with disabilities interact with a product. This can be testing with screen readers (NVDA, JAWS, VoiceOver), checking the keyboard instead of the mouse, or using simulators of various types of color blindness (for example, Color Oracle). Such testing is especially important because it allows you to see whether the interface is really convenient, and not just formally "correct".

Integrated development solutions are tools like Pa11y [7] or axe-core CLI, which can be built into the CI/CD process. They automatically check accessibility during each commit or build of the project. This approach allows you to catch errors at the development stage, rather than after the release, which significantly reduces the cost of fixing them.

Using these technologies in combination forms a multi-level control system: automated tests identify basic problems, emulators provide an idea of the real user experience, and integration into the development pipeline ensures that accessibility is checked constantly. Only in this way can you achieve not formal, but true inclusivity.

Accessibility methodologies and recommendations are described in standards such as WCAG, WAI-ARIA, and others. They often become the basis for legal requirements in highly developed countries, and common standards for IT and other companies.

Web Content Accessibility Guidelines (WCAG) aims to make web content accessible to people with various types of disabilities and describes the core principles of receivable, operable, understandable, and robust digital resources, which are called POUR principles. The principle of receivability requires that information be accessible for perception through alternative text for images, subtitles and transcripts for audio and video, and contrast of visual elements. The operable means managed interface by user, including keyboard navigation and interface personalization with the ability to adjust visual effects and components. The principle of understandability requires simple and clear language, consistent navigation, and error reporting through messages and symbols. Finally, a robust web service should provide compatibility with various technologies, such as support for screen readers, speech-to-text and text-to-speech systems, and keyboard navigators and alternative devices. The standard defines the levels of system compliance with the above requirements (A, AA, AAA), where level AA is considered the minimum for government websites in the EU and the USA. The international ISO/IEC 40500:2012 standard officially establishes WCAG 2.0 as a standard for countries and organizations, giving it official status in the international standardization system [8]. But in reality the international standard is WCAG 2.1/2.2, which most countries are guided by.

Accessible Rich Internet Applications (WAI-ARIA) is a set of W3C specifications, which complements HTML to improve the accessibility of dynamic web applications. The main problem is that JavaScript and modern SPAs (React, Angular, Vue) often make it difficult for screen readers to work. Therefore, it is important to adhere to semantics in the code, define the role of elements, their state, and properties [9].

Conclusion. The European standard Accessibility requirements for ICT products and services (EN 301 549), adopted by ETSI (European Telecommunications Standards Institute), is used to harmonize with the European Accessibility Act and the Web Accessibility Directive. It covers a wider range than WCAG, such as websites and mobile applications, software, office documents, hardware devices, e-books and multimedia. And it also contains sections with technical requirements for hardware, software and documentation, as well as accessibility testing methods.

Accessibility principles should also be taken into account through legal aspects. ADA (Americans with Disabilities Act, 1990) prohibits discrimination against people with disabilities in the USA, including in the digital environment. European Accessibility Act (2019/882/EC) is mandatory for electronic services, banking, e-commerce, e-readers, software. Accessible Canada Act (2019) mandates mandatory accessibility for federal agencies and companies. Australia, Japan, Israel, Norway, South Korea have national laws that are mainly based on WCAG.

In Ukraine, regulations for digital accessibility are still developing, but several important steps have been taken. The Law of Ukraine "On the Fundamentals of Social Protection of Persons with Disabilities" provides general rights and protections for people with disabilities. Additional provisions can be found in the Law "On Information" and the Law "On Electronic Communications", which establish the right to access information resources and communication systems. On the standards level, Ukraine has introduced DSTU ISO/IEC 40500:2015, which is the national adoption of WCAG 2.0. Accessibility is also included in the Digital Transformation Strategies (2021–2025) as one of the policy components. In practice, major government services, such as the Diia portal, are beginning to apply the requirements of WCAG 2.1, moving toward closer alignment with international accessibility standards.

References

1. Давиденко Г. Цифрова інклюзія та доступність: соціальна діджиталізація: монографія. Вінниця: ТВОРИ, 2023. 240 с. URL: cifrova-inklyuziya-ta-dostupnist-socialna-didzhitalizaciya.pdf.

2. Blindness and vision impairment. URL: https://www.who.int/news-room/fact-sheets/detail/blindness-and-visual-impairment.

3. Doyle, N., & McDowall, A. (2022). Diamond in the rough? An "empty review" of research into "neurodiversity" and a road map for developing the inclusion agenda. *Equality, Diversity and Inclusion: An International Journal,* 41(3): 352–382. https://doi.org/10.1108/EDI-06-2020-0172. URL: https://www.emerald.com/edi/article/41/3/352/289942/Diamond-in-the-rough-An-empty-review-of-research.

4. Цифрова адженда України – 2020 («Цифровий порядок денний» – 2020): концептуальні засади (версія 1.0). Першочергові сфери, ініціативи, проекти «цифровізації» України до 2020 року. *HITECH office*. Грудень, 2016. URL: https://ucci.org.ua/uploads/files/58e78ee3c3922.pdf.

5. Introduction to Lighthouse. URL: https://developer.chrome.com/docs/lighthouse/overview.

6. Level Access (2024). Assistive Technology in Digital Inclusion. URL: https://www.levelaccess.com/blog/assistive-technology.

7. Pa11y. URL: https://pa11y.org/

8. ISO (2012). ISO/IEC 40500:2012 Information technology – W3C Web Content Accessibility Guidelines (WCAG) 2.0. Geneva: *International Organization for Standardization*. URL: https://www.iso.org/standard/58625.html.

9. W3C (2023). Accessible Rich Internet Applications (WAI-ARIA) 1.2. *World Wide Web Consortium*. URL: https://www.w3.org/TR/wai-aria-1.2/