An Approach to Validating the Accessibility of Dynamic Web Pages

Hristo Hristov, Gencho Stoitsov, Milen Bliznakov, Stilian Minchev

Plovdiv University "Paisii Hilendarski", 236, Bulgaria Blvd, Plovdiv, Bulgaria

***Abstract*–The purpose of the publication is to present a web page validation approach that is part of a method for designing accessible web content. The validation approach is built on standards, principles, guidelines, success criteria and techniques for web content validation. The main standards to which the approach adheres are Web Content Accessibility Guidelines and Technical Specifications for Accessible Rich Internet Applications. The publication discusses the understanding of dynamic web content, dynamic web content design and implementation method and web content validation approach.**

***Keywords* – accessibility validation, dynamic web content, validation approach.**

1. Introduction

According to the Web Accessibility Initiative (WAI) "Web accessibility" means that web content are developed so that people with disabilities can use them. Web accessibility encompasses all disabilities that affect access to the Web, including: auditory, cognitive, neurological, physical, speech, visual [1], [2]. WAI as part of The World Wide Web Consortium (W3C) offers, supports and develops, standards, methods for accessibility, etc.

DOI: 10.18421/TEM121-30
https://doi.org/10.18421/TEM121-30

Corresponding author: Hristo Hristov,
Plovdiv University "Paisii Hilendarski", 236, Bulgaria Blvd, Plovdiv, Bulgaria.
Email: hth@uni-plovdiv.bg

Received: 12 September 2022.
Revised: 04 October 2022.
Accepted: 31 October 2022.
Published: 27 February 2023.

Accessible web page, layout or web content means that it can be clear, understandable and easy to perceive by a large number of users with different types and degrees of difficulty, such as reduced vision, colour blindness, motor difficulties, etc. Creating accessible web content is a challenge when developing dynamic web pages [3]. In the article, the concept of "dynamic" is associated - firstly with the automated generation of web content upon the occurrence of an event for the web page, and secondly "dynamic" refers to the responsiveness and adaptability of the web page's appearance and its content. The problem investigated in the article is the validation of the accessibility of dynamic web pages, their layout and content in the network of Paisii Hilendarski Plovdiv University. Validation of dynamic web content is a practice that, in modern agile and iterative software development methodologies, is most closely associated with the testing phase of the software process life cycle. The purpose of the publication is to present a web page validation approach that is part of a method for designing and implementing accessible web content for people with disabilities. In order to achieve the goal of the article, a model of the validation approach was created, which was tested when updating the web content of the university accessible through the main domain *https://uni-plovdiv.bg/* and many of its subdomains. The active web pages of Paisii Hilendarski Plovdiv University, which are indexed in the Google search engine, are more than 370,000, and approximately 2,000 of them generate dynamic web content through REST service and Vue framework.

WAI to W3C provides a set of guidelines that are internationally recognized as the standard for web accessibility [2], [4]. In this article, we consider the validation of web pages and content through the Web Content Accessibility Guidelines (WCAG), taking into account some basic provisions of the Accessible Rich Internet Applications (ARIA). WCAG is an international standard, the current versions of which are 2.1 – official and 2.2 – working draft [5].

WCAG 2.1 has several layers of guidance including overall principles, general guidelines, testable success criteria and a rich collection of

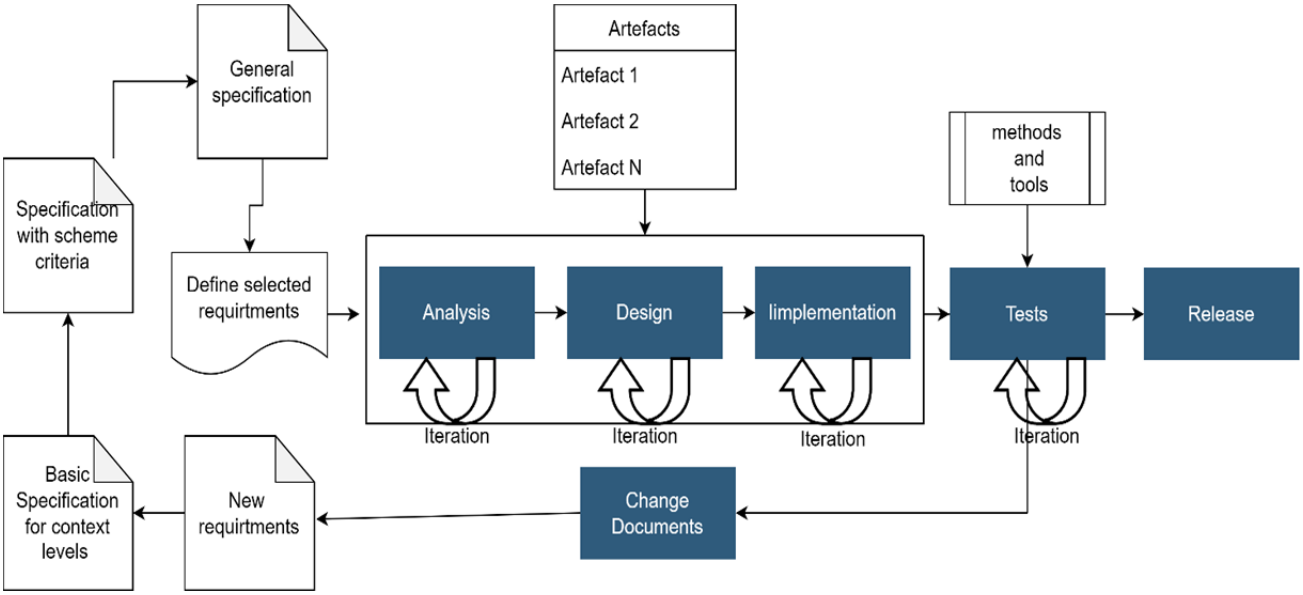

Figure 1. Accessibility development method

sufficient techniques, advisory techniques, and documented common failures with examples, resource links and code.

The *success criteria* are what determine "conformance" to WCAG. The means through which success criteria are implemented are Sufficient, Advisory and Failures Techniques, but also designers and software developers can develop different techniques.

2. Dynamic Web Pages

Dynamic web pages have web content developed with HTML, CSS, JavaScript, AJAX, XML, jQuery, PHP and related technologies [5], [7], [8], [9]. The dynamism of web pages and applications is related to the change of various properties and states of the web pages. Most often, the dynamism of the web page or application is associated with the automatic generation or modelling of web content, layout and structure upon the occurrence of a certain event or user action [10], [11]. The user can act as a web content author through web page authoring tools like HTML editors, Content Management Systems (CMS), Learning Management Systems (LMS), courseware tools, content aggregators, etc [12], [13], [14]. He can be a user of a social network like Meta, Twitter, Instagram, etc. where he creates web content on his profile, page or web store. In such cases, dynamic web content is generated using authoring tools. Similarly, in the thousands of web pages on *https://uni-plovdiv.bg/*, through authoring tools, an administrator or developer with a certain level of access can generate web content for a corresponding subdomain. Another common aspect of dynamism concerns the adaptability and responsiveness of the layout and visual components of the web page according to the device on which they are displayed. This is known in literature and practice as adaptive and responsive web page layout [15], [16], [17]. A dynamic web page layout is a combination of: content builders, visual presentation elements, and script elements for dynamically changing content. In this article, the subject of analysis and research are the two aspects of dynamism described above. Specifically, we investigate accessibility validation of dynamic web content by adhering to WCAG 2.1 [18] compliant standards and tools.

3. Research Approach

The web page and content validation approach is part of a method for designing and implementing dynamic web content (Fig. 1), which was used in updating the web infrastructure and content of *https://uni-plovdiv.bg/* during the period from June 2020 to June 2022. The life cycle of the method represented by its model is covered in a separate publication. It is aligned with modern agile and iterative practices in software processes. Its application runs parallel to the implementation of the software process according to a certain methodology. The accessibility development method is independent of a particular software process, but needs to be adapted to the methodology used in software development. The accessibility validation approach in Fig. 2, is part of the accessibility design and implementation method in Fig. 1. It serves to guide developers, designers or QA specialists during the implementation of the software process. It serves to guide developers, designers or QA specialists during the implementation of the software process. The validation approach is most closely related to the testing phase, but it also covers planning, requirements gathering, design, and implementation. It is a sequence of actions that run parallel to other activities of the software process.

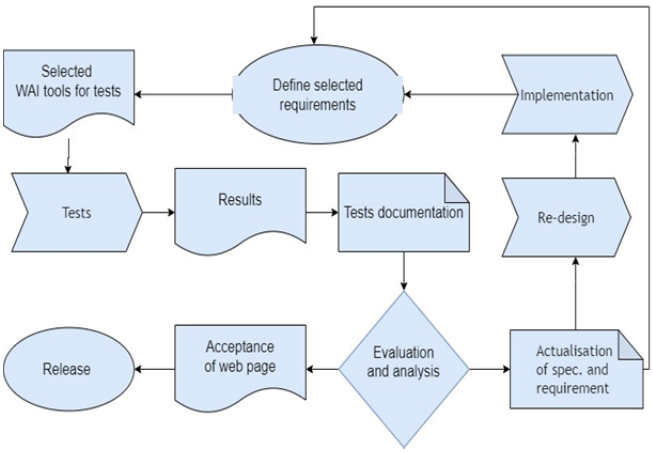

Figure 2. Validation approach model

The validation approach lifecycle model in Fig. 2, is composed of the steps: defining accessibility requirements, selecting inspection tools, inspecting with tools to check quality and level of accessibility, documenting results, analysing and evaluating inspection results, updating requirements and specification, page re-design and page implementation. The first stage of the validation approach is the selection of requirements from those defined in the development specification, followed by the selection of testing tools, performing inspection and documenting the results. Depending on the result of tests with one or more accessibility assessment tools, it is possible that the web page satisfies or fails the requirement specification. In cases where there are satisfactory results for a certain web page, its status becomes "acceptance of web page". The lifecycle of the validation approach for one iteration ends with the "acceptance of web page" status for a particular inspected web page. When this process is completed for all web site pages to be evaluated, then the application of the validation approach ends. In such a situation, the web development has reached the intended level of accessibility and the site with its applications and web pages is ready for operation (release). When the inspection results are not satisfactory for the web page, as a result of their analysis and evaluation, what is taken into account is which principles, guidelines, categories and techniques have been tested. Which of them led to success or failure. It is analysed whether the success criteria are satisfied – "Level A", "Level AA" or "Level AAA" according to WCAG. In cases of failure, barrier, or dissatisfaction with the test result, the recommendations of the test tool are first analysed, then it is considered whether there are alternative techniques by which to correct the development source code. It is often necessary to add requirements from several success criteria of a current guideline or another WCAG principle to the software specification for a particular page or its functionality with corresponding source code patching techniques. In some cases, developers have to create their own solution. Situations where an effective solution cannot be found are not exceptions. After the stage of evaluation and analysis of the results, in the case where there are no satisfactory results, it is possible to move on to the selection of a new test instrument. In any case, when there are no satisfactory results, it goes through stages of updating the specification with required re-design and implementation.

WAI, in addition to standards and guidelines, has created methodologies for accessibility planning, development, design, testing and evaluation, including reporting templates. [19], [20]. WIA documentation is voluminous and formalised. It must be adapted to the specific software process and in particular to the accessibility validation process. However, no tool alone can determine if a site meets accessibility standards. Knowledgeable human evaluation is required to determine if a site is accessible [20], [21]. For these reasons, the authors of the article have developed their own validation approach. The validation approach for accessibility of dynamic web pages and content has been tested with pages in a number of subdomains of *https://uni-plovdiv.bg/*. In the process of work, experiments were conducted with various information resources and data. In the course of conducting inspections, after making the necessary corrections and changes by applying a method for designing and implementing accessible web content to the source code level, the appropriateness and effectiveness of the considered validation approach has been confirmed. Through it, a number of errors, barriers and failures have been removed, some of which should be presented.

4. Results of Accessibility Inspection

To ensure the accessibility of web pages, it is recommended to use different validation tools [22], [23]. WAI develops and promotes accessibility design and testing resources and tools. The initiative maintains a list of tools that meet specific needs [24]. Tools are software programs, online services, or browser add-ons that help design and evaluate the accessibility of web content. The list of WAI tools [6] is not exhaustive. There are tools from corporations, communities, university research groups, etc. that also provide quality services. During the renovation of part of the web infrastructure of Paisii Hilendarski Plovdiv University through the accessibility validation approach of the method for design and implementation of dynamic web content, various tools were used. The publication presents validation results with the User-Agent Switcher, Responsive Design Checker, Google Chrome

Inspector, FireBug for Firefox, ax DevTools, WAVE Evaluation Tool, and Accessibility Checker tools. The results obtained are analysed and documented, using them for application in subsequent iterations of the method for designing and implementing dynamic web content.

It should be noted that the tools cannot find all accessibility obstacles and limitations. User test results are based on a limited set of trials. The evaluated pages may contain partitions that are not reported by any of the tools.

The web pages from subdomains of https://uni-plovdiv.bg/, for which an accessibility validation inspection was performed, are more than 2000, and what they have in common is that the generation of dynamite content is implemented with the Vue framework and a REST service.

4.1. Inspection with the User-Agent Switcher

User-Agent Switcher is a Chrome extension that simulates layout resolutions for different mobile and desktop browsers and their versions on different operating systems without the need to install them. The Chrome extension was used as part of the accessibility validation approach of web pages on the Plovdiv University web network for simulations with Android KitKat, all versions of Chrome, iOS (iPhone 6, iPad), Firefox from version 33 for Windows and Mac, Opera 12, Safari from version 7 and Windows Phone version 8. For the above listed operating systems, browsers and their versions with User-Agent Switcher, inspections were made with resolutions from 320px to 1200px with portrait and landscape screen layout. For the parameters thus specified, during the tests, no obstacles and changes to the implemented accessibility of the layout were found during the dynamic generation of the content in the web pages of the web infrastructure of the university.

4.2. Inspection with Responsive Design Checker, Google Chrome Inspector and FireBug for Firefox

Responsive Design Checker, Google Chrome Inspector and FireBug for Firefox are tools that can be used to inspect responsive design. Through them, developers and users can learn how well web pages adapt to different user devices and screen parameters. In Table 1, a conditional distinction is made between notebook, desktop monitor, tablet and smartphone devices, for which the Responsive Design Checker, Google Chrome Inspector and FireBug for Firefox tools were used to simulate the layouts of the web pages of the departments of the respective faculties at the university. The simulations performed are for resolutions in the range from 320px to 1400px with portrait and landscape screen placement. In these situations, accessibility is inspected when web content is dynamically generating. For the web pages of the departments, the scientific title, the academic title, the names of the members of the department, as well as the arrangement of the members according to certain criteria, are dynamically generated in the page layout. For this type of page (Fig. 3), due to the small volume of information, an adaptation approach has been adopted through the so-called horizontal "flattening" and changing the font size. During the inspections made, no deviations were found for the layouts of different devices and resolutions.

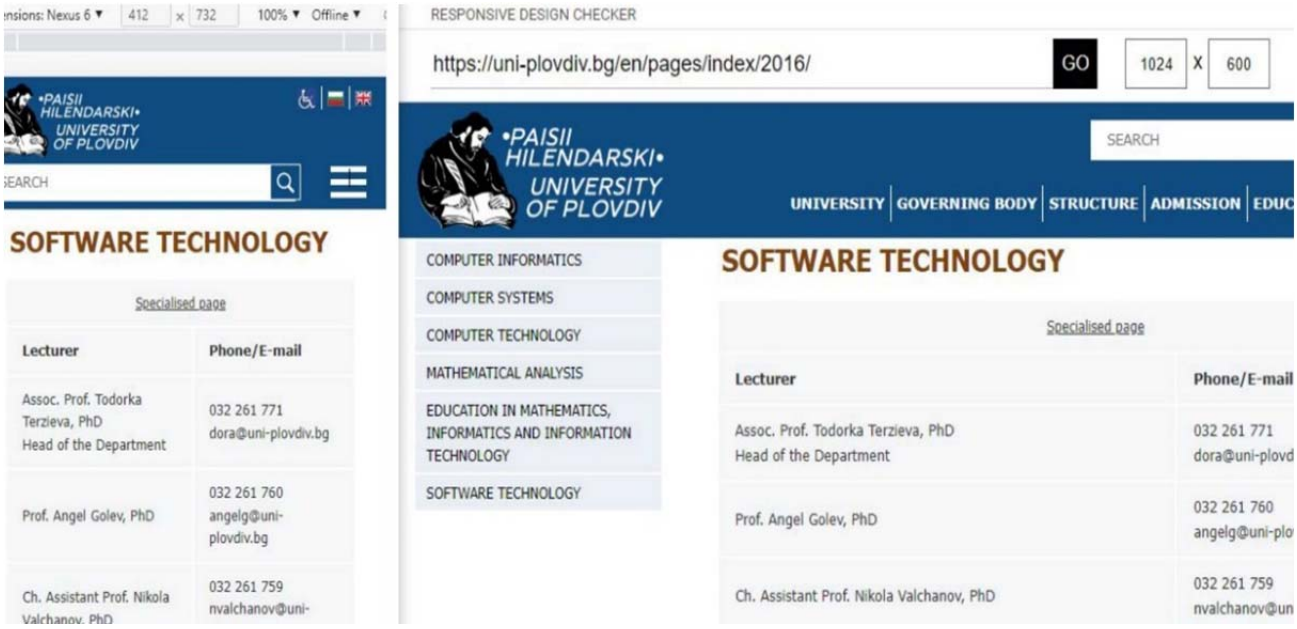

Figure 3. Inspection with Responsive Design Checker, Google Chrome Inspector and FireBug for Firefox

Table 1. Simulated devices and resolutions

Device	***Type/Resolution***
Device	From 10 to 24 inches
Tablet	Apple iPad (Mini, Retina, Pro), Amazon Kindle Fire, Asus E1000, Nexus (7, 9), Samsung Galaxy Tab 10
Smartphones	iPhone (3, 4, 4s, 5, 5s, 6, 6s, 7, 6s Plus, 7 Plus), Samsung Galaxy (S5, S6, S7), Sony Xperia (Z2, Z3), Google Pixel, Nexus (4, 5, 6), etc.

4.3. Inspection with axe DevTools

The "axe DevTools" is a Chrome extension and Accessibility Checker for Developers, Testers, and Designers. The browser extension automates individual page and component-level testing. Axe DevTools gives the ability to zoom in on and test specific parts of a page relevant to your immediate project. In tests with axe DevTools on more than 500 web pages of the university's main domain *https://uni-plovdiv.bg*, as shown in Fig. 4, several hundred WCAG 2.1 non-conformities related to heading levels, colour contrast, frames and input, zooming and scaling of web content, etc.

In several successive integrations of the application of the validation approach through the method of designing and implementing accessible dynamic web content, a large number of inconsistencies were prevented.

4.4. Inspection with Web Accessibility Evaluation Tool

Web Accessibility Evaluation Tool (WAVE) is a tool for evaluating the accessibility of web content for people with disabilities. It identifies errors and barriers according to WCAG accessibility guidelines. WAVE displays errors on the page by presenting and marking them with appropriate icons at the specific locations, provides additional information about detected errors, and provides user-accessible functionalities that are relative to WCAG. The tool provides the ability to inspect pages with CSS styles enabled and disables. After completing a test with WAVE, a menu with six sections is displayed: Errors, Contrast errors, Signals, Properties, Structural elements and ARIA elements.

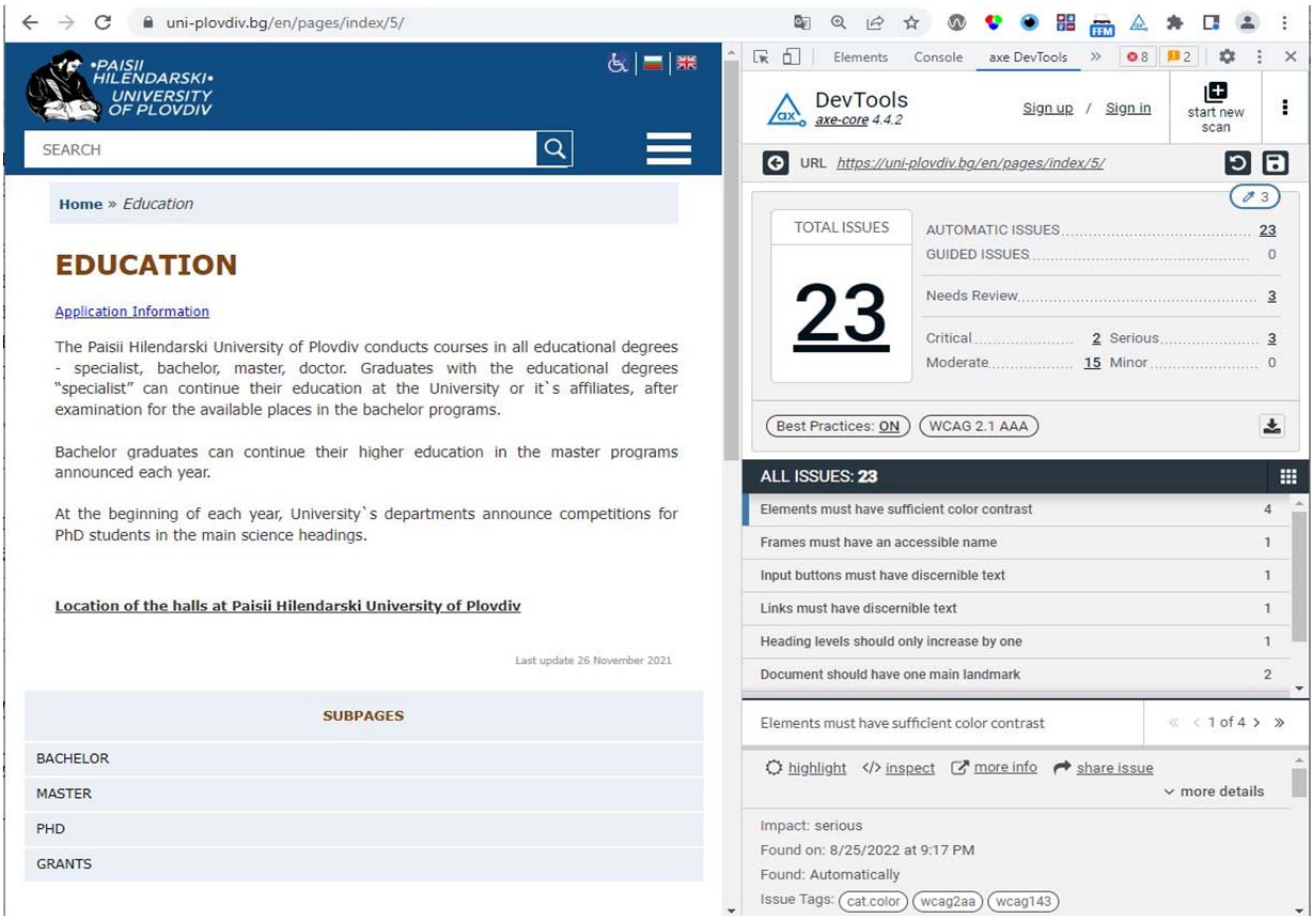

Figure 4. Inspection with axe DevTools

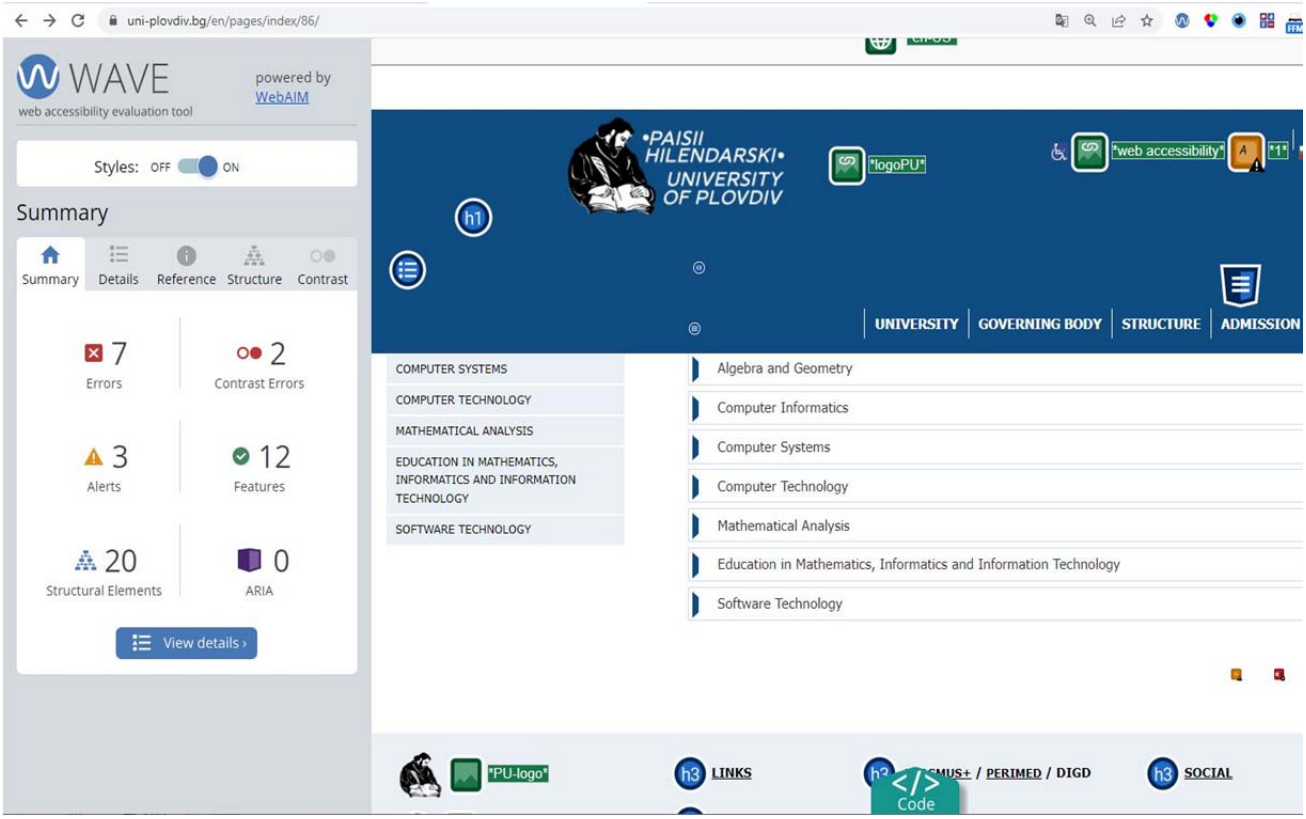

Figure. 5. Inspection with WAVE when styles are enabled

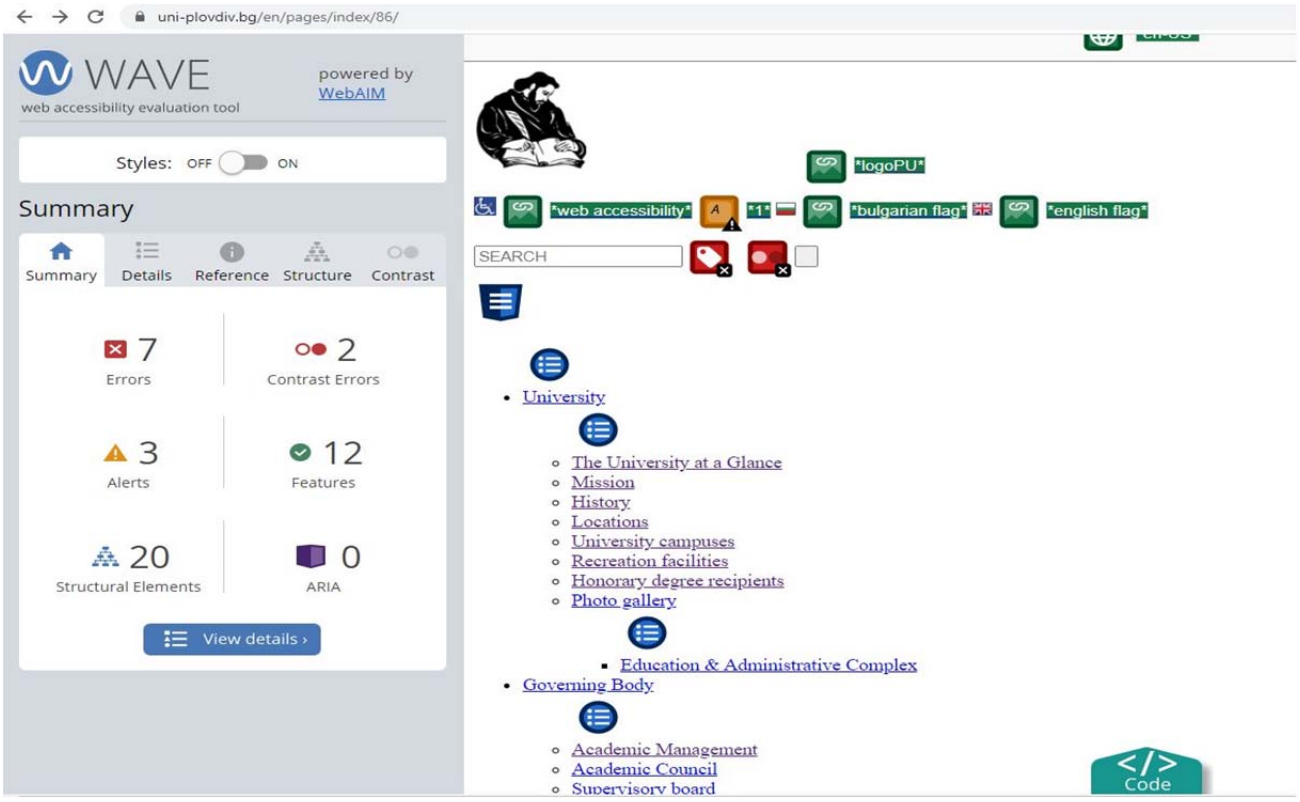

Figure 6. Inspection with WAVE when styles are disabled

Inspection of 400 web pages from the main domain https://uni-plovdiv.bg/ and its subdomains, as shown in Fig. 5 and Fig. 6, revealed 117 single-type errors, 94 contrast errors, 192 signals, 118 properties, 230 structural elements and 198 defined ARIA attributes that need to be addressed. The results of applying the accessibility validation approach with WAVE were used in several iterative cycles of the method to design and implement accessible dynamic web content, and most of the errors, barriers and inconsistencies were removed.

4.5. Inspection with Accessibility Checker

Accessibility Checker is an accessibility flaw identification tool that is among the most preferred due to its efficiency and functionality.

The advantage of Accessibility Checker over alternative tools is that it scans the source code at the website level, not just an individual webpage or its component, and also provides an analysis of barriers and flaws in the source code. Inspection with the tool provides an overview of the audit - status, assessment and results (Fig. 7) and sections for Critical Issues, Passed Audits and Required Manual Audits.

Figure 7. Audit with Accessibility Checker

The three sections Critical Issues, Passed Audits and Required Manual Audits are composed of two categories – type of problem and technical analysis (Fig. 8).

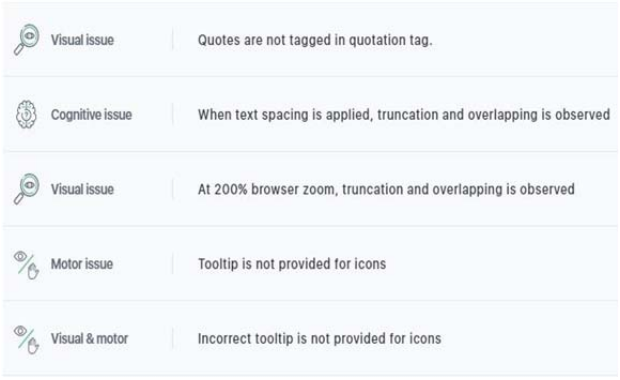

Figure 8. Accessibility Checker categories

Convenience for developers of accessible web content is that the tool's technical analysis follows scenarios of: Identification of a problem category, a technical description of the problem, an explanation to whom the problem applies, and technical analyses and recommendations on how to implement the solution (Fig. 9). Accessibility Checker was used by the development team when updating the website of Paisii Hilendarski Plovdiv University for the web pages in which dynamic content is generated using the Vue framework and a REST service for generalised auditing. The generalised audit is a web site inspection applied to the accessibility validation approach at the beginning of the iteration of the method for designing and implementing accessible dynamic web content. Identified errors from a generalised Accessibility Checker audit are matched with inspections from an additional web page or web page component level accessibility tool, then a solution is implemented in the source code.

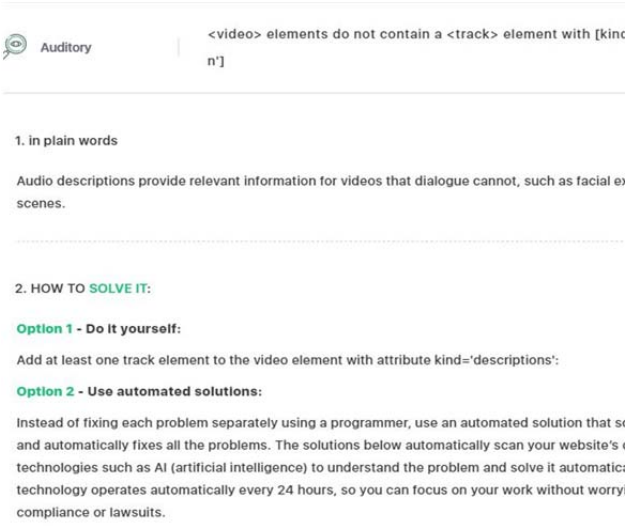

Figure 9. Accessibility Checker script to implement accessibility when identifying a problem

5. Conclusion

The article considered the development of an accessibility validation approach to dynamic web pages, which is part of a method of designing and implementing accessible dynamic web content. The approach is built on WCAG and ARIA principles, guidelines, success criteria, techniques, accessibility inspection tools and decision recommendations. The objective of the publication to present a web page validation approach as part of the software process in building web sites and applications has been achieved. The validation approach was presented through its model and description. The approbation of the approach was made during the accessibility validation of dynamic web pages, which were designed and implemented with a REST service and Vue framework during the renewal of the web infrastructure of Paisii Hilendarski Plovdiv University. Based on the discussion in the publication, it can be concluded that the accessibility validation approach of dynamic web pages and content is universal in terms of its applicability and compatibility with modern software development methodologies and processes.

Acknowledgements

The work is funded by the MU21-FMI-004 and MU21-FMI-011 projects of the Fund "Scientific research" at the University of Plovdiv "Paisii Hilendarski".

References

[1]. Henry, S. L. (2006). Understanding Web Accessibility. In: J. Thatcher, M. Burks, C. Heilmann, S. Henry, A. Kirkpatrick, P. Lauke, B. Lawson, B. Regan, R. Rutter, M. Urban, C. Waddell (Eds.), *Web Accessibility*. Apress Berkeley, CA. pp. 1-51.

[2]. World Wide Web Consortium. (2022). Making the Web Accessible. Retrieved from: https://www.w3.org/WAI/ [accessed: 10 August 2022].

[3]. Freire, A. P., & de Mattos Fortes, R. P. (2005, May). Automatic accessibility evaluation of dynamic web pages generated through XSLT. In *Proceedings of the 2005 International Cross-Disciplinary Workshop on Web Accessibility (W4A)* (pp. 81-84). https://doi.org/10.1145/1061811.1061826

[4]. Amado-Salvatierra, H. R., Hernández, R., & Hilera, J. R. (2012). Implementation of accessibility standards in the process of course design in virtual learning environments. *Procedia Computer Science*, *14*, 363-370.

[5]. World Wide Web Consortium. (2022). What's New in WCAG 2.2 Working Draft. Retrieved from: https://www.w3.org/WAI/standards-guidelines/wcag/new-in-22/, [accessed: 13 August 2022].

[6]. World Wide Web Consortium.(2022). WAI tools. Retrieved from: https://www.w3.org/WAI/ER/tools/ [accessed: 24 August 2022].

[7]. Heinle, N., & Peña, B. (2002). *Designing with Javascript: creating dynamic Web pages*. " O'Reilly Media, Inc.".

[8]. Tatroe, K. & MacIntyre, P. (2020). *Programming PHP: Creating dynamic web pages*. 4th edition CA, USA: O'Reilly Media Sebastopol.

[9]. Nixon, R. (2012). *Learning PHP, MySQL, JavaScript, and CSS: A step-by-step guide to creating dynamic websites*. " O'Reilly Media, Inc.".

[10]. Ravi, J., Yu, Z., & Shi, W. (2009). A survey on dynamic Web content generation and delivery techniques. *Journal of Network and Computer Applications*, *32*(5), 943-960.

[11]. Gamecho, B., Minón, R., Aizpurua, A., Cearreta, I., Arrue, M., Garay-Vitoria, N., & Abascal, J. (2015). Automatic generation of tailored accessible user interfaces for ubiquitous services. *IEEE Transactions on Human-Machine Systems*, *45*(5), 612-623.

[12]. Chittò, P., Baez, M., Daniel, F., & Benatallah, B. (2020, November). Automatic generation of chatbots for conversational web browsing. In *International Conference on Conceptual Modeling* (pp. 239-249). Springer, Cham.

[13]. Van Dijk, J. A. (2005). *The deepening divide: Inequality in the information society*. Sage publications.

[14]. Brake, D. R. (2014). Are we all online content creators now? Web 2.0 and digital divides. *Journal of Computer-Mediated Communication*, *19*(3), 591-609. https://doi.org/10.1111/jcc4.12042

[15]. Dubey, A., De, P., Dey, K., Mittal, S., Agarwal, V., Chetlur, M., & Mukherjea, S. (2013, May). Co-operative content adaptation framework: Satisfying consumer and content creator in resource constrained browsing. In *Proceedings of the 22Nd International Conference on World Wide Web* (pp. 221-222). https://doi.org/10.1145/2487788.2487901

[16]. Li, N., & Zhang, B. (2019, November). The design and implementation of responsive web page based on HTML5 and CSS3. In *2019 International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI)* (pp. 373-376). IEEE.

[17]. Mohorovičić, S. (2013, May). Implementing responsive web design for enhanced web presence. In *2013 36th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)* (pp. 1206-1210). IEEE.

[18]. Chochev, N. & Hristov, H. (2022). Design Techniques and Practices of Grid Layouts and Content of Web Pages. *Scientific Works of the Union of Scientists in Bulgaria - Plovdiv. Series C. Technics and Technologies.* Vol. XX, pp. 99-103.

[19]. World Wide Web Consortium. (2020). Web Content Accessibility Guidelines (WCAG) 2.1. Retrieved from: https://www.w3.org/TR/WCAG21/ [accessed: 05 September 2022].

[20]. World Wide Web Consortium. (2022). Template for Accessibility Evaluation Reports. Retrieved from: https://www.w3.org/WAI/test-evaluate/report-template/ [accessed: 07 September 2022].

[21]. Brajnik, G., Yesilada, Y., & Harper, S. (2012). Is accessibility conformance an elusive property? A study of validity and reliability of WCAG 2.0. *ACM Transactions on Accessible Computing (TACCESS)*, *4*(2), 1-28. https://doi.org/10.1145/2141943.2141946

[22]. Thatcher, J. (2006). Accessibility Testing. In: J. Thatcher, M. Burks, C. Heilmann, S. Henry, A. Kirkpatrick, P. Lauke, B. Lawson, B. Regan, R. Rutter, M. Urban, C. Waddell (Eds.), *Web Accessibility*. Apress Berkeley, CA. pp. 407–457.

[23]. Kurt, S. (2017). Accessibility of Turkish university Web sites. *Universal Access in the Information Society*, *16*(2), 505-515. https://doi.org/10.1007/s10209-016-0468-x

[24]. Akgül, Y. (2016, June). Quality evaluation of E-government websites of Turkey. In *2016 11th Iberian conference on information systems and technologies (CISTI)* (pp. 1-7). IEEE.