ENERGY EFFICIENCY OF THE WEB DESIGN BASED ON WEBSITE PERFORMANCE MEASUREMENTS

Yoanna Klinchova, Emel Shaban, Nikolay Kasakliev, Elena Somova*

University of Plovdiv "Paisii Hilendarski", Plovdiv
Bulgaria

* Corresponding Author, e-mail: kasakliev@uni-plovdiv.bg

Abstract: The article is oriented towards studying technical sustainability of UX/UI design, exploring the approaches to website creation: responsive, adaptive and mobile-first design. Measurements were made for the performance of the different web page elements that make up a web page, testing the various methods for creating an element, as well as structuring the web page as a whole. Experimental measurements were made for responsive and adaptive designs and for 3 types of mobile devices: smartphone, tablet and laptop. Based on the analysis of the collected measurement results, conclusions are drawn and specific recommendations are given for optimization in creating effective web page designs for different design approaches.

Key words: mobile UX/UI, energy efficiency, sustainable UX/UI design.

1. INTRODUCTION

As in many other fields, in software development, sustainability is becoming an increasingly important goal, guided by the 17 Sustainable Development Goals of the United Nations [1]. Models such as the Green software development model, which affects how software can be developed in a sustainable and environmentally friendly way [2].

Leading companies in the software industry in recent years have been trying to popularize this model. Companies like Google and Microsoft recognize that this industry plays a significant role in energy and resource consumption and are trying to incorporate this model into their business strategies, focusing on energy efficiency and sustainability [3].

The main goal of this development model is to minimize the negative impact on the environment, while creating effective, aesthetically pleasing and long-lasting solutions. Basic principles in development include [4]: energy efficiency and use of renewable sources; selection of environmentally friendly materials, including recycled and biodegradable; design for durability and minimization of waste; and social responsibility and improving the quality of life.

One of the most important aspects of sustainability is energy efficiency, not only because of reducing consumption, but also because the efficiency in energy management will also enhance the security of the power system [5]. Many authors [6] define software efficiency as an important non-functional characteristic that determines the quality of software.

Although sustainability has different aspects (with the basic pillars being environmental, social and economic), in this article we will focus on technical sustainability, achieved by reducing energy consumption. The main objective of the work is to investigate the energy efficiency of creating web user experience/user interface (UX/UI). The study aims to analyse the loading speeds of different web elements (as faster loading results in less energy consumption), implemented in different ways, by conducting multiple measurements. The measurements are done in simplified versions of websites designed through adaptive and responsive approaches.

In Section 2 of the article, an overview of research on achieving sustainability in UX/UI design is made. Section 3 provides an overview of the widespread approaches of designing UX/UI. Section 4 shows the methodology of the study and the results of the experiments on the energy efficiency of the different elements in UX/UI design. The loading speed of different implementations of web elements is measured for 3 types of screens (laptop, tablet and smartphone). In Section 5, the results obtained are discussed, as well as specific recommendations for optimization and creation of effective web designs are made. The paper ends with a conclusion.

2. SUSTAINABLE UX/UI – STATE OF THE ART

Creating sustainable websites with minimal negative consequences and maximizing positive contributions to the global world is a priority of sustainable web design. Efforts to achieve sustainability are put not only into the software product itself, but also into the processes of planning, coding, hosting, and maintaining the website [7].

Without a doubt, one of the important stages in the development of sustainable software is that its UI/UX design has a minimal effect on the environment, which can be implemented in two directions. Sustainable UX/UI design on the one hand aims at optimizing code, reducing unnecessary functionalities and using energy-efficient servers, and on the other hand, it focuses on creating digital products that are inclusive, efficient and responsible for the environment and society. Authors of [4] present the specifics of sustainable design such as: humanity-oriented design (satisfying consumer needs for better well-being and satisfaction); resource efficiency (minimizing energy and computing costs); durability (designing digital products with a long life); ethical data practices (ensuring transparency, protection of personal information and security for users); social justice (ensuring access to digital technologies for all, incl. vulnerable groups); environmental responsibility (reducing the carbon footprint through environmental technologies and optimized energy consumption); and economic viability (balancing financial goals with sustainable practices).

When developing a sustainable UX/UI design, the type of application (web or native) and the device on which it will be used should be considered. Considering that according to [8], there are over 7.2 billion smartphones in the world in 2025, the mobile user is becoming a key factor in UX/UI design. The study of various sources outlines the following main factors and principles for sustainable mobile UX/UI design in the direction of technical sustainability, which is the goal of this study.

Energy-efficient interfaces. Different approaches and techniques can be used to create an energy-efficient user interface (UI). For example, [9] found that scrolling using the so-called. Screen buttons, replacing long lists of data with tabs, or input caching can increase battery lifetime by up to 20%. Another study [10] found that the development of multiplatform mobile applications and the use of Hybrid UIComponents (native UI components and wrap a WebView inside, for displaying the content) can also improve energy efficiency. In this case, the Dialog and Sheet components have the potential to reduce consumption by an average of 38% and 54%, respectively.

Minimalistic interfaces. Minimalism is a strategy that aims to remove redundant elements from the UI by making the most essential elements easily accessible to the user. According to [11], minimalist design principles are particularly impactful for mobile app design, the most successful of which attempts to minimize the amount of user input required to provide users with the information they are looking for. The key features of minimalist design according to [12] are: design is focused on the primary content; using flat design elements; avoids unnecessary graphic elements; and uses substantial amount of negative space. To achieve a minimalist UI, it is also necessary to prioritize the functions of the UI and the content that users need for flowless user-system communication, which will increase user happiness [13].

Faster loading times. Faster loading times are very important for applications' CO2 Emissions Reduction and they result from using fewer images, scripts, and animations, which is advantageous for both users and the environment [14]. Tools like Google PageSpeed Insight are used to assess the loading speed of a website, provide a score, and offer recommendations for speeding up the site without negatively impacting the user experience [15].

Sustainable typography. Typography is often defined as design to facilitate reading and comprehension and is an essential part of visual design [16]. [17] identifies good practices that include: using built-in system fonts instead of external fonts; using the WOFF2 format for external fonts; and SVG icons over icon fonts, to reduce server requests and improve loading times.

Sustainable colour palettes. The relationship between fastness and colours has been well researched, as different colours have different power consumption, especially on OLED screens [19]. A study by [20] shows that the most energy-efficient colour is black, followed by red, green, blue and white, which consumes the most energy.

Reducing and optimizing images. Using unnecessary or particularly large images can be not very effective in terms of the weight of a web page Page Weight. [15] discusses the importance of size, format, quality, colour, contrast, and motion, which are all important considerations for optimizing files and vectors for energy efficiency. Research by [21] shows that the WebP and HEIF formats have good compression efficiency, especially at the low bit rates, to make the download of images from the web as quick as possible.

Design approach. By providing access to websites from different devices: desktops, tablets, smartphones or laptops, the need to create dedicated mobile applications or dedicated apps is avoided, which can reduce the use of resources and reduce the impact on the environment. [22]. Sustainable design often uses techniques such as responsive design to optimize resource efficiency across different devices and screen sizes [23]. This ensures consistent performance while reducing unnecessary resource usage. On the other hand, adaptive design can also improve sustainability, especially for large web applications, since only the appropriate UI design and resources will be loaded on the respective device.

3. UI/UX DESIGN APPROACHES

Adaptive design uses predefined different designs for different resolutions, allowing for better performance as only the necessary resources for the respective device are loaded. This provides a personalized user experience for different screens and leads to better optimization. The disadvantages include higher complexity in maintenance, as it works with multiple versions of the design, more development work, and limited flexibility on new, non-standard resolutions [25].

Responsive design provides flexibility, with the site automatically adapting to different screen sizes using CSS media queries. This makes maintenance easier, reduces development costs, and improves SEO, since, for example, Google prefers mobile-optimized sites [24]. In addition, the user experience is better as the site looks good on all devices. However, responsive design can slow downloading time because all resources are loaded regardless of the device. It is also more difficult to optimize the design for specific screens, and development requires more testing at different breakpoints [25]. There are two approaches to develop a responsive design: starting with the largest (computer) screens first, then gradually reducing the layout for smaller screens and the reverse approach (mobile-first) – from the smallest screens progressively enhancing to the larger ones.

Mobile-first design focuses on mobile first, which ensures better performance, better SEO, and optimized content for mobile users. This approach reduces unnecessary elements and makes the website faster. However, it can make it difficult to adapt to complex desktop designs, requires more attention from designers and developers, and can limit the possibilities for larger screens in the early stages of development.

4. STUDY OF THE SUSTAINABILITY OF WEBSITE'S ELEMENTS

4.1. Methodology of the study

The empirical study aims to analyse the loading speeds of web elements, implemented in different ways, by conducting multiple measurements. Simplified versions of websites have been created through adaptive and responsive designs in which measurements are made – each web element with particular realization is experimented in each design for each screen size.

The tests are done independently by 2 people on 2 different computer configurations (OS: Windows 10 and 11, CPU: AMD Ryzen 7 on both; 8 GB and 16 GB of RAM, and ISP internet speed ~ 60 Mbps and 250 Mbps) with different test websites that contain at the time of the experiment the item being tested. One website is used for the actual **test experiment** and the other for a **control experiment**. To ensure the reliability of the results, the tests are carried out under the same conditions. Each of the two experiments executes a test session at approximately the same time of day (in a short time interval) and in the same working environment. Each test session includes 10 separate measurements, and the obtained values are summarized and presented with their arithmetic mean. The same test session for an item is run as a control on the other computer, with the other website.

The same measurements are done for all different realizations of one web element (e.g. text with serif font and text with sans-serif font). Based on arithmetic means for different realizations of one web element can be concluded, which is the best way (the fastest in loading) for implementation of the element and this conclusion is done separately for both experiments. If the conclusions obtained are the same in the test and control experiments, then the results are considered significant, otherwise the experiments are repeated, and an in-depth analysis of the experimental conditions is made. Several of the experiments were repeated in full, because after analysing the test and control experiments, it was found that during the conduct of the experiments there was a significant deviation in the conditions (mostly the network speed) for conducting. The numerical data obtained from both experiments (test and control) are not compared, because they depend on the Internet speeds, the configuration of both computers and the content of the websites. If there are single measured values that have a large deviation, they are removed before the value of the arithmetic mean is calculated. For outlier removal, the IQR method is used.

Various speed monitoring and analysis tools have been used in the experiments: Google PageSpeed Insights, GTmetrix, WebPageTest, Google DevTools, and Brave DevTools. By selecting and using Google DevTools and Brave DevTools, objectivity and accuracy of measurements are ensured, even at the level of a web element.

The measurements were made for a smartphone with a viewport of 430 × 932 px, a tablet – a size of 768 × 1024 px and a desktop/laptop – a size of 1024 × 1366 px. These parameters are chosen to cover the most common screen resolutions and provide a reliable comparison between them.

4.2. Experimental results

The results from measurements are shown in Table 1. In the first column are presented measured web elements with their different realization. Second and third columns show results for one web element used respectively in responsive and adaptive design, as results illustrate measurements for three different devices: smartphone, tablet and laptop.

Responsive design

Text. It has been experimented with serif and sans-serif fonts, with minimal differences (see Table 1.), but sans-serif fonts have a slight advantage. Serif loads fastest on a smartphone (1 ms), slower on a laptop (2 ms) and slowest on tablets (3 ms). Sans-serif fonts load fastest on a laptop (1 ms), and on smartphones and tablets they load in 2 ms. The loading speeds of fonts of small and large sizes, as well as in bold and italic styles, were measured, and the results show that there is no difference in loading speeds.

Navigation bar (menu). The built-in horizontal menu loads fastest on tablets (20 ms) and slightly slower on smartphones and laptops (22 ms). The floating horizontal menu has the fastest loading on tablets (22 ms), while on a smartphone (23 ms) and laptop (26 ms), loading is slower. The vertical menu loads fastest on a smartphone (28 ms), and on tablets (29 ms) and laptop (31 ms) the speed is slower.

Button. The standard button and link as a button load quickly and equally on smartphone and laptop devices, but are slightly delayed on tablets. The shadow button loads quickly on laptops but slower on tablets due to graphics processing. The transparent button is fastest on a laptop (2 ms) and tablet (3 ms), but it charges more slowly on a smartphone – 17 ms.

Image. Different image formats were studied – APNG, GIF, ICO, JPEG, PNG, SVG, BMP, HEIC, TIFF and WebP. When using <img> all formats load the fastest on a smartphone. When using <picture> all formats except APNG and HEIC load the fastest on a smartphone (9 – 22 ms) and slightly slower on a tablet and laptop (11 – 36 ms).

Embedded image. For embedded images implemented by the tags <object>, <embed>, <iframe> and <figure>, the results show: <object> loads images fastest on a smartphone (3 ms), while on tablets (4 ms) and laptop (6 ms) the speed is slower. When <embed> loads, it is fastest on tablets (3 ms), but slower on a smartphone (7 ms) and laptop (9 ms). The tag <iframe> loads faster on a smartphone (5 ms), and on tablets and laptops (7 ms) the speed is slower. <figure> Tablets have the fastest loading (4 ms), while smartphones (6 ms) and laptops (7 ms) are slower.

Built-in webpage. With <object> tag, the website loads fastest on a smartphone (14 ms), but slower on tablets (16 ms) and laptop (22 ms). With <embed> tag, the website loads fastest on a laptop (12 ms), but slower on tablets (13 ms) and smartphone (16 ms). The speed of <iframe> tag is the same for all devices (16 ms).

Table 1. Results of measuring the effectiveness of individual elements in responsive and adaptive design (in ms)

<table>
<tr><th colspan="2" rowspan="2">Element / Approach</th><th colspan="6">Responsive Design</th><th colspan="6">Adaptive Design</th></tr>
<tr><th colspan="2">Smartphone</th><th colspan="2">Tablet</th><th colspan="2">Laptop</th><th colspan="2">Smartphone</th><th colspan="2">Tablet</th><th colspan="2">Laptop</th></tr>
<tr><td rowspan="2">Text</td><td>serif</td><td colspan="2">1</td><td colspan="2">3</td><td colspan="2">2</td><td colspan="2">1</td><td colspan="2">3</td><td colspan="2">2</td></tr>
<tr><td>sans-serif</td><td colspan="2">2</td><td colspan="2">2</td><td colspan="2">1</td><td colspan="2">2</td><td colspan="2">2</td><td colspan="2">1</td></tr>
<tr><td rowspan="3">Navi-gation Bar</td><td>inline horiz.</td><td colspan="2">22</td><td colspan="2">20</td><td colspan="2">22</td><td colspan="2">41</td><td colspan="2">46</td><td colspan="2">22</td></tr>
<tr><td>float horiz.</td><td colspan="2">23</td><td colspan="2">22</td><td colspan="2">26</td><td colspan="2">37</td><td colspan="2">42</td><td colspan="2">40</td></tr>
<tr><td>vertical</td><td colspan="2">28</td><td colspan="2">29</td><td colspan="2">31</td><td colspan="2">30</td><td colspan="2">32</td><td colspan="2">36</td></tr>
<tr><td rowspan="4">Button</td><td>standard</td><td colspan="2">5</td><td colspan="2">11</td><td colspan="2">5</td><td colspan="2">3</td><td colspan="2">3</td><td colspan="2">3</td></tr>
<tr><td>link</td><td colspan="2">3</td><td colspan="2">12</td><td colspan="2">3</td><td colspan="2">5</td><td colspan="2">4</td><td colspan="2">7</td></tr>
<tr><td>shadow</td><td colspan="2">11</td><td colspan="2">22</td><td colspan="2">6</td><td colspan="2">4</td><td colspan="2">4</td><td colspan="2">4</td></tr>
<tr><td>transpa-rent</td><td colspan="2">17</td><td colspan="2">3</td><td colspan="2">2</td><td colspan="2">3</td><td colspan="2">3</td><td colspan="2">10</td></tr>
<tr><td rowspan="3">Image</td><td></td><td>img</td><td>picture</td><td>img</td><td>picture</td><td>img</td><td>picture</td><td>img</td><td>picture</td><td>img</td><td>picture</td><td>img</td><td>picture</td></tr>
<tr><td>APNG</td><td>13</td><td>10</td><td>21</td><td>10</td><td>21</td><td>8</td><td>25</td><td>17</td><td>26</td><td>17</td><td>32</td><td>19</td></tr>
<tr><td>GIF</td><td>13</td><td>10</td><td>21</td><td>12</td><td>21</td><td>13</td><td>24</td><td>14</td><td>25</td><td>10</td><td>25</td><td>16</td></tr>
</table>

	ICO	13	10	21	13	21	13	25	9	18	12	26	10
	JPEG	13	10	22	13	21	13	17	13	20	12	17	11
	PNG	13	10	22	12	13	13	25	17	31	17	31	17
	SVG	9	9	22	11	14	11	36	24	39	21	37	22
	WEBP	16	9	20	12	18	12	25	12	23	12	23	15
Built-in image	object	3		4		6		5		3		3	
	embed	7		3		9		3		13		3	
	iframe	5		7		7		17		14		3	
	figure	6		4		7		3		4		13	
Built-in web site	object	14		16		22		7		5		14	
	embed	16		13		12		3		3		5	
	iframe	16		16		16		4		4		4	
Element positioning													
Flexbox		64		85		73		20		29		21	
Float	none	58		41		34		14		9		32	
	left	46		41		52		25		20		20	
	right	47		45		38		26		20		26	
	initial	49		49		47							
	inherit	33		52		42		19		23		26	
Posi-tion	static	48		45		40		24		45		40	
	relative	54		43		42		45		44		54	
	fixed	70		78		73		49		44		51	
	absolute	59		57		56		29		29		37	
	sticky	51		51		51		31		30		35	
Layout													
CSS Float		57		66		60		93		102		94	
CSS Grid		53		55		53		89		102		97	
CSS Flexbox		62		60		58		106		71		23	
CSS Framework		20		24		21		106		73		109	

Positioning of elements. The different ways of positioning the elements in a web page with their different values are explored: Flexbox, Float and Position. Float with *inherit* (for smartphone) and *none* (for laptop) provides the lowest loading times and is suitable for lightweight UIs. Position with *static* and *sticky* values balances good performance and control.

Page layout. Table 1. presents the measurements made in different ways of arranging the page layout: CSS Float, CSS Grid, CSS Flexbox and CSS Frameworks (such as Bootstrap), where it is visible that CSS Framework gives the best results.

Web page. When measuring the loading speed of the entire web page, the tablet version is the slowest (825 ms), indicating that it contains the most resources. The laptop version is the most optimized (793 ms) and the mobile version is balanced (794 ms). This indicates that the page is most optimized for a laptop, while the tablet version has more load. Also, the loading speeds of text in different colors have been measured.

Adaptive design

Text. Serif fonts load the fastest on a smartphone (1 ms) and the slowest on tablets (3 ms) as shown in Table 1. Sans-serif fonts load fastest on a laptop (1 ms), while smartphones and tablets have similar speeds (2 ms). Features such as size and style show no differences.

Menu. The built-in horizontal menu loads the slowest on tablets (46 ms) and fastest on laptops (22 ms). The floating horizontal menu is the slowest on a tablet (42 ms) and the fastest on a smartphone (37 ms). The vertical menu shows better performance on smartphones (30 ms) and tablets (32 ms), while on a laptop it charges a little slower (36 ms).

Button. The standard button and shadow button load faster on all devices, with loading speeds almost the same and optimized. The standard button charges equally fast on all devices (3 ms). Link as a button is the slowest on a laptop (7 ms), but the fastest on a tablet (4 ms). The shadow button charges evenly (4 ms) on all devices. The transparent button charges significantly slower on a laptop (10 ms) compared to a smartphone and tablets (3 ms).

Image. Different image formats were examined. When using <img> ICO loads fastest on tablets (18 ms), HEIC is the fastest format on all devices: smartphone (14 ms), tablet (18 ms) and laptop (15 ms), and SVG images load significantly slower on all devices.<picture> GIF loads fastest on a tablet (10 ms), and ICO and HEIC load fastest on a smartphone (9 ms) and laptop (10 ms), both photo formats have the same speed on these devices.

Embedded image. The <object> tag loads images faster on tablet and laptop (3 ms), <embed> loads the slowest on tablets (13 ms), while on smartphone and laptop it is relatively fast (3 ms), <iframe> is the slowest method for embedding images on a smartphone (17 ms) and tablets (14 ms), and <figure> is faster for a smartphone.

Built-in webpage. The loading speed of a built-in webpage is measured using different tags: <object> is the slowest on a laptop (14 ms) compared to a smartphone (7 ms) and a tablet (5 ms), and <embed> <iframe> they have minimal differences in speed.

Positioning of the elements. Different ways of positioning the elements (Flexbox, Float and Position) are measured. For a smartphone, the fastest option is float:none with a speed of 14 ms. For a tablet, float:none is also rendered the fastest at a speed of 9 ms. For a laptop, float:left works fastest at a speed of 20 ms.

Page layout. For a smartphone, arranging elements in a web page through CSS Grid Layout (89 ms) is the fastest, making it preferred for mobile UIs. For tablets, the fastest is CSS Flexbox Layout (71 ms), which shows that the network structure works efficiently on medium screens. For desktops, CSS Flexbox Layout (23 ms) is again the fastest.

Web page. When measuring the loading speed of the entire web page, the laptop version is the slowest (1193 ms), indicating that it contains the most resources. The tablet version is the most optimized (1021 ms) and the mobile version is balanced (1070 ms).

This indicates that the page is most optimized for tablets, while the laptop version has more load. Also, the loading speeds of text in different colors have been measured.

5. DESIGN RECOMMENDATIONS AND DISCUSSION

Based on the analysis of the tests performed, the following conclusions and specific recommendations for optimization can be made when creating effective web designs.

For **responsive design**, sans-serif fonts are faster to load, making them ideal for smartphones where speed and efficiency are critical. Using a built-in horizontal menu, which loads the fastest, is an excellent choice for mobile devices. It allows users to easily navigate through the site without burdening the site with unnecessary loading time. Standard buttons and links such as buttons are the best choice for smartphones and laptops, as they load quickly and provide clear and intuitive actions for users. Transparent buttons, although they may look stylish, load slowly on smartphones, which can lead to waiting and a negative user experience. Using optimized image formats such as WebP and SVG provides the fastest loading times and a better user experience. Avoiding heavy images with large sizes and formats that slow downloading can improve speed and reduce wait times, which is critical for mobile users. For smartphones and laptops, it is recommended to use embedded images through <object> a tag, as they provide the fastest loading times. This allows mobile users to have fast-loading pages without having to wait for the images to fully load. Using an <embed> tag on websites is a good solution when looking for a balance between loading speed and user-friendliness. It doesn't require additional scrolling or <iframe> loading containers, resulting in a smoother interaction with mobile devices. For an optimal smartphone user experience, simpler and less burdensome color schemes, such as black and blue, should be used, which will ensure fast loading times and minimize visual strain for the user if used excessively. For a faster and more efficient responsive design, it is advisable to use Float and Position methods for positioning the elements. When building responsive web pages, the best choice is to use a CSS Framework like Bootstrap, which offers fast loading times and adaptability to different devices. While Flexbox and Grid are both powerful stacking tools, they can require more resources, which can degrade the user experience, especially on slower drives or in unstable network conditions. For better performance in a responsive design, it is advisable to optimize the tablet version by reducing heavy resources and eliminating unnecessary elements from the laptop version. The use of adaptive images (via srcset and WebP), lazy loading of invisible content, and the removal of unnecessary animations and scripts contribute to faster loading times and a better user experience.

In **adaptive design**, it is recommended to use sans-serif fonts, as they provide the best performance on most devices, as well as local font loading, instead of using external resources (Google Fonts), for faster initial loading times. A good approach is also to minimize the number of different fonts and styles to achieve better speed. The vertical menu is the better option for achieving optimal speed for smartphones and tablets, and the built-in horizontal menu – for laptops. It is recommended to use standard buttons on all devices for faster loading and avoid transparent buttons as they slow downloading, especially on a laptop. The use of complex CSS effects (box-shadow, opacity, and filter) to improve speed should be reduced. It is better to use WebP or HEIC formats instead of PNG/JPEG for better compression and fast loading, SVG optimization through tools such as SVGO to reduce file size, as well as lazy-loading images that are not visible at initial loading. It is recommended to use it <object> for embedding images on tablets and laptops and to avoid <iframe>it, especially on a smartphone, and can also be used <embed> <figure> for a smartphone. It is appropriate to use it <embed> to embed a website on a smartphone and tablets for better performance, but for a laptop it is better to avoid it <object>as it is slower, but it can be replaced by <iframe>, which shows the fastest speed in testing. For smartphones and laptops, measurements show (although a significant number of colours have not been examined) that it is good to use pure blue text colour, for tablets – black text color is the best choice. Using float:none provides the highest performance on a mobile device and tablet, and float:left on a laptop. On the other hand, positioning with position:relative and position:fixed are relatively slow, especially on laptop devices. CSS Flexbox Layout is optimal for larger screens, while Grid is better for a smartphone as it handles dynamic space allocation well. It is recommended to optimize the laptop version by reducing the use of heavy effects and built-in resources, using lazy-loading content that is not visible on first loading, and reducing unnecessary JavaScript effects and animations.

Finally, when designing a website, you should first consider the intended users, as well as analyse market trends, budget, and future business development. If the majority of users use desktops, responsive design is the better choice, but if the main traffic comes from mobile devices, the responsive design has to be done through mobile-first approach. And the adaptive design is suitable for specific cases with fixed resolutions and for widely used sites by different groups of users who use different devices.

6. CONCLUSION

Sustainability is now becoming a critical factor for high efficiency in software development. At the global level, there is an energy shortage and prices are rising. In this regard, energy-efficient applications are a must, especially for mobile devices. Test websites were made using the methods of adaptive, responsive and mobile first design and experiments were conducted on the effectiveness of using different UI elements, measuring the load times for each method. The conclusions drawn after the many experiments are summarized and some general recommendations for the design of sustainable UIs are defined.

ACKNOWLEDGEMENT

This study is carried out and funded in the frame of the project FP25-FMI-010, “Innovative Interdisciplinary Research in Informatics, Mathematics, and Pedagogy of Education” of the Scientific Fund of the Paisii Hilendarski University of Plovdiv, Bulgaria, 2025-2027.

REFERENCES

[1] *United Nations. The 17 Goals*. 2015. URL: https://sdgs.un.org/goals (Visited on 21.05.2025).

[2] Shenoy, S.S., Eeratta, R. Green software development model: An approach towards sustainable software development. *2011 Annual IEEE India Conference (INDICON), Hyderabad, India*, 2011, pp. 1-6. DOI: 10.1109/INDCON.2011.6139638.

[3] Harrison, M. *Sustainability and Green Software Engineering*. Microsoft. URL: https://www.microsoft.com/en-gb/industry/blog/technetuk/2021/08/19/sustainability-and-green-software-engineering/ (Visited on 21.05.2025).

[4] *Sustainable UX: Designing for Tomorrow.* URL: https://www.nagarro.com/en/blog/sustainable-ux-designing-for-tomorrow (Visited on 21.05.2025).

[5] Zoraida, B.S.E., Magdalene, J.J.C. Enhancing energy efficiency in a smart home through window-based support vector regression for energy consumption prediction. *International Journal on Information Technologies and Security (IJITS)*, vol.15, no.3, 2023, pp. 23-32. DOI: 10.59035/ENQO9045.

[6] Pelivani, E., Besimi, A., Cico, B. An Empirical Study of User Interface Testing Tools. *International Journal on Information Technologies and Security (IJITS)*, vol.14, no.1, 2022, pp. 37-48. ISSN: 1313-8251.

[7] *IBM. Sustainable Design*. URL: https://www.ibm.com/think/topics/sustainable-design (Visited on 21.05.2025).

[8] *Smartphone Usage Statistics 2025 (Worldwide Data)*. URL: https://www.demandsage.com/smartphone-usage-statistics/ (Visited on 20.06.2025).

[9] Vallerio, K.S., Zhong, L., Jha, N.K. Energy-efficient graphical user interface design. *IEEE Transactions on Mobile Computing*, vol.5, no.7, 2006, pp. 846-859. DOI: 10.1109/TMC.2006.97.

[10] Huber, S., Döller, M., Felderer, M. On the Energy-Efficiency of Hybrid UI Components for Mobile Cross-Platform Development. *In: Garrigós, I., Murillo Rodríguez, J.M., Wimmer, M. (eds), Web Engineering. ICWE 2023. Lecture Notes in Computer Science*, vol.13893, Springer, Cham, 2023, pp. 246–261. DOI: 10.1007/978-3-031-34444-2_18.

[11] Maddirala, S. *Minimal Criteria: Minimizing User Input and User Interface for Faster Output in Minimalistic Mobile Applications*. Master's thesis, University of Cincinnati, 2019. URL: http://rave.ohiolink.edu/etdc/view?acc_num=ucin1560867841585446 (Visited on 21.05.2025)

[12] Meyer, K.L. *Towards a Definition of Minimalism: Principles of Minimal Visual Design in Web Interfaces*. Master's paper for the M.S. in I.S., University of North Carolina, March 2015, 58 p. Advisor: Dr. Ronald Bergquist. DOI: 10.17615/trf0-ya50.

[13] Durgekar, S.R., Rahman, S.A., Naik, S.R., Kanchan, S.S., Srinivasan, G. A Review Paper on Design and Experience of Mobile Applications. *EAI Endorsed Transactions on Scalable Information Systems*, vol.11, no.3, 2024. DOI: 10.4108/eetsis.4959.

[14] Kiourtis, A., Mavrogiorgou, A., Zafeiropoulos, N., Mavrogiorgos, K., Karabetian, A., Kyriazis, D. UI/UX Sustainable Design: Best Practices for Applications CO2 Emissions Reduction. 2024 *9th International Conference on Smart and Sustainable Technologies (SpliTech), Bol and Split, Croatia*, 2024, pp. 01-06. DOI: 10.23919/SpliTech61897.2024.10612495.

[15] Fahlström, E. *Sustainable Web Design - How much can environmental friendly design principles improve a website's carbon footprint?* URL: https://www.diva-portal.org/smash/get/diva2:1775278/FULLTEXT02.pdf (Visited on 21.05.2025).

[16] Grabska, E., Grabska-Gradzińska, I., Frodyma, T. The role of typography in visual design. *Craft, Technology and Design*, 2022, pp. 118-130. ISBN: 978-951-784-832-9.

[17] Granum, H.L., Hansen, S.G., Hulleberg, O., Moen, M. *Sustainable Web Design Guidelines.* Bachelor's thesis, NTNU, 2023. URL: https://ntnuopen.ntnu.no/... (Visited on 21.05.2025).

[19] Han, Y., Wu, M., Roth, R. Toward green cartography & visualization: a semantically-enriched method of generating energy-aware color schemes for digital maps. *Cartography and Geographic Information Science*, vol.48, no.1, 2020, pp. 43–62. DOI: 10.1080/15230406.2020.1827040.

[20] Greenwood, T. *The dark side of green web design.* Wholegrain Digital. URL: https://www.wholegraindigital.com/blog/dark-colour-web-design/ (Visited on 21.05.2025).

[21] Testolina, M., Upenik, E., Ebrahimi, T. Comprehensive assessment of image compression algorithms. In: *Applications of Digital Image Processing XLIII*, vol.11510, 2020, pp. 469-485. SPIE.

[22] *Sustainable web design: recommendations and examples for accessible site design.* URL: https://smowl.net/en/blog/sustainable-web-design/ (Visited on 21.05.2025).

[23] Li, W., Zhou, Y., Luo, S., Dong, Y. Design Factors to Improve the Consistency and Sustainable User Experience of Responsive Interface Design. *Sustainability*, vol.14, no.15, 2022, p. 9131. DOI: 10.3390/su14159131.

[24] *Designshot. Sustainable Design: 6 Things You Can Do as a Designer.* URL: https://www.designshot.co/insights/sustainable-design-6-things-you-can-do-as-a-designer (Visited on 21.05.2025).

[25] *Arts University Bournemouth. What Is Sustainable Design and Why Is It Important?* URL: https://online.aub.ac.uk/resources/what-is-sustainable-design-and-why-is-it-important (Visited on 21.05.2025).

Information about the authors:

Yoanna Klinchova, student at Faculty of Mathematics and Informatics, University of Plovdiv.

Emel Shaban, student at Faculty of Mathematics and Informatics, University of Plovdiv.

Assoc. Prof. Dr. Nikolay Kasakliev, ORCID ID: 0000-0003-4010-144X, Dept. "Computer Science", University of Plovdiv, Areas of scientific research: information security, programming languages, web development, mobile development, etc.

Prof. Dr. Elena Somova, ORCID ID: 0000-0003-3393-1058, Dept. "Computer Science", University of Plovdiv, Areas of scientific research: responsible and sustainable software, green gamification, game-based learning, etc.

Manuscript received on 25 May 2025