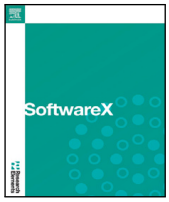
SoftwareX 25 (2024) 101633

Contents lists available at ScienceDirect

SoftwareX

journal homepage: www.elsevier.com/locate/softx



Original software publication

MAWA: Enhancing mobile web browsing through web augmentation for improved user experience

Iñigo Aldalur *, Alain Perez, Felix Larrinaga, Miren Illarramendi

Electronics and Computing Department, Mondragon Unibertsitatea, Loramendi 4, Arrasate-Mondragon, 20500, Gipuzkoa, Spain

ARTICLE INFO

Keywords:
Human–computer interaction
End-user development
Web augmentation
Customization
Mobile browser

ABSTRACT

In recent years, mobile devices have witnessed an exceptional surge in web browsing, emerging as the primary information source. Developers, however, struggle to cater to every user's needs, compelling users to customize web content. Moreover, mobile interactions, including URL typing, scrolling, tapping, and tab-switching, are more cumbersome due to limited screen size, leading to an unpleasant web experience. Web Augmentation offers a solution by enhancing the user experience, reducing interactions, and meeting information needs. We introduce MAWA, a Firefox mobile extension that empowers users to adapt websites. Results demonstrate MAWA's effectiveness in reducing interactions, battery consumption, and information retrieval time.

Code metadata

Current code version	0.4.6
Permanent link to code/repository used for this code version	https://github.com/ElsevierSoftwareX/SOFTX-D-23-00710
Permanent link to Reproducible Capsule	
Legal Code License	GPLv3
Code versioning system used	Git
Software code languages, tools, and services used	JavaScript, HTML and CSS
Compilation requirements, operating environments & dependencies	Firefox Nightly
If available Link to developer documentation/manual	https://github.com/ialdalur1/MAWA/blob/main/README.md
Support email for questions	ialdalur@mondragon.edu

1. Motivation and significance

The proliferation of mobile phones and web applications has led to a surge in web users, making mobile traffic a dominant force on the internet [1]. As a consequence, there is a growing need for customized web experiences tailored to individual preferences [2].

Web Augmentation (WA) emerges as a vital technique, enabling end-users to personalize websites without requiring extensive programming skills [3]. WA, as initially coined by Bouvin in 1999, involves enhancing web pages by adding or modifying content through user-friendly tools integrated into web browsers or servers [4]. Among various methods for implementing WA, visual programming techniques are popular due to their simplicity and effectiveness [5].

End-User Development (EUD) encompasses methods, tools, and techniques enabling non-professional users to create, modify, or extend software artifacts [6]. WA aligns with the principles of EUD, allowing gradual implementation and adaptation by end-users [7]. This approach holds promise for mobile users, who share concerns with desktop users regarding web accessibility and usability [8].

Responsive Web Design (RWD), initially developed to adapt websites from desktop to mobile, faces challenges in preserving user experience and avoiding usability issues [9]. Personalization techniques, although valuable, might not always accurately anticipate individual preferences [10]. WA, on the other hand, empowers users to make specific adjustments themselves, reducing issues like excessive scrolling and tapping [11,12].

* Corresponding author.
E-mail addresses: ialdalur@mondragon.edu (Iñigo Aldalur), aperez@mondragon.edu (Alain Perez), flarrinaga@mondragon.edu (Felix Larrinaga), millarramendi@mondragon.edu (Miren Illarramendi).

https://doi.org/10.1016/j.softx.2024.101633
Received 19 October 2023; Received in revised form 12 December 2023; Accepted 9 January 2024
Available online 16 January 2024

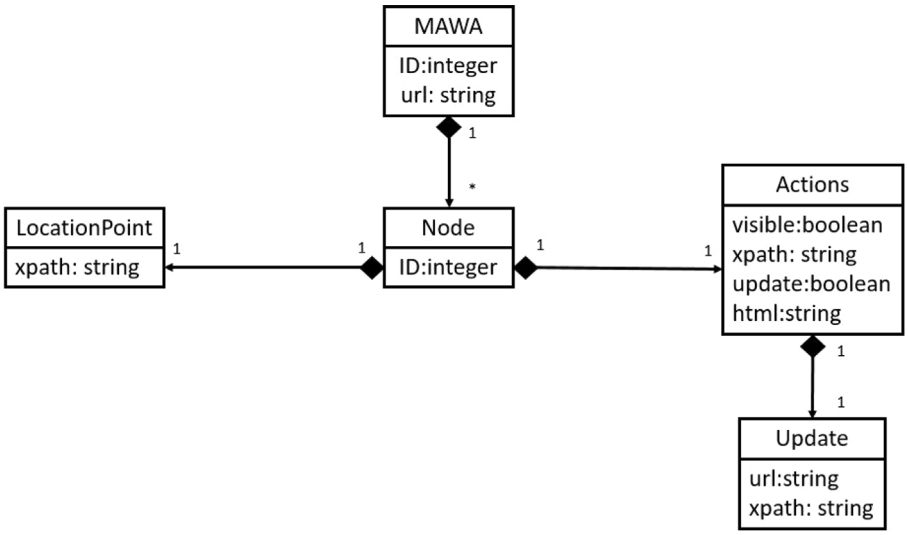


Fig. 1. MAWA data model.

The problem stems from the impracticality of developing a unique web version for each user and the limitations of RWD [13]. Several causes contribute to this challenge: the prohibitive cost of individualized web versions, the impossibility of foreseeing all user requirements, the lack of programming skills among end-users, and the inherent limitations of small screens on mobile devices [14–17].

These challenges result in various consequences, including increased scrolling [18] and tapping [19], frequent tab-switching [20], difficulties in managing multiple open tabs [21], errors in URL typing [22], heightened battery consumption [23], longer search times [24], cluttered navigation [25], and the display of unnecessary or hidden content [26]. These issues collectively undermine the user experience and hinder efficient web browsing on mobile devices. In order to mitigate these issues, we have developed a browser extension, MAWA.

2. Software description

This section introduces the architecture of MAWA and its functionalities.

2.1. Software architecture

The data model of MAWA is depicted in Fig. 1. This architecture comprises five distinct components: MAWA, Node, LocationPoint, Actions, and Update. The software is implemented in JavaScript and stores all essential information required for correct execution in the browser's local storage in JSON format.

MAWA component serves as the central element of the model, storing key information such as the web page's address where the automation process initiates, an identification (ID) for augmentation, and a URL indicating when the augmentation process should take effect.

Nodes encompass web elements from both third-party sites and local elements, which can be inserted, removed, or relocated within the adapted web page.

LocationPoint utilizes a web locator to uniquely identify nodes within the Document Object Model (DOM) [27]. This component collects the locator that specifies where the element should be inserted into the adapted web page.

Actions describe whether the augmented node is local or external, and it also stores the locator for moving or removing the component. When a node requires removal, the visible element attribute is set to true. For nodes that should not be updated, HTML elements store the node's content. Conversely, when updating a node is necessary, the extension saves this value to facilitate the update process.

The *Update* component collects the locator in XPath format, determining from which point to extract the node when the user intends to update the content. Additionally, it contains the URL specifying the source from which this node should be extracted.

2.2. Software functionalities

MAWA is a Firefox Nightly browser extension,[1] one of the few mobile browsers that accepts installing extensions. Once the application is installed, the user must double-tap to enact the application. Once the augmentation has been performed, every time that the user visits the modified website, the extension automatically executes the augmentation process performing all the changes. An example has been accomplished using all the actions allowed in MAWA. All the needed actions are described in Table 1. It is described how to add, move and remove elements from the "The Boston Globe" website.[2] Fig. 2A show the initial state and 2B shows the final state of this website after using MAWA. Table 2 illustrates the number of actions, battery consumption, rendering time and execution time needed with and without MAWA to perform the example. The reduction of this metrics in performed thanks to three actions (move, remove and add nodes):

2.2.1. Scrolling reduction

The size of mobile screens means that the content of web pages is structured in such a way that the user has to scroll frequently. By moving all the nodes that the user consults to the top of the web page, the user does not have to scroll to consult that information.

2.2.2. Tapping, tab-switching, branching and URL typing reduction

Web browsing is more complex on a mobile phone than on a laptop. Opening a new tab to search for additional information makes mobile browsing more complex and requires the user to make many taps to perform the action. Adding content that the user needs on another web page eliminates the need for the user to open a new tab, type the URL, tab-switching and all the taps associated with this action.

[1] https://addons.mozilla.org/es/firefox/addon/mawa/
[2] https://youtu.be/pULYEBtXmOs

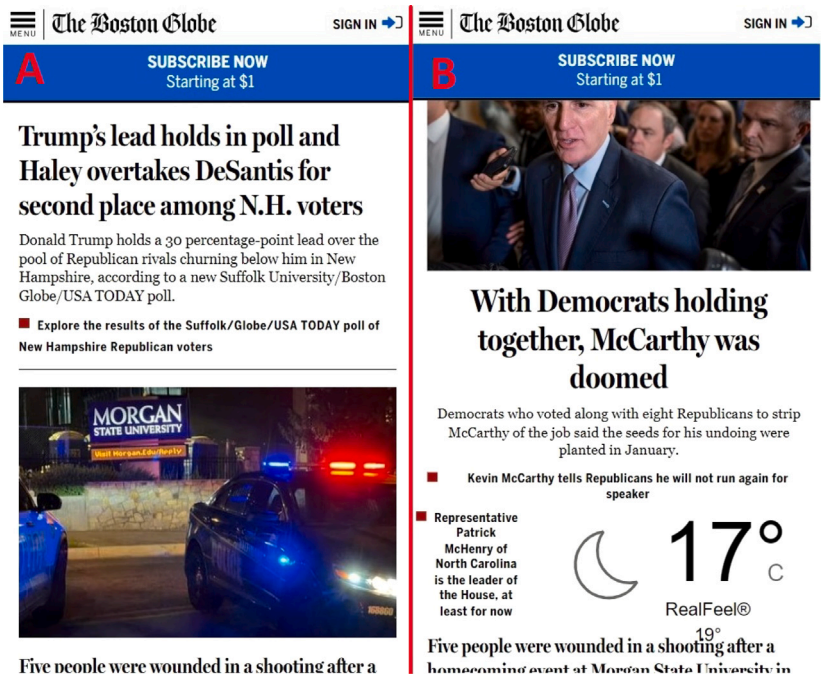


Fig. 2. Initial website (A, on the left) and final version after using MAWA (B, on the right).

Table 1
Description of MAWA actions.

Action	Description
Enact menu	The menu is enacted each time the user double-taps on the mobile phone screen. Fig. 3A shows the menu for the first time. The "New" option allows starting modifying the current page. "Mineit" allows copying and saves an element of the current page to be able to insert it later, and "Cancel" allows continuing browsing. This option is useful every time the user unintentionally double-taps. In Fig. 3B it is shown the MAWA menu when we are augmenting a web page. "Save" allows recording the changes made by the user, reloading the page and executing the page augmentation. Fig. 3C shows the MAWA menu after executing the augmentation. The "Update" option allows the user to continue modifying the web page, and "Remove" allows the user to remove the augmentation and reload the page to remove the changes made.
Select nodes	When augmenting a web page, the user can select the nodes on the page by simply tapping on them. At that moment, the background of the node becomes a reddish color and top, bottom and side arrows are added. These arrows allow the user to select nodes that are difficult to select by tapping on them. The upper arrow allows selecting the father node of the current node. The bottom arrow, selects the first child node of the current node. The side nodes allow the user to select brother nodes, the left arrow selects the previous node and the right arrow the next node. If the selected node has no parents, children or siblings, the corresponding arrows do not appear. We can see examples in Figs. 4 and 6.
Remove nodes	Once a node is selected, the user must tap on the "X" at the top-right margin of the selected node and the node will be deleted. The example is shown in Fig. 4.
Move nodes	When a node is selected, it can be moved by drag&drop. In Fig. 5B it is shown that when the node is moved, the purple background indicates where the node will be inserted. Once dropped, Fig. 5C shows how the node is relocated (in this case to the top of the web page).
Mine Nodes	Once the user taps on the "Mineit" option in the menu, he/she can select a node and save a copy to be inserted later in a page to be augmented. On this occasion, all the selected nodes have a "Copy" button (see Fig. 6A) which, once tapped, asks the user to introduce a name (see Fig. 6B). Once the name is accepted, the page returns to its natural state and MAWA saves a copy of the node.
Insert nodes	In order to insert a new node from another page into the current page, the user must tap on the "+" button, which is located at the top-right of the web page. Once tapped, all saved elements are displayed (see Fig. 7A). Once the desired button is tapped, the page goes back to the page that is being augmented and once tapped on the screen, the node is inserted at that place. These nodes can be constant (the user wants the content to be always the same) or they can be dynamic (the content should be updated with the current content). The content inserted in Fig. 7 is a Boston weather element. The content must be updated as the weather varies every day. Once tapped on "Update" (see Fig. 7B), the background of the node turns green (see Fig. 7C) and this will provoke that in the augmentation process, MAWA will update its content. Otherwise, the content will always remain constant.

2.2.3. Battery time reduction and rendering increase

Battery consumption is important in any mobile device. The screen is what causes the battery to decrease the most [28]. For this reason, the time the user needs to consult the information must decrease. By removing content from the web page, the user only has the information he wants and discards the information that is not useful to him/her. Adding and moving content also helps to reduce the time of use. The consequence is that the time needed to render the page increases.

3. Related works

This contribution has been previously published in the INTERACT conference [29], showcasing its recognition within the academic community and its adherence to rigorous peer-review standards. The conference venue served as a platform for disseminating this research, enabling it to reach a broader audience. The acceptance and presentation of this work at this conference affirm its scholarly significance and its potential to contribute to the ongoing discourse in the respective domain of study.

4. Impact overview

Numerous contributions related to WA have surfaced in recent years. A comprehensive survey [30] delves into the characteristics of WA tools, analyzing tools developed from 2010 to 2022. WA has demonstrated its versatility in aiding end-users across diverse domains.

Table 2
Number of actions completed by users in mobile phones without MAWA and with MAWA.

Scroll	Tap	Tab-switching	Branching	URL typing	Battery consumption (mAh)	Rendering cost (ms)	Time (s)
Without MAWA							
2	6	2	1	1	22.8	1281	78
With MAWA							
1	0	0	0	0	1.6	1469	10

Fig. 3. MAWA extension menus for Web Augmentation.

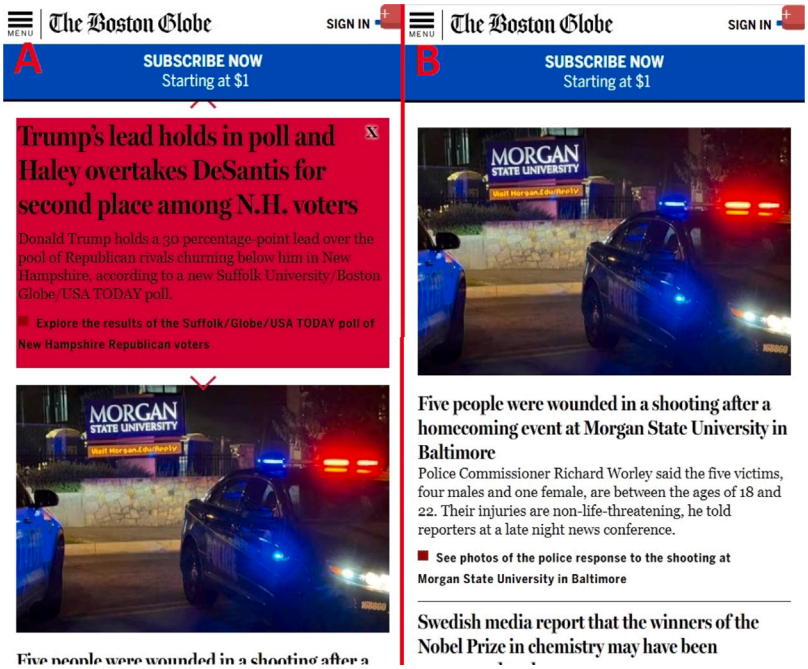


Fig. 4. Removing the first news from the website.

A burgeoning domain for WA is mobile browsers; mobile extensions have only become feasible in recent years. The unique challenges posed by smaller screens and intricate interactions make WA tools particularly valuable in reducing user interactions, minimizing errors, changing the layout and expediting information retrieval. To minimize interactions, it is common practice to introduce new functionalities [31] while also reducing the occurrence of errors in web interactions [32–34]. Incorporating content from alternative origins [35–37] is a frequent occurrence in the context of WA, and it is especially crucial in the realm of mobile WA, given the limitations imposed by smaller screens. Furthermore, the page layout is typically adjusted to cater to user preferences [38,39]. The visual style can also undergo modifications through the integration

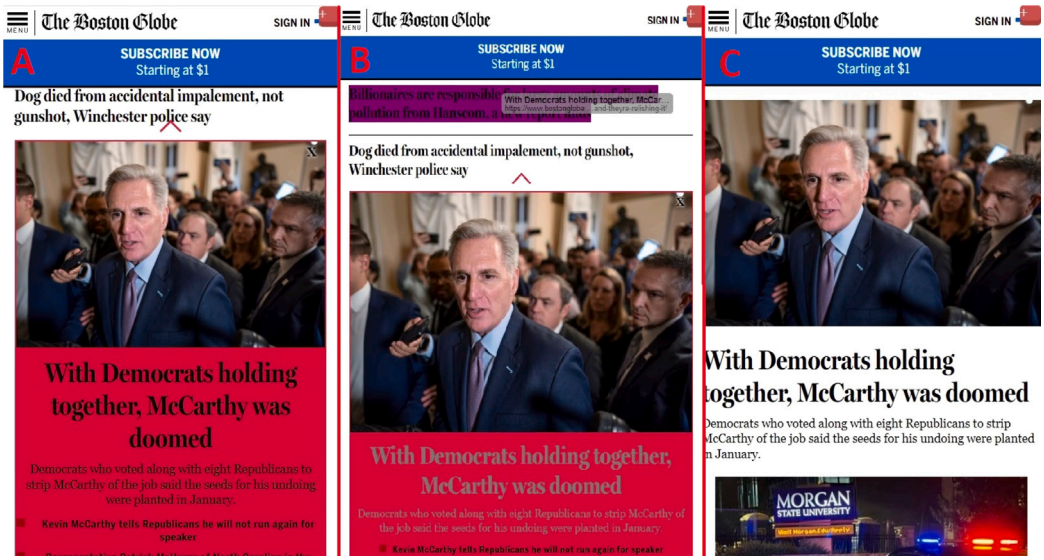


Fig. 5. Moving news from the bottom to the top of the website.

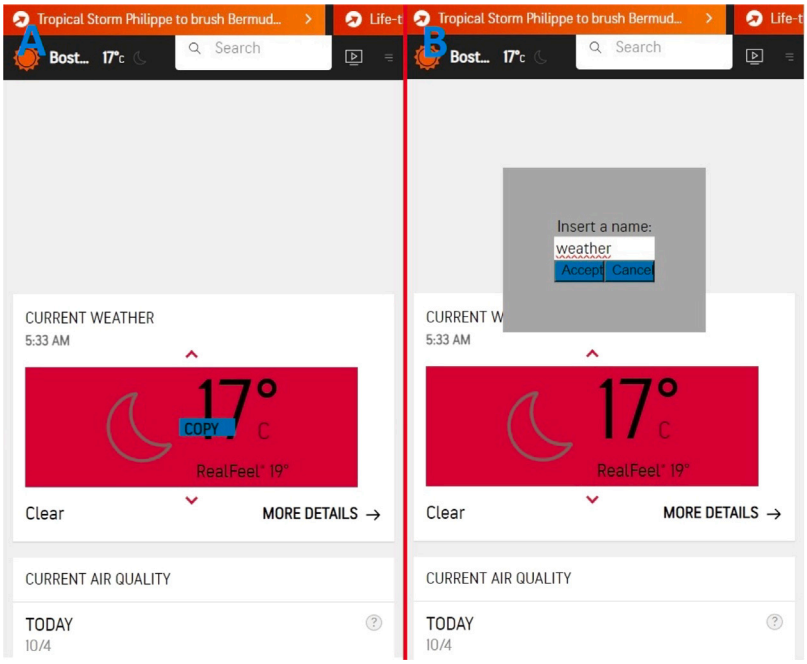


Fig. 6. Mining foreign content to be inserted in the augmented website afterward.

of content from diverse sources [40]. An 50% of web page revisitation [41] justifies the importance of the personalization of this content [42].

Our research on MAWA significantly contributes to the community by addressing challenges in mobile web browsing, enhancing user experience, and empowering end-users through WA. The impact extends to potential advancements in the broader field of mobile browser extensions, paving the way for future research in improving customization, reducing user interactions, and optimizing battery consumption for a more efficient mobile browsing landscape.

In the present day, the Firefox mobile browser holds the seventh position in global usage, accounting for 0.51% of the market share.[3] The option to employ browser extensions on mobile devices has only recently become available, and other browsers are also considering enabling this feature. Consequently, in the near future, it is expected that this extension will become compatible with various other web browsers.

5. Limitations and future work

The primary constraint of this tool pertains to a recurring issue: webpage updates. These modifications disrupt the efficacy of the methods employed to pinpoint desired nodes, resulting in locator failures and halted automation. In our forthcoming endeavors, we aim to incorporate mechanisms that enhance the durability of locators [43–47]. Additionally, we intend to explore and develop strategies for further prolonging locator longevity.

As part of future endeavors, we aspire to delve into the realm of security and privacy in Mobile WA. Our focus will include the

[3] https://gs.statcounter.com/browser-market-share/mobile/worldwide

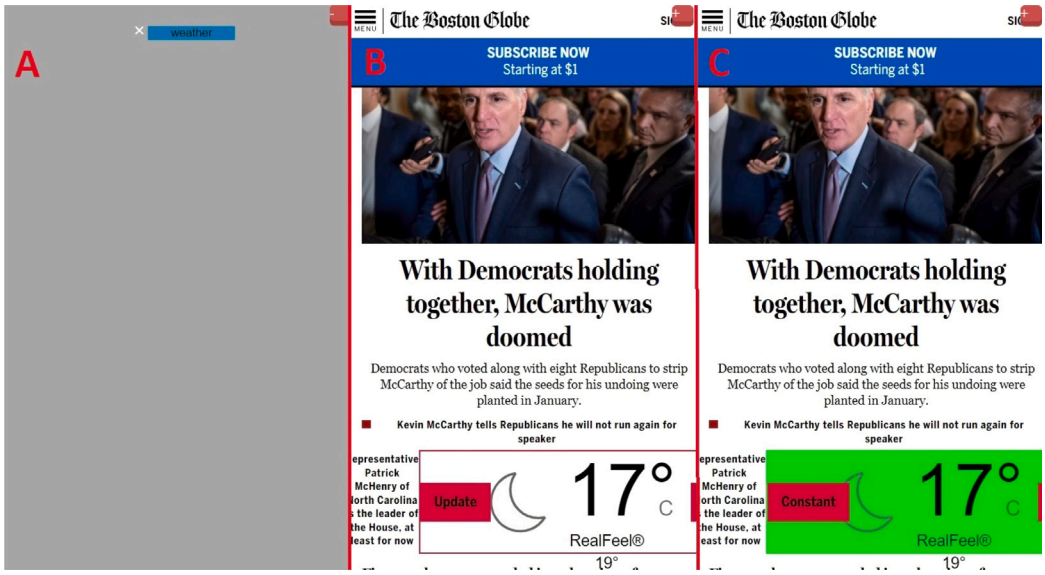


Fig. 7. Adding mined content into the augmented website.

exploration of cutting-edge encryption techniques and the development of robust privacy controls within MAWA, paving the way for enhanced user data protection.

6. Conclusions

The surge in mobile web browsing has created challenges as websites struggle to adapt to various screen sizes, primarily designed for desktops. This mismatch between design priorities and user needs leads to issues like increased scrolling, tapping, and longer action times, resulting in higher battery consumption for mobile users.

To address these challenges, we introduced MAWA, a Firefox mobile browser extension. MAWA's development offers promise for enhancing mobile web browsing and addressing the evolving needs of users in an increasingly mobile-centric digital landscape. MAWA empowers users to customize websites by removing unnecessary content, repositioning elements to reduce scrolling, and adding new content. Our evaluation showed that MAWA significantly improves the mobile browsing experience, reducing battery consumption by 92.98% and user interactions by 91.66%, while decreasing action completion time by 87.18%.

Finally, mobile browsers generally lack extension support. Adding extension support would make MAWA and other extensions more accessible and provide their benefits to a wider audience.

CRediT authorship contribution statement

Iñigo Aldalur: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Project administration, Software, Supervision, Validation, Visualization, Writing – original draft, Writing – review & editing. **Alain Perez:** Data curation, Writing – review & editing. **Felix Larrinaga:** Methodology, Writing – review & editing. **Miren Illarramendi:** Funding acquisition, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgments

This work was carried out by the Software and Systems Engineering research group of Mondragon Unibertsitatea (IT519-22), supported by the Department of Education, Universities and Research of the Basque Government, Spain.

References

[1] Nurshuhada A, Yusop ROM, Azmi A, Ismail SA, Sarkan HM, Kama N. Enhancing performance aspect in usability guidelines for mobile web application. In: 2019 6th international conference on research and innovation in information systems. IEEE; 2019, p. 1–6. http://dx.doi.org/10.1109/ICRIIS48246.2019.9073617.

[2] García-Lopez E, García-Cabot A, Manresa-Yee C, de Marcos L, Pagés-Arévalo C. Validation of navigation guidelines for improving usability in the mobile web. Comput Stand Interfaces 2017;52:51–62. http://dx.doi.org/10.1016/j.csi.2017.01.011.

[3] Díaz O, Arellano C, Aldalur I, Medina H, Firmenich S. End-user browser-side modification of web pages. In: Web information systems engineering - WISE 2014 - 15th international conference, thessaloniki, greece, october 12-14, 2014, proceedings, part I. Lecture notes in computer science, vol. 8786, Springer; 2014, p. 293–307. http://dx.doi.org/10.1007/978-3-319-11749-2_23.

[4] Bouvin NO. Unifying strategies for web augmentation. In: Proceedings of the tenth ACM conference on hypertext and hypermedia: Returning to our diverse roots: returning to our diverse roots. 1999, p. 91–100. http://dx.doi.org/10.1145/294469.294493.

[5] Aldalur I, Winckler M, Díaz O, Palanque P. Web augmentation as a promising technology for end user development. In: New perspectives in end-user development. Springer; 2017, p. 433–59. http://dx.doi.org/10.1007/978-3-319-60291-2_17.

[6] Lieberman H, Paternò F, Wulf V, editors. End user development. Human-computer interaction series, Springer; 2006, http://dx.doi.org/10.1007/1-4020-5386-X.

[7] Spahn M, Dörner C, Wulf V. End user development: Approaches towards a flexible software design. In: 16th European conference on information systems. 2008, p. 303–14.

[8] Pinandito A, Az-zahra HM, Fanani L, Putri AV. Analysis of web content delivery effectiveness and efficiency in responsive web design using material design guidelines and user centered design. In: International conference on sustainable information engineering and technology. IEEE; 2017, p. 435–41. http://dx.doi.org/10.1109/SIET.2017.8304178.

[9] Punchoojit L, Hongwarittorrn N. Usability studies on mobile user interface design patterns: A systematic literature review. Adv Hum-Comput Interact 2017;2017:6787504:1–22. http://dx.doi.org/10.1155/2017/6787504.

[10] Mahajan S, Abolhassani N, McMinn P, Halfond WGJ. Automated repair of mobile friendly problems in web pages. In: Proceedings of the 40th international conference on software engineering. ACM; 2018, p. 140–50. http://dx.doi.org/10.1145/3180155.3180262.

[11] Radeckỳ M, Smutnỳ P. Evaluating user reaction to user interface element using eye-tracking technology. In: Proceedings of the 2014 15th international carpathian control conference. IEEE; 2014, p. 475–80. http://dx.doi.org/10.1109/CarpathianCC.2014.6843651.

[12] Athukorala K, Glowacka D, Jacucci G, Oulasvirta A, Vreeken J. Is exploratory search different? A comparison of information search behavior for exploratory and lookup tasks. J Assoc Inf Sci Technol 2016;67(11):2635–51. http://dx.doi.org/10.1002/asi.23617.

[13] Nebeling M, Norrie MC. Responsive design and development: Methods, technologies and current issues. In: Web engineering - 13th international conference, ICWE 2013, aalborg, denmark, july 8-12, 2013. proceedings, vol. 7977. Springer; 2013, p. 510–3. http://dx.doi.org/10.1007/978-3-642-39200-9_47.

[14] Rhodes BJ. Margin notes: building a contextually aware associative memory. In: Proceedings of the 5th international conference on intelligent user interfaces. ACM; 2000, p. 219–24. http://dx.doi.org/10.1145/325737.325850.

[15] Aldalur I, Perez A, Larrinaga F. Customizing websites through automatic web search. In: Human-computer interaction - INTERACT 2019 - 17th IFIP TC 13 international conference, Paphos, Cyprus, September 2-6, 2019, proceedings, Part II, vol. 11747. Springer; 2019, p. 598–618. http://dx.doi.org/10.1007/978-3-030-29384-0_36.

[16] Winckler M, Cava RA, Barboni E, Palanque PA, Freitas CMDS. Usability aspects of the inside-in approach for ancillary search tasks on the web. In: Human-computer interaction - INTERACT 2015 - 15th IFIP TC 13 international conference, Bamberg, Germany, September 14-18, 2015, proceedings, part II. Lecture notes in computer science, vol. 9297, Springer; 2015, p. 211–30. http://dx.doi.org/10.1007/978-3-319-22668-2_18.

[17] Syed ZA, Helmick J, Banerjee S, Cukic B. Touch gesture-based authentication on mobile devices: The effects of user posture, device size, configuration, and inter-session variability. J Syst Softw 2019;149:158–73. http://dx.doi.org/10.1016/j.jss.2018.11.017.

[18] Liu C, Liu C, Mao H, Su W. Tilt-scrolling: A comparative study of scrolling techniques for mobile devices. In: Intelligent computing methodologies - 15th international conference, ICIC 2019, Nanchang, China, August 3-6, 2019, proceedings, part III. 11645, Springer; 2019, p. 189–200. http://dx.doi.org/10.1007/978-3-030-26766-7_18.

[19] Lehmann F, Kipp M. How to hold your phone when tapping: A comparative study of performance, precision, and errors. In: Proceedings of the 2018 ACM international conference on interactive surfaces and spaces. ACM; 2018, p. 115–27. http://dx.doi.org/10.1145/3279778.3279791.

[20] Chang JC, Hahn N, Kim Y, Coupland J, Breneisen B, Kim HS, et al. When the tab comes due: Challenges in the cost structure of browser tab usage. In: Proceedings of the 2021 CHI conference on human factors in computing systems. 2021, p. 1–15. http://dx.doi.org/10.1145/3411764.3445585.

[21] Hahn N, Chang JC, Kittur A. Bento browser: Complex mobile search without tabs. In: Proceedings of the 2018 CHI conference on human factors in computing systems. ACM; 2018, p. 251. http://dx.doi.org/10.1145/3173574.3173825.

[22] Tahir R, Raza A, Ahmad F, Kazi J, Zaffar F, Kanich C, et al. It's all in the name: Why some URLs are more vulnerable to typosquatting. In: 2018 IEEE conference on computer communications. IEEE; 2018, p. 2618–26. http://dx.doi.org/10.1109/INFOCOM.2018.8486271.

[23] Cao X, Masood A, Luqman A, Ali A. Excessive use of mobile social networking sites and poor academic performance: Antecedents and consequences from stressor-strain-outcome perspective. Comput Hum Behav 2018;85:163–74. http://dx.doi.org/10.1016/j.chb.2018.03.023.

[24] Pourghassemi B, Sani AA, Chandramowlishwaran A. What-if analysis of page load time in web browsers using causal profiling. Proc ACM Meas Anal Comput Syst 2019;3(2):27:1–23. http://dx.doi.org/10.1145/3341617.3326142.

[25] Andreadis I. Comparison of response times between desktop and smartphone users. Mobile Res Methods 2015;63. http://dx.doi.org/10.5334/bar.e.

[26] Marcotte E. Responsive web design: A book apart N 4. Editions Eyrolles; 2017.

[27] Ricca F, Leotta M, Stocco A, Clerissi D, Tonella P. Web testware evolution. In: 15th IEEE international symposium on web systems evolution. 2013, p. 39–44. http://dx.doi.org/10.1109/WSE.2013.6642415.

[28] Lewerentz A, Lindvall J. Performance and energy optimization for the android platform. 2012.

[29] Aldalur I, Perez A, Larrinaga F. MAWA: A browser extension for mobile web augmentation. In: Human-computer interaction - INTERACT 2021 - 18th IFIP TC 13 international conference, Bari, Italy, August 30 - September 3, 2021, proceedings, part IV. Lecture notes in computer science, vol. 12935, Springer; 2021, p. 221–42. http://dx.doi.org/10.1007/978-3-030-85610-6_14.

[30] Aldalur I. Web augmentation: A systematic mapping study. Sci Comput Programm 2024;232:103045. http://dx.doi.org/10.1016/j.scico.2023.103045.

[31] Valderas P, Torres V, Pelechano V. A social network for supporting end users in the composition of services: definition and proof of concept. Computing 2020;102(8):1909–40. http://dx.doi.org/10.1007/s00607-020-00796-8.

[32] Marticorena LG, Morales L, Antonelli L, Rossi G, Firmenich D. Development iterations based on web augmentation and context tasks. Multim Tools Appl 2022;82(8):11793–817. http://dx.doi.org/10.1007/s11042-022-13694-2.

[33] Bosetti GA, Firmenich S, Gordillo SE, Rossi G. An approach for building mobile web applications through web augmentation. J Web Eng 2017;16(1&2):75–102.

[34] Aldalur I. BEAUD: A browser extension to automatize end-user deeds. Softw Impacts 2023;17:100516. http://dx.doi.org/10.1016/j.simpa.2023.100516.

[35] Koidl K, Conlan O, Wade V. Cross-site personalization: assisting users in addressing information needs that span independently hosted websites. In: 25th ACM conference on hypertext and social media. ACM; 2014, p. 66–76. http://dx.doi.org/10.1145/2631775.2631798.

[36] Lv J, Zhong J, Chen W, Xiao Q, Yang Z, Li Q. Homepage augmentation by predicting links in heterogenous networks. In: Proceedings of the 27th ACM international conference on information and knowledge management. ACM; 2018, p. 1611–4. http://dx.doi.org/10.1145/3269206.3269249.

[37] Goffe L, Chivukula SS, Bowyer A, Bowen SJ, Toombs AL, Gray CM. Web augmentation for well-being: the human-centred design of a takeaway food ordering digital platform. Interact Comput 2021;33(4):335–52. http://dx.doi.org/10.1093/iwc/iwac015.

[38] Nebeling M, Speicher M, Norrie MC. CrowdAdapt: enabling crowdsourced web page adaptation for individual viewing conditions and preferences. In: ACM SIGCHI symposium on engineering interactive computing systems. ACM; 2013, p. 23–32. http://dx.doi.org/10.1145/2494603.2480304.

[39] Rivero JM, Urbieta M, Firmenich S, Witkin M, Serrano R, Cajas VE, et al. Improving legacy applications with client-side augmentations. In: Web engineering - 18th international conference, ICWE 2018, Cáceres, Spain, June 5-8, 2018, proceedings. Lecture notes in computer science, vol. 10845, Springer; 2018, p. 162–76. http://dx.doi.org/10.1007/978-3-319-91662-0_12.

[40] Gobert C, Todi K, Bailly G, Oulasvirta A. SAM: a modular framework for self-adapting web menus. In: Proceedings of the 24th international conference on intelligent user interfaces. ACM; 2019, p. 481–4. http://dx.doi.org/10.1145/3301275.3302314.

[41] Zemla JC, Tossell CC, Kortum P, Byrne MD. A Bayesian approach to predicting website revisitation on mobile phones. Int J Hum Comput Stud 2015;83:43–50. http://dx.doi.org/10.1016/j.ijhcs.2015.06.002.

[42] Tossell C, Kortum P, Rahmati A, Shepard C, Zhong L. Characterizing web use on smartphones. In: Proceedings of the SIGCHI conference on human factors in computing systems. 2012, p. 2769–78. http://dx.doi.org/10.1145/2207676.2208676.

[43] Biagiola M, Stocco A, Ricca F, Tonella P. Diversity-based web test generation. In: Proceedings of the ACM joint meeting on european software engineering conference and symposium on the foundations of software engineering. ACM; 2019, p. 142–53. http://dx.doi.org/10.1145/3338906.3338970.

[44] Hammoudi M, Rothermel G, Stocco A. WATERFALL: an incremental approach for repairing record-replay tests of web applications. In: Proceedings of the 24th ACM SIGSOFT international symposium on foundations of software engineering. 2016, p. 751–62. http://dx.doi.org/10.1145/2950290.2950294.

[45] Aldalur I, Larrinaga F, Perez A. ABLA: An algorithm for repairing structure-based locators through attribute annotations. In: Web information systems engineering - WISE 2020 - 21st international conference, Amsterdam, the Netherlands, October 20-24, 2020, proceedings, part II. Lecture notes in computer science, vol. 12343, Springer; 2020, p. 101–13. http://dx.doi.org/10.1007/978-3-030-62008-0_7.

[46] Kirinuki H, Tanno H, Natsukawa K. COLOR: Correct locator recommender for broken test scripts using various clues in web application. In: 26th IEEE international conference on software analysis, evolution and reengineering. 2019, p. 310–20. http://dx.doi.org/10.1109/SANER.2019.8667976.

[47] Stocco A, Yandrapally R, Mesbah A. Visual web test repair. In: Proceedings of the 2018 ACM joint meeting on european software engineering conference and symposium on the foundations of software engineering. ACM; 2018, p. 503–14. http://dx.doi.org/10.1145/3236024.3236063.