Enhancing Performance Aspect in Usability Guidelines for Mobile Web Application

Amirul Nurshuhada Romle
Universiti Teknologi Malaysia
Kuala Lumpur, Malaysia
shu_amirul@yahoo.com

Othman Mohd Yusop
Universiti Teknologi Malaysia
Kuala Lumpur, Malaysia
othmanyusop@utm.my*

Azri Azmi
Universiti Teknologi Malaysia
Kuala Lumpur, Malaysia
azriazmi@utm.my

Saiful Adli Ismail
Universiti Teknologi Malaysia
Kuala Lumpur, Malaysia
saifuladli@utm.my

Haslina Md Sarkan
Universiti Teknologi Malaysia
Kuala Lumpur, Malaysia
haslinams@utm.my

Nazri Kama
Universiti Teknologi Malaysia
Kuala Lumpur, Malaysia
mdnazri@utm.my

***Abstract*—The rapid growth of technology has boosted the usage of mobile devices to access a web application. These devices have various specifications such as screen sizes and resolutions. Responsive Web Design (RWD) approach bridges the gap of these differences by adapting the flexibility concept in screen sizes and resolutions, thus ensuring better website usability. Poor performance in website usability resulting in declination in number of users. Therefore several existing Usability Guidelines (UG) were invented mainly for a desktop-based website and unfortunately lesser studies have been carried out for a Mobile Web Application (MWA) especially the performance aspect of it. Henceforth, this paper focuses on enhancing the performance element of an existing UG by analysing, identifying, proposing, implementing, and measuring the new additional performance attributes in UG for MWA. We used *First Contentful Paint* (FCP), *Speed Index* (SI), *Time to Interactive* (TtI), *First Meaningful Paint* (FMP), *First CPU Idle* (FCI) and *Estimated Input Latency* (EIL) to measure the performances of two case studies and the result shows better score at 90-100 (fast-GREEN) with the proposed performance attributes compared to another website without it which averages at 50-89 (average-ORANGE) range.**

Keywords—Responsive Web Design, Usability Mobile Website, Mobile Web Application, Usability Guidelines

I. INTRODUCTION

Nowadays, there is a steady increase in the usage of smartphones and tablets to access the web application [1]. In fact, the usage of mobile devices to access web application is higher compared to a desktop user [2]. These mobile devices come with a different set of specifications in terms of screen sizes and resolutions to cater to the different types of end-users' preferences.

The introduction of Responsive Web Design (RWD) in 2011 has helped to provide the flexibility of user interface on a single website to adapt to different screen sizes of various mobile devices. Therefore RWD provides adaptability on any screen devices, be it on desktop or on mobile [3]. RWD is responsible to ensure the website content is presented in perfect layout in any device. Prior to this, the web designer needs to fix the width of the website for the desktop user and then zooms out to rearrange the component to adapt to mobile devices screen [4] and thus, the owner of a website created two different versions of the website to comply with different sizes of desktop and mobile devices screen [1]. As a result, this may cause misaligned of design and the possibility of the important information, links, buttons, and component of the website that should have been presented to users to be hidden [1]. With the abilities of RWD [4] such as fluid and flexible grid, flexible images and media, and media queries, ready to be implemented on a website across multiple devices, has helped the web designer to only focus on a single website rather than designing multiple layouts for various screen devices.

Due to the rapid growth of supported web application using RWD approach, people are starting using the web application regularly on mobile devices. A web application with good usability could help the user to complete their task effectively. According to [5], "usability is a necessary condition for survival on the web", and it is one of the most important criteria besides security concern [6]. There are a lot of usability guidelines (UG) that have been established for a desktop-based web application, however, there are only a few UGs for an MWA and only one UG covers the performance aspect for MWA [7]. Poor performance leads to losses of users and affects web usability such as page loading time [14]. Hoehle (2015) stated that effective performance in usability for MWA is a benchmarking of two alternative mobile apps using page loading time, success rate, number of attempts, number of errors, task accuracy, and task completion time. There are only a few structured UGs [6] and not clearly formulated UG in MWA that can be found in the scientific literature especially emphasising on the performance aspect of MWA as recommended in [7]. Hence, the UG in MWA should be improved in order to effectively deliver the content of web application to end-users and reduce users' disinterest. Few techniques will be used to improve the performance aspect of UG in MWA particularly in page loading time aspect which will be explained in subsection IV.

II. LITERATURE REVIEW

The International Standard Organization [10], defined usability as "the degree to which product can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a given context". [10] highlights usability measures through three components which are effectiveness, efficiency, and satisfaction.

According to [5], usability guidelines show slightly different elements of usability as compared to the ISO's definition of usability. Nielsen has divided usability into five main elements that are i) *Learnability* – how easy a user can complete the basic task for first time visit, ii) *Efficiency*

978-1-7281-6726-8/19/$31.00 ©2019 IEEE

– how quickly a user can complete a task, iii) *Memorability* – measure how easy to remember and use the website for another visit after some time, iv) *Errors* – measure the number of errors and its severity when a user uses the website, and v) *Satisfaction* – how pleasant for a user to use the website.

[9] in their finding emphasize that there is a number of aspects that are lacking in mobile usability guidelines and it does not cover all the usability aspect in mobile learning. The usability attributes that were not covered enough in mobile usability guidelines such as the feedback performance of the website. In addition, their usability guidelines are prone to mobile learning for a student to acquire knowledge through mobile devices.

[3] in their research has identified five common usability principles in evaluating RWD on e-commerce websites. In their research, they have analysed and compared three widely known usability principles. From their findings, they highlighted five shared keywords among the three principles are consistency, familiarity, flexibility, efficiency and aesthetically pleasing [3]. Their findings showed lesser feedback in performance under keyword effective feedback.

[9] in their research, has proposed a set of usability guidelines. The usability guideline for mobile applications was proposed by performing a comprehensive and systematic literature review, direct observation of the problem and the outcomes of usability evaluation with a mobile website. They later demonstrated the application of the proposed usability guidelines into two mobile applications scenarios that are i) the development of mobile application and ii) a research portal website. During the design stage of both mobile web applications, the usability guidelines are established and implemented into it.

Table I summarizes all the principles used in the usability guidelines for the mobile web application created by the researchers' understudy. Each principle may present the first layer of usability attributes whereas others [8],[9] have second layers of attributes which represent the details attributes of a principle. Mobile web application guideline attributes such as Effectiveness, Satisfaction, Efficiency and Feedback, and Performance from [5],[3],[9],[10] are the web application performance aspect in these usability guidelines.

TABLE 1. USABILITY GUIDELINES FOR MOBILE WEB APPLICATIONS

[10]	[5]	[8]	[3]	[9]
• Effective-ness • Efficiency • Satisfaction	• Learnability • Memorability • Errors • Effectiveness • Efficiency • Satisfaction	• Adjustability • Funability • Reliability • Satisfaction	• Consistency • Familiarity • Flexibility • Efficiency and Feedback • Aesthetically Pleasing	• Layout • Navigation • Design • Content • Performance

The performance attribute will be discussed next.

III. METHODOLOGY

The main primary goal of this research is to determine the issues in UG for RWD in MWA and further proposed an enhanced version of UG particularly in performance aspect. Therefore, the initial process of data collection and analysis is guided by the question: What is the usability guideline on responsive web design and its current issues? In general, the methodology in conducting this study will be divided into several processes as in Fig. 1.

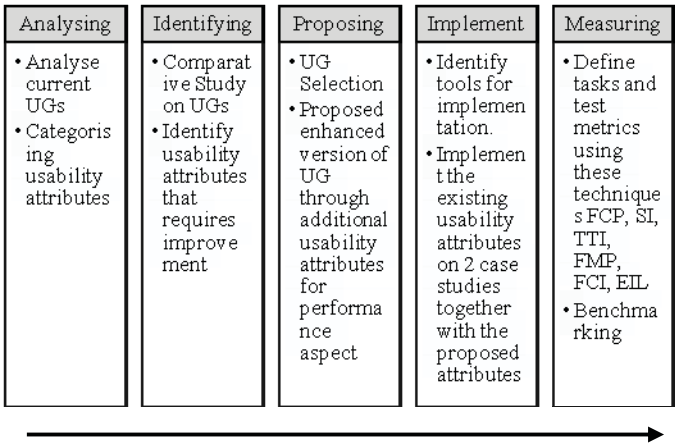

Fig. 1. The Methodology

A. Analysing

Analyses the current UGs for MWA from high prestige journals and comparative study was tabulated as in table 1. Each of the guidelines has its own attributes and category. The details for each attribute are omitted as detailing the information may exceed the maximum page allowed for this proceeding.

B. Identifying

Identified usability attributes that require improvement especially contributing to performance aspect. All elements/details of the involved usability attributes were outlined for next step.

C. Proposing

Identified the elements and combined them with existing UG [9] and label them as the proposed elements under performance usability attributes. The new proposed elements are explained further in subsection IV.

D. Implement

The proposed of enhanced version of UG is implemented on a selected website. Two case studies (website A and B) were derived from the same site and the experiment was conducted to capture results before and after implementation of a new additional elements for performance attribute in the UG.

E. Measuring

The performance of the enhanced version of UG is measured and benchmarked to determine its effectiveness based on Google Lighthouse Metric Index.

IV. PROPOSED PERFORMANCE ATTRIBUTES IN USABILITY GUIDELINES

The mobile usability guideline as established by [9] is used in this study as the base guideline. Based on the output from the previous objective of this project, the proposed enhanced mobile usability guideline seeks to improve the

performance attributes in mobile usability guidelines. Hence, four new items are added into performance attributes in the base guideline. The newly added items are:

- (G40) Bundle and minify JavaScript and stylesheet
- (G41) Add GZIP compression
- (G42) Use 'lazy loading' on images
- (G43) Place CSS at the top of the page and JavaScript at the bottom of the page

Table II summarizes the new enhancement attributes that should be added in the existing mobile usability guidelines i.e. [9]

requested asset, which is CSS and JavaScript. This is important to significantly reduce the page load time as current browser limits the number of simultaneous connections per each hostname into six requests. The remaining requests will be queued by the browser and thus affecting the page load time. Hence, combining multiple files into a single file could help in reducing the number of requests to the server and helps in reducing the page load time. The minification of JavaScript and CSS will remove unnecessary white space, comments and shortening the variable names to reduce the characters and eventually reduce the file size.

TABLE II. A PROPOSED OF AN ENHANCED VERSION OF MOBILE USABILITY GUIDELINE ATTRIBUTES

Layout	Navigation	Design	Content	Performance
(G1) Position the content at the centre of the screen (G2) Avoid using horizontal scroll (G3) Avoid use tabs and arrange the content vertically (G4) Place login button at the top of the screen (G5) Ensure the search bar to be visible (G6) Avoid table in input fields and use clear structure (G7) Use maximum of three buttons on a popup (G8) Place interactive button in the central position or at the bottom of a popup (G9) Place button on the right-hand side of the screen	(G10) Design navigation on menu to be simple and clear as possible (G11) Design menu to be one level of navigation (G12) Use less option item (max 8 items) and avoid long scrolling on the menu (G13) Make significant pages to be accessible from the homepage (G14) Give option to swiftly navigate back from the start page (G15) Reduce number of clicks required to go to every page (G16) Provide breadcrumbs to show current page location and enable user to switch between pages quickly (G17) Target external link to open in a new window (G18) Design menus by topics and usage scenarios to reduce search (G19) Make elements such as titles, links and navigation items descriptive and unambiguous self-explanatory	(G20) Make the design consistent, simple, clear and uniform (G21) Use a uniform design for different representations of the same product (e.g. website and mobile app) (G22) Focus more on crucial content (G23) Use icons that are easy to identify, memorable and visible (G24) Use size range between 7mm and 10mm for button	(G25) Ensure that the content, features and functionalities are similar for desktop version and mobile version (G26) Use simple texts and avoid long sentences (G27) Show minimal important content on homepage (G28) Arrange content by its priority (G29) Use automatic suggestion in search bar (G30) Enable user quick corrections by allowing revertible action (G31) Add confirmation popup whenever user perform change, edit or delete actions (G32) Remind user of the unsaved changes to prevent data loss (G33) Enable application to save current state and resume (G34) Provides function to fix typing errors automatically (G35) Correct non precise touch control by optimizing the interface (G36) Provide response on user touch actions (G37) Provide support common gestures to control apps (G38) Enhance passcode function on mobile app	(G39) Reduce redirects **PROPOSED ATTRIBUTES:** (*G40) Bundle and Minify JavaScript and Stylesheet* *(G41) Add GZIP compression* *(G42) Use 'lazy loading'on images* *(G43) Place CSS at the top of the page and JavaScript at the bottom of the page*

The proposed improvements focus on the performance attributes in mobile usability guidelines. The proposed improvements (G40-G43) are further discussed next.

A. (G40) Bundle and Minify JavaScript and Stylesheet

Bundling and minification of JavaScript and Stylesheet are a technique to improve request load time. Bundling is a feature that combines multiple files into one, while minification is a process to optimize the codes in a script or CSS. This technique improves the page load time by reducing the number of requests to the server and reduce the

B. (G41) Add GZIP Compression

GZIP is a data compression algorithm that compresses and decompresses files faster. It can compress almost any file type such as plain text, images, and code files. The main benefit of GZIP compression is that the compressed webpage will produce a smaller file size as compared to the uncompressed version. This will lower the loading time of the website. This performance boost will benefit the user of mobile devices and subsequently, give a positive impact on search engine ranking. When a browser visits a web server, it will send a header that contains information such as host,

and referrer. It also includes header 'Accept-Encoding' which contains information on the type of content it accepts. If the header 'Accept-Encoding' sent by the browser accept GZIP or compressed file, the server will return the compressed file if it is available. If the server does not have a compressed file, it will return the original file, which is the larger size from the compressed file.

C. (G42) Use Lazy-Loading on Images

Lazy loading is a technique that differs from the loading of the image to reduce the website loading time. This technique enables the images on the website to load only when the user needs to display them or when the user scrolls further down of the webpage. This will prevent the webpage from loading all images at once and avoid the user from waiting for too long for the webpage to be ready. Lazy loading will benefit the mobile user as it helps in saving the user bandwidth and speed up the webpage loading time[18]. Some of the innovation in the lazy loading technique is by presenting the low-resolution image first and then the actual image begins loading in the background. The actual images are then fully loaded when the user scrolls down the webpage. Lazy loading works by checking the current viewport that the user is currently using and then only load the images that are visible to the viewport. There are a lot of JQuery plugins that offer lazy loading functionality. This project used JQuery LazyLoad plugin to implement the image lazy loading feature in our case studies.

The function lazyload(); is used to initialize the lazy loading on images and the code is placed in JavaScript section. The CSS class 'lazyload' is then assigned in the image tags for the effect to take place on image files. There are two attributes in the image tag that this plugin uses: i) src and ii) data-src. The 'data-src' attribute contains the URL of the original high resolution of the image, meanwhile 'src' attribute contains the URL of the image in low resolution.

D. (G43) Place CSS at the Top of the Page and JavaScript at the Bottom of the Page

CSS or Cascading Style Sheets is a language that is used to describe reusable styles for presenting documents written in a markup language. It is a good practice to add the CSS stylesheet at the top of the page in the <head></head> tags. A browser needs instruction in the form of HTML and CSS to start laying out a webpage content. The rendering cannot begin until all the stylesheets have been downloaded. This is crucial in the mobile view to ensure that the user gets presented with proper layout of the website content rather than a blank page until the CSS finished downloaded.

JavaScript is a client-side scripting language that is mainly used for enhancing the interaction of the user with the webpage. It is widely used in game development and in mobile application development. JavaScript provides interactivity to the website but the downside of it is JavaScript prevents parallel downloads. When loading the JavaScript code, the browser will not begin to download other files. Hence, it is recommended to place JavaScript code at the bottom of the page and let the HTML content to display first and the placement of JavaScript in _Layout.vbhtml file of website version B. The JavaScript is placed at the bottom and right before the end of <body> tag.

V. CASE STUDIES APPLICATION, PERFORMANCE METRICS

A. Case Studies Application

An existing website developed by an XYZ company is used in this project. The XYZ company is an Australian company which is based in Malaysia. The website is a stall booking website that serves as a platform for the user to book a booth to sell their goods in the marketplace located in Brisbane, Australia. The website was built using VB.NET in MVC pattern on top of Microsoft .NET Framework 4.5. In this study, the website is modified and produced into two versions of websites. Website version A is modified based on the based guideline and website version B is modified based on the based guideline with the proposed new guideline.

The evaluation process is conducted by accessing selected five webpages from website version A and website version B using a mobile device. The five webpages are:

- Homepage
- List of Stalls Page
- Cart Page
- Terms & Conditions Page
- Gallery Page

Due to the nature of the project website domain, not all items in each attribute are fulfilled in this project. Table 3 shows the applicability of the items in each element that were fulfilled and applied on both websites.

TABLE 3. APPLIED ATTRIBUTES FOR BOTH VERSION OF WEBSITES (NEW ATTRIBUTES IS TICKED AND BOLDED)

Layout	**G1**	**G2**	**G3**	**G4**	**G5**	**G6**	**G7**	**G8**	**G9**	
Version A	✓	✓	-	✓	-	✓	✓	✓	✓	
Version B	✓	✓	-	✓	-	✓	✓	✓	✓	
Navigation	**G10**	**G11**	**G12**	**G13**	**G14**	**G15**	**G16**	**G17**	**G18**	**G19**
Version A	✓	✓	✓	✓	✓	✓	-	-	-	✓
Version B	✓	✓	✓	✓	✓	✓	-	-	-	✓
Design	**G20**	**G21**	**G22**	**G23**	**G24**					
Version A	✓	✓	✓	✓	✓					
Version B	✓	✓	✓	✓	✓					
Content	**G25**	**G26**	**G27**	**G28**	**G29**	**G30**	**G31**	**G32**	**G33**	**G34**
Version A	✓	✓	✓	✓	-	-	✓	✓	-	-
Version B	✓	✓	✓	✓	-	-	✓	✓	-	-
Content	**G35**	**G36**	**G37**	**G38**						
Version A	✓	✓	✓	-						
Version B	✓	✓	✓	-						
Performance	**G39**	**G40**	**G41**	**G42**	**G43**					
Version A	✓	-	-	-	-					
Version B	✓	✓	✓	✓	✓					

The mobile device used in the test is Huawei Mate 10 running on Android version Pie. The mobile device is connected by USB to the machine that hosted the website by using the 'Remote Debugging Android Device' feature that is provided by Chrome DevTools. Once the mobile device is connected, all of the inputs and interactions from the mobile device on the website can be monitored on the hosted/development machine through Chrome DevTools. Chrome DevTools is a tool that is built in the Chrome browser and is used to help to debug a website. Google Lighthouse is one of the available features in Chrome DevTools and it provides an audit function which analyses and measure website performances. The interface of the tools and connected mobile device are shown in Fig. 2.

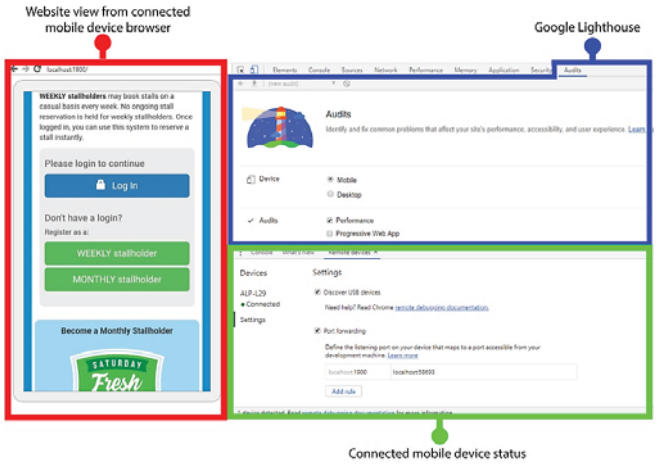

Fig. 2. Chrome DevTools Interface

B. Performance Metrics

The evaluation process between the website version A and website version B were conducted by using Google Lighthouse tool that is available as an extension in Chrome DevTools functionality. The tool is automated and is widely used by a web developer to analyze and measure the performance of a website [19]. The tool takes into consideration of six performance metrics and produces an overall performance score based on each metric individual result. There are six performance metrics that are measured by using Lighthouse tool. Each of these metrics captures some aspect of page load speed. The metrics are:

1) First Contentful Paint (FCP): FCP is a performance metric that measures the time from the start of the navigation until the browser renders the first bit of content from Document Object Model (DOM).

2) Speed Index (SI): SI indicates the average time for the contents of a page to visibly populated. The lower the SI, the better is the performance. SI is formulated by [20] as the following equation:

$$\text{Speed Index} = \int_0^{end} 1 - \frac{VC}{100} \qquad (1)$$

end = end time in milliseconds
VC = % visually complete

3) Time to Interactive (TTI): TIT metric indicates the time taken for a page to become interactive for the user. Low TTI of a webpage contributes to better performance.

4) First Meaningful Paint (FMP): FMP measures the time taken for a page's primary content to appear on the screen and visible to a user.

5) First CPU Idle (FCI): FCI measures the time when the webpage is minimally interactive. Both FCI and TTI are measured when the page is ready for user input.

6) Estimated Input Latency (EIL): EIL measures the time taken for the website to respond to user input during the initial time of page load. If the time taken is more than 50ms, the user may assume the webpage is broken or lagging (Gambhir, 2018).

VI. Result Findings

The evaluation results of mobile web application performance from both website version A and B are shown one after another in Fig. 3, 4, 5, 6 and 7 below:

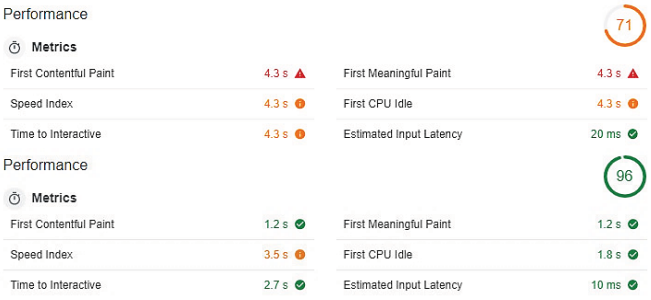

Fig. 3. Homepage Performance Results for Ver. A & B

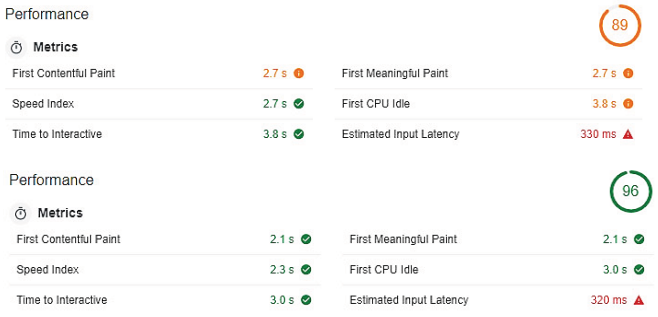

Fig. 4. List of Stalls Page Performance Results for Ver. A & B

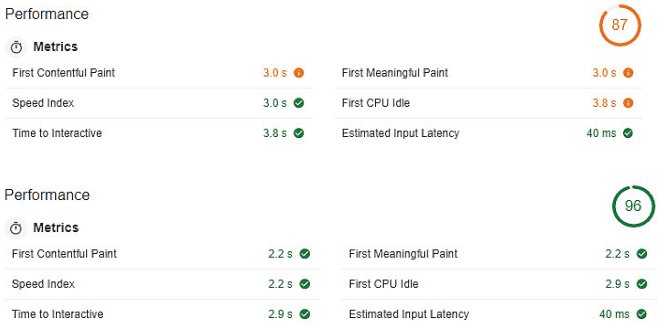

Fig. 5. Cart Page Performance Results for Ver. A & B

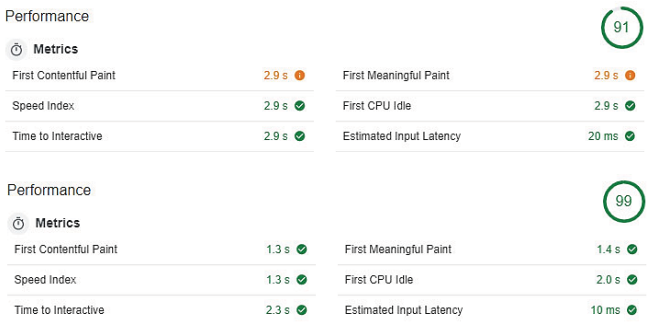

Fig. 6. Terms and Conditions Page Performance Results for Ver. A & B

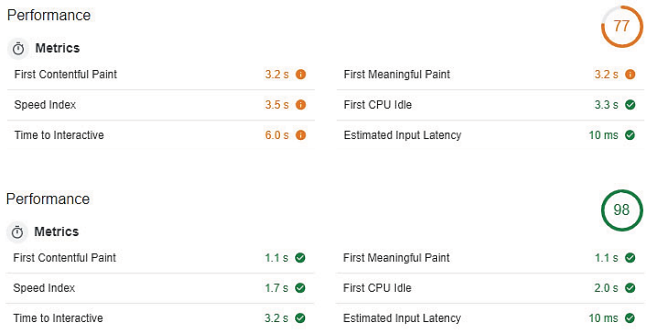

Fig. 7. Gallery Page Performance Results for Ver. A & B

Based on our experiment, the overall performance scores for website B are better than performance scores for website A. The average performance scores of website version B is 97 (GREEN) which is higher and whereas the overall performance scores for website A, without implementing new suggested attributes as shown in table 3 (Case Studies Application section), the average score is 83 (ORANGE). According to Google Lighthouse benchmarking, the performance scores are indicated as; 0-49 (slow-RED), 50-89 (average– ORANGE), and 90-100 (fast-GREEN). Therefore with the implementation of proposed performance attributes (G40-G43), yield a better result in the performance of the mobile web application.

VII. Conclusion

The results show that the performance scores for website version B are significantly better than website version A. Hence, it can be concluded that the proposed mobile usability guidelines help in improving the performance and usability of a mobile website. There are a constraint and limitation during conducting this study. There are very few resources and journals available that study about mobile usability guidelines. Most of the related topics were regarding usability guidelines on a static website. This study focuses on the use of responsive web design for a mobile website. The proposed enhanced mobile usability guideline has the possibility to be adopted in different scopes such as on native mobile applications.

VIII. Acknowledgment

The authors would like to thank Universiti Teknologi Malaysia (UTM) TransDisciplinary Research Grant under cost center number Q.K130000.3556.05G62 for funding this research. The fund is managed by Research Management Center of UTM.

References

[1] Pinandito, A. A.-z. (2017). Analysis of web content delivery effectiveness and efficiency in responsive web design using material design guidelines and User Centered Design. *nternational Conference on Sustainable Information Engineering and Technology (SIET)* , (pp. 435-441).

[2] Paternò, F., Schiavone, A. G., & Pitardi, P. (2016, June). Timelines for mobile web usability evaluation. In *Proceedings of the International Working Conference on Advanced Visual Interfaces* (pp. 88-91). ACM.

[3] Majid, E., Kamaruddin, N., & Mansor, Z. (2015). Adaptation of usability principles in responsive web design technique for e-commerce development. *Proceedings - 5th International Conference on Electrical Engineering and Informatics: Bridging the Knowledge between Academic, Industry, and Community, ICEEI 2015*

[4] Jiang, W., Zhang, M., Zhou, B., Jiang, Y., & Zhang, Y. (2014). Responsive web design mode and application. *2014 IEEE Workshop on Advanced Research and Technology in Industry Applications (WARTIA)* (pp. 1303-1306). IEEE.

[5] Nielsen, J. (2012, January 4). *Usability 101: Introduction to Usability*. Retrieved from Nielsen Norman Group: https://www.nngroup.com/articles/usability-101-introduction-to-usability/.

[6] Shitkova, M. H. (2015). Towards Usability Guidelines for Mobile Websites and Applications. *Wirtschaftsinformatik*, pp. 1603-1617.

[7] Hoehle, H. &. (2015). Mobile application usability: Conceptualization and instrument development. *Mis Quarterly*, 39(2).

[8] Hujainah, F. &.-h. (2013). Usability Guidelines of Mobile Learning Application. *Journal of Information Systems Research and Innovation*, 70-77M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[9] Shitkova, M., Holler, J., Heide, T., Nico, C., & Becker, J. (2015). Towards Usability Guidelines for Mobile Websites and Applications. *Wirtschaftsinformatik Proceedings.*

[10] International Organization for Standardization (1998). Ergonomic requirements for office work with visual display terminals – Guidance on usability (ISO Standard No. 9241-11:1998. Retrieved from http://www.iso.org/standard/16883.html.

[11] Sánchez, J. A., Starostenko, O., Castillo, E. A., & González, M. (2005, October). Generation of usable interfaces for mobile devices. In *Proceedings of the 2005 Latin American conference on Human-computer interaction* (pp. 348-348). ACM.

[12] Webcredible. 2007. 7 usability guidelines for websites on mobile devices. Webcredible Ltd.

[13] Chan, S. S., Fang, X., Brzezinski, J. R., Zhou, Y., Xu, S., & Lam, J. (2002). Usability for mobile commerce across multiple form factors. *J. Electron. Commerce Res.*, *3*(3), 187-199.

[14] Gallino, S. K. (2018). Why Retailers Should Care about Net Neutrality: The Impact of Website Performance on Online Retail. *Available at SSRN.*

[15] Pansanato, L. T. (2015). An evaluation with web developers of capturing user interaction with rich internet applications for usability evaluation. *Int. J. Comput. Sci. Appl, 4(2)*, 51-60.

[16] Isa, W. A. (2015). Evaluating the usability of homestay websites in malaysia using automated tools. *International Conference on Soft Computing in Data Science* , 221-230

[17] Kumar, R. &. (2017). Evaluating usability of a web application: A comparative analysis of open-source tools. *2017 2nd International Conference on Communication and Electronics Systems (ICCES)*, 350-354

[18] Brocato, M. (2019, April 10). *Preparing Your React App for Chrome's Native Support for Lazy Loading Images*. Retrieved from MoovWeb: https://www.moovweb.com/react-app-lazy-loading-image-chrome/

[19] Saverimoutou, A. M. (2018). Web Browsing Measurements: An Above-The-Fold Browser-Based Technique. *2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS)*, pp. 1630-1635.

[20] Hakim, M. (2016). Speed index and critical path rendering performance for isomorphic single page applications. *The 16th Winona Computer Science Undergraduate Research Symposium* , p. 41