THE EVALUATION OF THE TECHNICAL QUALITY OF WEB SITES WITH CONTENTS IN ARABIC AND SERBIAN LANGUAGES

Ali Elsadai*,
Milan Tair,
Nikola Savanović

Singidunum University,
Belgrade, Serbia

Abstract:

In this paper, authors present the result of an experimental study which covers the technical evaluation of 200 websites. Of the total, 100 are with content in the Arabic language and 100 are with content in the Serbian language. The technical evaluation covers the most common web development practices which are usually neglected. These include the application of the responsive page design, page download speed, the analysis of the most frequent HTML code errors and warnings, the use of CSS and JavaScript compression and the use of HTTPS certificates for secure connections. The test data is obtained with a script which collects information from a list of domains. The script processes the acquired information and performs code validation using the W3C HTML Validator tool. This paper presents a comparison of the evaluation results of web sites in Arabic and web sites in Serbian. The paper points out the most common technical problems and inadequacies of web sites in Arabic and Serbian languages. This research aims to help developers to maintain and improve their web page development processes and to avoid making common errors, which lead to increased incompatibility of web page with different browsers.

Keywords:

technical evaluation, browser compatibility, HTML code validation, web sites in Arabic, web sites in Serbian.

1. INTRODUCTION

Quality has become an essential requirement for creating web pages. Technical quality is as important as aesthetic and functional quality. Valid and clean code, free of errors is only one of the parameters which define valid web page code. In the past, the World Wide Web lacked standards for coding, web page design and browsers compatibility. Programmers adhered to their own experience to use practices they themselves deemed good. Because there were no standards to provide these guidelines, many web pages faced troubling issues with web browser incompatibility and even incompatibility between different versions of web browsers made by the same vendor. Alongside these incompatibilities, there was a general lack of interest in writing clean and maintainable code. This was mostly due to the fact that web sites were mostly static

Correspondence:

Ali Elsadai

e-mail:

aelsadai@singidunum.ac.rs

DOI: 10.15308/Sinteza-2019-351-359

and seldom required updating. Due to the fact that the Internet was rapidly expanding and the number of web sites was grooving almost exponentially, these issues became apparent early on.

In the effort to unify the development practices concerning the web, the World Wide Web Consortium (W3C) invests resources to create web standards, guidelines, and other supporting materials through a number of work groups. [1]

Presently, W3C is engaged in the process of specifying and standardising a number of APIs which should be made available to web pages in all browsers which adhere to these standards. [2]

Historically, web browser vendors have been quick to implement certain standards and specifications, but not all. [3] Nowadays, browser vendors have become motivated to implement newer standards and specifications more promptly than before. W3C has defined standards which provide a framework that allows all web pages to be rendered and handled in the same manner by all web browsers, regardless of version and vendor. These standards bring the World Wide Web to its maximum potential. They act as guidelines for both the developer and the browser vendors.

When web pages are developed in accordance with these guidelines, their behaviour is predictable and the graphical rendering should be unified. Because of this, it is important to comply fully with standard ways of using web components defined in the HTML specifications. W3C has created tools which check if web pages comply with these specifications. Those tools are called code validators. W3C provides validators for both the HTML and CSS code. These tools make compliance checks easier for developers. [4]

Even non-professionals can create web sites using web pages authoring tools. Such tools employ the WYSIWYG principle (What You See Is What You Get) to allow users to visually design their web pages without ever needing to type code. These authoring tools can generate web page HTML code with errors. These errors can be less or more critical in terms of standard compliance. Also, some professional web site developers do not use validation tools. This may lead to unclean code which might cause web site rendering errors. [5]

Another important factor is the download speed. This measurement depends on many factors. Some of the factors are the current traffic load on the server, network latency, communication channel speed etc [6]. In this paper, we explain how this issue was dealt with.

In this research we have compared web sites with content in Serbian and Arabic languages. Aside from the HTML code quality, we have analysed the compression ratio of CSS, and JavaScript files as well as a number of different metrics, among which is the text to HTML ratio. The text to HTML ratio is calculated by measuring the total number of visible/rendered characters on page with the amount of HTML code in the body, responsible for generating the rendered view showing that text. This ratio is always less than one. In text heavy web pages, the greater the value of this ratio the better the web page is in terms of this metric. [7] [8]

Usually, the better text to HTML ratio for pages with less HTML code overall improves the load speed and has positive influence on the user experience [9]. Search engines determine rankings of web pages in their search result indices by their quality of code as well as the content which is indexed [7].

One of the web site optimization techniques involves CSS and JavaScript code compression. Code compression is the process of removing any and all unnecessary characters from CSS and JavaScript files. This process must be conducted in such a way so that the functionality of the code remains absolutely the same, while the text size is reduced. In most cases, removing unnecessary whitespaces is sufficient. Whitespace characters include spaces, tabs, new lines and carriage returns. Additionally, in certain places in the code, semicolons can be removed when their removal will not affect the validity of the syntax [10]. Code compression is sometimes called minification. Ultimately, the final result is the reduction of the file size. This, in term, reduces the time required to download the file over the Internet from the server to the client. The reduction of the number of whitespaces in certain conditions help increase the page rendering speed because of the way web browsers parse CSS and JavaScript files before interpreting the code. [11]

This paper aims to give an insight into the quality of web sites created in different regions for different language and culture groups and compare their technical aspects, performance, usability and compliance with search engine optimisation guidelines. We give a comparative analysis of the results for each language group at the end of the paper alongside the results shown for each of the language groups separately. This is done in order to determine if there is a notable difference in quality and usability. The analysis of the comparison will show if there are clear indications where these metrics mostly differ. Also, we will list the most notable causes of the observed technical quality differences.

2. METHODOLOGY

The main purpose of this research is to analyze and compare the difference between evaluated technical qualities of web sites with contents in the Serbian and Arabic languages. The paper tests the technical aspects, code validity, performance, optimisation and usability of rendered web site pages on two screen sizes, one for the Desktop and one for the mobile device types.

The process of acquiring research data was automated using a script written in PHP. This script collects data from a predefined list of Internet domains. The list was compiled using web site ranking platforms and portals such as Alexa and a list of different web site directory indices.

For each of the domains in the list, a number of steps are performed in order to acquire the needed data. The steps in the data acquisition process are:

1. Download the HTML content of the home page;
2. Acquiring the screenshot of the home page in two screen size settings: Desktop and Mobile;
3. Calculating text to HTML ratio;
4. Download all required CSS and JavaScript files;
5. Perform HTML code validation;
6. Analyse CSS and JavaScript for compression;

The first step stores additional information such as the download time, latency etc. which is used in the future for better sorting web sites with similar evaluation results, for additional distinction.

The screen shots are made in typical screen resolutions of 1366x768 [12] and 750x1334 [13].

The text length for the text to HTML ratio is acquired using the JavaScript innerText property of the topmost page element, compared to the HTML code length. Both text values are trimmed before determining their length. However, these two string values are not normalized by removing unnecessary whitespaces before taking their respective lengths. The text to HTML ratio result is kept as a floating point value with all decimal points, without rounding. The tool used for performing the HTML code validation is the W3C offline HTML5 code validator tool. [14] The W3C Markup Validation service is a free service and is available online or as a standalone tool that can be ran on the user`s computer. Our automated analysis script uses the standalone program version.

The CSS and JavaScript analysis step gives additional information asides from determining if the CSS and JavaScript files are compressed or minified. The script stores data such as the total lines of CSS, total length of the CSS file, the number of whitespace characters, the ratio of whitespace characters to the total file length, the number of lines, the number of non-empty lines, the average number of characters per line, the ratio of the average characters per line to the total file length. The same types of measurements are taken from the JavaScript files as well. These measurements are used to determine the quality of the file compression or minification process. In general, files with large average characters per line and the smaller number of lines are considered minified if they have a small number of whitespace character to total file length ratios. These observations are made based on empirical observations and by comparing minified and non-minified source code files for both CSS and JavaScript documents. Code minification or compression can be used to influence the overall score of the web site in regards to the performance. This is due to the fact that compressed files are downloaded faster that non-compressed ones which saves bandwidth and takes less time to complete. Additionally, minified files are parsed more quickly before processing by web browsers.

One other piece of information is acquired about web sites during the first step. When attempting to download the HTML code of the web site at a certain domain, it is possible that the web server would redirect the browser to an SSL encrypted path. This information is used to confirm that the web site primarily uses the SSL encrypted HTTPS protocol for transferring information. This piece of information can be used to additionally differentiate two web sites with similar evaluation scores in the final step of the sorting process.

After all of the domains have been processed, an analysis of the acquired data is automatically performed. During this analysis, additional information is generated for each of the domains, relative to the average, total or other aggregate values generated from the collective data analysis.

Information that is additionally determined for each of the domains include the differences and deviations of CSS and JavaScript metrics as well as the differences and deviations of HTML and text related metrics from the appropriate aggregate value for those metrics, determined during the final analysis of the acquired data.

Key analytical insights for the evaluation process derive from the list of detected HTML code validation errors. Upon the completion of data acquisition for all domains, the list of all errors is collected and grouped and counted by their type. After performing this reduction for the first time, the result was manually evaluated and a list of common HTML errors was compiled. In order to easily create an automated mechanism for counting these errors in the final report, per domain, regular expressions were made for each of the common errors. The final list of error includes the count of the type of error and not the count of the specific error. We have made regular expressions that group errors by type. The grouping of HTML validation errors is done using the preg_match function in PHP for each of the detected errors. Errors that match any of the regular expressions are counted against the type of that specific error in the report.

3. EXPERIMENTAL RESULTS

After running the automated acquisition and analysis script with our list of domains, we have reviewed the results and created a report. This section presents our research results.

HTML code validation analysis results

Modern web browsers have made progress in parsing and rendering HTML code even when it has many errors and inconsistencies. This fault tolerance does allow most web pages to be rendered even though there might be errors with certain elements. However, not all broken HTML code can be rendered properly. [5]

Invalid code can greatly impact the user experience. This occurs when code segments with error behave in different ways from expected in different web browsers [15]. Because of this, we consider the HTML code validity an important measure. The results of the analysis of all the tested domains are presented herein.

The average number of HTML validation errors of all 200 tested web sites is 23.29. Of all the web sites, 67% have less errors that this average value. Even though 67% of web sites with less than 23.29 errors statistically show that the majority of web sites are well developed in terms of HTML code validity, the real issue is the type of the error and not the error count. In this research, we have not conducted an analysis of the impact of particular types of errors on usability of the rendered website.

This result can be further broken up into web sites in Serbian and web sites in Arabic languages. The average error count of web sites in Serbian is 23.75 and 66% of those web sites have less than the mentioned average error count. The average error count of web sites in Arabic is 22.82 and 67% of those web sites have less than the mentioned average error count. According to this, there is no significant difference between web sites in Serbian and web sites in Arabic.

Text to HTML ratio analysis results

A small text to HTML ratio can indicate that a web site has an unnecessarily large amount of HTML code that is rendered into a page with small amounts of text. This can be an issue because web browsers need more time to process, parse and render a lot of HTML elements that interact and influence each other when styled with different CSS rules. This can impact the page load speed. Page load speeds impact the user experience is a significant ranking factor for search engines, such as Google's Panda update [10]. This impacts text heavy web sites, such as blogs. However, some web sites are oriented towards multimedia such as images and videos. Those web sites might not have a lot of text, but do require a lot of HTML code to render all the required elements that hold the multimedia content. Since there is no clear way to automatically determine the type of the content, this metric is taken into account with reserve. Regardless, the results can be aggregated and presented. The presentation of the results of the analysis of this metric is given herein.

By analysing the acquired results, we have determined that most web sites have a text to HTML ratio which is close to the average value of 5.813%. Also, the average text content length is 7867 characters, while the average HTML content length is 202406 characters. Some web pages optimize their HTML code. Optimization reduces the HTML code length by removing unnecessary whitespaces [16]. Therefore, not all web sites can be directly compared by this metric. This should be taken into account when reviewing the results of the analysis. By analysing these results, we can conclude that 32% of web sites have a text to HTML ratio which is higher than the average value. Higher values of the text to HTML ratio are better. Therefore, we can conclude that, regarding this metric, the majority of web sites are technically worse as they have text to HTML ratios which are lower than the average.

We can break these results into those for web sites in Serbian and web sites in Arabic. The data is show herein.

Serbian websites achieved the following results:

Table 1. Comparison of page text and code analysis results

Average values for:	Arabic sites	Serbian sites
Text length (characters)	7883	7850
HTML length (characters)	198432	206379
Text to HTML ratio	5.76%	5.86%
Average load speed (sec)	1.11	1.33

Along with these comparative results, we have made additional conclusions:

- 31% of the Serbian web sites have higher text to HTML ratio than the overall average.
- 35% of the Arabic web sites have higher text to HTML ratio than the overall average.

According to these results, we can conclude that there is no significant difference between the text to HTML metrics of web sites in Serbian and Arabic.

CSS optimization analysis results

Since web pages are styles using CSS, having CSS files downloaded separately or in a web package alongside the page`s HTML content is unavoidable. To reduce the page load and rendering speeds, it is important to deliver these files as quickly as possible [9]. It is important to parse them, process them and apply the defined styles as quickly as possible. Otherwise, the user experience would be impacted. In order to reduce this impact, CSS should be optimized, regardless of the method of CSS delivery. When web sites observe good practice where CSS files are delivered as one combined CSS file or at least as a few separate files, those files need to be downloaded before they can be processed. It is recommended that those files are optimised to reduce the load time during the first page load, until the files are cached. [17]

There are different methods of CSS code optimisation and compression. As explained earlier in the paper, most are based on removing unnecessary characters such as whitespaces, the last semicolon in the style sheet set etc. This method is the one that our script is considering when determining if the CSS code was optimised. However, there are other methods and steps that can further deduce the syntax while retaining the original intent of the code.

Consider the example of a CSS style definition for a div element given in the code listing below.

```
div {
    margin-top: 5px;
    margin-right: 10px;
    margin-bottom: 5px;
    margin-left: 10px;
}
```

Listing 1. An example of CSS before optimization

The CSS code shown in Listing 1 can be optimized by replacing the separate edge margin properties with a combined margin property as shown in Listing 2.

```
div {
    margin: 5px 10px 5px 10px;
}
```

Listing 2. An example of CSS after optimization

Further, this code can be compressed by removing unnecessary whitespaces and symbols. The final optimized and compressed CSS code which retains the initial intent and function is shown in Listing 3.

```
div{margin:5px 10px 5px 10px}
```

Listing 3. An example of optimized CSS after compression

The combination of CSS optimization and compression is referred to as code minification. The idea is that the original source code footprint can be minified (minimized) to increase the speed of delivery over the network and the time of processing, parsing and applying to the loaded page [18].

For better performance, all the external CSS files should be included in one file and minified. In a study done on the 10 most visited web sites in US, the process of CSS code minification resulted in a 21% file size reduction. [19]

When we observe statistical information from the CSS analysis, we can see that the average number of CSS lines per domain is 561.6. The average number of characters per domain is 112467.6 and the average ratio of whitespaces to total file length is 2.58%, while average ratio of the average number of characters per line to total file length is 9.37%. Only 22.5% of web sites have a higher than average white space to character ratios. This indicates that most web sites probably employ a method of CSS file minification. This is confirmed by the fact that 20.5% web sites have a higher than average character per line to total length ratios.

The following table shows average values for this metric.

Average values for:	Arabic sites	Serbian sites
CSS line count	206	926.53
CSS character count	130030	94926.75
White space count	3046	4937.33
Characters per line	17883	3835.97
White space to character ratio	1.738%	3.445%
Characters per line to total length ratio	10.539%	7.18%

JavaScript optimization analysis results

JavaScript is an interpreted scripting language, usually executed on the client side. An example of an implementation used on the server side is Node.js. We focus on the JavaScript code that we can acquire, from the client side. When the web page is delivered, it usually requests more resources to be obtained in order for it to function properly. Asides from the already mentioned CSS style sheets, the page can request additional JavaScript files. When these JavaScript files are delivered from the server to the client, they are executed on the client side, in the browser.

JavaScript can be used to enhance many functionalities of a web page. Some of them include the possibility of dynamically loading additional content into the page. This allows the original page to be smaller and hence faster to download and render the initial structure, while all additional data is loaded asynchronously and shown at a later time. This increases the load time of the initial web page, but can sometimes lead to an impression that the web page takes longer to be ready for use after it seems to have been loaded.

As with CSS files, it is important to deliver JavaScript files as soon as possible. Therefore, the practice of minification can be applied to JavaScript files as well. The purpose is the same as with CSS files. However, the method is slightly different, but not as much.

One step in JavaScript file minification is the removal of unnecessary white space characters and symbols. Other steps include the minification of local variable names within functions. This is done to decrease the overall size of the file.

Our results show that JavaScript files have an average of 1413 lines and 630592 characters. The average whitespace count is 26218. This means that the average whitespace to total length ratio 3.341%. The average number of characters per line of code is 2876. And 1.57% average characters per line to total length ratio.

Only 24% of web sites have a higher than average white space to character ratios. This indicates that most web sites probably employ a method of JavaScript file minification.

Average values for:	Arabic sites	Serbian sites
JS line count	1050	1764
JS character count	660979	601051
White space count	24080	28289
Characters per line	4277	1476
White space to character ratio	2.5888%	4.079%
Characters per line to total length ratio	2.3796%	0.7738%

Responsive design analysis

Responsive web design is the ability of a web site to adapt its content to different screen sizes. This improves user experience. It is especially important for devices with smaller screen sizes. Responsive web sites allow for better web site discoverability. The responsive web design analysis was the only step that was done manually and not with an automated tool. The acquisition of two screenshots of each web page in two characteristic resolutions was automated. After analysing all web sites, we have found that 80% of web sites are responsive. This means that they do provide a visually adequate representation of their content to users of devices with smaller screen sizes. When we compare web sites in Arabic and web sites in Serbian, the results are almost identical. We have found that 81% of Arabic web sites are responsive and that 80% of Serbian web sites are responsive. In conclusion, Arabic and Serbian web sites do not differ significantly regarding this metric.

Secure communication analysis

Hyper Text Transfer Protocol Secure (HTTPS) provides a method of connecting to a web server in a secure manner, using encryption. Browsers help users distinguish between secure and insecure web sites by showing the HTTPS protocol in the URL and visually by marking the URL in a different colour. Most commonly, secure web sites have their URLs marked in green. [20] According to Statoperator, 48.71% of Alexa top 1,000,000 web sites use the HTTPS protocol and are secured using valid SSL certificates. [21] ISRG has found that 73.7% of the web sites that have been visited using Firefox are secured and use the HTTPS protocol. [22]

From our results, we can conclude that 74.5% of analysed web sites do use the HTTPS protocol and are secured with a valid SSL certificate. Of all Arabic web sites that we have tested, 78% are secured using SSL certificates. Of all Serbian web sites that we have tested, 71% are secured using SSL certificates. Once again, this shows that there is not a very significant difference between web sites in Arabic and web sites in Serbian.

4. DISCUSSION

By reviewing the experimental results, we can conclude that there is no significant difference regarding the technical quality of Serbian and Arabic web sites. As we have elaborated in the experimental results section, on average, there were 24 HTML code errors in Arabic and 23 HTML code errors in Serbian web sites. Of all 200 tested web sites, 67% had less than the average number of HTML code errors. Since at least a portion of the content is generated manually, and not as a result of program generated structures, we can assume that developers either use validation tools or are reasonably well trained to write clean and good quality source.

As for the page content analysis, we have assumed that web sites are text dominant. Therefore, the text to HTML ratio was considered a factor. The average text to HTML ratio for all tested web sites was 5.813%. Only 33% of web sites had a higher than average text to HTML ratio. As with the previously discussed metric, the difference between Serbian and Arabic web sites concerning the text to HTML ratio is also insignificant. Arabic web sites have had a 5.86% text to HTML ratio, while the result for Serbian web sites was 5.76%. These results show that the tested web sites require a lot of HTML code in order to render a small amount of textual content. Such results can even affect page load speed and web site ranking. [8]

Regarding the CSS minification, Serbian web sites have, on average, four times the number of CSS lines than the average of all tested web sites, but less than the average number of characters in all CSS files per web site. As shown in the experimental results section, the difference between Serbian and Arabic web sites in regard to this metric are not significant. Also, the average number of whitespace characters and the ratio of whitespace characters to the total file lengths indicate that, on average, web sites do employ minification.

The results are similar in regard to JavaScript minification, as both Serbian and Arabic web site groups show signs that they use source code minification techniques.

Compared to the statistics on the use of HTTPS by the top one million web sites on the Internet, which concludes that 48.71% of them use encryption, the top Serbian and Arabic web sites show much better results in this regard, because 74.5% of analysed web sites use SLL to encrypt their traffic.

5. RECOMMENDATIONS

Based on the results of our analysis, we can make certain recommendations that can help increase the technical quality of web sites. Since the validation of the HTML code leads to cross browser consistent behaviour, faster web site rendering, and fewer errors, it is an important aspect in technical evaluation. Therefore, we suggest that all developers should frequently check the output HTML code of their web pages using HTML code validation tools such as the World Wide Web Consortium`s HTML Validator. This tool is free and it is available as both an online service hosted by the W3C and a standalone executable package for major platforms. Developer can integrate this tool into their test environments and have it check generated web pages whenever tests are automatically performed upon committing new changes to the repository.

From our results, it would seem that CSS and JavaScript minification is already a practice employed by developers. Those who do not yet minify their files should consider the benefits of this practice. Also, automation of code minification is a well documented and easy to set up. The automation can be integrated into build scripts that can perform automated CSS and JavaScript minification at the same time while testing HTML code validity, as mentioned in the previous paragraph.

This practice does not only reduce the file size and decrease the file download speed, but also leads to faster web page rendering on the client side. It can have a direct positive impact on the user experience. Therefore, it is not only a metric that leads to the better technical score, but also the overall usability and quality of the produced web site.

Generally, it is observed that web sites use much more HTML code than is needed to generate a smaller amount of textual content. This practice is bad if we consider web pages as primarily sources of textual content. However, modern web sites also use a lot of multimedia content which cannot be treated as text and therefore a lower text to HTML ratio is expected. Still, it is not recommended. We recommend that developers take the time to better design their web page components in such a way that less HTML code is required to generate the same full visual representation of their target content, be it text, images, animations or videos. Less HTML code impacts the overall web page size, which leads to less bandwidth use, faster web page delivery and faster parsing and rendering. Overall, higher text to HTML ratios with less HTML code in general is a target developer should aim for.

6. CONCLUSION

We can confirm that the technical quality of top web sites in Serbian and Arabic languages is good. The difference in quality of these two groups is insignificant. The majority of web sites have less than the average number of HTML code validation errors. We recommend the use of HTML code validation tools to check for error before publishing web sites. CSS and JavaScript file minification is a practice that the majority of top Serbian and Arabic web sites already employ. Both the HTML code validation and CSS and JavaScript minification are important metrics in the technical evaluation. Another technical metric that we have analysed is the use of encryption. Compared to the top web sites on the Internet, both Serbian and Arabic sites show better results. Asides from technical metrics, we have evaluated one of the most important user experience metrics for modern sites. Responsive web page design is an important factor in determining the quality of a web site. It directly impacts web site accessibility and therefore user experience. We have concluded that the great majority of Serbian and Arabic web sites are responsive. Their share is around 80%. The top one hundred Serbian and the top one hundred Arabic web sites are technically very well made. There is still space for improvement. We have given recommendations regarding steps that would lead to solving the most common issues. Applying recommended steps in the development process would help produce technically better quality sites.

REFERENCES

[1] W3C, “W3C Groups,” [Online]. Available: https://www.w3.org/Consortium/activities. [Accessed 25 03 2019].

[2] M. Jasim, Building Cross-Platform Desktop Applications with Electron, Birmingham: Packt Publishing Ltd, 2017.

[3] R. W. Sebesta, Programming the World Wide Web, Boston: Addison Wesley, 2002.

[4] S. Jenkins, Web Design All-in-One For Dummies, Hoboken: John Wiley & Sons, 2013.

[5] W3C, “Why Validate?,” W3C, 2009. [Online]. Available: https://validator.w3.org/docs/why.html. [Accessed 11 2 2019].

[6] A. Harvey, “Why page download speed is important for your SEO,” 11 02 2019. [Online]. Available: https://b2bquotes.com/en/page-download-speed-important-seo. [Accessed 28 02 2019].

[7] B. Demaria, “Are Text to HTML Ratios Important?,” 6 3 2018. [Online]. Available: https://www.woorank.com/en/blog/are-text-to-html-ratios-important. [Accessed 11 3 2019].

[8] S. Draganus, “What is Text to HTML Ratio? Does it Affect Your Content?,” 2017. [Online]. Available: https://www.georanker.com/how-to-determine-texthtml-ratio-for-your-website. [Accessed 24 2 2019].

[9] A. B. King, Website Optimization, O'Reilly Media, 2008.

[10] Various, “Minification (programming),” 13 2 2019. [Online]. Available: https://en.m.wikipedia.org/wiki/Minification_(programming). [Accessed 15 3 2019].

[11] N. C. Zakas, High Performance JavaScript: Build Faster Web Application Interfaces, O'Reilly Media, 2010.

[12] W3Schools, “Browser Display Statistics,” W3Schools, [Online]. Available: https://www.w3schools.com/browsers/browsers_display.asp. [Accessed 18 3 2019].

[13] JKielty, “Most used smartphone screen resolutions in 2019,” 9 1 2019. [Online]. Available: https://deviceatlas.com/blog/most-used-smartphone-screen-resolutions. [Accessed 11 3 2019].

[14] W3C, “W3C Markup Validator,” W3C, [Online]. Available: https://validator.w3.org/docs/. [Accessed 11 1 2019].

[15] S. Chen, D. Hong and V. Y. Shen, “An Experimental Study on Validation Problems with Existing HTML Webpages,” in *Proceedings of the 2005 International Conference on Internet Computing, ICOMP'05*, Las Vegas, 2005.

[16] Google, “Optimize HTML,” 3 8 2015. [Online]. Available: https://developers.google.com/speed/pagespeed/service/OptimizeHtml. [Accessed 11 3 2019].

[17] S. Thapar, S. K. Chowdhary and D. Bahri, “Compression and Optimization of Web-Contents,” in *Proceedings of 3rd International Conference on Advanced Computing, Networking and Informatics. Smart Innovation, Systems and Technologies*, New Delhi, 2015.

[18] M. Hague, A. W. Lin and C.-D. Hong, “CSS Minification via Constraint Solving (Technical Report),” CoRR, vol. abs/1812.02989, 2018.

[19] Yahoo!, “Best Practices for Speeding Up Your Web Site,” [Online]. Available: https://developer.yahoo.com/performance/rules.html. [Accessed 5 3 2019].

[20] Mozilla, “How do I tell if my connection to a website is secure?,” [Online]. Available: https://support.mozilla.org/en-US/kb/how-do-i-tell-if-my-connection-is-secure. [Accessed 11 2 2019].

[21] StatOperator.com, “HTTPS usage statistics on top 1M websites,” 3 6 2017. [Online]. Available: https://statoperator.com/research/https-usage-statistics-on-top-websites/. [Accessed 18 3 2019].

[22] Internet Security Research Group, “Let's Encrypt Stats,” [Online]. Available: https://letsencrypt.org/stats/. [Accessed 9 3 2019].