VAWKUM Transactions on Computer Sciences
http://vfast.org/journals/index.php/VTCS@ 2023 ISSN(e): 2308-8168, ISSN(p): 2411-6335
Volume 11, Number 1, January-June 2023 *pp: 123-137*

Identification of an Optimized Google Page Speed Audit-Rule-Sequence to Optimize Page Speed

Abdul Ghafar[1], **Fazeel Abid**[1*], **Mohsin Ashraf**[2*], **Abdul Jamil**[1], **Ali Abbas**[2], **Farah Rauf Malik**[2]

[1]Hassan Shoaib School of Management, University of Management & Technology, Lahore, Paksitan; [2]Faculty of Information Technology & Computer Science, University of Central Punjab, Lahore, Pakistan

Keywords: *Search Engine Optimization, Page Speed Optimization, Page Load Time Optimization, Search Engine Ranking, Digital Marketing*

Journal Info:
Submitted:
January 15, 2023
Accepted:
March 16, 2023
Published:
April 2, 2023

Abstract World Wide Web is a collection of online resources and websites, including e-commerce, social sites, educational content, etc. To find relevant online resources, people search these by using search engines by providing their desired keywords. After filtering those keywords, search engine list the most relevant websites which are more optimized and efficient in terms of loading speed. Search engine optimization (SEO) is a set of techniques used to make a website optimized and relevant to those keywords and set the website's rank. An online resource or website will be on the top of the search result set if it has a higher rank in search engines. Page speed is one of the most important on-page search engine optimization techniques that is used to make web site efficient in load time, so the user will get the content of the websites in a minimum time. Google has set page speed as the main factor in a higher ranking in search engines. Getting higher page speed is not an easy task, as several performance matrices must be optimized to get efficient loading time. There are many audit rules which are irrelevant or have less impact on the performance score. So selecting audit rules to be optimized is one of the main decisions before starting page speed optimization work. It will be a waste of time to investigate audit rules for their impact on performance scores. In this paper, we have analyzed all of the audit rules and identified the most important and relevant audit rules in optimizing page speed. A tool is used to generate the best sequence of relevant audit rules based on weighted performance benefit scores in the execution of each audit rule. The same audit rule sequence is applied on five different websites and results in more than 80% improvements in performance scores by applying the first three audit rules only and above 90% performance scores obtained by using the first five to seven audit rules in our proposed audit rules sequence.

***Correspondence Author Email Address:**
fazeel.abid@umt.edu.pk; mohsin.ashraf@ucp.edu.pk

1 Introduction

Search engine optimization (SEO) is a technique to optimize the code and content of a website within the guideline provided by search engines and to achieve a higher rank in search engine result pages so that search engine crawl, index, and understand the content of a website easily [1]. The search engine is the main gateway to access any website, as 80% of new users come from web searches, and if the website is not available on search engine result pages, then it loses 80% of users. The first page of the search result is more than any other page, as 99% of users just like the first page, and only 1% of users visit the second and third pages [2]., so users accessing the page beyond the third are negligible. Therefore rank in search engines is the most important element to get visitors from search engines, which can be achieved by SEO [1].. The search engine returns results against organic as well as inorganic searches. Organic search is the normal search engine process to index the web pages of sites based on various keywords and site metrics such as loading speed, keyword relevancy, backlinks, etc. Whereas inorganic search is the process of listing paid advertisements on web pages based on keywords. SEO is further divided into On-Page and Off-Page SEO. On-Page SEO is a collection of various techniques to optimize the website by configuring various components available on a web page, which are normally controllable by the developer [3]. The purpose of On-Page SEO is to normalize the web page content in such a way that makes search engines relate its content with different keywords as well as optimize and reduce the time of loading those content to the user in an efficient way. Optimizing the page load time is one of the major tasks in on-page SEO to rank it better in search engines. Off-Page SEO is the process of setting various components that are outside the web page, such as back-linking, use of social sites, etc. [4]. In this paper, we are discussing page speed optimization, which is a component of On-Page SEO to increase the load speed and improve the load time of various elements of the web page.

Web Speed optimization is used to improve the website's performance so that it loads within a reasonable time and the user does not complain about the slowness of the web page. If a site responds slowly, then its users spend less time and skip to another site, so page loading speed is an important ranking factor considered by Google [5]. Normally the mobile user will leave the site if it takes more than 3 seconds to load [5], and more than 53% of visitors leave the site if it does not load within 3 seconds and never return to the same site in the future if it takes more than 4 seconds to load [6].

The objective of Web Speed optimization is to reduce the page load time, which is the time to display all of the content on the browser. Web site Performance and speed mainly depend on optimized CSS, JavaScript, images, and HTML, as well as removing render blocking scripts in above-the-fold, responsive sites, use of the cache, the minimum number of server requests, and fast server response [7]. To find the optimization score of a web page, several tools such as Google PageSpeed, YSlow, Pingdom, Gtmatrix, and WebPageTest can be used to perform the score of audit rules provided by them. Pingdom, Gtmatrix, and WebPageTest are based on Google PageSpeed and YSlow, but with a more detailed and comprehensive optimization analysis report. Some of these services are paid, but most of them are open-source and free. The selection of tools to optimize web site never affects the optimization process significantly. Audit rules in Google PageSpeed and YSlow are different, but they are the same in meaning; for example, Google PageSpeed has a different rule to minify CSS and JavaScript, whereas YSlow has a single rule to minify CSS and JavaScript.

In this paper, we follow the audit rules given by Google PageSpeed, as Google is one of the major search engines having a market share of 70% which is higher than the share of any other search engines such as Baidu, Bing, Yahoo, etc. [2].

2 Related Work

Website load speed could serve as a success parameter for websites and also explain slow websites will severely affect users' experience [8]. Various types of optimization, such as server, application, and database to improve the performance and speed by using various optimization rules and found an accumulative result of distinct improvement in load time and website response [9]. Different optimization guidelines were given by PageSpeed and YSlow on a sample portal, and found improvement in page speed

by adjusting different website components [10]. They just tried to identify the overall impact of optimization and compare them with government mobile applications. Website must be optimized for desktop as well as for mobile devices, but now Google has preferred to optimize the website for its high ranking in the context of the mobile device only, as the number of mobile users increased more than for desktop systems [5].

Perceived page load time is a more realistic value, but optimization with it is not an easy task, as it is based on the importance of the content from the user's point of view [11]. It is greatly affected if there are heavy JavaScript and images in HTML, so optimization of JavaScript and images must be the top priority instead of CSS. The value of perceived page load time is better on a high-speed network instead on a low-speed network, optimization of the network is not more flexible, so it will not be considered a good matrix than Speed Index. [12] mentioned a speed index matrix for optimizing a website, but calculating the speed index is a difficult task, as it involves some complex unrealistic calculations.

To reduce the load time, content on the servers and proxy servers, such as images, scripts, and HTML, must be optimized with an efficient mechanism of a multi-level cache control system [13]. The time to load is highly dependent on the network latency and the size of the content to download. They suggested that different types of cache, such as web application cache and browser cache, must be implemented to reduce the number of HTTP requests [14]. The work in [15] identifies the web performance by using audit rules given by Google PageSpeed and YSlow and categorizes them into high-scoring, medium-scoring, and low-scoring and focuses more on high-scoring audit rules for improvement in the performance of a website. He investigated that page performance is depended on three measures such as page structure, page weight, and page request, so they mentioned three types of optimization, i.e., front-end optimization, back-end optimization, and network optimization, to optimize a sample website. The mode in [15] found the main barrier in optimizing mobile devices is the load speed, so optimizing a website for load speed is important for mobile applications.

With the development of new technologies and web development techniques, the average page size is becoming larger concerning page weight, page request, and page structure as compared to the past, so a website with new technology is much slower without proper optimization [15]. So slow page loading leads to a negative user experience and, ultimately, a lower rank in search engines. Site loading speed is much important as the user never accept slow-loading sites, which leads to high bounce rates and low performance of the website, and these inversely affect the site ranking in search engines [16]. So search engines consider web performance as an important metric, which is based on page speed, responsiveness, and server parameters, to rank a web page [17]. Amazon and Google report losses in the 0.6-1.2% range for an increasing delay of 0.4-1 second [12]. To improve web performance, several efforts, such as network protocols, web architecture, use of best practices, and tools to develop better websites, must be implemented [8]. In this paper, we try to find those audit rules which give maximum performance benefit depending upon the page structure, page weight, and many page requests so that effort to improve page speed in less time instead of focusing on audit rules with less performance benefit as in [15]. So just focusing on less number of audit rules, instead of exploring all of the audit rules, will optimize the website with less amount of time. A non-optimized website is tested with different audit rules to identify the most performance benefits audit rules. An automated tool is developed which identifies the important audit rules for that website. The same procedure is applied to 10 websites, and all of the results are accumulated to find the average to find a sequence of audit rules with maximum performance benefit. The same sequence is applied to five other websites to authenticate the results.

3 Problem Statement

rules to be optimized. These audit rules are difficult to understand and not easy to optimize with proper analysis. Most of the audit rules are not under the user's control, so working on them will be useless and time-wasting. Some of the audit rules depend upon others, so working for those is also ineffective. In most cases, people struggle to optimize the website with irrelevant audit rules and work uselessly. Time is money, so wasting time in optimizing increases the cost of optimization and is also not effective in presenting our website to our target clients, so indirectly losing business if unable to rank the website

within a specific time. It becomes much more difficult to select the most relevant audit rule which returns maximum performance benefit.

In this paper, an optimized sequence of audit rules is identified by assigning a higher weight to those audit rules, which return maximum performance benefits and avoid irrelevant audit rules. By using a proposed sequence of audit rules, it becomes easy to optimize page speed in less time and without struggling with irrelevant tasks to optimize the page speed. It can be also helpful to develop an automated application to improve the page speed and loading time using artificial intelligence techniques. Ultimately, we try to find out the most effective sequence of audit rules to optimize the website with less effort and time. Then a website will be optimized within less time and cost by applying the proposed sequence of audit rules.

4 Proposed System

The proposed system is an optimized best sequence of audit rules based on performance benefit score and priority of audit rule based on its importance. The same sequence of audit rules, when implemented to optimize a website, it optimizes the website with time and cost. To find this sequence, different audit rules defined by Google PageSpeed are discussed below which are used to calculate the performance score of various page load matrices and finally return a page speed performance score. It is a weighted formula to calculate the net score on a 0-100 scale and categorize a web page into slow (0-49), average (50-89), and fast (90-100) [18]. Based on this range, score weight is assigned to each value in the range, and the average group is also divided into two sections, as in Table 1, that will be used in calculating performance percentage.

Table 1. Performance Score Weight

Performance	Score Range	Score Weight
Slow	0-49	1
Average	50-69	2
Above Average	70-89	3
Fast	90-100	4

4.1 PAGE LOAD METRICS

Google PageSpeed returns several page load metrics to estimate the speed of the page. These are First Contentful Paint, Speed Index, Largest Contentful Paint, Time to Interactive, Total Blocking Time, and Cumulative Layout Shift. To rank a web page higher, all of these matrix values must be zero or lower up to an extent. The First Contentful Paint (FCP) are good indicators that change the user experience as feel the primary content is visible. The First Contentful Paint is just a time when the browser renders the first bit of content from the DOM and shows the loading of the page. Its value can be optimized by configuring Render-Blocking CSS and JavaScript, use of HTTP Caching, minifying resources to load fast, and code splitting to improve the initial load.

Speed Index (SI) is a good matrix instead of page load time. It is the time to load the visible content of the page, also called Above the Fold content [19]. Optimizing content loading and reducing JavaScript execution time are the most important audit rules to minimize speed index value [18].

Largest Contentful Paint (LCP) is a user-centric metric to measure perceived load speed by measuring the load time of the main content in the viewport. It can be improved by the optimized server response time, render-blocking resources, and optimizing the CSS and JavaScript to render the main content [18].

Time to Interactive (TTI) is the time when the user is ready to interact with the page, so its value can be improved by optimizing the First Meaningful Paint and registering the event handlers for visible content, which can be improved by code splitting, reducing render-blocking, optimizing JavaScript execution time [18].

Cumulative Layout Shift (CLS) provides a good user experience by testing visual stability by avoiding layout shifts. Its value can be improved by reserving space for content such as images, videos, or text. This matrix will be optimized by improving the above-the-fold content and avoiding the dynamic DOM insertion to avoid layout shift.

Total Blocking Time (TBT), an important metric, is the blocking time to respond to user input. It can be improved with optimized JavaScript by removing unused CSS and JavaScript, code splitting, efficient loading of third-party scripts, and removing the render-blocking resources [18].

Google PageSpeed calculates the performance matrix separately for desktop and mobile devices. Due to the increased use of mobile devices, Google gets an SEO ranking of the web pages based on the mobile performance matrix only [5]. So a website optimized for desktop only is getting a lower rank in search engine result pages now; nonetheless, it retains its rank to some extent if its content relevancy is higher. Google has assigned different weights to these matrices, as given in Figure 1 , as the highest weight is assigned to Total Blocking Time, whereas the lowest weight is to First Contentful Paint and Time to Interactive [18].

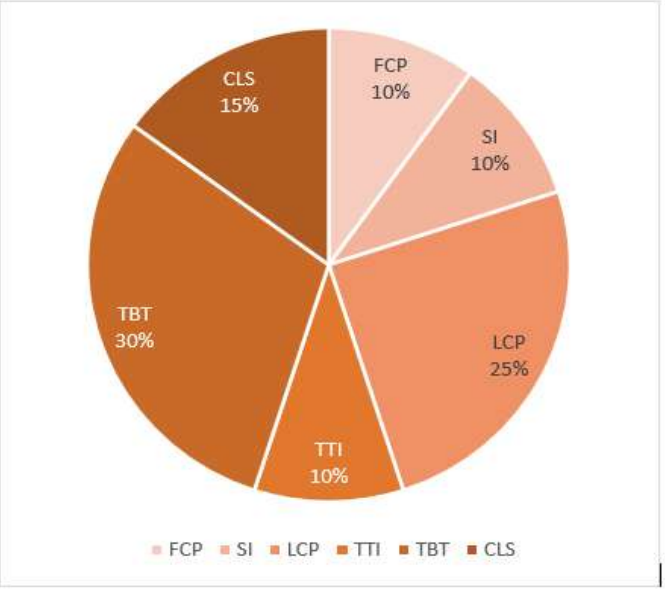

Figure 1. Performance Matrices Weight given by Google

4.2 AUDIT RULES

the matrix in Google PageSpeed. These audit rules must be configured and adjusted to optimize a website to load it faster. The most important of them are JavaScript and CSS scripts, image optimization, DOM size, cache, and compression techniques. As a website contains more rich content and dynamic elements handled with JavaScript and images, it becomes slow, and developers must optimize its design and layout by Hogan [20]. Improvement of any audit rule can impact one or more performance matrices, so our research objective focuses on audit rules, not on performance matrix in optimizing the website.

In this research, selected audit rules are categorized as shown in Table 2, having some common functionality as in [9] to optimize a website to categorize optimization tasks. The selection of these audit rules is based on the most common contents and functionalities in most of the websites. We set a priority for each category, so the selection of audit rule to be optimized must be from the high-priority category first, as the executive order is shown in Figure 2. So, minify CSS has a higher priority audit rule to be optimized against preload key requests. Then we identify audit rules with maximum performance benefits. If the performance score of the two audit rules is the same, then it will arrange by its priority. While finding

the best sequence of audit rules to be optimized first, we have assumed quality standards in developing the website, such as no duplicate content, scripting error, etc. These audit rules are divided into opportunities that can be used to improve performance metrics. Diagnostics are the additional information to improve the page with best practices and passed audits which are already implemented for optimized performance.

Table 2. Priorities of Audit Rules Optimization

Audit Rules	Description	Priority
Resource Loading	Minify CSS, Minify JavaScript, Reduced Unused JavaScript, Properly size images, Serve images in next-gen formats, Enable text compression, Reduce unused CSS, Efficiently encode images, Preload the largest contentful paint image, Image elements have explicit width and height	**3**
Network Access	Uses efficient cache policy on static assets, Defer off-screen images, Avoid multiple page redirects, Avoid request counts low and transfer size small, Preload key requests, Pre-connect to required origins, Avoid enormous network payloads	**2**
Script Execution	Eliminate render-blocking resources, JavaScript execution time, Lazy load third-party resources with facades	**1**

4.3 OPTIMIZING PAGE LOAD SPEED

Google recommends many techniques to optimize a website page load speed against each audit rule; for example, Google recommends the use of SVG images, responsive images, images with a media query, or image set function to optimize the “Properly Size Images“ audit rule. To pass these audit rules, various types of content on the website are managed by different techniques discussed here.

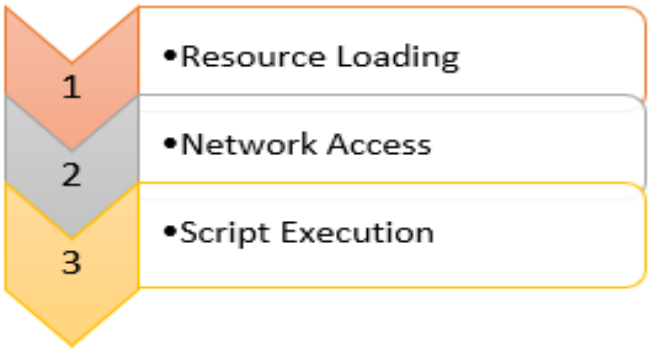

Figure 2. Execution Priority of Audit Performance Rules

4.4 RESOURCE LOADING

CSS and JavaScript are major components of the website to be loaded before displaying the content of the web page, so the larger sized CSS and JavaScript can be the major reason for slow loading speed if

those are not properly optimized. In interactive websites, the use of JavaScript, jQuery plugin, and Ajax increased tremendously, which put an extraordinary burden while loading different functionalities of the web page. So “Minify CSS” and “Minify JavaScript” are major audit rules to reduce the page loading time. There are several online and API services to minimize the size of CSS and JavaScript.

Images are the most important resource, which must be loaded first to render the visual presentation of the web page. To reduce the page load time, all of the images must be resized and optimized. An image that is larger but renders on a small size device severely affects the performance issue and increases the load time. The use of new image formats and responsive images is effective in improving page load time.

HTTP caching is an important technique to reduce the load time on the next visit; the browser can access the local resource instead of a network request. The web application can maintain a web app cache storing data objects in a web page; if data is not available in the web app cache, it moves to the browser cache, and if data still missing there, then initiate a network request [14]. The cache policy can be defined in the htaccess file by setting the Cache-Control with the max-age parameter [21]. To avoid multiple loads of the same resource, the server must also be configured for Etag to identify the browser cache with the server data that will be used to identify requests to various servers simultaneously. The ETag indicates if the source has changed since the resource was stored in the cache, then it reloads the content from the server [22].

4.5 SCRIPT EXECUTION

To reduce the processing burden on the main thread and decrease the page load time, the most important audit rule that must be optimized is DOM size. Larger DOM size causes different problems related to network efficiency, runtime performance, and memory performance. Different techniques can be used to optimize the size of the DOM, such as paging the content, dynamic loading, and minimizing third-party scripts. A larger size JavaScript increases the execution cost and consumes a lot of memory cost, higher compile cost, which also blocks to respond user input, and higher download increases network cost. It is automatically optimized by eliminating unused JavaScript and deferring uncritical JavaScript. Any useless CSS rule puts a heavy processing burden on the main thread and delays the page load. Critical CSS is part of CSS rules those must be loaded to render the page, and the remaining CSS rules are uncritical CSS that require later on, so it can delay loading to improve the page speed. Avoiding many render-blocking, CSS, JavaScript, and fonts also reduces the execution time and improves page load time [23], which is achieved by deferring uncritical resources.

4.6 NETWORK ACCESS

With third-party JavaScript snippets, the number of files containing CSS and JavaScript is increasing now, which generates several HTTP requests, which results in a slow website. To avoid these HTTP requests, some scattered CSS and JavaScript files must be merged into fewer files and avoid redirects in resources required for critical rendering paths [24]. To reduce network requests, off-screen images are deferred and can be loaded later on by using several lazy load techniques. Responsive images also reduce network requests. Network payload is automatically adjusted by applying other audit rules with defer requests, optimizing requests by minifying CSS and JavaScript, use of compression for text and images, and applying cache to reduce network requests. Uncritical resources are loaded with preload option, so the browser can load these resources at any time to improve the web performance [25]. The use of modern-style web fonts, such as Google fonts, can be optimized by loading web fonts asynchronously to improve performance score and load time [23]. Content Delivery Networks are high-speed data centers to provide the data nearest to the user to reduce the delay in processing the network requests to load resources and to improve bandwidth usage. It takes time to connect CDN and affects the website’s performance, so the pre-connect option will be effective for connecting the server before loading the resources and avoiding multiple connections to the same server.

Server Response Time is important to audit rules to optimize the page load time, but it is less flexible to control, such as slow routing, availability and power of CPU, and memory availability. Changing the

physical server is one the most hectic jobs, with several issues such as finance charges, management issues, migration costs, etc. Some of the aspects of getting slow server responses are due to application logic and database access logic, which can be improved by having the latest and optimized algorithms and scripts [26].

4.7 TYPES OF AUDIT RULES

In this research, audit rules are categorized in the following categories as given in Figure 3 with a high-performance score in optimizing a website. Each category of audit rule is further classified with the performance audit weight, as Google PageSpeed has categorized performance matrices by weights [18], but in this research, we have assigned five different weights to these performance audit rules. This study focuses on using only performance audit rules, which give maximum performance benefit and also optimize other audit rules, whereas null audit is avoided in optimizing a website to improve page speed score, whereas Google PageSpeed does not give any weight based on the impact on other performance audit rules. Rigid and Null audit rules are not good for further research, so zero weight is assigned to them.

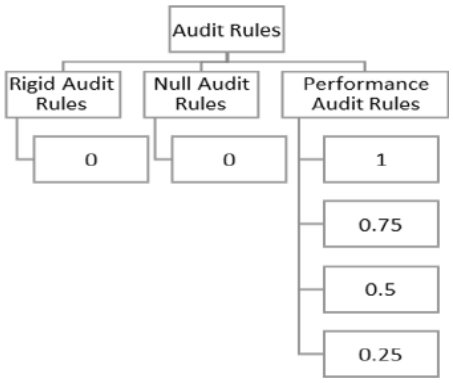

Figure 3. Classification of Audit Rules in Proposed Audit Rules Sequence

4.8 RIGID AUDIT RULES

Audit rules are not flexible to control and not easy to implement. Therefore site optimizer cannot optimize it, i.e., Server Response Time and Minimize Main Thread Work immediately, as given in Table 3. These rules are ignored for further processing, as they are not good candidates to optimize load time. These are identified manually, and these are not mentioned in Table 3.

Table 3. List of Rigid Audit Rules

Sr.No	Audit Rule
1	Server Response Time
2	Minimize Main Thread Work
3	Minimize Third Party Usage
4	Avoid Non-Composite animation
5	Avoid Excessive DOM size
6	Use video formats for animated content

4.9 NULL AUDIT RULES

Audit rules which are totally depended upon other audit rules and automatically optimized while optimizing other audit rules are called null audit rules, and there is no impact on the performance score due to

them or less than 1% performance benefit. Our script can identify those null audit rules by using Google PageSpeed API, and these are ignored to find the final set of audit rules to optimize the load time of a website. Zero weight is assigned to them. These are a set of those audit rules which are automatically optimized to a great extent for optimizing some other audit rules. As the guideline given by PageSpeed documentation, optimization of these depends upon various other audit rules, as given in Table 4. Audit rules which just affected little, are not included in the list. Therefore these are not good candidates to optimize the load time of a website, as it is a real waste of time to try to optimize them.

Table 4. List of Null Audit Rules

Sr.No	Null (Dependant) Audit Rule
1	Avoid enormous network payloads
2	Eliminate render-blocking resources
3	Avoid request counts low and transfer sizes small
4	JavaScript execution time

4.10 PERFORMANCE AUDIT RULES

Audit rule which affects the performance score of other audit rules that is called the Performance audit rule. They return the maximum performance score, which is the sum of its performance score and the performance score of all dependent audit rules. These audit rules are assigned 1 weight if improvement in Weighted Performance Benefit Score (wpbs) is greater than 30% as shown in Figure 4. If improvement in wpbs is between 20-30% and the number of other audit rules affected is more than or equal to 3, then 0.75 weight is assigned, and 0.5 weight is assigned if affected audit rules are less than 3. If the wpbs is more than 10% and the number of other audit rules affected is more than or equal to 4, then 0.5 weight is assigned. Otherwise, 0.25 weight is assigned to it as the lowest priority is selecting the audit rule to optimize load time. So Performance Audit rules with 1 weight will be the starting point in optimizing the load time of a website, which further prioritizes with the wpbs and priorities given in Figure 2. Execution of auto rules in such a way will help to optimize a website with a minimum number of audit rules configuration.

5 METHODOLOGY

To find the performance score and dependency of various audit rules, five different websites were installed on one Linux Server with different sub-domains, given in Table 5, with different contents constructed in the Codeignator PHP framework and MySQL configuration. Website content will contain images, videos, CSS, JavaScript, database, Ajax call, etc. Every website has several pages that are optimized for one audit rule given in Table 2and one non-optimized page to compare the result of optimization. One week test is performed on all of the pages eight times in 24 hours after every four hours. All pages are tested and compared with the non-optimized page, and all of the results are stored in a CSV file against all of the audit rules. All of the tests of one week, including all of the websites, are accumulated to find the average performance benefit of each audit rule that will be used to identify the most important audit rules and arrange them in maximum to minimum performance scores. Then proposed sequence of audit rules are applied to the other five non-optimized websites to test performance score.

5.1 PROPOSED AUDIT RULES SEQUENCE

The proposed system, as shown in Figure 5 is used to determine the proposed audit rules sequence, a simple script to find the performance matrix against audit rules for sample websites to test the Google PageSpeed performance score by performing a Google PageSpeed API call with CURL [27]. The same script is executed multiple times with the help of a CRON job. It stores the results of various tests with their performance data returned by PageSpeed API response [28] in a CSV file.

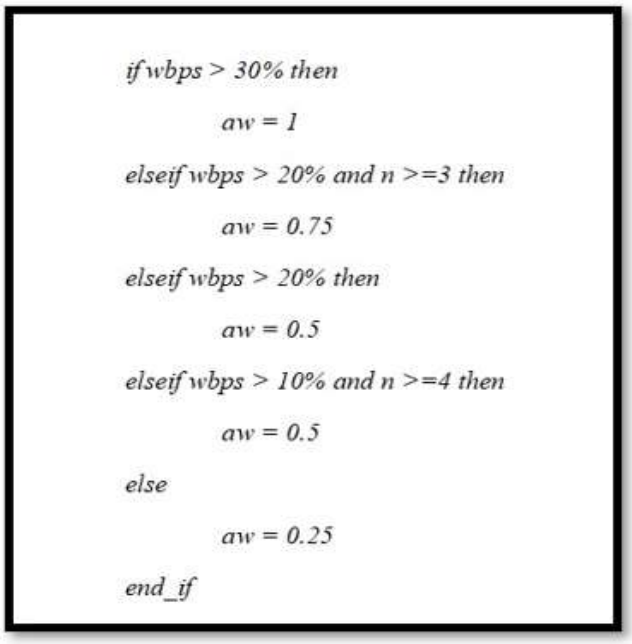

Figure 4. Calculation of Weighted Performance Benefit Score for an Audit Rule

Table 5. List of Sample Websites

Sr.No	Sample Websites
1	https://silksoft.us/sample1
2	https://silksoft.us/sample2
3	https://silksoft.us/sample3
4	https://silksoft.us/sample4
5	https://silksoft.us/sample5

CSV file includes the performance score of each of the audit rules as well as the performance score of the non-optimized web page. The CSV file or dataset includes the list of all of the audit rules performed on the sample sites to optimize its speed with previous non-optimized scores and optimized scores. It also shows the size of the page in bytes as well as the time to load, but we are concerned with the accumulated parameter, which is the performance score. The same information is created for all of the sample sites. A CRON job is scheduled to run after every four hours for three consecutive seven days to identify the effect of different timings in a day. Performing test for seven consecutive days also helps to normalize data by eliminating outliers due to slow networks, bad server response, or unexpected high server response. It shows the importance of various audit rules, as shown in Figure 6, to be used in priority to optimize the page load speed. The performance benefit score is multiplied by the audit rule priority, as shown in Table 2and using equation 1 to find the performance percentage.

The required weight is assigned to the audit rule to identify the proposed audit rule type, as in Figure 4discussed above. After performing the above tests, all results are compiled in Microsoft Excel to find performance gain by comparing the performance score of each audit rule with the performance score of the non-optimized web page. After that performance gain score of each audit rule is accumulated to find the average performance gain score by optimizing the same audit rule, and the audit weight is assigned to prioritize the execution order. The average results of audit rules, as given in Table 6, are sorted in ascending order regarding audit weight and the wpbs.

$$wfbs = (s1 - s0) * p \quad (1)$$

Audit rules with the same audit weight will be arranged in the order of the wpbs to find out the best sequence of audit rules in terms of higher improvement in obtaining maximum performance score with less effort and time to optimize load time for a website.In our study, the performance score is already calculated by Google PageSpeed by taking all of the audit rules and other parameters, but we still assign some weight to these with improvement in load time score along with the priority weight as assigned to resource loading, network access, and script execution, to identify the most important audit rules to optimize a website.

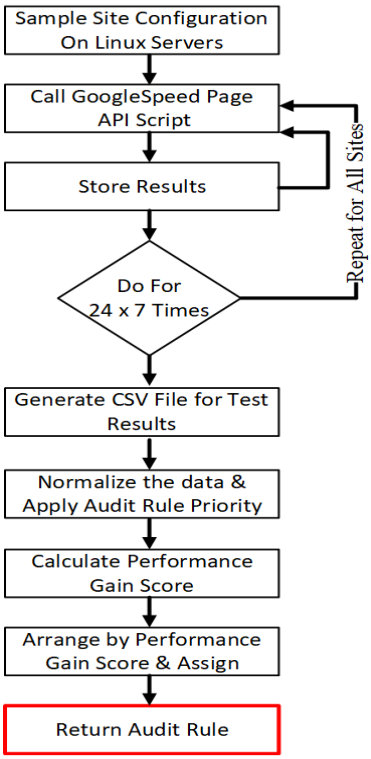

Figure 5. Proposed System Flowchart

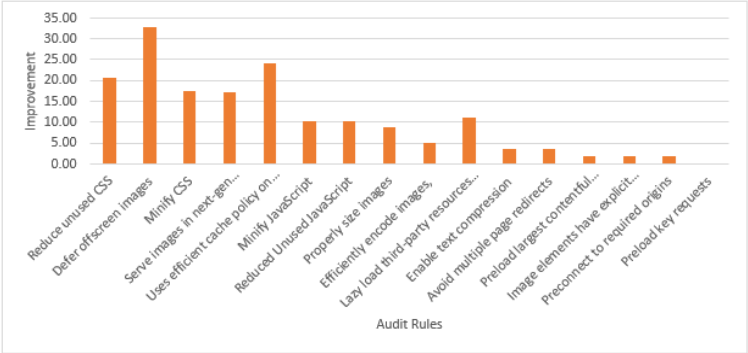

Figure 6. Speed Optimization Performance of different Audit Rules

Table 6. Audit Rules Performance Score on Sample Sites

Audit Rules	Before score (s0)	After Score (s1)	Audit Rule Priority (p)	Audit Rules Affected (n)	Weighted Performance Benefit Score (wpbs)	Audit Weight (aw)
Reduce unused CSS	58	70	3	5	36	1
Defer off-screen images	49	65	2	4	32	1
Minify CSS	57	67	3	5	30	0.75
Serve images in next-gen formats	58	68	3	3	30	0.75
Uses efficient cache policy on static assets	58	72	2	4	28	0.75
Minify JavaScript	58	64	3	5	18	0.5
Reduced Unused JavaScript	59	65	3	4	18	0.5
Properly size images	57	62	3	2	15	0.25
Efficiently encode images,	59	62	3	2	9	0.25
Lazy load third-party resources with facades	54	60	1	2	6	0.25
Enable text compression	57	59	3	4	6	0.25
Avoid multiple page redirects	56	58	2	1	4	0.25
Preload largest contentful paint image	57	58	3	2	3	0.25
Image elements have explicit width and height	56	57	3	2	3	0.25
Preconnect to required origins	59	60	2	1	2	0.25
Preload key requests textbf	57	57	2	0	0	0

6 RESULTS & DISCUSSION

The sequence of audit rules identified in the previous step will be used to optimize five non-optimized websites. Audit rules are applied with the audit weight and wpbs score in the sequence. If both are the same, they use the audit rule priorities to execute it. By following this proposed audit rules sequence of executing audit rules, there will be no wait time to investigate various audit rules and consuming time in selecting to start optimizing the load speed. Without any effort, all of the audit rules in the audit rule sequence are applied individually to optimize the website.

The proposed audit rules sequence is tested on five different websites developed in Codeignator and WordPress by using various technologies such as JQuery, Bootstrap, and AMP. The results of these tests are satisfactory as they achieved above 90% performance score just after executing seven audit rules in the proposed sequence with three test websites, and found the results in given Table 7. It shows that all of the sites return more than 80% performance score after the execution of a maximum of seven audit rules without any random selection of audit rules to optimize page load speed. . After performing these rules, a reasonable difference, as shown in Figure 7, can be observed as compared to the un-optimized websites, which indicates the validity of the proposed audit rule sequence. Using this proposed sequence of audit rules to optimize the loading speed can improve SEO score as it becomes a necessary requirement by Google to improve the site loading time before the site is available to search engines. So a site with less load time will have more chance to get a higher rank in google search results.

Figure 7. Performance Score Improvement of Proposed Audit Rule Sequence

Table 7. Proposed Audit Rules Sequence Test Results

Sr.No	Test Site	Performance Score (New)	Audit Rules Applied
1	https://acheron-instruments.com	96%	6
2	https://inodes.pk	98%	5
3	https://mojoask.com	92%	6
4	https://beautyofpakistan.com	88%	7
5	https://chemicalformulaservices.com	82%	7

7 CONCLUSION & FUTURE WORK

Page speed optimization is one of the major tasks to get a higher rank in search engines, but doing various activities to optimize page load time is not an easy task. Several complex and time-consuming tasks must be performed to optimize a web page to improve its load time. It consumes the valuable time of the website optimizer before starting the actual search engine optimization process. Google PageSpeed is a tool that guides various performance metrics to be optimized by using a long list of audit rules. Selecting an audit rule to start the optimization process is much more difficult, and sometimes, all of the task performed to optimize that audit rule is useless. So in our search, we have identified the best sequence of audit rules to optimize to get maximum performance benefit within minimum time and cost. Five different websites developed with various technologies are tested, and a more than 80% performance score is achieved by applying the first seven audit rules in the proposed sequence. More than 90% performance score is achieved with less than seven audit rules. In this way, we have optimized websites in terms of load time by working on only a few audit rules without wasting time in the selection of audit rules from a long list of audit rules. In this research, the audit rules are used by Google PageSpeed, but the outcomes of this research are easily applicable to the audit rules given by YSlow.

Author Contributions

Abdul Ghafar: Conceptualization, Methodology, Software **Fazeel Abid**: Data curation, Writing- Original draft preparation. **Mohsin Ashraf**: Visualization, Investigation. **Abdul Jamil**: Supervision.: **Ali Abbas**: Software, Validation. **Farah Rauf Malik**: Writing- Reviewing and Editing

Compliance with Ethical Standards

It is declared that all authors don't have any conflict of interest. It is also declared that this article does not contain any studies with human participants or animals performed by any of the authors. Furthermore,

informed consent was obtained from all individual participants included in the study.

References

[1] S. Prakash, "Search engine optimization techniques for attracting organic traffic to websites," *International Journal of Recent Trends in Engineering & Research*, vol. 6, no. 5, pp. 1–6, 2020.

[2] M. Nagpal and J. A. Petersen, "Keyword selection strategies in search engine optimization: how relevant is relevance?" *Journal of retailing*, vol. 97, no. 4, pp. 746–763, 2021.

[3] S. Krrabaj, F. Baxhaku, and D. Sadrijaj, "Investigating search engine optimization techniques for effective ranking: A case study of an educational site," in *2017 6th Mediterranean Conference on Embedded Computing (MECO)*. IEEE, 2017, pp. 1–4.

[4] S. Kaur and K. Kaur, "Search engine optimization techniques with website perspective." *Int. Arab. J. e Technol.*, vol. 4, no. 3, pp. 103–108, 2017.

[5] A. Osmani and I. Grigorik, "Speed is now a landing page factor for google search and ads," *Google Developer Updates*, 2018.

[6] S. ANDERSON, "How fast should a website load in 2022?" *hobo SEO Consultancy*, vol. 16, 2021.

[7] P. H. Shroff and S. R. Chaudhary, "Critical rendering path optimizations to reduce the web page loading time," in *2017 2nd International Conference for Convergence in Technology (I2CT)*. IEEE, 2017, pp. 937–940.

[8] C. Kelton, J. Ryoo, A. Balasubramanian, and S. R. Das, "Improving user perceived page load times using gaze." in *NSDI*, vol. 17, 2017, pp. 545–559.

[9] H. Chen, J. Wan, and J. Zhao, "Optimization research and application of enterprise website based on web service," in *2017 2nd International Conference on Materials Science, Machinery and Energy Engineering (MSMEE 2017)*. Atlantis Press, 2017, pp. 557–563.

[10] H. O. Al-Sakran and M. A. Alsudairi, "Usability and accessibility assessment of saudi arabia mobile e-government websites," *IEEE Access*, vol. 9, pp. 48 254–48 275, 2021.

[11] J. Cito, D. Suljoti, P. Leitner, and S. Dustdar, "Identifying root causes of web performance degradation using changepoint analysis," in *Web Engineering: 14th International Conference, ICWE 2014, Toulouse, France, July 1-4, 2014. Proceedings 14*. Springer, 2014, pp. 181–199.

[12] E. Bocchi, L. De Cicco, and D. Rossi, "Measuring the quality of experience of web users," *ACM SIGCOMM Computer Communication Review*, vol. 46, no. 4, pp. 8–13, 2016.

[13] A. V. Nori, J. Gaur, S. Rai, S. Subramoney, and H. Wang, "Criticality aware tiered cache hierarchy: a fundamental relook at multi-level cache hierarchies," in *2018 ACM/IEEE 45th Annual International Symposium on Computer Architecture (ISCA)*. IEEE, 2018, pp. 96–109.

[14] T. S. Shan, J. H. Obit, R. Alfred, and A. Tahir, "Enhancing the performance of university's website for mobile devices based on responsive web design approach," *Advanced Science Letters*, vol. 23, no. 11, pp. 10 969–10 973, 2017.

[15] S. W. Young, "Speed matters: Performance enhancements for library websites," *Weave: Journal of Library User Experience*, vol. 1, no. 4, 2016.

[16] P. Lo, A. Cho, M.-h. Leung, D. K. Chiu, E. H. Ko, and K. K. Ho, "Use of smartphones by art and design students for accessing library services and learning," *Library Hi Tech*, vol. 34, no. 2, pp. 224–238, 2016.

[17] M. Gavalda, "A beginner's guide to website speed optimization," *Viitattu*, vol. 13, p. 2016, 2015.

[18] J. Danielsson and O. Danielsson, "My media favorites on any device: A study about the development and evaluation of a partly cloud-based and partly on-premise solution based on microsoft platform," 2021.

[19] E. Nichifor, R. C. Lixăndroiu, I. B. Chițu, G. Brătucu, and A. Trifan, "How does mobile page speed shape in-between touchpoints in the customer journey? a research regarding the most trusted retailers in romania," *Journal of Theoretical and Applied Electronic Commerce Research*, vol. 16, no. 5, pp. 1369–1389, 2021.

[20] L. C. Hogan, *Designing for Performance: Weighing Aesthetics and Speed*. " O'Reilly Media, Inc.", 2014.

[21] D. J. Harkness, "Getting started with apache," in *Apache Essentials: Install, Configure, Maintain*. Springer, 2022, pp. 1–28.

[22] J. Jokinen, "Personal internet privacy and surveillance: Implementation and evasion of user tracking," 2018.

[23] S. K. Shivakumar and S. K. Shivakumar, *Mobile Web Performance Optimization*. Springer, 2020.

[24] Z.-C. Dong, X.-M. Fu, Z. Yang, and L. Liu, "Redirected smooth mappings for multiuser real walking in virtual reality," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 5, pp. 1–17, 2019.

[25] S. K. Shivakumar and S. K. Shivakumar, *Mobile Web Performance Optimization*. Springer, 2020.

[26] A. P. Lalengke and I. Nurhaida, "Performance analysis of cloudlinux-based web server at the embassy of the kingdom of morocco in jakarta," *Jurnal Sisfokom (Sistem Informasi dan Komputer)*, vol. 10, no. 2, pp. 250–258, 2021.

[27] V. Vasilijević, N. Kojić, and N. Vugdelija, "A new approach in quantifying user experience in web-oriented applications," in *4th International Scientific Conference on Recent Advances in Information Technology, Tourism, Economics, Management and Agriculture–ITEMA*, 2020, pp. 9–16.

[28] K. I. Roumeliotis, N. D. Tselikas, and C. Tryfonopoulos, "Greek hotels' web traffic: A comparative study based on search engine optimization techniques and technologies," *Digital*, vol. 2, no. 3, pp. 379–400, 2022.