Speed Optimisation For Group Websites

CASE STUDY

colorwhistle

Industry : Hospitality

Tools Used : WPRocket, GTMetrix
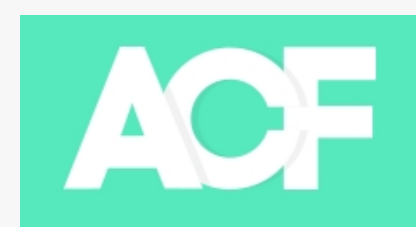
Region : United Kingdom

Technology Used : WordPress CMS with Multisite, Twig – a PHP Template Engine, Timber – Theme Builder, ACF

The Team

Project Manager : 1

Developer : 1

Project Duration : 4 Months

CLIENT'S REQUIREMENTS

As the client's website was slow – an average loading time of 5.82 seconds with 76% of performance, they approached us for website speed optimization and performance. In addition to it, they also came up with a few dependencies on website maintenance works as well.

The client decided to enhance all the 30 websites' speed, performance, and loading time with the web developer's support. While looking up for agencies to work with, they checked our ColorWhistle site's speed in GTMetrix, and on looking at the score, they reached us.
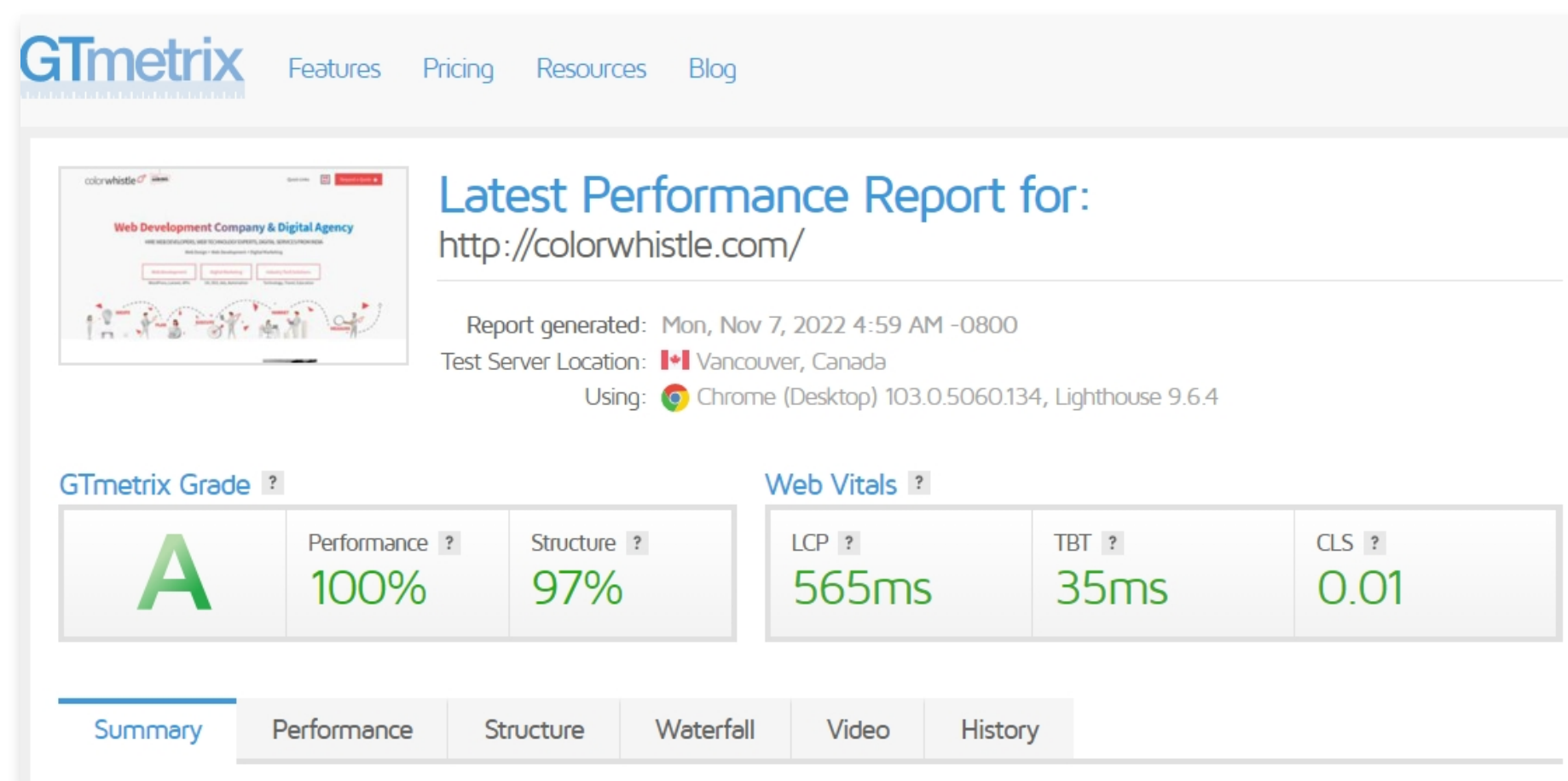
BEFORE SPEED OPTIMIZATION – CHECKLIST

Optimizing the speed of live websites is a sensitive concern. So, at ColorWhistle, while working on speed-related activities, we follow a checklist. This will make us ensure that we're not missing out anything.

1. Test the Speed

The client had 30 websites for their hotels, restaurants, spas, and so on. Among all these multiple sites, one was the parent site – Cairn Collection. And they managed the rest of the child websites from this parent site using the super admin login credentials.

> Super admin is an admin type with a dedicated username and password. It greatly helps in managing multiple sites. Instead of creating individual usernames and passwords for all the sites, you can access them from a single username.

We got the credentials from the client and ran a quick speed test for all websites. The image below shows the average GTMetrix grade, performance, and load time status of the websites before implementing speed optimization techniques.

2. Take Backups & Screenshots

When working on speed improvements, backup is a crucial step. For normal single sites, we use All-in-one WP Migration plugin, we tried to use the same.

Unfortunately, the plugin didn't support multiple websites. So, we manually, signed-in to the MyAdmin portal and took backups of the database of all websites.

Then, we logged-in to the cPanel of each website and took backups of the website files. The existing website had a file manager with all the databases in it. We downloaded the files and exported them to the new file manager with '.dev' extension.

In addition to it, we took screenshots of all the key pages on all 30 websites. This helped us to check and compare the before and after status. We created a folder in the local directory and saved all the screenshots with the proper names.

3. Create a Demo Version

Since this is speed implementation, we cannot work on the live site nor put off the site in maintenance mode. So, what we did? We built a demo environment and cloned all the websites.

We kept the URLs as it is just with '/dev'. We ensured the same is followed all through the sites and pages. Also, confirmed all the demo sites work similarly as current sites without anything being missed out.

Inorder to avoid the serious issue of URLs without the proper extension being counted as the live site, we updated the URLs with the proper '/dev' extension using the Better Search Replace plugin.

4. Upgrade WordPress & Plugins

The outdated versions will also make the websites slow. On both the demo and live sites, we found that the versions of WordPress and plugins were outdated. We updated to the latest versions on both live and demo sites.

After updating the plugins, we checked if it was working well or facing any issues. For that, we did a few test form submissions and checked if emails are sent properly. Everything worked fine on both the demo and live versions.

WHAT DID WE DO?

MANUAL SPEED OPTIMIZATION WORKS

1. Optimize the Images

On average, the optimal size of an image on a website gallery can be below 200KB because when the image size is high, it will have a great impact on website load time. As our client is a hotel-based project, the websites had too many images.

We're already familiar with an image optimization plugin, called Smus. In our previous projects, after activating this, we have seen positive results, so opted for the same.

Without damaging the quality, the plugin helps to compress the height & width of the images. Moreover, this plugin can be a great option to enable lazy loading. When lazy loading is activated, the text content loads fast. Later on, the images load. The website will load right when the text loads, it will not wait for the images to load.

So, we decided to add the new WordPress plugin exclusively for image optimization.

2. Optimize the Database

If a website needs to load quickly, then it's imperative that its database be optimized as well. Form submissions, user details, tables, fields, and every small to big detail on a website will be stored in the database.

We did a thorough audit of the database and removed all the unwanted form submissions, tables, fields, etc. For this, we installed a plugin named WPOptimize. With just a few clicks in the plugin settings, we were able to optimize the database in a matter of minutes.

3. Clean-up Cache

Cache is a major issue that heavily impacts website performance. To clear all the cache, we were looking for an ideal plugin solution for multiple sites. Our previous experiences helped us to opt for the right plugin, which is, AutoOptimize.

3. Create a Demo Version

Since this is speed implementation, we cannot work on the live site nor put off the site in maintenance mode. So, what we did? We built a demo environment and cloned all the websites.

We kept the URLs as it is just with '/dev'. We ensured the same is followed all through the sites and pages. Also, confirmed all the demo sites work similarly as current sites without anything being missed out.

Inorder to avoid the serious issue of URLs without the proper extension being counted as the live site, we updated the URLs with the proper '/dev' extension using the Better Search Replace plugin.

4. Disable Unwanted Scripts and CSS Files

Unwanted CSS files and Java scripts are nothing, for example, if 'request a quote form' is added to the 'contact us' page. To make the form work, certain coding scripts will be added. The same scripts will be replicated across all the pages including the homepage. This is not needful in any way, instead slow downs the site's performance.

On the other side, manually disabling the code section would be tedious, so we came up with an alternate plugin option. The WordPress plugin Asset CleanUp helped us to get rid of all the unwanted scripts and CSS files.

5. Remove Unnecessary Plugins

Any difficult task can be accomplished with a plugin installation. As plugins help in serving different purposes, web developers generally download a myriad of them. At the same time, more plugins will break the site's performance too.

When we did a complete audit of all the websites, we came to know that certain plugins were sitting without purpose. So, we removed the unused plugins from the list.

6. Implement HTTP2

After the update of HTTP2, enabling it makes websites load faster. Since the client approached us specifically for speed optimization, we planned to activate HTTP2 for all 30 websites.

For this, we signed into the respective cPanel of the hub site and enabled the HTTP2 Server Push mechanism. Previously, the client used the HTTP extension.

MOBILE RESPONSIVENESS

When we were working on speed-related tasks, we found that a few sites were facing responsiveness issues. To fix them, we learned the twig template engine and re-visited the style and layout of the designs in mobile view and fixed. What we did for mobile responsiveness?

- *Optimized the toggle menu in mobile view*
- *Changed background color in contrast to the text*

SERVER OPTIMIZATION

Usually, when the server is loaded with more files and data, it will make the site load slower. We examined the client's website server had more files, so planned to delete unwanted files. As a result, we were able to remove 108 GB of data from the server.

BUG FIXING

When we were assessing the sites, we found certain bugs and fixed them.

- *The pop-up for cookies should appear only once and when users click on 'Accept', it should not pop up again. But here even after clicking 'Accept', the pop-up was showing. We fixed this bug by adding cookie-based popup functionality.*
- *In the Timber theme, when certain requests were made, indifferent functionalities worked. We fixed it*
- *The Revolution Slider was not functioning properly as CDN was disabled. So, we enabled the CDN*
- *Two forms cannot work on a single page, so removed the form and added a button. When users click the button, it will trigger a request to the header book now form*

MANUAL SPEED OPTIMIZATION – RESULT

After manually optimizing all the 30 websites' speed, there was an improvement in a few sites as expected, but unfortunately, few websites loaded slower than before.

After Speed Improvement - Manual		
Grade	Performance	Fully Loaded Time (in Sec)
	78%	5.04

To resolve this, we installed the WPRocket plugin. As a result, we were able to see improvements.

This is a paid plugin. Initially, the client was not ready to pay the price. As we explained to them the perks of it, they were convinced.
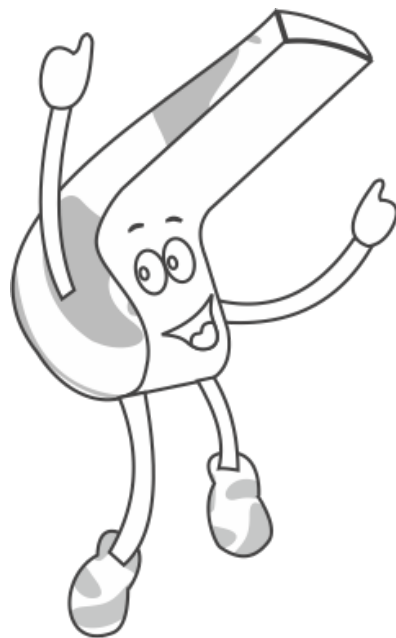
WPRocket is the best suitable plugin for speed optimization. This one can replace all the other plugins, as it can on itself handle all the speed-related tasks.

We added the new WordPress plugin to the parent website. It automatically got installed across all the other child websites. But, the problem we faced is, when we configure a setting option in the parent site, it didn't get reflected on the other sites.

We've to manually configure it for all sites. This was the only issue we faced, but still, we managed. We configured the settings as required. We removed the other unnecessary plugins that we installed before.

THE END-RESULT

After completing the quality checks, we measured all 30 websites' speed. And, the client was happy with the improvement. ☺ This was the overall GTMetrix performance rate and load time.

After configuring WP-Rocket plugin	
Performance	Fully Loaded Time (in Sec)
84%	2.50

▶ Performance was increased from 76% to 84%

▶ Acquired 8% improvement rate in the overall performance

▶ Load time got decreased from 5.82 sec to 2.50 sec
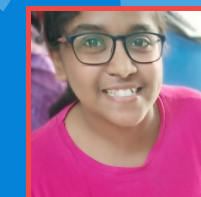
▶ Got 43% enhancement rate in the overall load time

At ColorWhistle, we've got in-depth knowledge and experience in website speed optimization works. So, working on this project didn't seem overwhelming for us. But what seemed new to us was, the multiple sites with a SuperAdmin panel.
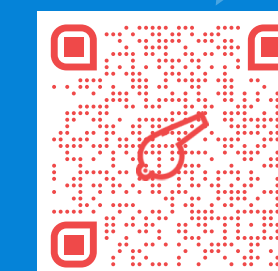
It was again a happy learning experience for our team.

Alright, sounds like a plan!
Let's go!!

AUTHOR
PAVITHRA SAMUEL
Content Writer

SEND DIRECT MAIL
hi@colorwhistle.com