

A. Introduction

Nous entrons dans le cœur de notre analyse, en résumé, jusqu'à présent vous avez :

- fait un point complet du périmètre du projet,
- décelé les points de blocage au crawl du site,
- paramétré Screaming Frog et réalisé un crawl du site, en plusieurs tentatives si besoin,
- analysé l'environnement organique du site,
- analysé le statut de son indexation dans Google Search Console.

Nous allons maintenant passer à l'analyse purement technique du site. Cette étude sera subdivisée en plusieurs parties que nous allons traiter.

Sachez cependant que ce plan peut être adapté aux besoins du client, notamment en ce qui concerne les sites e-commerce ou multilingues.

Pour rappel, en réalisant ces analyses, vous rédigerez également **le rapport d'audit** et de **cahier de préconisations techniques**. Commençons par faire un recueil des informations essentielles du site.

B. Analyses techniques de base

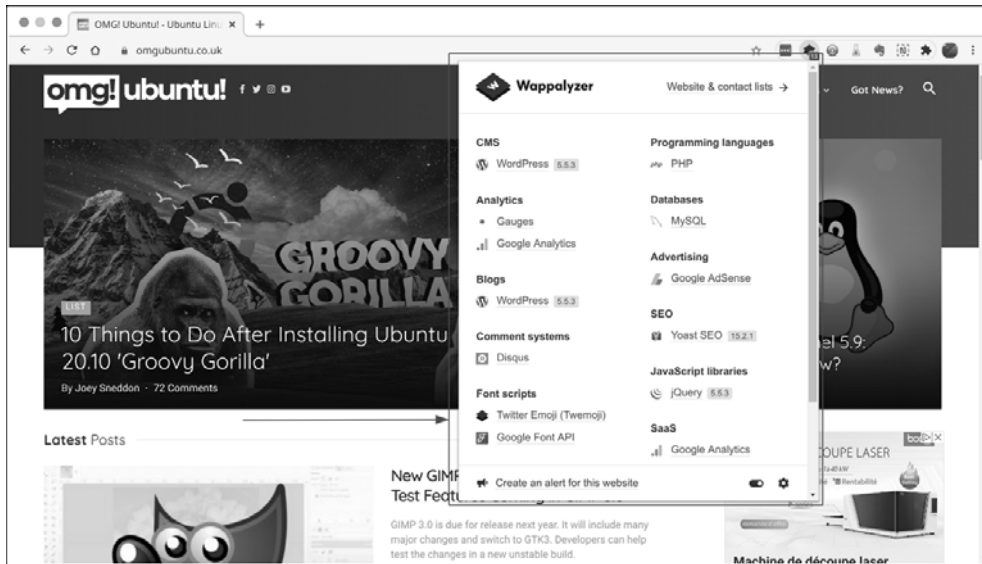
Après avoir passé en revue l'environnement organique du site, il s'agit maintenant de faire une analyse générale de ses **aspects techniques**. Bien sûr, la taille du site aura un impact sur vos recherches. Un site de petite taille (avec quelques centaines de pages) n'a pas les mêmes problématiques que le site d'un groupe international. Ce sera à vous d'adapter vos recherches dans ce sens.

Prenons un exemple. Les choses sont relativement plus simples pour le site d'une petite entreprise : peu d'URL, site monolingue, sous WordPress voire sous Joomla. Cela est différent d'un site dont le trafic a chuté après une migration et la fermeture de plusieurs sites satellites. Et cela est encore différent de l'analyse du site d'un groupe international, décliné en plusieurs langues (comme le chinois et le russe...) ou qui repose sur une base technique plus complexe (Drupal par exemple).

Un petit aparté : malgré la complexité de ces derniers types de sites, ce sont souvent ceux sur lesquels il est possible de mettre en place des actions correctives de grande ampleur, leurs propriétaires disposant de fonds conséquents.

Voici quelques pistes et outils qui vous donneront un aperçu des technologies employées sur un site.

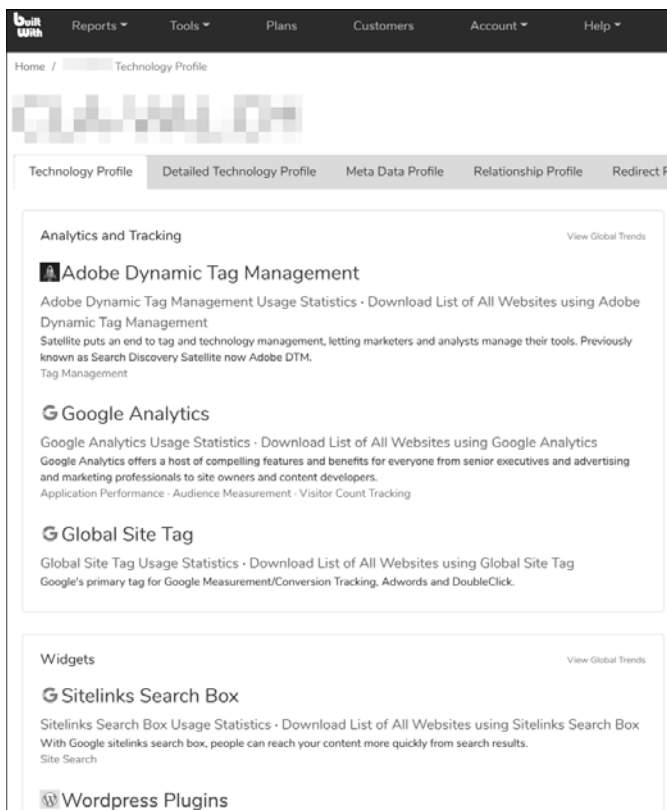
Wappalyzer : plug-in disponible sur Chrome et Firefox, il donne un aperçu de la base technique du site, notamment type de CMS, langage de programmation, type de serveur, solution analytics.



Un autre plug-in nommé **WhatRuns** est également disponible sur Chrome et Firefox (rendez-vous sur les sites des extensions des navigateurs).

- **BuiltWith.com** : ce site est une véritable mine d'or pour les apprentis chercheurs. Il présente les différentes technologies en place sur le site analysé et maintient l'historique des modifications effectuées : changements de CMS au cours des années, d'adresse IP, de code de suivi Google Analytics, etc. Il vous propose également les sites faisant une redirection depuis un nom de domaine autre. Il se peut d'ailleurs que le mandataire de l'audit ne les connaisse même pas.

Site web : <https://builtwith.com/>



- **PublicWWW** : réservé à des recherches plus approfondies, ce site vous permet de réaliser des investigations sur des éléments non disponibles via les moteurs de recherche, mais disponibles librement dans leur code source :
- Codes de suivi Google Analytics (pour vérifier si un code UA a pu être utilisé sur différents sites au cours de son existence).
- Quels sites utilisent un thème WordPress particulier.
- Quels sites font mention d'une marque (ou de votre nom, si vous souhaitez analyser votre e-réputation, mais c'est un autre sujet).
- Ou encore la liste des designers ou développeurs web qui détestent Internet Explorer : <https://publicwww.com/js/%22stupid+ie%22+filetype%3Ajs/>

Site web : <https://publicwww.com/>

The screenshot shows the PublicWWW Source Code Search Engine interface. At the top, there's a navigation bar with links for 'PublicWWW', 'Examples', 'Clusters', and 'Pricing', along with 'Sign Up' and 'Log In' buttons. The main heading is 'Source Code Search Engine' with a subtitle 'Find any alphanumeric snippet, signature or keyword in the web pages HTML, JS and CSS code.' Below this is a search bar with the placeholder text 'Search for code snippet, signature or keyword...' and a 'Search' button. To the right of the search bar, it displays '536 159 029 web pages' and 'November 10, 2020'. On the left side, there's a section titled 'Ultimate solution for digital marketing and affiliate marketing research...' followed by a list of use cases. On the right side, there's a section titled 'Usage Examples' with a grid of logos and search snippets for various services like AngularJS, Bootstrap, Add This, reCAPTCHA, Akamai, Algolia, HubSpot, and comScore.

Source Code Search Engine
Find any alphanumeric snippet, signature or keyword in the web pages HTML, JS and CSS code.

Search for code snippet, signature or keyword... **Search**

query syntax: RegEx, ccTLDs, etc.

Usage Examples

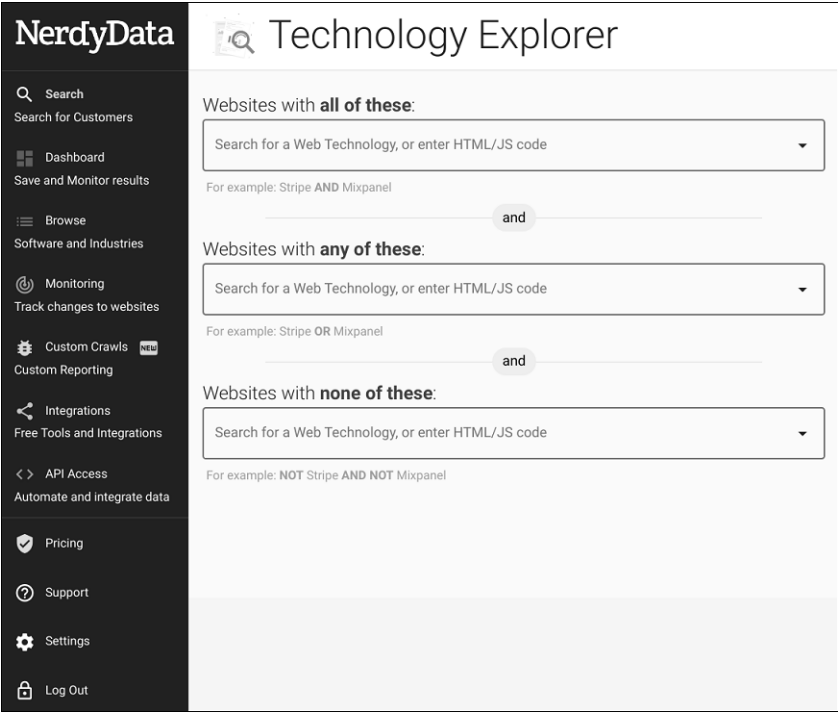
- AngularJS: "angular.min.js"
- Bootstrap: "bootstrap.min.js"
- Add This: "addthis_widget.js"
- reCAPTCHA: "recaptcha/api.js"
- Akamai: "X-Akamai-Transformed"
- Algolia: "AlgoliaSearch"
- HubSpot: "hubspot"
- comScore: "Begin comScore Tag"

Ultimate solution for digital marketing and affiliate marketing research, PublicWWW allow you to perform searches this way, something that is not possible with other regular search engines:

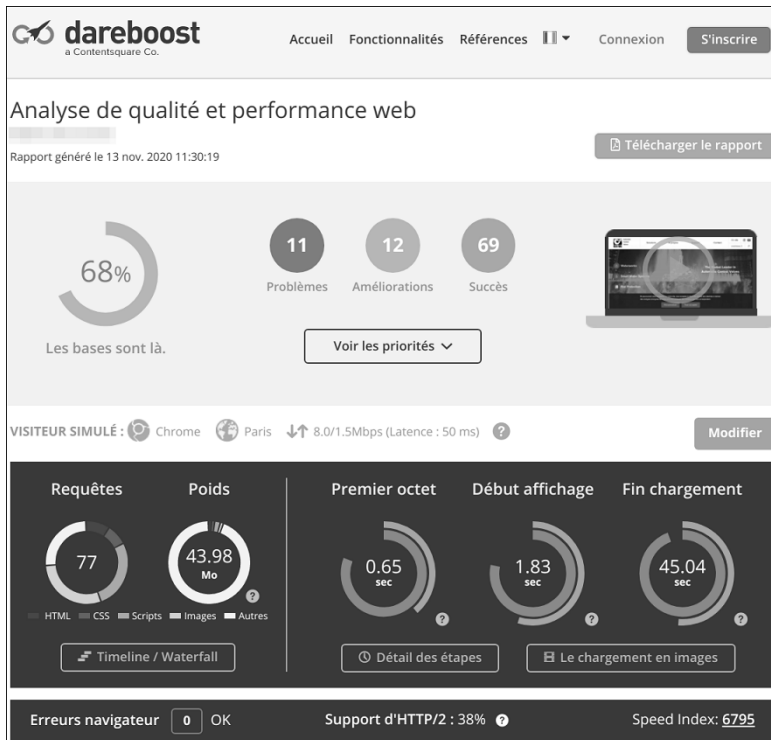
- Any HTML, JavaScript, CSS and plain text in web page source code
- References to StackOverflow questions in HTML, CSS and JS files
- Web designers and developers who hate IE
- Sites with the same analytics id: "UA-19778070-"
- Sites using the following version of nginx: "Server: nginx/1.4.7"
- Advertising networks users: "adserver.adtech.de"
- Sites using same adsense account: "pub-9533414948433288"
- Wordpress with theme: "/wp-content/themes/twenty sixteen/"
- Find related websites through the unique HTML codes they share, i.e. widgets & publisher IDs
- Identify sites using certain images or badges
- Find out who else is using your theme
- Identify sites that mention you
- References to use a library or a platform
- Find code examples on the internet
- Figure out who is using what JS widgets on their sites.

- **NerdyData** : à l'image de PublicWWW, NerdyData comme son nom l'indique, vous permet également de retrouver toutes sortes d'informations sur un site ou des technologies employées. Une extension pour Chrome est disponible, mais son intérêt est limité en comparaison avec Wappalyzer ou WhatRuns par exemple. En revanche, l'intérêt de cette solution repose sur son site web qui vous permet de faire des recherches approfondies sur des codes HTML (un code HTML précis se retrouve-t-il sur plusieurs sites de votre client ?) ou sur des secteurs entiers.

Site : <https://www.nerdydata.com/>

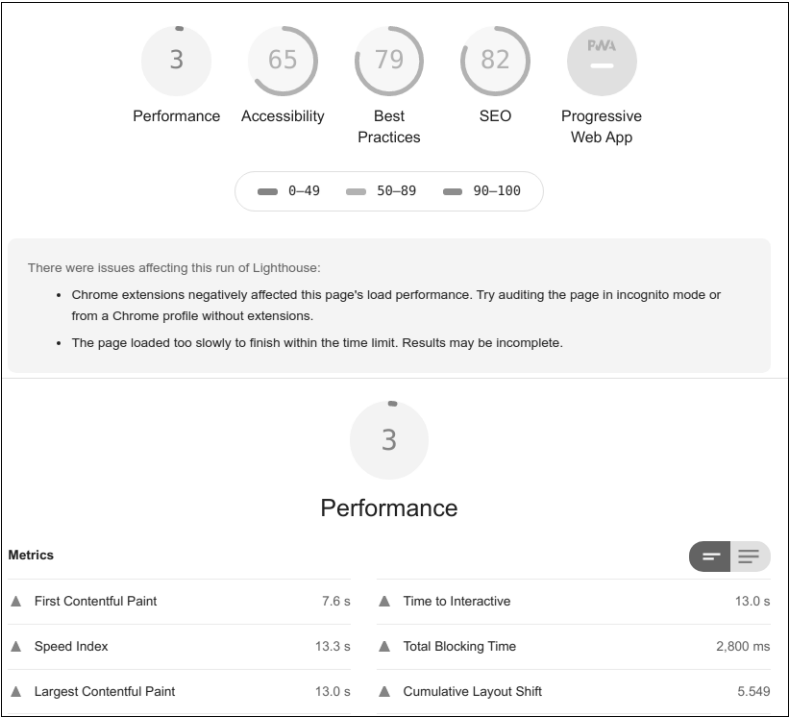


- **Dareboost** : créé initialement par une société française, Dareboost vous permet de réaliser une analyse détaillée de très nombreux points techniques et vous propose des solutions clé en main concernant des points de SEO, sécurité, UX, etc. Avec son système d'évaluation, vous pouvez vérifier quelles sont les performances d'un site. Dareboost offre d'autres services, notamment de monitoring, mais ils n'entrent pas dans le cadre de cet audit.



- **Google Lighthouse** : ce logiciel est très connu et aussi très visible, puisqu'il est intégré à Google Chrome (dans les outils de développeurs). Il est disponible comme plug-in pour Chrome et Firefox et accessible via le site **Google PageSpeed Insights dit PSI** : <https://developers.google.com/speed/pagespeed/insights/>

Il peut être aussi utilisé comme module sur un serveur Node.js. Notons qu'il s'intègre à Screaming Frog. Il fournit des indications sur les performances d'un site. Elles ne sont pas à prendre au pied de la lettre, mais peuvent être intéressantes pour un aperçu « high level ». Les données peuvent être exportées au format JSON pour être exploitées plus tard grâce au Lighthouse Report Viewer : <https://googlechrome.github.io/lighthouse/viewer/>



Pour un rendu plus concret des résultats, choisissez les options suivantes :

- **Appareil (Device)** : « **Mobile** », Google Lighthouse va ainsi simuler une connexion plus lente (la majeure partie des connexions au web se faisant via appareils mobiles).
- **Vider le cache (Clear Storage)** pour débiter les analyses avec un cache navigateur vide, comme lors d'une première connexion au site.
- **Simulate throttling** : pour simuler le changement de la page testée sur un smartphone moyen de gamme (Motorola 4G).

! Ces informations sont disponibles depuis la version 80 de Google Chrome, d'autres options étaient auparavant disponibles, mais peu fiables.

Attention : le problème avec Google Lighthouse/PSI est que l'on ne connaît pas l'emplacement du serveur de test ! Et vous obtiendrez des résultats bien différents selon que vous testiez le site web à Sydney, Saint-Brieuc ou à São Paulo.

La section dédiée à l'analyse des performances de sites web vous donnera de plus amples renseignements.