
Handbuch Webcontent

Version 0.1

Valentin Bruch

02.03.2024

Inhaltsverzeichnis

1	Einleitung	3
2	Was ist Webcontent?	5
3	Barrierefreiheit	7
3.1	Design und Struktur	8
3.2	Visuelle Elemente	8
3.3	Checkliste	8
3.4	Automatische Tests	9
4	Anzeige auf verschiedenen Geräten	11
4.1	Checkliste	12
4.2	Automatische Tests	12
5	IT-Sicherheit	13
5.1	Checkliste	13
5.2	Automatische Tests	14
6	Emails und Newsletter	15
6.1	Sicherheitsrisiken in Emails	15
6.2	Zwei Alternativen in einer Email	16
6.3	Anhänge	16
6.4	Verteiler und Datenschutz	16
6.5	Checkliste	17
7	Effizienz	19
7.1	Codierung von Medien	19
7.2	Checkliste	21
7.3	Automatische Tests	22
8	Datenschutz	23
8.1	Checkliste	23
9	Automatische Tests	25

9.1	Lighthouse	25
9.2	Website Carbon Calculator	26
9.3	WPScan	26
9.4	Pa11y	26

Abstract

Das Bespielen von Webseiten, Newslettern und anderen digitalen Kommunikationswegen erfordert meist kein technisches Fachwissen. Trotzdem hilft etwas Hintergrundwissen, um die Ergebnisse besser zugänglich, effizienter und sicherer zu gestalten. Dieses Dokument soll dieses Wissen vermitteln, ohne technisches Fachwissen vorauszusetzen.

Dieses Dokument als Webseite

<https://vbruch.eu/docs/webcontent>

Zu diesem Dokument beitragen

Anmerkungen, Korrekturen, Ergänzungen zu diesem Dokument: <https://github.com/stiglers-eponym/HandbuchWebcontent/issues>

KAPITEL 1

Einleitung

Es gibt heute zahlreiche Werkzeuge, um Webseiten und Newsletter mit optisch ansprechenden Inhalten zu füllen. Viele davon erfordern keine Programmierkenntnisse. Richtig angewandt erhält man eine ansprechende Darstellung auf fast allen Arten von Endgeräten. Darüberhinaus macht eine barrierefreie Gestaltung die gleichen Inhalte z.B. Menschen mit eingeschränktem Sehvermögen leicht zugänglich. Um diese Möglichkeiten zum Ansprechen einer möglichst breiten Zielgruppe nutzen zu können, hilft ein Grundverständnis davon, was Webcontent ist und worauf man bei dessen Erstellung achten sollte. Dieses Grundverständnis soll mit diesem Dokument geschaffen oder verbessert werden.

Neben der Barrierefreiheit und Zugänglichkeit gibt es weitere Themen, die beim Erstellen von Webcontent in Grundzügen benannt sein sollten. Hier gehen wir kurz auf Effizienz bzw. Geschwindigkeit, IT-Sicherheit und Datenschutz ein. Ziel ist es, eine Übersicht für Menschen ohne Interesse an den technischen Details bereitzustellen. An einzelnen Stellen werden wir deshalb für Aufgaben, die über das alltägliche Erstellen von Webcontent hinausgehen, an Admins verweisen. Denn wer ohne Hintergrundwissen mit IT-Systemen wie Webseiten oder Newslettern arbeitet, sollte immer eine Ansprechperson mit mehr technischem Wissen haben, beispielsweise um die Sicherheit der IT-Systeme zu prüfen.

Dieses Dokument ist folgendermaßen strukturiert. Nach einem Überblick über den Begriff "Webcontent" in [Abschnitt 2](#) erklären die Abschnitte [3 \(Barrierefreiheit\)](#) bis [8 \(Datenschutz\)](#) voneinander unabhängig einzelne Themen. Zu den einzelnen Abschnitten sind automatische Tests aufgelistet, die in [Abschnitt 9 \(Tests\)](#) genauer erklärt werden.

Was ist Webcontent?

Das World Wide Web hat seit den 90er Jahren unsere Welt verändert. Webseiten und Emails zeigen interaktiv und flexibel Inhalte basierend auf einem gemeinsamen Darstellungsformat an. Diese Inhalte¹ bezeichnen wir als *Webcontent*.

Intern besteht Webcontent im Wesentlichen aus einer Textdatei, die Anweisungen zur Struktur des Dokuments und Hinweise zur Darstellung dieser Struktur enthält. Ergänzt wird diese Datei durch Bilder, weitere Anweisungen zur Darstellung und Programmcode für interaktive Elemente. Wichtig ist, dass die Darstellung stets einer Struktur folgt — oder zumindest folgend sollte. Die definierte Struktur kann nicht nur zur Darstellung auf einem Bildschirm genutzt werden, sondern auch zum automatisierten Vorlesen durch einen Screenreader. Wenn man Webcontent erstellt, ist also nicht nur die Darstellung auf dem Bildschirm wichtig, sondern auch dem zugrundeliegende Dokumentenstruktur.

Den Unterschied zwischen Struktur und Darstellung kann man an einer Überschrift sehen. Wird eine Zeile als Überschrift formatiert, so ist sie über den Screenreader auch für Menschen mit eingeschränktem Sehvermögen direkt als Überschrift erkennbar. Wird dagegen die gleiche Darstellung erreicht, indem händisch eine andere, größere Schrift ausgewählt wird, so geht die Information über die Struktur verloren und es ist nicht mehr für jede*n erkennbar, dass es sich um eine Überschrift handeln soll.

¹ Technisch handelt es sich um HTML-Dokumente mit CSS-Stil.

Barrierefreiheit

Barrierefreiheit bei längeren Dokumenten wie Webseiten beinhaltet insbesondere folgende Aspekte:

1. Das Dokument hat eine klare, logische Struktur.
2. Die visuelle Darstellung ist auch für Menschen mit schlechten Augen erkennbar. Beispielsweise ist der Text groß genug und hat einen ausreichenden Kontrast zum Hintergrund.
3. Für rein visuelle Elemente wie Bilder oder Logos sind kurze Text zur Beschreibung hinterlegt, die von Screenreadern vorgelesen werden können. Alternativ können Elemente auch als irrelevant für Screenreader gekennzeichnet werden.
4. Das Dokument hat eine Navigation, die mit verschiedenen Eingabegeräten genutzt werden kann. Üblich sind Maus, Touchscreen und Tastatur (mit der Tab-Taste). Menüs sind als solche auch für Screenreader erkennbar.
5. Texte sind verständlich geschrieben, klar strukturiert und einfach zu lesen. Fachbegriffe werden vermieden oder erklärt. Das ist zwar kein technischer Punkt, aber trotzdem wichtig.

3.1 Design und Struktur

Die Punkte 1 und 4 werden üblicherweise beim Erstellen einer Webseite oder eines Designs bedacht. Beim Hinzufügen neuer Inhalte sollten sich diese Punkte ohne Mehraufwand ergeben.

Auch Punkt 2 zur visuellen Darstellung muss bereits beim Erstellen eines Designs bedacht werden. Aber auch bei einzelnen Inhalten, etwa bei der Darstellung von Text vor einem Bild, muss ein ausreichender Kontrast und ausreichend große Schrift bedacht werden.

3.2 Visuelle Elemente

Wesentlich beim Erstellen von Inhalten ist Punkt 3 zur Beschreibung von visuellen Elementen. Für jedes hochgeladene Bild sollte eine kurze Beschreibung erstellt werden, um den Zweck des Bildes für Menschen mit visueller Beeinträchtigung erkennbar zu machen. Üblicherweise geschieht das noch von Hand.

Der sogenannte "alt"-Text für Bilder wird angezeigt, wenn ein Bild nicht angezeigt werden kann, und er beschreibt das Bild beim Vorlesen der Seite durch einen Screenreader. Bei WordPress kann man für jedes Bild einen alt-Text anlegen und auch bei Facebook und Instagram kann ein alt-Text für Bilder angegeben werden.

Falls Videos eingebettet werden, können Untertitel zur Barrierefreiheit beitragen. Es gibt Software zum automatischen Hinzufügen von Untertiteln, die aber in der Regel noch eine manuelle Korrektur brauchen.

3.3 Checkliste

- Haben alle Bilder einen alt-Text mit einer kurzen Beschreibung?
- Hat der Text immer einen ausreichender Kontrast zum Hintergrund?
- Falls mehrere Absätze mit Text vorhanden sind: Gibt es eine klare Struktur?
- Ist der Text auf allen Geräten (Computer, Smartphone, evtl. Tablet) gut zu lesen?

Fortgeschritten

- Sind Links ausreichend weit voneinander entfernt, damit man auf einem Touchscreen nicht versehentlich das falsche anklickt?

- Bei einer Webseite: Kann man mit der Tab-Taste gut durch die auf der Webseite vorhandenen Links und Menüs navigieren?

3.4 Automatische Tests

- *Lighthouse*, <https://pagespeed.web.dev>
- *Pa11y*, <https://pa11y.org>

Anzeige auf verschiedenen Geräten

Webcontent soll meist auf verschiedenen, teils sehr unterschiedlichen Geräten angezeigt werden können: vom Computer (“Desktop”) mit großem, breitem Display bis zum Smartphone mit kleinem, schmalen Display (“mobil”). Üblicherweise hat eine Webseite ein Design, das dafür sorgt, dass die Inhalte auf allen Geräten gut dargestellt werden. Damit das funktioniert, sollte beim Erstellen der Inhalte nicht zu einseitig auf die Darstellung auf einem bestimmten Gerät geachtet werden. Wenn man manuell Einstellungen zur Darstellung wie die Schriftgröße anpasst, oder wenn man komplexere Elemente wie Spalten oder Tabellen einfügt, sollte man die Ansicht auf verschiedenen Geräten prüfen. Verwendet man nur die vorgegebenen Designelemente ohne manuellen Anpassungen, so sollte die Ansicht automatisch auf allen Geräten funktionieren.

Hinweis: Auch ein Webseite-Design ist meistens nicht perfekt. Es fällt aber in der Regel in den Aufgabenbereich der Admins, an die man sich bei Problemen wenden sollte.

Zum Testen der Darstellung kann man Webseiten mit verschiedenen Geräten öffnen. Es hilft aber auch, am Computer die Größe des Browserfensters zu verändern und zu prüfen, ob die Webseite weiterhin gut aussieht. Darüberhinaus bieten viele Desktop-Browser die Möglichkeit, auf eine mobile Darstellung (also Smartphone-Ansicht) umzustellen. Bei Firefox gibt es dafür die Tastenkombination Strg+Shift+M. In Google Chrome und ähnlichen Browsern kommt man über die Entwicklertools an eine ähnliche Ansicht (erste Strg+Shift+I, dann Strg+Shift+M). So kann man die Ansicht auf Smartphones verschiedener Größe simulieren.

4.1 Checkliste

- Sieht die Darstellung am Computer bei verschieden großem Browserfenster gut aus?
- Sieht die Darstellung auf dem Smartphone gut aus?
- Genau getestet werden sollte insbesondere wenn...
 - manuelle Anpassungen z.B. der Schriftgröße vorgenommen wurden
 - Tabellen, Spalten oder andere komplexere Designelemente verwendet werden

4.2 Automatische Tests

- *Lighthouse*, <https://pagespeed.web.dev>

IT-Sicherheit auf Webseiten ist primär Aufgabe der Admins. Wichtig ist, dass es jemand mit dem nötigen Fachwissen gibt, die oder der sich für die Webseite verantwortlich fühlt und sich darum kümmert. Wer sich nur um Inhalte kümmert, hat insbesondere die Verantwortung, mit den eigenen Zugangsdaten und Berechtigungen verantwortungsvoll umzugehen.

5.1 Checkliste

- Nie neue Plugins installieren oder aktivieren, ohne das mit Admins abzusprechen
- keine iframes einbinden, ohne das mit Admins abzusprechen
- Weiß ich, wer für IT-Sicherheit verantwortlich ist und an wen ich mich mit Fragen wenden kann?
- Zugangsdaten immer geheim halten
- Verwende sichere Passwörter. Empfehlung: Verwende einen Passwortmanager
- Habe ich mehr Berechtigungen als ich brauche? Dann sollten die Berechtigungen reduziert werden.

Fortgeschritten

- Bei Webseiten: Werden regelmäßig Backups gemacht?

- Bei Webseiten: Werden regelmäßig Updates eingespielt?

5.2 Automatische Tests

- nur eingeschränkt: *Lighthouse*, <https://pagespeed.web.dev>
- nur für WordPress: *WPScan*, <https://wpscan.com>

Emails und Newsletter

Email ist ein Standardprotokoll, das zum Versenden von Nachrichten entwickelt wurde. Es erlaubt aber auch das Versenden von HTML-Code, also von Webcontent im gleichen Format wie auf Webseiten. Beim Verwenden von Emails über reinen Text hinaus sollte man folgendes beachten.

6.1 Sicherheitsrisiken in Emails

Per Email kann man jemandem direkt Webcontent schicken, oft ohne dass eine echte Überprüfung des Absenders stattfindet. Deshalb sind Emails beliebte Einfallstore für Cyberangriffe. Um dem entgegenzuwirken, ist es vernünftig, nur unbedenkliche Inhalte in Emails direkt anzuzeigen. Ein so eingestelltes Emailprogramm wird also Emails möglicherweise abgewandelt darstellen. Deshalb sollte man in Emails auf Elemente verzichten, die potentiell für Hackerangriffe genutzt werden könnten (z. B. JavaScript). Sonst wirkt eine Email schnell potentiell gefährlich.

6.2 Zwei Alternativen in einer Email

Die sicherste Darstellung von Emails beschränkt sich nur auf den Text. Oft besteht eine Email technisch aus zwei alternativen Teilen: einem nur-Text-Teil und einem HTML-Teil (Webcontent). Das Emailprogramm der Empfänger*in entscheidet, welcher Teil davon angezeigt wird. Programme zum Versenden von Newslettern sind meistens so eingestellt, dass der damit erstellte Webcontent automatisch in einen nur-Text-Teil konvertiert und in der Email mitgeschickt wird. Das funktioniert dann gut, wenn man sich an die Empfehlungen zur *Barrierefreiheit* hält.

6.3 Anhänge

Eine praktische Funktion von Emails sind Anhänge. Jedoch ist zu beachten, dass der Datentransfer per Emailanhang grundsätzlich ineffizienter ist als andere Formen des Datentransfers. Große Anhänge verbrauchen Speicherplatz bei Sender*in und Empfänger*in, und machen das Laden von Emails langsam. Deshalb ist typischerweise eine Maximalgröße für eingehende Emails eingestellt, sodass zu große Emails einfach nicht zugestellt werden — teils ohne dass der oder die Absender*in das merkt.

Wenn große Dateien verschickt werden sollen, gibt es meist effizientere Lösungen als Emails. Die zu verschickende Datei wird bei einem Cloudservices (z. B. Nextcloud) oder auf einer Webseite hochgeladen und per Email wird nur der Link zu der Datei verschickt. Das ist effizienter und lässt die Email schneller laden, insbesondere wenn es sich um eine große Datei handelt.

6.4 Verteiler und Datenschutz

Eine weitere Gefahr bei Emails sind versehentliche Datenlecks. Wenn man eine Email an viele Empfänger*innen (also einen großen Verteiler) schickt, sollte man diese Empfänger*innen immer in BCC packen, sodass die Emailadressen gegenseitig nicht einsehbar sind. Das verschicken von Emails an einen offenen Verteiler kann einen Verstoß gegen den Datenschutz bedeuten. Außerdem erhöht es das Risiko von Spam- und Phishingemails und damit auch von erfolgreichen Cyberangriffen.

6.5 Checkliste

- große Verteilung immer in BCC, nicht in Adressat oder CC
- keine großen Anhänge per Email verschicken
- niemals große Anhänge per Email an einen großen Verteiler schicken
- Emails sollten im nur-Text-Format und in HTML lesbar sein

Fortgeschritten

- Wenn Newsletter verschickt werden, sollten DKIM und SPF-records angepasst sein. (Aufgabe für Domain-Admins)

Die Effizienz von Webcontent ist aus mehreren Gründen wichtig. Effizienz verringert die Ladedauer von Webseiten und erhöht damit die Nutzerfreundlichkeit. Außerdem müssen weniger Daten übertragen werden und der Energieverbrauch ist geringer. Das ist durchaus relevant, denn der Energiebedarf des Internets ist beachtlich¹.

Zu bedenken ist die Effizienz immer, wenn größere Datenmengen übertragen werden, etwa bei Bildern oder Videos. In anderen Fällen ist die Effizienz eher eine Aufgabe der Admins.

7.1 Codierung von Medien

Bei Bildern, Videos und Audiodaten sucht man typischerweise einen Kompromiss zwischen Dateigröße und Qualität. Entscheidend sind die Auflösung von Bildern und Videos, aber auch das Dateiformat und andere Einstellungen. Durch ein grundlegendes Verständnis davon kann man vermeiden, dass Ladedauer und Energieverbrauch der Webseite ein Vielfaches des Nötigen betragen.

¹ Laut Website Carbon Calculator ist der Energiebedarf des Internets höher als der Stromverbrauch von Großbritannien (<https://www.websitecarbon.com>).

7.1.1 Bilder

Das häufigste Problem bei der Effizienz von Webcontent ist das Format von Bildern. Verschiedene Dateiformate von Bildern sind für verschiedene Anwendungen geeignet. Für Logos und Zeichnungen, bei denen viele Pixel die exakt gleiche Farbe haben, eignet sich das PNG-Format, mit dem Bilder verlustfrei komprimiert werden.² Für Fotos eignet sich das JPEG-Format, bei dem ein Kompromiss aus Dateigröße und Bildqualität meist so gewählt werden kann, dass ohne sichtbare Qualitätsverluste eine deutlich kleinere Datei erzeugt wird.

Speichert man ein Logo als JPEG statt als PNG, so wirkt es bei genauer Betrachtung leicht verpixelt. Wenn man ein Foto als PNG statt als JPEG speichert, dann kann die Datei schnell zehn mal so groß wie nötig werden. Das kann eine Webseite erheblich verlangsamen und sehr ineffizient machen.

Neben PNG und JPEG gibt es das modernere Format WebP. WebP kann Bilder entweder verlustfrei (wie PNG) oder verlustbehaftet (wie JPEG) komprimieren und ist insbesondere bei starker Komprimierung besser. WebP wird von ca. 97% der Endgeräte unterstützt.³

Neben dem Format ist die Auflösung von Bildern entscheidend für die Dateigröße. Zu geringe Auflösung von Bildern lässt diese verpixelt wirken. Eine zu hohe Auflösung verzögert das Laden einer Webseite oder Email. Es gibt Webseiten, die Bilder automatisch in der passenden Auflösung bereitstellen. Wenn das aber nicht der Fall ist, müssen diejenigen, die Bilder hinzufügen, auf die passende Auflösung der Bilder achten.

Übrigens: Technisch ist es möglich, Bilder in verschiedenen Auflösungen und Dateiformaten bereitzustellen, sodass je nach Endgerät und Webbrowser die passende Version geladen wird. Das Einrichten dieser Funktion ist aber Aufgabe der Admins.

7.1.2 Videos

Noch wichtiger als bei Bildern ist die Komprimierung (oder Codierung) bei Videos, da für Videos deutlich mehr Daten übertragen werden müssen. Die Komprimierung ist hier grundsätzlich ein Kompromiss zwischen Qualität, Datenrate bzw. Dateigröße und der zum Komprimieren benötigten Rechenleistung.

Wenn man Streamingdienste wie YouTube oder PeerTube zum Einbetten von Videos verwendet, muss man sich um die Komprimierung in der Regel keine Sorgen machen. Man lädt das Video in hoher Qualität hoch und der

² Bei falscher Verwendung kann die Komprimierung im PNG-Format ausgeschaltet werden, wodurch das Übertragen von Bildern sehr ineffizient wird.

³ Quelle: <https://caniuse.com/webp>, Stand Februar 2024

Streamingdienst codiert es passend.⁴ Außerdem haben diese Dienste in der Regel viele Server, um die Videos bereitzustellen. So wird der eigene Server entlastet. Jedoch muss ggf. in der Datenschutzerklärung einer Webseite darauf hingewiesen werden, dass externe Anbieter zum Streamen von Videos verwendet werden, da dadurch Daten an diese externen Dienstleister übertragen werden.

Wenn man keinen Streamingdienst verwendet und sich selbst um die Komprimierung kümmert, gibt es einige Einstellungen zu beachten. Die Auswahl des Videocodecs ist entscheidend für die Effizienz der Codierung und dafür, welche Endgeräte das Video abspielen können. Zu beachten ist, dass das Videocodec nicht gleichbedeutend mit dem Dateiformat und der Dateiendung ist: MP4-Videos können beispielsweise unterschiedliche Videocodecs nutzen, von denen nicht alle auf allen Geräten funktionieren. Verbreitete Videocodecs für Webcontent sind:

- VP9 (in WebM-Videos): von den meisten Endgeräten unterstützt
- H.264 (in mp4-Videos): alt und weniger effizient, von quasi allen Endgeräten unterstützt

Modernere Codecs wie AV1 und H.265 erreichen bei gleicher Qualität geringere Dateigrößen, werden jedoch von weniger Endgeräten unterstützt.⁵

Bei der Codierung gibt es zudem Einstellungen zu Datenrate oder Qualität von Video und Audio sowie teils zum Rechenaufwand der Codierung. Dazu kommen Einstellungen zur Auflösung und Framerate, die in der Regel bereits beim Aufzeichnen des Videos gesetzt werden. Allgemeine Empfehlungen dazu findet man bei Software zu Videocodierung wie etwa [ffmpeg](#)⁶.

7.2 Checkliste

- Sind Bilder im richtigen Format?
- Haben Bilder die richtige Auflösung?
- Sind eingebettete Videos effizient komprimiert und nicht zu groß?

⁴ Empfohlene Einstellungen zum Hochladen bei Youtube: <https://support.google.com/youtube/answer/1722171>

⁵ Es ist technisch möglich, verschiedene Formate bereitzustellen, sodass das Endgerät sich das passende Format aussucht. Dann können moderne Formate wie AV1 verwendet werden. Die begrenzte Unterstützung von H.265 durch Browser hat (auch) lizenzrechtliche Gründe.

⁶ <https://trac.ffmpeg.org/wiki/Encode/VP9>

7.3 Automatische Tests

- *Website Carbon Calculator*, <https://www.websitecarbon.com>
- *Lighthouse*, <https://pagespeed.web.dev>

Der Schutz personenbezogener Daten ist immer dann wichtig, wenn persönliche Daten verarbeitet werden, wie etwa zum Anzeigen einer Webseite oder beim Verschicken von Newsletters per Email. Zwar kümmern sich in der Regel Admins um den Datenschutz einer Webseite, doch das erfordert, dass ohne Abstimmung mit den Admins keine Plugins installiert oder externen Inhalte eingebunden werden. Potentielle Datenschutzverstöße können bereits durch das Einbinden von Schriftarten von Google möglich.

Besonders zu beachten ist der Datenschutz beim Umgang mit Listen von Emailadressen, etwa zum Versenden von Newsletters. So sollten niemals Emails an große offene Verteiler verschickt werden, damit keine Emailadressen ohne Zustimmung weitergegeben werden.

8.1 Checkliste

- Habe ich Zugriff auf personenbezogene Daten? Dann sollte ich mich detaillierter mit Datenschutz befassen.
- Vorsicht bei Emails an offene Verteiler!

Automatische Tests

Für Webseiten gibt es einige automatische Tests. Diese Tests dienen zum Erkennen von Fehlern und zum Optimieren etwa von Webseiten. Sie können jedoch das bewusste Beachten der hier formulierten Empfehlungen nicht ersetzen. Die folgende Auflistung ist zudem unvollständig.

9.1 Lighthouse

Lighthouse⁷ ist ein Werkzeug zum automatischen Testen von Webseiten. Getestet werden die Aspekte Leistung, Barrierefreiheit, Best Practice und die Bewertung durch Suchmaschinen (search engine optimization, SEO).

Man kann den Test sowohl für Computer, als auch für die Ansicht auf Smartphones durchführen. Die Ergebnisse von Lighthouse geben einen guten Überblick über wichtige Aspekte einer Webseite und konkrete Handlungsempfehlungen. Allerdings sind diese Handlungsempfehlungen etwas technisch.

Zum Nutzen von Lighthouse gibt es zwei Methoden¹

- ganz einfach: über die Webseite <https://pagespeed.web.dev>
- fortgeschritten: über die [Chrome-Entwicklertools](#)⁸

⁷ <https://github.com/GoogleChrome/lighthouse>

¹ Für Expert*innen gibt es noch mehr Methoden.

⁸ <https://developer.chrome.com/docs/lighthouse/overview?hl=de#devtools>

Vorsicht: Beim Nutzen der Chrome-Entwicklertools sollte man vorsichtig sein. Insbesondere sollte man die Konsole (Texteingabe) nur verwenden, wenn man wirklich weiß was man tut.

9.2 Website Carbon Calculator

Der [Website Carbon Calculator](https://www.websitecarbon.com)⁹ schätzt ab, wie viel Energie eine Webseite braucht und welcher CO₂-Ausstoß damit einhergeht. Webseiten werden auf einer leicht verständlichen Skala bewertet, die Ergebnisse sind also stark vereinfacht und nicht detailliert.

9.3 WPScan

Mit [WPScan](https://wpscan.com)¹⁰ können WordPress-Webseiten automatisch auf bekannte Sicherheitslücken geprüft werden. Ein solcher Test ersetzt nicht das Prüfen von Plugins oder das regelmäßige Einspielen von Sicherheitsupdates.

9.4 Pa11y

[Pa11y](https://pa11y.org)¹¹ ist ein Werkzeug zum Testen der Barrierefreiheit von Webseiten. Dieses Werkzeug zu nutzen erfordert technisches Hintergrundwissen und sollte nicht ohne technisches Wissen zu JavaScript eingerichtet werden.

Vorsicht: Bitte nicht versuchen, dieses Werkzeug zu installieren, wenn man die Hintergründe einer Anleitung zur Installation nicht versteht!

⁹ <https://www.websitecarbon.com>

¹⁰ <https://wpscan.com>

¹¹ <https://pa11y.org>