Overview of the Web Environmental Sustainability Guidelines

This document provides a simple set of guidelines for the development of more environmentally friendly websites. The scope of this document is focussed mostly on minimising the amount of energy that a website consumes. It is not a set of broader “sustainability” standards that would cover more of the UN Sustainable Development Goals. We acknowledge that whilst sustainability does need to be considered more widely a set of standards to help minimise the increase in digital footprints is needed quickly.

We also acknowledge that the carbon impact of a digital solution is much broader than its digital footprint and that digital solutions can have offline carbon benefits as well as causing significant offline carbon emissions. And whilst we would encourage teams to consider these wider impacts, often driven by behavioural changes caused by digital technologies, these are outside the scope of what a simple set of standards can achieve.

There is an existing, broad community who are looking at different areas of digital footprints. We are inviting as many from that community to feed into these standards as possible and would like to launch them as a broad collaboration with the community in order to give them wider reach and credibility. We also acknowledge that these standards may be subsumed into a wider set of standards in the future. For example potentially into a set of standards from the W3C who we believe are perhaps best placed to own these types of standards.

We would invite you to comment on any aspect of these standards including, but not limited to:

- The best practices that were collated via multiple workshops into a Miro Board: https://miro.com/app/board/uXjVO6kQXh0=/
- The top level categories for the standards
- Any of the wording or examples given
- The weighting / score applied to each guideline*

*Rather than following a levelled approach to each element of the standard (e.g. the A, AA, AAA format of the WCAG) our suggestion is to have a simple scoring system that will enable us to define thresholds that can be met in order that the site can be shown to adhere to a certain level with regards to its impact on the environment.

Vision Statement (wording TBC)

Our goal is that the planet is included as a design constraint for all website builds. Whilst these guidelines cover core areas in a lot of detail the spirit of these guidelines are to ask whether what you're adding is actually required. And if it is required, is there a better, lower-carbon way of doing it?

0. Governance

Reference	Description	Example / Notes	Weighting / Score
0.1	The team has a documented policy for low-carbon design patterns in place	E.G. Documented policy to minimise CSS, follow a Design System, re-use fonts and UI components.	8

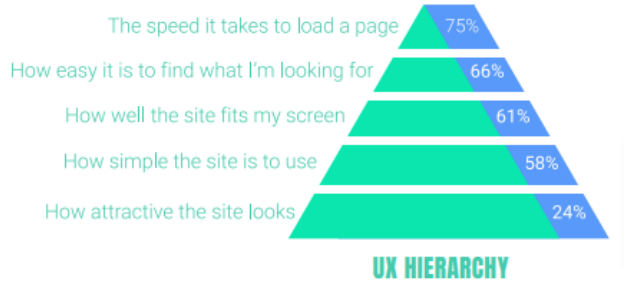
1. UX / Design

Reference	Description	Example / Notes	Weighting / Score
1.2	The team has a content decision tree to minimise unnecessary use of rich content	Decision gates to include (as a minimum): 1. Do we need this content? 2. Can it be text based? 3. Can an illustration delivered via CSS or vector be used? 4. Can a web animation or interactive element be used? 5. Can audio or image(s) be used? Video will only be considered when the above questions have been genuinely considered.	5
1.3	The number and dimensions of images used in the design are actively reduced	Images are one of the biggest sources of data storage and transfer, and therefore emissions. Images are vital in many circumstances, but	5

		where they're not, remove them. Stock imagery doesn't add any value to people using a site, particularly to those with accessibility needs. Ecosia's recent rebrand is a great example of minimal use of imagery. It's used sparingly, where it will have impact and is appropriately sized. Their illustrations have taken centre stage in a brand that could have easily been filled with photos of their funded tree planting projects. INSERT EXAMPLE OF OVER USE OF IMAGES *E.g. Do you need full-bleed header or background images? Could smaller images be used to convey the same message?*	
1.4	The website has content which raises awareness of digital emissions	Awareness of the emissions caused by digital products is very low. To help raise that awareness websites must have content helping to explain how the internet causes emissions and what can be done about it.	5
1.5	Website displays amount of data transfer or estimated emissions	The Website Carbon badge is a great example of this and is most frequently implemented in the footer of pages. Other tools are available to measure carbon and data transfer. What gets measured gets done! Explaining what has been done on the website to reduce emissions or data transfer is a great way to help people understand this issue.	3
1.6	Website uses "carbon aware design"	Branch magazine is an amazing example of carbon aware design. Organic Basics is another. Both websites know the emissions factor of the grid the person using the site is on, and as a result change colours, image resolutions, image	8

		appearance (consider dithering like Waste Not) and logos, depending on whether the grid is being mainly powered by renewable energy or fossil fuels. Lookups such Electricity Maps or the Green Software Foundation SDK can be used.	
1.7	Website is designed with colours that use less energy on a screen	As research from Google and Android shows, certain colours on screens use more energy. White uses about 6 times as much as black. Blue uses twice as much as green and red.	2
1.8	Avoid dubious user retention tactics	The lowest emitting webpage is the one that is never created. This is a potentially flippant comment but the digital industry is driven by attention metrics; our industry employs dark patterns to control people's attention. Get people on and off your site as quickly as possible by enabling them to do what they came to do. If people find what they need quickly and leave sooner they will be creating less data transfer and therefore less emissions. The types of retention tactics we mean are brilliantly covered by Humane Tech. Their ledger of harm (particularly "Attention and Cognition") and their video on extractive technology are great ways to understanding this issue.	2
	User journey optimisation	Get people on and off the website as soon as possible	

2. Images

Reference	Description	Example / Notes	Weighting / Score
2.1	Use the most efficient file compression formats	WebP and AVIF are the two formats enabling greatest compression whilst being available to most people, especially now that Safari supports AVIF as of September 2022. This tips the scales in favour of AVIF as the best choice seeing as it compresses better than WebP. ShortPixel is a great compression plugin for Wordpress. ImageMagick is great and is management system agnostic. ImageSharp is great .NET.  Remember, speed of loading is still the most important thing to people using your site; image compression is a huge part of that.	13
2.2	Optimise images prior to uploading to the CMS or server	Content management systems and content delivery networks will (usually) compress and convert images as needed for 2.1. Therefore, this is about making image files smaller before	2

		they get loaded into a CMS or CDN. This type of optimisation includes removing metadata, removing complexity in the image, removing colours, blurring backgrounds etc. Tools such as ImageOptim can help here.	
2.3	Images will be stored and served at the largest dimensions that they are used on the website	Taking into account image density of target devices	5
2.4	Avoid cropping techniques in the browser, where areas of an image are not being displayed		3

3. Video

Reference	Description	Example / Notes	Weighting / Score
3.1	Do not autoplay videos	Playing a video when the person using the site hasn't indicated they want to watch it is a waste of data transfer. This is also an attention extracting technique that people are better without.	13
3.2	Video should only start loading when requested by the user	Just the presence of a third party video embed on a page causes data transfer (and therefore emissions) that only benefits the third party video provider. Therefore, video embeds should only be loaded once the person using the site actively indicates they want to use a video. This means using a fake video thumbnail that, when clicked, loads the video. The Climate Group uses this technique. You might want to consider auto playing the video once the embed is loaded to avoid the person having to click twice. Don't worry, auto playing is allowed under these circumstances!	8
3.3	Video defaults to streaming at the lowest practical resolution	Using a video streaming service e.g. Vimeo could solve a lot of size and technical issues	5
	Recommendation from 10 Degrees		

	Recommendation from EY		

4. Content

Reference	Description	Example / Notes	Weighting / Score
4.1	A documented content audit takes place (minimum quarterly) in order to remove redundant, out of date, trivial, no longer used and excessively large content	If content is any of these things, having it live on the internet is a waste of energy to both store and serve to people. Having this type of content available for people to find is also a poor user experience as it will distract people from the task(s) they have visited the site for. “Excessively large” relates to the file size of a webpage; for example, pages can become bloated over time as images are added to them. Tools such as Sitebulb, Siteimprove, Monsido and analytics can help with content audits.	3
4.2	Use a digital-first content approach	The website avoids downloadable files where a low-carbon webpage is a viable alternative. This avoids the need for additional files needing to be uploaded, such as PDFs. PDFs are often difficult for users to find and harder for content teams to manage, maintain and store. Digital-first content is more accessible, allows users to easily translate into another language and users can still save a copy and print if they need to. The CDR Primer is a fantastic example of something which could have easily served as a downloadable PDF but is instead an accessible, lightweight website. *We need to just nod to the downloads of invoices and certificates etc*	5

4.3	Make important information clear and quick to find	Don't make users have to search for important information, such as contact details. If a user needs to spend several minutes trying to find a phone number, they'll spend longer on the site but for the wrong reasons. The RNLI is a good example with their Supporter Experience Team number under the homepage banner image.	2
4.4	Write descriptive headings	People read differently online and mostly only read 20% of what's on a webpage. Well-written, structured headings help users to find the information they need quickly and easily, regardless of what device they're using. If someone's using a screen reader, they'll also be able to navigate the content more easily and can 'tab' through different headings. All of this means less time on pages, creating fewer emissions. The Citizens Advice writes particularly clear content. *Nod to the fact this is also good for accessibility and screen readers*	1

5. Fonts

Reference	Description	Example / Notes	Weighting / Score
5.1	Font variations on the site are kept to a minimum	Reduce the number of fonts to only the bare minimum required. Putting even one word in italics requires loading an entire italic font file for that font weight.	2
5.2	Website uses the most efficient font file format available	If you are using custom fonts in your website, only use .woff and .woff2 web fonts, nothing else. (https://stackoverflow.com/a/11002874) https://web.dev/variable-fonts/	3
5.3	All unnecessary characters are removed from font files	Do not load other language glyphs (eg. Cyrillic or Greek) if the site does not support translation options for those languages. Same is true for alternate characters, flourishes, ligatures or special symbols if the site design does not call for them. Add this before existing sentence: "You can reduce the file size of each font file by subsetting it to only include the characters or range of characters you need. (https://css-tricks.com/three-techniques-performant-custom-font-usage/, https://markoskon.com/creating-font-subsets/)	2
5.4	Site design uses only system fonts	No custom fonts loaded on the site Consider using a system font stack to make use of fonts already installed on users' computers to eliminate the need for downloading any font files.	3

		Your developer's choice of CSS reset tool may have this option already built in. Resources: https://github.com/csstools/sanitize.css#typography, https://systemfontstack.com/, https://css-tricks.com/snippets/css/system-font-stack/	
	Serving of fonts	*Could use of font platform be relevant here. Fontshare vs Google Fonts for privacy and data collection for instance. Same arguments as Google Analytics vs Fathom etc* *If you are using Google Fonts, use this tool (https://google-webfonts-helper.herokuapp.com/fonts) to download local copies and host on your own website to prevent Google from tracking your users when they load the font.* *Other resources for making fonts more performant: https://web.dev/optimize-webfont-loading/, https://css-tricks.com/the-best-font-loading-strategies-and-how-to-execute-them/*	

6. Web Development

Reference	Description	Example / Notes	Weighting / Score
6.1	Images should only be loaded when they are within the visible area of the browser (“lazy loading”)	Browser support for native lazy loading is on the way (https://web.dev/browser-level-image-lazy-loading/)(https://caniuse.com/loading-lazy-attr), but a popular, more powerful, performant and SEO friendly lazy loader (https://github.com/aFarkas/lazysizes) is available now with JavaScript fallbacks.	8
6.2	Ensure that images are responsive (e.g. by utilising image srcsets) so that images are optimised for the screen size and pixel density that they are being displayed on.	https://developer.mozilla.org/en-US/docs/Learn/HTML/Multimedia_and_embedding/Responsive_images This extensive article by Addy Osmani, an engineering manager on Google Chrome, is an excellent resource to develop an approach for image optimisation and loading. (https://www.smashingmagazine.com/2021/04/humble-img-element-core-web-vitals/) Also see *Maximally optimizing image loading for the web* (https://www.industrialempathy.com/posts/image-optimizations/)	5
6.3	All web pages should be measured and tracked by data transfer size (Kb)		8

6.4	Website follows a modular design system in order to encourage reuse of existing assets	Ideally the design system should be built into the CI / CD process to ensure that deployments are blocked that could add bloat to the front-end. E.g. bloated CSS files	5
6.5	All JS and CSS is minified	Make appropriate use of JS dynamic imports	5
6.6	Where possible overwrite CMS templates, 3rd party scripts and plugins that do not adhere to these guidelines	Ridding them of excess code, calls and libraries where possible E.g. Susty, Granola, Underscores etc - however, even these can be further stripped back / optimised.	13
6.7	Only load the minimum assets needed to render a page	JS and CSS files are only loaded if the web page requires them No duplicate resources are loaded - JS and CSS are prime suspects here. Avoid 3rd party libraries that embed JS libraries that you have already included. Use tools to assist with dead code detection. Remove code that is no longer in use. E.g. purgeCSS (look this up)	8
6.8	Consider the lightest weight analytics platform that will meet the needs of the product	E.g. Plausible, GA Lite, Cabin Analytics, Koko, Netlify Consider if a 3rd party party analytics tool is even needed; some CMS and hosting providers will provide their own high level stats that are useful enough for most small orgs/websites. GA4 - what reductions has GA4 bought in regards to data transfer? Also mention about turning off things like hotjar	2

		when they're not being used.	
6.9	Practice data minimisation; only capture data that will enable you to make a tangible difference to users	How will knowing a particular thing help the people you serve better achieve their goals? Then define the bare minimum data you need to capture to deduce what you need to know. Don't capture data and then define what you want to know from it or mine the data to try and uncover interesting facts. This approach needs to cover all data points from forms being filled in, to web analytics.	2
6.10	Use a print stylesheet to minimise ink and paper consumption	Add example of using browser dev tools to simulate print media query, otherwise it's hard to test	1
6.11	Website uses text compression[1]	E.g. Compression for text-based resources served over HTTP with gzip, deflate, brotli or equivalent.	1
6.12	Practice carbon aware development	Carbon aware development is where the current carbon intensity of the electricity grid drives the decision as to when energy intensive processes (such as deployments, data transfers, file uploads/downloads) should be run. With the premise of: do more when renewable energy is flooding the grid, do less when fossil fuels are. Tools such as ElectricityMap highlight grid intensity. Open Source projects such as the Carbon Aware SDK are written specifically for aiding carbon aware development. N.B. this could be a server-side batch job (for	8

[1] https://web.dev/uses-text-compression/

		example) or front-end adaptations.	
	Recommendation from Cyber-Duck Ltd		
	Recommendation from Unic AG		
	Recommendation from EY		

7. Development Operations

Reference	Description	Example / Notes	Weighting / Score
7.1	Prevent bad robots	Different research shows different results for bot traffic vs human traffic on the internet. However, most seem to agree that bad bot traffic is at about 25%, doing things ranging from content scraping to DDoS attacks to getting user data and much more, sophisticated bots are mimicking human behaviour and are becoming more difficult to detect. Hosting providers and specialist tools such as Radware can help prevent bot traffic.	13

		Cloudflare, Blackhole Bad Bots This bot blocking needs to be paid particular attention to if your site has ads.	
7.2	Ensure that environments are only running when they are actively needed	The multiple environments that can make up a website (Dev, Test, UAT, Prod, DR) need to all be managed according to the habitats of those who use them. If non-production environments are only used in certain hours then shut them down at other times. Use a serverless environment; these tools will transparently spin servers up and down and allocate you resources on a per-request basis.	5
7.3	Ensure that environments have an appropriate dataset and content	It is important to have representative content and data across non-Prod environments to aid testing. However, aim for the bare minimum to reduce unnecessary use of resources.	3
7.4	Optimise the site architecture to minimise resource use to serve the people visiting the site	There are many different architecture approaches for minimising a site's carbon footprint. Peer reviewed and well accepted research is still lacking as to what the best approaches are but the summary is: reduce the amount of resources needed to serve the site to people. E.g. Microservices Static generation of content *Possibly include PWA*	8

7.5	Implement a caching strategy that minimises the amount and distance of data transfer	Things such as a CDN (with the aim of being as close to your users geographically as possible), static site delivery and event driven architecture could all be considered here.	13
7.6	Only collect data that is genuinely needed in order to improve the experience of the people using the site approach	Do not harvest data and decide what to do it with later. First define why you want to know something; how will knowing that help the people you serve better achieve their goals? Then define the bare minimum data you need to capture to deduce what you need to know Implement a data management strategy to reduce storage and workload: https://docs.aws.amazon.com/wellarchitected/latest/sustainability-pillar/data-patterns.html	2
7.7	Measure the carbon footprint of code in the development pipeline	In order to ensure sustainability is thought of as a non functional requirement in the same way as security, performance and accessibility, it must be actively measured and monitored during the development cycle, as per checking security with Snyk, or accessibility with Lighthouse. Digital sustainability is a very active market but hasn't yet reached a comparative level of maturity. There is no universally accepted calculation method and no tool is capable of measuring the multiple aspects of what makes up a website's carbon footprint. Tools worth mentioning at the moment are Greenframe.io, Green Metrics, Green Software	5

		Foundation Pipeline, SDIA DEF and EcoPing.	
7.8	Website should be hosted on a data centre that is running on 100% renewable energy	Remember, not all web hosts claiming to be 'green' are actually running off 100% renewable energy, a lot of claims are based on offsetting and carbon credits. Be sure to do your research and make sure your web hosts' claims are genuine. Some good providers to check out: - Positive Internet (https://positive-internet.com/) - Krystal (https://krystal.uk/)	13
	Recommendation from 10 Degrees		
	Http2 / 3		

Collaborators

Cyber-Duck Ltd

Screenmedia LTD

Wholegrain Digital

Unic AG

10 Degrees

EY

Sitegeist Web Services

Those That Dare

Potato

Aline

State University of New York
TPXimpact
MSQ