

Low carbon web General

Devices

- Design mobile-first

Design mobile-first to force yourself to design for the (usually) most-viewed and most energy-efficient screen size.
- Design for greener devices

Make sure that your solution works just as well or better on devices produced by more planet-friendly materials for long lives, repair and recycling like e.g. the Fairphone.
- Design for the oldies

Make sure your solution is compatible with older versions of devices to avoid forcing users to buy new hardware.

IA

- Help users find their way

Design a clear navigation to reduce pages loaded and time spent online by users. A search feature can also be helpful.
- Remember error pages

Inform the user if they end up in a dead end and guide them from there.
- Understand the flows

Help users quickly find what they need by mapping user journeys to designing better user flows.

Content

- Work user-centric

Conduct user research and test with users to understand their needs to decide what to include and, most importantly, exclude in your solution.
- Reduce pages and content

Reduce number of pages, page templates, and amount of content in general.
- Clean up

Delete outdated and unnecessary content and only upload new content if it is valuable to the user. Define a life cycle for deleting outdated content on a regular basis.
- Archive the old-school way

If you do need to store old content, use USB (offline) storing instead of storing in the "cloud".

Low carbon web Visuals

Images

- Avoid
● Reduce use of images or replace them with vector illustrations to reduce the page weight.
- Simplify motive
● Use simple imagery with shallow depth of field and a reduced and darker colour palette.
- Load lazy but don't hide
● Don't load an image before the user has scrolled to where it is and don't hide images in carousels.
- Compress and optimise
● Compress and optimise images and use dithering technique to reduce size but maintain quality.
- Convert to the best format
● Use WebP format for images with lots of shading and PNG for partial transparent images.
- Consider HTTP/2
● Use HTTP/2 technology.

Illustrations

- Use SVG format
● Use SVG format for anything vector-based.
- Simplify motive
● Minimise colours and details to reduce the weight of the file. Avoid special effects (brush strokes, textures etc.).
- Load lazy
● Don't load an illustration before the user has scrolled to where it is.
- Make them out of text
● Consider using visuals entirely made out of text characters (ASCII).

Colour

- Use dark shades
● Go for darker shades or even black to save a little bit of device power since black pixels are turned off on OLED displays (most smartphones and newer desktops).
- Increase contrast
● Ensure good contrast to make sure users don't increase device brightness (and energy usage) to be able to see the content.
- Use red, yellow and green hues
● On OLED screens, use colour hues closer to red on the visible spectrum since they are less bright and therefore consumes 25 % less energy than blue hues. But be aware that the impact of colours is quite small in the bigger picture.

Low carbon web

Visuals

Video

- Minimise use or replace with animations
As a rule of thumb, avoid videos. Consider removing the sound or replacing them with SVGs animated in code.
- Provide alternatives
Always provide the user with alternatives for watching a video.
- Only show upon request
Don't autoplay a video and make the users ask for it to load.
- Lower definition
Use half the resolution to reduce the weight by three times without the users noticing a difference.
- Avoid looping
Unless the video is small enough for the browser to cache it, don't loop videos.
- Use the big players
Use Vimeo or Youtube for webinars and longer videos to only load a few seconds ahead.
- Convert to WebM format
Use GIF or WebM format.

Animation

- Avoid
Avoid unnecessary animations to reduce page weight.
- Code SVGs
Animate SVG illustrations in code instead of uploading videos to reduce weight.
- Load lazy
Don't load an animation before the user has scrolled to where it is.
- Put user in control
Allow users to pause and start animations.

Font

- Convert to WOFF2 format
Minimise font file size by using modern web formats like WOFF2 that use higher compression methods than TTF and OFT.
- Reduce fonts and weights
Reduce number of fonts and font weights to reduce page weight.
- Choose system fonts
Avoid custom fonts. Use open-source fonts or even standard system fonts (Times New Roman, Arial etc.) instead.
- Consider a variable font
Consider using a variable font (adapting weight with code) if using more than 4 font weights.
- Reduce number of characters
Evaluate what languages your web solution needs to support and cut down to the 98 most common characters if possible.

Low carbon web Communication

Copy

- Optimise SEO and copy
Write clear efficient messages and optimise copy for search so users find what they need faster to reduce total time spent online and bounce rate.
- Maintain links
Continuously identify broken/ outdated links and fix them.
- Spread the word
Digital weight is invisible - make it visible. Talk about it and show it e.g. by having a low-carbon mode on your solution, or an explanation of what you have done to save energy on your solution e.g. in the footer. Consider how much users should be in control of the site's energy usage and when you want to educate or just decide for the users without informing.

Marketing

- Only send emails to those who want them
Make subscription to newsletters transparent and unsubscribing easy. Put a time limitation on newsletter sign-up and remind receivers to delete the email after reading it. Consider using services like [Ecosend](#) to automate newsletters more sustainably.
- Lighten campaigns
Be mindful about the impact of communication activities related to your solution. Design data light email and SoMe campaign templates and avoid using video and heavy image files.
- Reduce notifications
Use notifications with care, reduce amount of non-essential notifications and allow the user to control them.

Low carbon web Development

Programming

- Compress and optimise visuals
Compress and optimise all visuals, set media queries to use minimum width and load images at the correct scale.
- Code clean
Write efficient, clean, reusable and modularised code and remove bottlenecks and valueless code with e.g. tree shaking and native features.
- Select the best languages
Select languages and frameworks with low emissions and use the latest stable version of them.
- Optimise caching and rendering
Use browser caching to reduce http-requests and optimise performance. Render pages server-side on run time or build time.
- Ensure coherence
Enable tools to interpret your code by e.g. using HTML and metadata correctly and following the latest standards.

Third-parties

- Use light-weight
Avoid or choose light-weight options. Consider if heavy third-parties and tools bring enough value to be worth the size and uncontrollable nature of them.
- Reduce analytics
Don't collect data you won't use - not even cookies - and allow users to always decline tracking.
- Avoid plugins, widgets, ads...
Reduce amount of plugins, widgets and ads in general.
- Block the bots
Block the robots - especially the ill-intended ones - scanning through your site. But be mindful that you need search engines to be able to scan your solution for users to find it.

Low carbon web Artificial intelligence

AI training

- Use only necessary data
Use the smallest model needed for the desired output. Often there is no need for e.g. a chatbot to be able to answer everything.
- Choose non-peak hours
The carbon intensity can be three times more during peak hours compared to non-peak hours so choose your timing carefully.
- Use energy-conserving computational methods
Use as little computationally expensive approaches as possible to process the data.

AI use

- For meaningful purposes
Make sure there is a real need. Using AI just for creating amusing stories may not be the best use for such computation-heavy tools.
- Improve your prompting
The fewer queries used, the less energy spent. Furtherly, good prompt engineering provides more valuable results in a shorter time.
- Measure energy use
There are several online tools available like CodeCarbon, Green algorithms, and ML CO2 Impact, which can be included in your code at runtime to estimate your emissions.

Hosting

- Shorten distance
Host your solution in data centres close to the users. If the users are in different places, use content delivery networks (CDNs) with servers close to all users.
- Select green hosting
Host your solution in data centres and countries running on renewable energy. Don't use more server space than you need.
- Down time
Build a shutdown mechanism or remove non-critical elements at times where the energy is less green or traffic is limited.
- Kill zombies
If you have a server not used for anything shut it down. Also shut down e.g. test environments that are no longer needed.
- Use cloud and big-scale hosting
Always store on the cloud and not on your own personal server. Bigger providers like Google and Azure are most likely more optimised.

Low carbon web Accessibility

User Engagement

- Limit Failures and Extended Sessions

An accessible site may encourage users to stay longer, engage more, and accomplish tasks with less effort. This, in turn, can reduce the environmental impact associated with repeated visits or extended sessions.

Optimized Code

- Server Resources

Optimized code and resources can result in more efficient use of server resources and reduce the environmental impact associated with data storage, data transfer, and server processing.

- Speed and Efficiency

Improving the efficiency and speed of a website can indirectly contribute to reduced energy consumption. Users on accessible sites may experience faster loading times and more efficient interactions, which can lead to lower energy usage.

- Device Compatibility and Updates

The product or service should be designed to work well, also on older or less energy-efficient sites. Avoid requiring users to upgrade their devices frequently, indirectly contributing to a reduction in electronic waste and associated environmental impacts.

Society

- Economic and Social Inclusion

Web accessibility supports the economic and social inclusion of individuals with disabilities. When people with disabilities have equal access to online resources, they are better positioned to participate in society.