TI616I - Green IT

The Greenest Website
Car recommandation website

Noah Hemon - Thomas Marchal - Maxime Tang

Github repository: https://github.com/ThomasMrchl/switch2green
Please read the README.md file to set up the project.

Limited version deployed:
https://switch2green.vercel.app/components/index.html

Summary

Introduction

The objective of this project was to conceive "The Greenest Website" related to Green IT courses we had. We wanted to create a website where we would all be happy to work on, something connected to our shared interests while still aligning with the principles of Green IT. After some discussions, creating a website on cars and their carbon footprint was obvious as we all share a strong interest in vehicles. But we wanted to create a site whose purpose would be to help young people like us find their car, not to be only informational and that they would make even more searches.

To achieve this, we created a form, asking technical questions on people's requirements on their cars and based on their responses, an algorithm then recommends the vehicles that best match their criteria.

Even if the goal of the website was clearly defined, we had to think about the technologies used. Firstly, we wanted to code in the React framework but we understood that it was too energy-consuming. So we decided to stay simple by using HTML5, CSS3 and Javascript, light technologies that consume fewer resources.

For the backend, we wanted to use a huge database to better recommend people so our website suits user's needs well. So we used the JSON format of the following database : Link of the database.

The dataset says that :

" Since 2001, ADEME has been acquiring this data every year from the Union Technique de l'Automobile du motocycle et du Cycle UTAC (in charge of approving vehicles before they go on sale), in agreement with the French Ministry of Sustainable Development.

For each vehicle, the original data (transmitted by Utac) is as follows:

- fuel consumption
- carbon dioxide (CO2) emissions
- emissions of air pollutants (regulated under the Euro standard)
- all vehicle technical characteristics (range, make, model, CNIT number, type of energy, etc.)."

The database covers a wide range of car brands, including Mercedes, Volkswagen, Audi, BMW, and many others. It also features various vehicle types, such as hybrids and conventional vehicles, using different types of energy sources like diesel, gasoline, electricity-hydrogen (EH), gas-hydrogen (GH), and more.
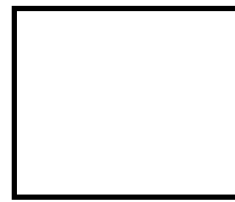
This was the first design of our idea:

Home page:

BIG TITLE

START THE FORM

FEW BLOCKS TO INTRODUCE THE WEBSITE

The form:

FORM

QUESTIONS 1

QUESTIONS 2

QUESTIONS 3

QUESTIONS 4

SUBMIT

Results page:

RESULTS

OPTIONS

NAME
YEAR
MODELE
HORSEPOWER
ECO SCORE

1

Division of work

At the beginning of the project, in class, we worked together to find a concept for the website. Then we designed mock-ups on a draft which served as a guide to ensure we were aligned and help us to go in the right direction.

During the project sessions in class, we began the first steps of implementing our website. Using Live Share on VSCode, we worked together in real time. We had the time to code together the HTML part. We didn't have time to finish CSS during the last project session.

And since two of us went on vacation and weren't sure we would have wifi connection, we divided the work to successfully finish the work.

Noah finalized the design to get the same as we had. Maxime started working on the ca recommandation functions. And we sent what we've done in a ZIP folder. After that, Thomas merged the work by finishing the functions and put the JSON file in the folder and put everything on Github.
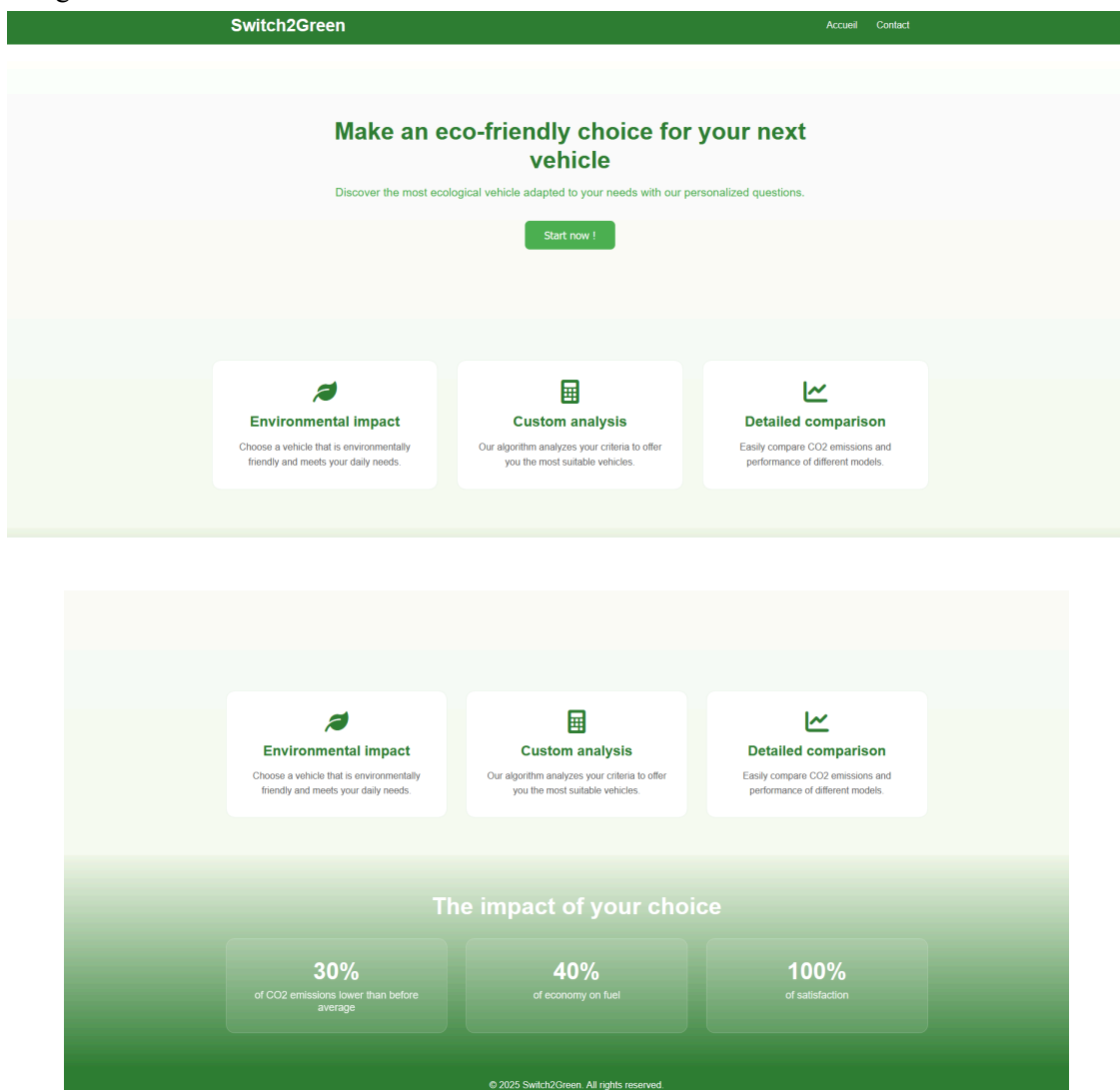
File organization

We have 6 different html files but only 4 pages.
header.html : appears as header on all pages
footer.html : appear as footer on all pages

index.html : This is the home page, it contains several informations on the website and the impact of choosing a vehicle.

form.html : It allows the user to answer four questions about the cars that they are looking for.

results.html : According to the answers of the form, it recommends the most suitable cars.

contact.html : This page shows the members of the team and allows user to write a message to us.

styles.css : We majoritarily used green and white colors to recall the environment's importance.

recommendation.js : It contains the functions to get answers of the form and to recommend vehicles

vehicules-commercialises.json : It contains all the database

For further information, please visit the github repository (link above)

Database:

For our database we use a dataset coming from the french government :
https://public.opendatasoft.com/explore/dataset/vehicules-commercialises/table/?flg=
fr-fr&sort=puissance_maximale&dataChart=eyJxdWVyaWVzIjpbeyJjb25maWciOnsiZ
GF0YXNldCI6InZlaGljdWxlcy1jb21tZXJjaWFsaXNlcyIsIm9wdGlvbnMiOnsiZmxnIjoiZ
nItZnIiLCJzb3J0IjoicHVpc3NhbmNlX21heGltYWxlIn19LCJjaGFydHMiOlt7ImFsaWdu
TW9udGgiOnRydWUsInR5cGUiOiJsaW5lIiwiZnVuYyI6IkFWRyIsInlBeGlzIjoicHVpc3
NhbmNlX2FkbWluaXN0cmF0aXZlIiwic2NpZW50aWZpY3BsYXkiOnRydWUsIm
NvbG9yIjoiI0ZGNTE1QSJ9XSwieEF4aXMiOiJhbm5lZSIsIm1heHBvaW50cyI6IiIsInR
pbWVzY2FsZSI6InllYSIsLCJzb3J0IjoiIn1dLCJkaXNwbGF5TGVnZW5kIjp0cnVlLCJhb
Glnbk1vbnRoIjp0cnVlfQ%3D%3D

This dataset is very complete and large and allows us to have really precise results for our recommendations. However as it is very large (200 mo+) , we were not able to use it inside our API as it would be expensive and not matching our objective of doing a Green website. We decided it would be easier for everybody to simply download it from the original website and place it to the right place for the moment. One of the future solutions could be to implement the API that they provide, but it would increase the complexity, length, and energy consumption of our code a lot.

Why is this website green ?

This website is pretty simple, offering only one functionality and achieving it well.

We could not implement a cache memory system because our recommendation algorithm needs to search through the entire database every time. Caching would risk omitting or overlooking certain data, which could lead to inaccurate or incomplete recommendations. Since every piece of information is important for matching users with the best vehicle, we had to prioritize complete and real-time searches over performance optimization.

We chose to develop the website using HTML5, CSS3, and JavaScript because they are lightweight technologies that consume fewer resources compared to more complex frameworks such as React.
Using these fundamental web technologies, we were able to create an efficient, accessible, and eco-friendly website while still delivering a smooth user experience.

In order to analyze our carbon footprint, we used several tools recommended, such as Website Carbon Calculator, which analyzes our website globally and compares it with the rest of the world's performances!

We were quite happy and not so surprised that our website was better than 96% of the whole internet with an A+ grade. (url of our test:
https://www.websitecarbon.com/website/switch2green-vercel-app-components-index-html/)
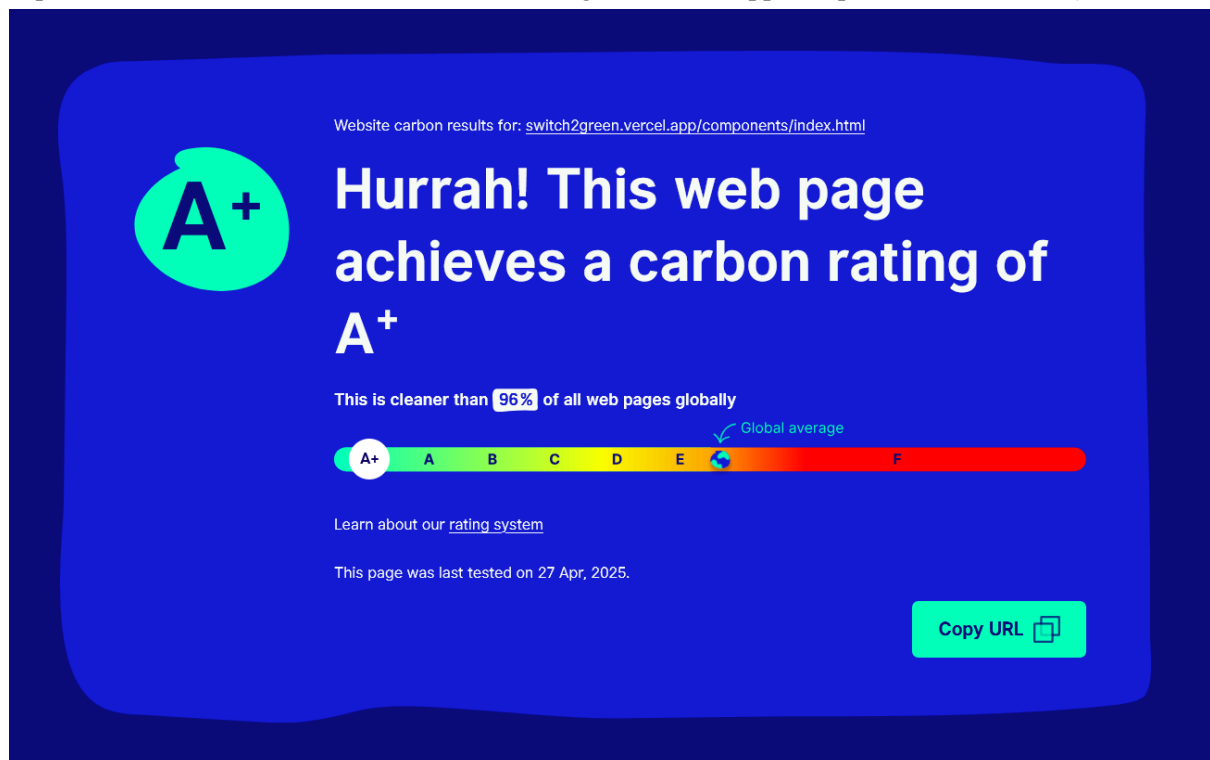
This website was quite nice but did not provide detailed information about what we had to improve to match the green IT standart so we used the Greet IT Analysis extension to provide several

informations in a more technical way.

This helps us a lot during the development phase to consider which path to use and not to use in order to maximize our score on those tests.

One of the most carbon consuming emissions is from our database which is in the case of your website downloaded in local as it is a large dataset and was very expensive and not really ecological to upload it only for our project. It is stored in a local JSON and limits the HTTP requests to the API from the website which was one of the major energy consuming thing possible.

The choice of using only basic HTML, CSS and JS was a good benefit for our results. At the beginning we thought we would use React or NextJS to build a nice looking modern website with all the frameworks we now but it was so massive in term of performance loss and energy consuming that we decided to avoid those frameworks quite quickly.

We also did not import any librairies ! They are often overkill for this type of small project and import a lot of useless code that we would not have to use. So we are back with the basics!

One last score was from the EcoPing website and it was quite nice to see a score of 99!!!

Your Sites

A list of the sites you've added on EcoPing sorted by EcoScore.

Site Name ⌄	EcoScore 🟢 ⌃	CO2 (g) ⌄	Transfer size ⌄	Load time (s) ⌄	Actions	Status
🟦 🟥 Switch2Green	99	-	20.4 KiB	4.09	⧉ ✏ ⌄	✅

Previous page   25 ⌄   Next page

Conclusion

In conclusion, this project allowed us to combine our passion for vehicles and Green IT. Using HTML5, CSS3 and Javascript which are lightweight and energy-efficient technologies, we created a website that is helpful and eco-friendly.

It helps people to make more sustainable choices by recommending cars based on their criteria and carbon footprint. This project doesn't only show the importance of consumption but also raises awareness about environmental impact. It also highlighted the fact that even if eclectic vehicles are not suitable options for our budget or our needs we may always find a solution that can match the planet's requirements.

With this project, we rethink our way to implement algorithms, without using the easiest framework or technologies but just using more ecological technologies. We learned to pay attention to the performance and the ecological impact of the thing we made and we learned to measure and fix usual mistakes.