

# Documentation

Outil de mesure de l'impact carbone du numérique

Etudiant

Alexis Mérienne

**Encadrant**

Guillaume Urvoy-  
Keller

## Sommaire

- i. Outils existants**
- ii. Maquette du site**
- iii. Prospection technologique**
- iv. Méthode de travail**
- v. Structure de l'application**
- vi. Fonctionnement de l'outil de mesure de l'impact carbone du numérique**
- vii. Adaptabilité mobile**
- viii. Mesure qualités**
- ix. Axes d'améliorations**

## i. Outils existants

Des outils de calculs de l'impact carbone orientés sur des scénarios établis par un utilisateur sont accessibles sur internet.

### a. Nos Gestes Climats [6]

Notamment le service public *Datagir* a développé plusieurs simulateurs d'émission carbone pour différents domaines. Le simulateur le plus complet s'appelle *Nos Gestes Climats* et englobent l'ensemble des usages émetteur de GES d'un citoyen. La partie numérique ne rentre cependant pas autant dans les détails que l'outil que nous avons développé.

### b. CALCUL DE MON IMPACT ENVIRONNEMENTAL PRO [7]

L'INR (institut du numérique responsable) a développé un outil qui permet de calculer son empreinte carbone du numérique. Cet outil calcule plus finement et selon des scénarios plus précis que *Nos Gestes Climats* les émissions en GES d'une utilisation du numérique. Cependant, cet outil se cantonne à une utilisation professionnelle et propose peu d'explications sur la nature des émissions carbone.

## ii. Maquette du site

Dans un premier temps, nous avons divisé la structure du site en 3 pages web : **Appareils, Actions et Bilans**. Ces trois pages reprennent l'architecture développée pour le prototype et permet d'être fidèle à l'approche abordée dans la documentation. En effet, les rapports présents dans notre documentation différencient la phase de production liée aux **appareils** à la phase d'usage liée aux **applications numériques**.

Pour les **Appareils** et **Actions**, la structure de la page est similaire. En effet, sur ces deux pages, l'utilisateur renseigne les éléments de son utilisation du numérique.

Ainsi, sur la page Appareils, il devra :

- Ajouter des appareils en fonction du type
- Supprimer des appareils
- Renseigner le temps d'utilisation de l'appareil sur une journée
- Renseigner l'âge de l'appareil

Sur la page Action, il devra :

- Ajouter des applications numériques
- Supprimer des applications numériques
- Renseigner l'unité d'utilisation d'une application numérique sur une journée.

Header	Header
<p><b>Appareil 1</b> J'utilise appareil x heures dans la journée</p> <p>Je possède cet appareil depuis x années</p> <p><b>Supprimer</b></p>	<p><b>Action 1</b> Je réalise l'action 1 pendant x heures dans la journée</p> <p><b>Supprimer</b></p>
<p><b>Appareil 2</b> J'utilise appareil x heures dans la journée</p> <p>Je possède cet appareil depuis x années</p> <p><b>Supprimer</b></p> <p>+</p>	<p><b>Action 2</b> Je réalise l'action 2 pendant x heures dans la journée</p> <p><b>Supprimer</b></p> <p>+</p>

La page **Bilan** présente la partie pédagogique de notre outil. Donc, en plus de présenter les résultats du bilan carbone, cette page doit offrir des explications sur la provenances des émissions carbone, expliquer comment les calculs ont été faits et présenter des solutions pour réduire ces émissions carbone.

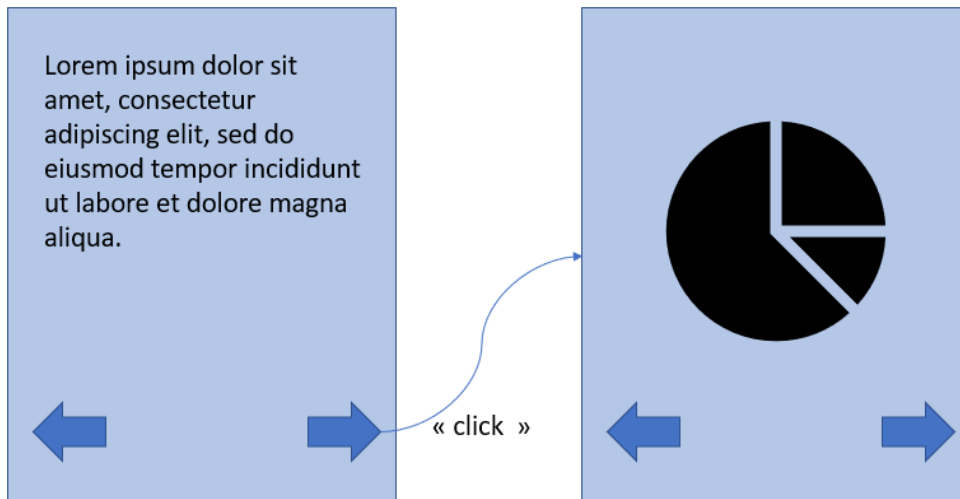
Cette partie pédagogie se déroule selon un narratif dont chaque figure représente une étape du cycle de vie du numérique.

Ce narratif se déroule ainsi :

- émissions en GES de la consommation électrique des appareils
- émissions en GES des actions numérique
- émissions en GES de la production des appareils
- rapport entre la phase d'utilisation et la phase de production

Les résultats des émissions en GES sont présentés sous la forme de graphiques circulaires, puisque l'information que nous voulons communiquer à l'utilisateur est les rapports entre les différentes émissions carbone. Les graphiques circulaires permettent de visualiser au mieux ces rapports.

Chaque graphique est précédé par un texte expliquant d'où viennent les émissions carbone.



### iii. Prospection technologique

#### a. Les critères

L'outil doit se présenter sous la forme d'une interface web. En effet, il doit facilement être accessible aux public concerné. La base de donnée carbone établis lors de la lecture de la documentation est de petite taille, il n'est donc pas nécessaire de créer un service dédiée à celle-ci. Les données seront alors accessibles directement par l'application via un fichier json.

De plus, dans un besoin de limiter les ressources utilisée lors de la conception du site, il est préférable de concentrer la partie *client* et la partie *métier* de l'outil sur une même service. En d'autres termes, l'outil ne présentera pas de *back-end*. Les calculs d'émission carbones seront fait via le site web. Nous pouvons nous autoriser à choisir cette stratégie car les calculs ne sont pas d'une grande complexité et ne nécessite pas d'appel à une grande quantité de données.

Enfin, pour avoir une interface claire et intuitive, nous avons divisé les pages de l'outil en plusieurs composants. Chaque composant répond à une fonction et est disposé sur la page selon des relations hiérarchique avec les autres composants.

#### b. Les solutions

Nous nous sommes naturellement dirigés vers des *framework* web pour déterminer la technologie que nous avons utilisé. En effet, les *framework* permettent de développer des interfaces web en montant rapidement en complexité et en fonctionnalité.

Les trois *frameworks* les plus cités sont :

- React.js

- Angular.js
- Vue.js

Ces trois *frameworks* sont orientés composant, donc répondent aux critères établis ci-dessus.

Vue.js est le *framework* que je maîtrise le mieux étant donné que j'ai déjà développé plusieurs projets avec cette technologie. Choisir Vue.js m'a donc permis de développer plus rapidement l'outil et d'avoir une version fonctionnelle dès les premiers jours, sur laquelle se reposer pour y apporter des améliorations.

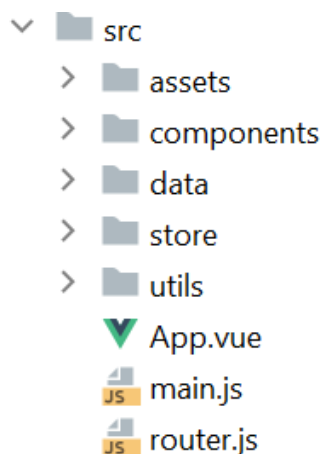
#### iv. Méthode de travail

Après avoir conçu un MVP (minimum valuable product), nous avons instaurer une méthode de travail se rapprochant de la méthode scrum. C'est-à-dire en planifiant des réunions hebdomadaires avec Mr Urvoy-Keller pour faire le point sur l'avancé de l'outil et ensuite déterminer les fonctionnalités à ajouter.

De plus, nous demandions régulièrement à des personnes extérieures au projet de tester le site pour vérifier que l'interface restait intuitive et que les données représentées étaient pertinentes.

#### v. Structure de l'application

Le code de l'application est structuré selon les conventions de Vue.js. Les fichiers sont donc organisés dans le dossier *source* de la sorte :



**assets** : contient l'ensemble des fichiers liés au style de l'application. (les logos, les icones, les images)

**components** : contient les composants présents dans l'application.

**data** : contient la base de donnée carbone sur laquelle s'appuient les calculs d'émissions de GES.

**store** : le store permet de gérer les échanges de données au sein de l'application.

**utils** : regroupe des fonctions utiles au fonctionnement de l'application. De plus, contient le fichier *computedges.js* qui calcule les émissions en GES du numérique en fonction du scénario établis par l'utilisateur.

**main.js** : entrée de l'application

**App.vue** : Composant à la racine de l'application.

**router.js** : fichier que gère les différentes routes de l'application.

**Le Store** : Le store est l'implémentation du gestionnaire d'état de la librairie *VueX* [8]. Il nous permet d'avoir une interface centralisée dans l'application par laquelle on peut facilement récupérer les données depuis l'ensemble de nos composants.

## vi. Fonctionnement de l'outil de mesure de l'impact carbone du numérique

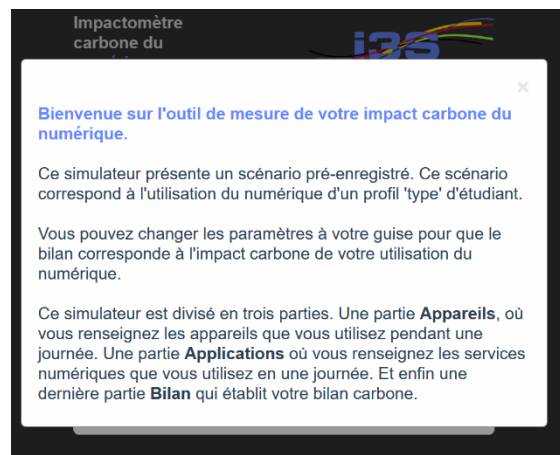
Pour présenter la conception de l'outil, donc présenter les relations entre les différents composants, les communications de données et les méthodes de calculs des émissions de GES, nous allons nous reposer sur un scénario utilisateur.

Le scénario se déroule de la sorte :

1. *L'utilisateur arrive sur le site de l'impactomètre.*
2. *L'utilisateur renseigne les appareils qu'il utilise dans sa journée.*
3. *L'utilisateur renseigne les applications numériques qu'il effectue dans sa journée.*
4. *L'utilisateur consulte son bilan carbone*
5. *L'utilisateur s'informe sur les solutions*

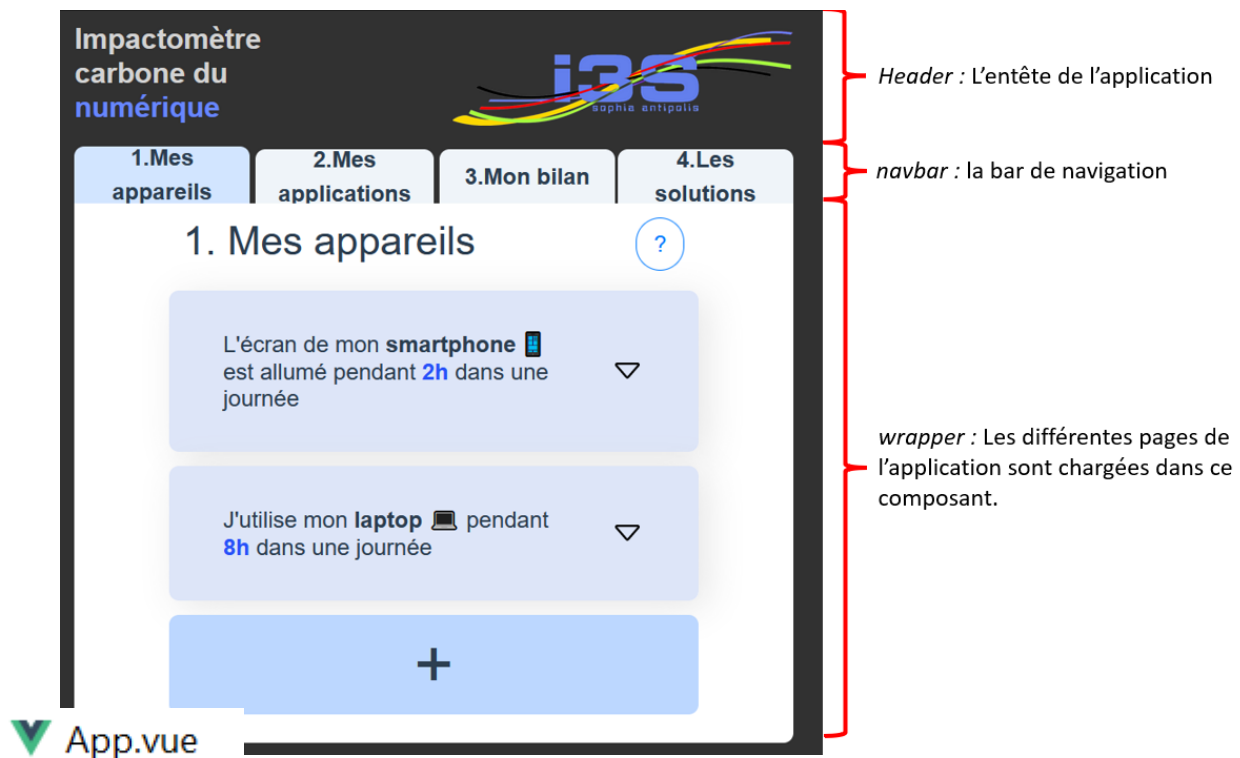
1. *L'utilisateur arrive sur le site de l'impactomètre.*

La page qui s'affiche à l'arrivée est la page *home*. Une popup s'affiche à l'écran qui explique le fonctionnement de l'outil.



Après avoir fermé la fenêtre popup, l'utilisateur peut accéder aux fonctionnalités de l'outil.

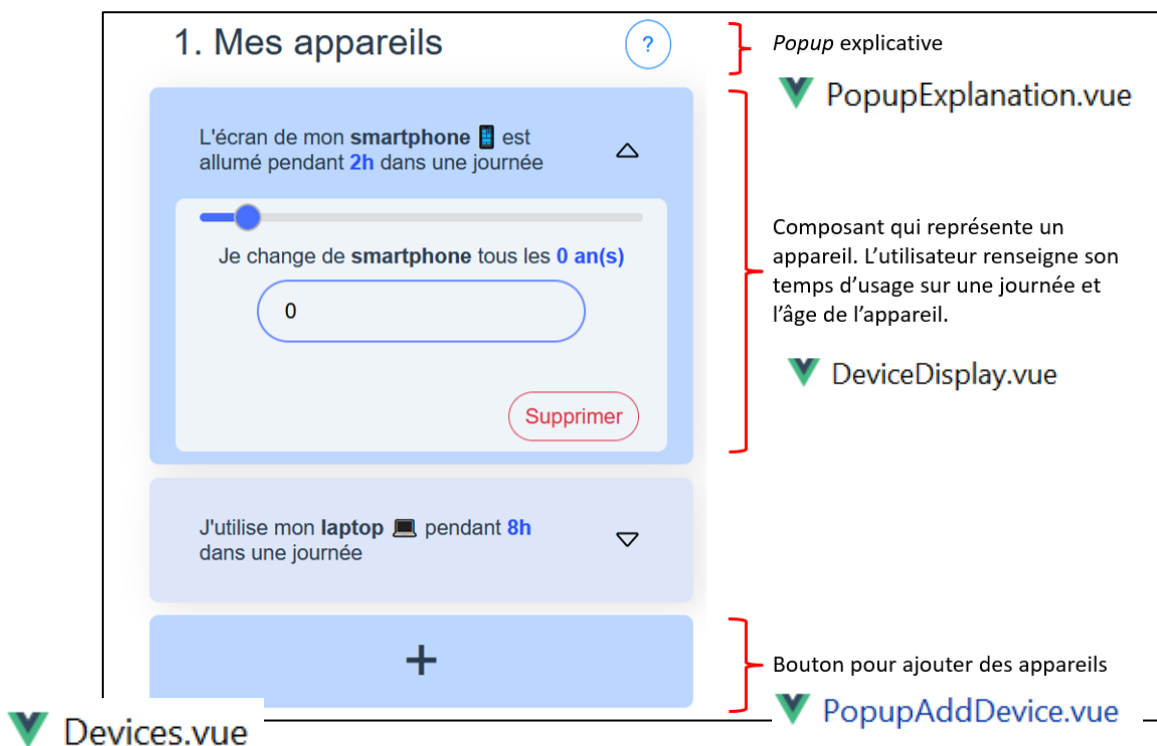
La structure de la page est expliquée ci-dessous.



2. L'utilisateur renseigne les appareils qu'il utilise dans sa journée.

Cette action s'effectue sur la page *home*, qui est la première page qui s'affiche quand on arrive sur le site.

La page est composée des éléments suivants :





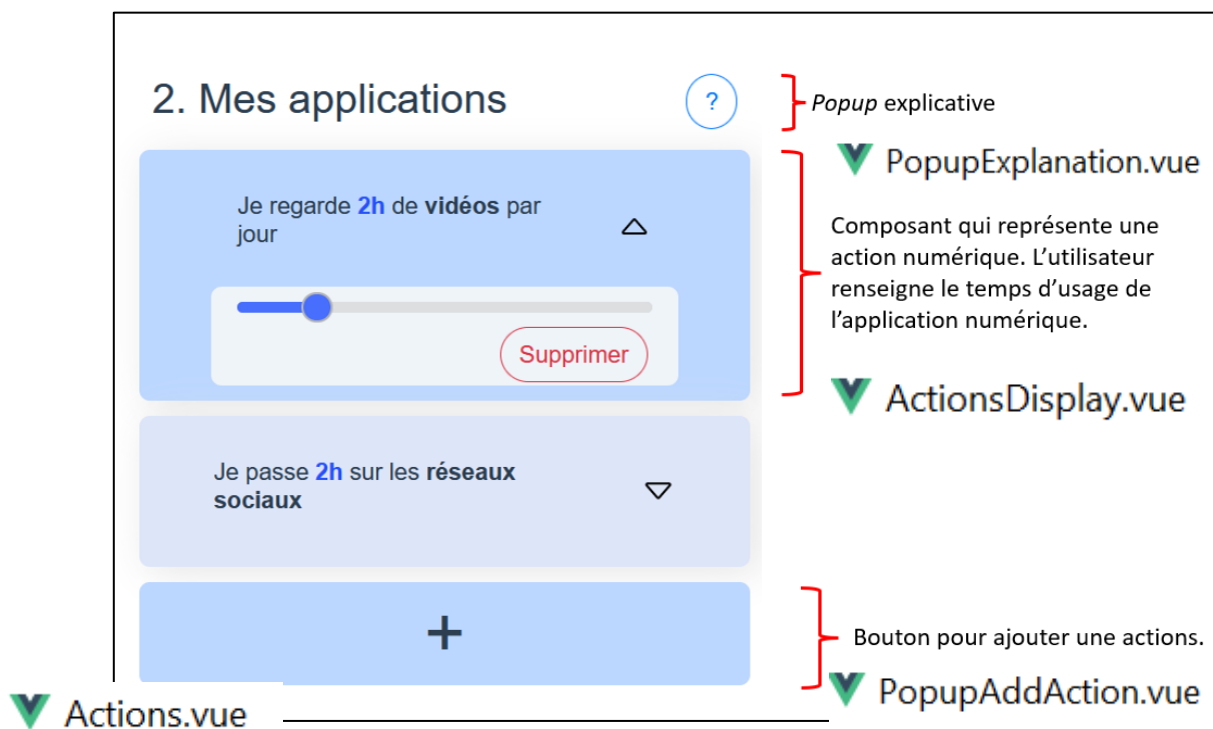
Lorsque l'utilisateur entre les données dans les champs dédiées, *DeviceDisplay.vue* communique avec le store les données.

```
this.$store.commit( type: 'SET_VALUE_DEVICE', this.device)
```

Ensuite, le store s'occupe d'enregistrer ces données dans le fichier *scenario.json*.

### 3. L'utilisateur renseigne les applications numériques qu'il effectue dans sa journée.

Cette action se déroule sur la page *actions*, que l'on accède depuis l'onglet *Mes applications*. La page est composée des éléments suivants :



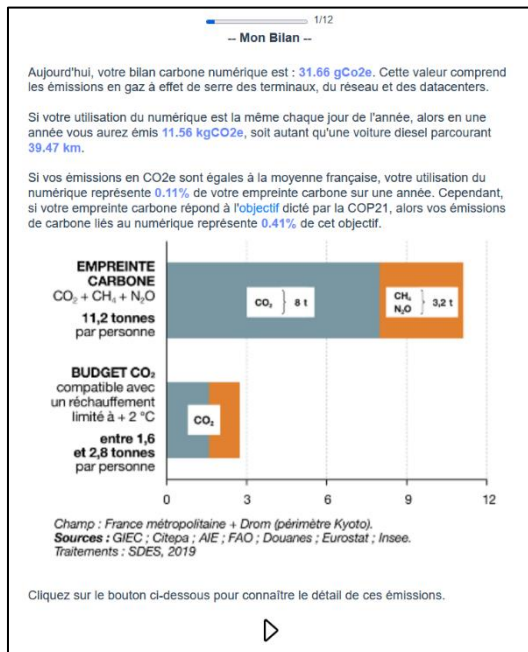
De la même manière que pour les appareils, lorsque l'utilisateur rentre les données dans les champs dédiées, *ActionsDisplay.vue* communique les données avec le store.

```
this.$store.commit( type: 'SET_VALUE_ACTION', this.action)
```

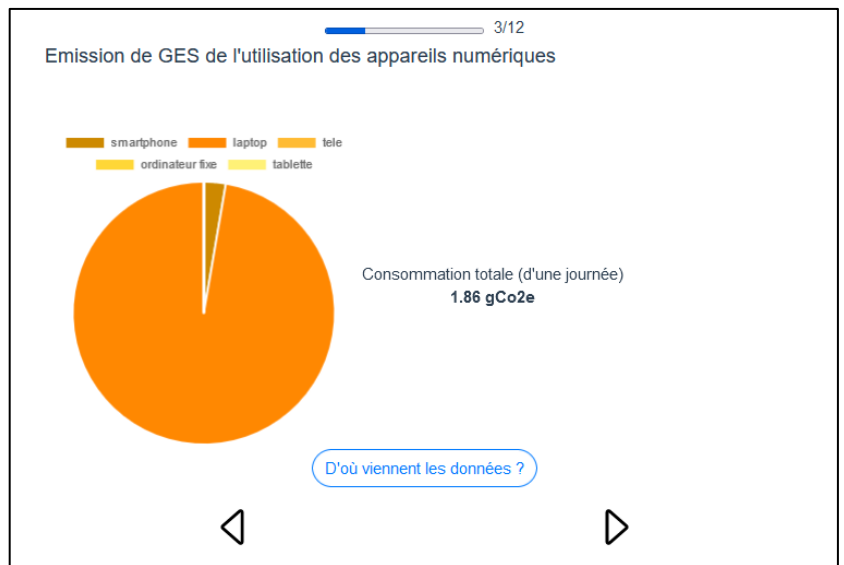
Ensuite, le store s'occupe de mettre à jour le fichier *scenario.json*.

#### 4. L'utilisateur consulte son bilan carbone.

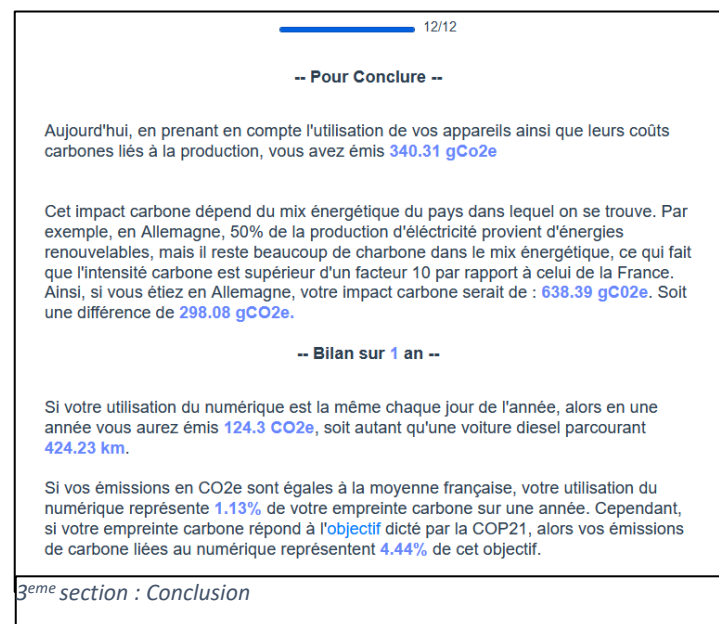
L'utilisateur peut consulter son bilan carbone via la page *Bilan*. Cette page est divisée en trois sections. La première est un résumé du bilan carbone, avec une comparaison à l'année et des valeurs en équivalent de carbone émis par une voiture diesel. La seconde section explique les détails des émissions carbone en fonction des différentes infrastructures sollicités lorsqu'on utilise le numérique. Cette section est composée de 5 figures, qui sont respectivement : Les émissions en GES de l'utilisation des appareils, les émissions en GES des actions numériques, les émissions en GES de la production des appareils, la comparaison entre les émissions en GES de la phase d'utilisation avec la phase de production et cette même comparaison en faisant l'hypothèse d'une utilisation « sobre » du numérique. Enfin, la dernière section est une conclusion, qui affiche la valeur d'émission carbone, une comparaison avec un cas où l'utilisateur consommerait de l'électricité allemande et une comparaison avec les objectifs dictés par la COP21.



1<sup>er</sup> section : Introduction du bilan



2<sup>ème</sup> section : Les détails des émissions.



C'est dans cette partie que l'application calcule les émissions en GES du numérique de l'utilisateur via les fonctions javascript présentes dans le fichier *computedges.js*.

Ces fonctions sont :

```
function getGESAction(data, scenario){  
  
function getGESDevice(data, scenario){  
  
function getGESProduction(data, scenario){  
  
function getGESProdUtilisation(data, scenario){  
  
function getGESProdUtilisationDoubleDureeDeVie(data, scenario){
```

Elles prennent en paramètre le fichier *data.json* qui fait office de base de donnée carbone et le fichier *scenario.json* qui est le fichier dans lequel se trouve les données rentrées par l'utilisateur.

Chaque calcul est expliqué dans la popup « D'où viennent les données », et les valeurs utilisés pour ces calculs sont sourcés dans cette même popup.

#### 5. L'utilisateur s'informe sur les solutions de sobriété.

Les solutions que l'on propose pour diminuer notre impact carbone lié au numérique sont regroupées sur la page *solution*, accessible via l'onglet *Mes solutions*. Cette page est une simple page html est ne présente aucun élément dynamique.

-- Les solutions --

L'action de **sobriété énergétique** ayant le plus d'influence sur la baisse des émissions carbone du numérique est d'**augmenter la durée de vie de nos appareils**.

Cependant, nous devons également réduire nos consommations de GES de la phase d'utilisation, puisqu'en réalité, pour le numérique **nous ne pouvons pas dissocier l'utilisation de la production**.

En effet, lorsque l'on regarde une vidéo en streaming, les données consommées proviennent de centres de données et sont transportées dans un réseau. Ces infrastructures sont coûteuses en énergie lors de leur fabrication et de leur maintenance.

De plus, une **utilisation accrue du numérique** implique un développement croissant d'infrastructures.

Voici quelques conseils pour diminuer votre impact carbone lors de l'utilisation :

- Limiter le visionnage et l'écoute en streaming
- Diminuer la résolution des vidéos pour le visionnage
- Stopper la pub sur les sites web
- Télécharger avec modération
- Utiliser le wifi plutôt que le réseau mobile

#### vii. Adaptabilité mobile.

Etant donné que l'outil a été développé à des fins pédagogiques, il était donc nécessaire de le rendre accessible au plus grand nombre et notamment aux utilisateurs de smartphone. Ainsi, une version de l'interface a été développée spécialement pour les mobiles.

L'enjeu était de rendre l'interface plus simple, c'est-à-dire avec moins d'éléments visuels étant donné qu'une dimension mobile nous offre moins de place pour afficher les informations. De plus, les éléments dynamiques de l'application devaient être développés de telle sorte qu'ils soient utilisables avec un écran tactile.

Ainsi, les choix ont été :

- D'ajouter un menu déroulant pour accéder aux pages *Mes Appareils* et *Mes actions*.
- De simplifier les interfaces de saisies pour les appareils et les actions.
- De supprimer l'élément *Résumé* de la page *Mon Bilan*.

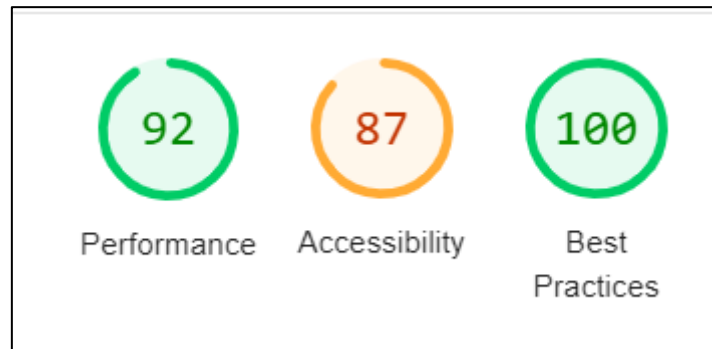
#### viii. Outils qualités.

Lorsqu'on développe une application web, il existe des outils permettant de mesurer la qualité de notre site web. Les métriques qui nous intéressent sont :

- La performance
- L'accessibilité
- Les bonnes pratiques

L'outil que nous avons utilisé pour mesurer ses métriques est Lighthouse[9], qui est une extension chrome se greffant à l'outil de développement web du navigateur.

Selon Lighthouse, nous avons les résultats suivants :



Ces résultats sont satisfaisants. La valeur de l'accessibilité s'explique par le fait que certains éléments ne présentent pas assez de contrastes par rapport au second plan.

Étant donné que notre outil vise à informer sur la question de l'impact carbone du numérique, il est nécessaire de se soucier que ce même outil ne présente pas de consommation anormale. Green-it a développé un plug-in permettant de mesurer la consommation de site web [9] et d'attribuer un ecodindex allant de A à F. Nous avons donc décidé d'utiliser ce plug-in pour mesurer la consommation de notre application.

Notre application affiche le résultat suivant :

EcoIndex <span>A</span>		
EcoIndex	Eau (cl)	GES (gCO2e)
87.43	1.88	1.25
Nombre de requêtes	Taille de la page (Ko)	Taille du DOM
0	0	161

Ce bon résultat provient en partie du fait que Vue.js est un framework qui optimise la taille de ces fichiers html et présente au navigateur des pages légères.

## ix. Axes d'améliorations

En termes de développement informatique, il y a des bonnes pratiques que je n'ai pas totalement acquies. Par exemple, sur le fait de nommer les fichiers, les variables et les fonctions. A ce jour, le code de l'application est un mélange de français et d'anglais, ce qui n'est pas optimale d'un point de vu maintenabilité.

De plus, nous aurions pu intégrer des expériences utilisateurs plus tôt dans le processus de développement, cela nous aurait permis d'éviter de persévérer dans des fonctionnalités qui n'étaient pas pertinentes. Par exemple, initialement, dans la partie *Bilan*, nous commençons par afficher les détails des émissions de GES par infrastructures. Cependant, les utilisateurs s'attendaient à connaître directement leurs émissions carbone.

## x. Plan de développement opérationnels.

Dans un premier temps, lors de la phase de développement de l'outil, le site était disponible sur le service *Firebase* qui est un hébergeur cloud appartenant à Google. Ce service à l'avantage d'être gratuit et de proposer une API simple pour déployer son site rapidement.

Cependant, cette solution n'est pas pérenne, étant donné que cet hébergement dépendait de mon compte google personnel. De plus, le code a été produit dans un laboratoire de recherche publique, il n'est donc pas souhaitable que le site soit disponible via une service externe privé.

La solution choisie a donc été d'héberger l'outil sur le serveur web d'i3S. Il est donc disponible sur un serveur apache présent au sein du laboratoire. Cette solution a rencontré quelques problèmes liés aux redirections de routes et à la réécriture de l'URL que nous avons pu résoudre en modifiant le code et en configurant avec de bons paramètres le serveur apache.

## xi. Open source

Le code de l'application est disponible en Open Source sur le dépôt Github suivant :

<https://github.com/AlexisMerienne/calculateur-ges-num>

Le code est publié sous une licence MIT.