

## Chapitre 3

# La conception du site

### 1. Introduction

La conception du site est une étape cruciale pour créer un service numérique avec une faible empreinte environnementale. Les plus gros leviers de gain green IT sont accessibles durant cette étape. Par la suite, il sera possible d'effectuer des améliorations, mais elles auront des impacts beaucoup plus faibles.

La conception d'un service numérique peut s'appuyer sur des solutions utilisant peu de technologies ou des technologies anciennes et avec un faible impact écologique. Ces solutions sont dites *low-tech* (par opposition à *high-tech*). Par exemple, ce peut être un service numérique consultable même sur un matériel ancien ou avec un réseau de faible débit...

Il est également possible de recourir à des solutions n'utilisant pas du tout de technologie : c'est la *no-tech*. Par exemple, l'utilisation de chiens pour détecter précocement, à l'odeur, certains cancers donne de meilleurs résultats qu'une intelligence artificielle.

Bien souvent, les meilleures solutions allient plusieurs niveaux technologiques parmi les trois (no-tech, low-tech et high-tech). Il existe au Rwanda et au Ghana des drones assurant des livraisons de sang, de médicaments et de vaccins à des communautés éloignées. Les commandes sont passées par SMS, ce qui est une bonne solution puisque ces pays n'ont pas une couverture suffisante pour utiliser Internet. La solution alliant high-tech (drone) et low-tech (SMS) est très écologique puisqu'elle a une empreinte environnementale bien moindre que l'acheminement par voie terrestre et elle permet de sauver beaucoup de vies grâce à une bien meilleure rapidité de livraison.

Lorsque le service numérique est repensé de fond en comble pour trouver une solution novatrice et à fort impact écologique, cela se nomme une action de facteur 4, par opposition à une action ayant un impact plus faible dite de facteur 1.

Pour illustrer cela de manière imagée, prenons l'exemple d'un déplacement avec un véhicule. La voiture thermique correspond à une solution qui n'a pas été écoconçue. La voiture électrique est une solution de facteur 1 : elle ne remet pas en question l'utilisation de la voiture, mais elle cherche à optimiser le déplacement écologiquement parlant. Enfin, l'utilisation d'un vélo est une solution de facteur 4 : l'impact est réduit drastiquement.

Dans nos choix, il faut se méfier des reports. Cela peut se produire lorsqu'une solution réduit l'empreinte environnementale directe, mais augmente l'empreinte indirecte. C'est par exemple le cas d'un véhicule électrique. À l'usage, son empreinte est bien moindre par rapport à un véhicule thermique, mais il faut créer et recycler la batterie du véhicule, ce qui est plus polluant que la création et le recyclage d'un réservoir de carburant.

Enfin, une solution peut être globalement meilleure, mais si son utilisation est décuplée, au final elle aura une empreinte écologique plus importante. C'est par exemple ce qui s'est passé avec la mise en place de la 5G en Corée du Sud.

## 2. Les fonctionnalités utiles

Dans un article de son blog (<https://lebondigital.com>), Alizée Colin écrit : "Un site est fini lorsqu'il n'y a plus rien à enlever – et non à ajouter." Cette phrase surprenante résume très bien la qualité première attendue d'un site internet écoconçu : la frugalité. Nous avons tous tendance à vouloir enrichir nos sites internet de nouvelles fonctionnalités, de nouvelles pages. Le terme "enrichir" laisse entendre que le site s'améliore, mais en fait c'est comme les aliments trop riches : ils font grossir, avec à la clé une surcharge pondérale, voire une obésité ! Les logiciels et en particulier les sites internet en sont sujets également : on parle d'ailleurs d'obésiciel et de gras numérique ! Il est donc nécessaire de faire subir une cure d'amincissement à nos sites existants et de faire attention au régime alimentaire de nos futures créations.

Près de la moitié (45 %) des fonctionnalités d'une application ne sont jamais ou sont très rarement utilisées. Il est donc stratégiquement plus intéressant de se concentrer sur les fonctionnalités vraiment utiles. Moins de lignes de code à produire, c'est plus de temps à consacrer à la qualité du code, à l'accessibilité, à la sécurité, au respect des règles du green IT...

Cette bonne pratique est difficile à mettre en œuvre, car il faut convaincre ses clients, sa hiérarchie et ses collègues et faire preuve de beaucoup de pédagogie ! Pourtant, sur le papier, c'est ce qu'il y a de plus simple à réaliser puisque, justement, il y a moins de choses à créer. Et, sans l'ombre d'un doute, c'est cette bonne pratique qui a le plus gros impact sur l'empreinte numérique du site internet ! Comme quoi, la paresse est une bien belle qualité !

Pour chaque page internet ou fonctionnalité, il est important de se poser la question de sa réelle utilité, et une fois en ligne ou en service, de consulter les statistiques d'utilisation pour vérifier sa pertinence.

### Remarque

*Il est préférable d'adopter une approche "mobile first", c'est-à-dire de concevoir le site en premier lieu pour les smartphones avant d'envisager les plus grands écrans. La faible taille de l'écran nous incite à nous concentrer sur l'essentiel.*

## 3. La structuration du site

Une fois le contenu réellement utile trouvé, il faut se poser la question de l'organisation de celui-ci dans les différentes pages du site. Quelles pages faut-il créer ? Comment naviguer de l'une à l'autre ? Cette section répond à ces questions.

### 3.1 La profondeur du site

La profondeur d'un site internet est le nombre de clics nécessaires pour accéder à l'une de ses pages depuis la page d'accueil. Par exemple, sur un site d'e-commerce, il peut être nécessaire d'effectuer un premier clic sur une catégorie, puis un second sur un article pour pouvoir visualiser la page détaillant les caractéristiques de ce produit. La profondeur est alors de 2.

Il faut veiller à ce que cette profondeur soit la plus faible possible. Cela est bon à tous points de vue. La personne rencontrant des difficultés à naviguer sur Internet en raison d'une déficience sera heureuse d'accéder rapidement et simplement à l'information souhaitée. Ensuite, le nombre de pages demandées et de ressources téléchargées sera faible, donc bon d'un point de vue green IT. Enfin, cela améliorera le référencement naturel. Une profondeur de 1 ou 2 est souhaitable pour un petit site et cela peut aller jusqu'à 4 pour un gros site.

### 3.2 L'historique de navigation

Pour ne pas dérouter les internautes, il est important de respecter l'historique de navigation. C'est-à-dire que lorsque l'internaute demande à son navigateur de revenir à la page précédente, il doit retrouver la page précédemment consultée. Dans la plupart des cas, cela se fait naturellement. Mais pas sur un site web monopage (*SPA - Single Page Application*) : l'internaute reste toujours sur une seule et même page, le contenu évolue avec des requêtes asynchrones (AJAX) en fonction des actions du visiteur. Il est alors nécessaire de manipuler manuellement l'historique de navigation. En JavaScript, cela peut être réalisé grâce à la fonction `history.pushState()`. Il faut également écouter l'évènement `window.onpopstate` (déclenché lors d'un retour en arrière dans l'historique) pour restituer la page attendue.

Il est également nécessaire de modifier la valeur de la balise `<title>` afin d'aider l'utilisateur à se repérer dans sa navigation. Cela se fait avec la propriété `document.title` en JavaScript.

### 3.3 La personnalisation des pages d'erreur

Il est très classique de personnaliser la page d'erreur 404 retournée quand la ressource demandée n'existe pas. Néanmoins, il est important que cette page soit la plus simple et légère possible. Le seul message à indiquer est que l'internaute demande une page qui n'existe pas ! Il en est de même pour les autres pages d'erreur. Même s'il est tentant de faire une page web dynamique pour mutualiser le code, il est préférable de faire des pages web statiques pour alléger la charge du serveur.

Voici un exemple avec un serveur Apache :

Fichier `.htaccess` à la racine du site :

```
ErrorDocument 403 /erreur403.html
ErrorDocument 404 /erreur404.html
ErrorDocument 500 /erreur500.html
```

Fichier `erreur403.html` à la racine du site :

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="UTF-8">
    <meta name="robots" content="noindex">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Accès interdit</title>
  </head>
  <body>
    <h1>Vous n'êtes pas autorisé à accéder à cette page !</h1>
    <a href="/">Retour à la page d'accueil</a>
  </body>
</html>
```

Fichier erreur404.html à la racine du site :

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="UTF-8">
    <meta name="robots" content="noindex">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Page non trouvée</title>
  </head>
  <body>
    <h1>La page demandée n'existe pas !</h1>
    <a href="/">Retour à la page d'accueil</a>
  </body>
</html>
```

Fichier erreur500.html à la racine du site :

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="UTF-8">
    <meta name="robots" content="noindex">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Problème sur le serveur</title>
  </head>
  <body>
    <h1>Notre site web a rencontré une erreur !</h1>
    <a href="/">Retour à la page d'accueil</a>
  </body>
</html>
```