

Mobile first

Less is more!

Objectif : réalisation d'un site statique en mobile first

Le responsive web design est une méthode de conception web qui permet au web de s'adapter automatiquement aux écrans de différents appareils, en affichant le contenu de manière à ce que les gens se sentent à l'aise.

Cela réduit considérablement les opérations des utilisateurs telles que les zooms et les défilements lors de la navigation sur le web.

Les stats d'utilisation mobile vs desktop

<https://gs.statcounter.com/platform-market-share/desktop-mobile-tablet>



L'avancement progressif

L'avancement progressif implique de construire d'abord une version pour le device le plus petit (mobile). Celle-ci comprend les fonctions et caractéristiques les plus basiques.

Ensuite, on passe à une version pour tablette puis pour desktop.

On ajoute des interactions, des effets plus compliqués, du contenu, etc.

La dégradation élégante

La dégradation élégante au contraire commence par le desktop puis on retire des fonctions, du contenu, des interactions pour les versions de devices plus petits (tablette, mobile)

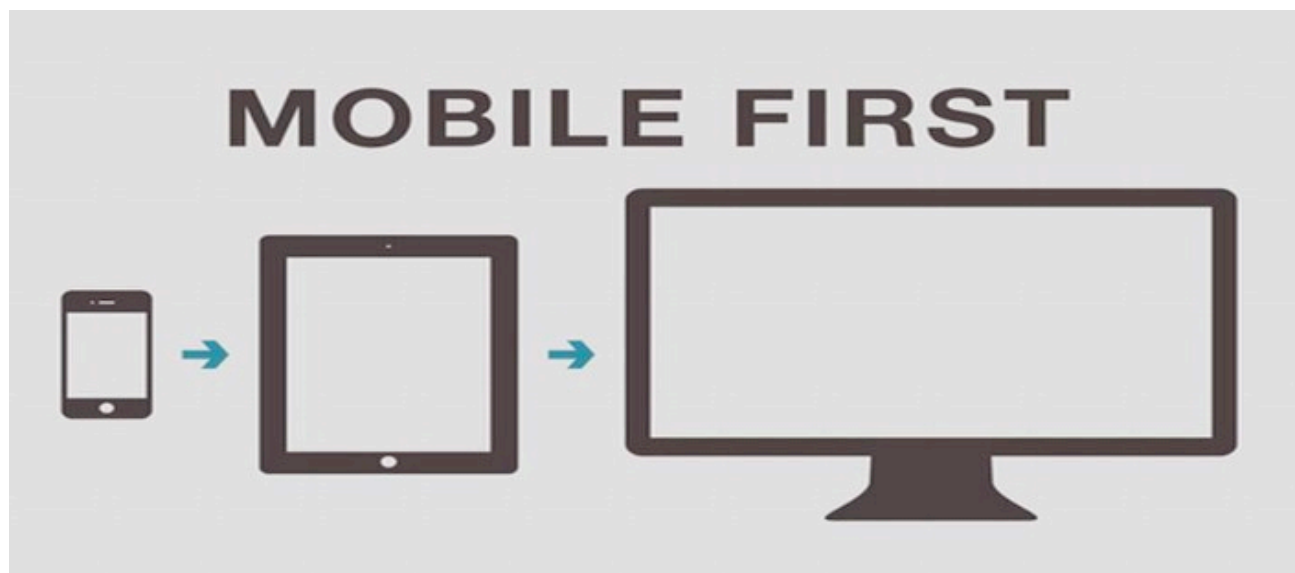
Mobile first

Mobile first, comme son nom l'indique, signifie que nous commençons la conception du produit du côté mobile, qui a plus de restrictions, puis nous étendons ses fonctionnalités pour créer une version pour tablette puis ordinateur de bureau.

Nous pensons notre site pour les utilisateurs de mobile.

Une approche de style "mobile-first" signifie que les styles sont appliqués en premier lieu aux appareils mobiles.

Les styles pour les écrans plus grands sont ensuite ajoutés à la feuille de style par le biais de media queries.



Cette approche utilise `min-width` dans les media-queries

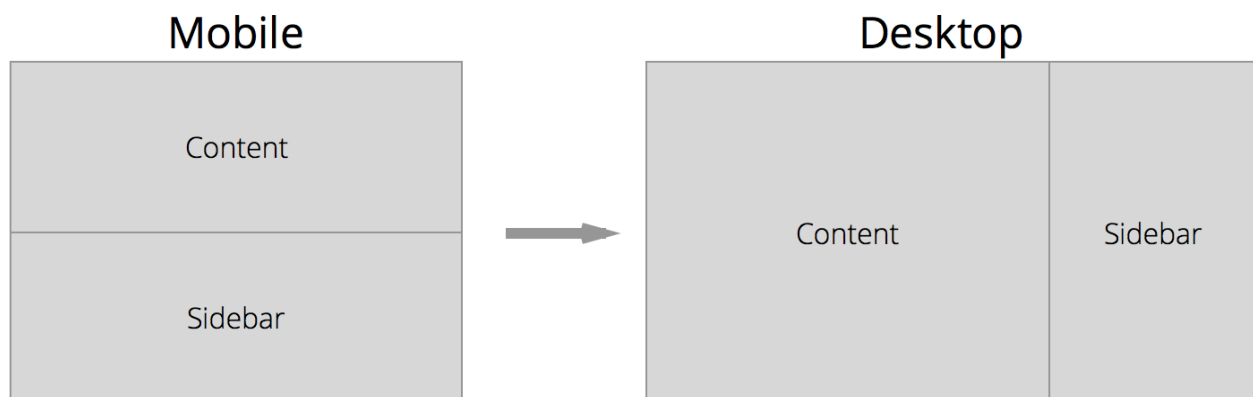
Exemple

```
body {  
  background:red;  
}  
  
// device d'une largeur > 600px  
@media (min-width: 600px) {  
  body {  
    background: green;  
  }  
}
```

Le code pour desktop est généralement plus complexe que celui des petits écrans.
C'est pourquoi le codage en mode "**mobile first**" permet de simplifier votre code.

La plupart du temps, nous pouvons nous fier aux propriétés par défaut pour styliser le contenu pour mobile.

Par exemple une `<div>` a une largeur de 100% par défaut.



Code pour mobile first

```
.content {  
  // Propriétés pour mobiles  
  // Ici rien n'est requis puisque le style par défaut suffit  
  
  // Propriétés pour écrans plus larges  
  @media (min-width: 800px) {  
    float: left;  
    width: 60%;  
  }  
}
```

Code pour dégradation élégante

```
.content {  
  // Propriétés pour écrans plus larges  
  float: left;  
  width: 60%;  
  
  // Propriétés pour mobiles  
  // Nous devons écraser les déclarations précédentes pour adapter la mise en page pour  
  // mobile  
  @media (max-width: 800px) {  
    float: none;  
    width: 100%;  
  }  
}
```