

Application web progressive

Plan et objectifs

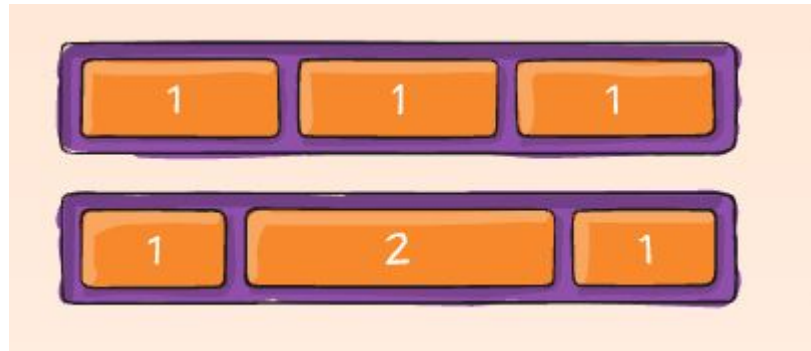
- Applications web progressives (PWA)
- Réactive vs Progressive
- Manifeste
- *Service Worker*

Application web réactive (responsive web app)

Une application web réactive ajuste son affichage en fonction de la taille de l'écran.

Habituellement l'affichage utilise les media queries (css) et flex box (css) afin d'y parvenir.

```
@media screen and (min-width: 80rem) {  
  .container {  
    margin: 1em 2em;  
  }  
}
```



Application web progressive

*Une progressive web app (PWA) est une **application web qui consiste en des pages ou des sites web, et qui peuvent apparaître à l'utilisateur de la même manière que les applications natives ou les applications mobiles.** Ce type d'application tente de combiner les fonctionnalités offertes par la plupart des navigateurs modernes avec les avantages de l'expérience offerte par les appareils mobiles.*

*Une PWA se consulte comme un site web classique, depuis une URL sécurisée mais permet une **expérience utilisateur similaire à celle d'une application mobile, sans les contraintes de cette dernière (soumission aux App-Stores, utilisation importante de la mémoire de l'appareil...).***

Elles proposent de conjuguer rapidité, fluidité et légèreté tout en permettant de limiter considérablement les coûts de développement.

https://fr.wikipedia.org/wiki/Progressive_web_app

Caractéristiques d'une PWA

Progressive

Les applications web progressives fonctionnent sur n'importe quel périphérique en intégrant les fonctionnalités disponibles du navigateur et de l'appareil utilisé.

Sécurisée

Afin de répondre aux problématiques de sécurité des échanges entre les utilisateurs et les sites, les PWA doivent être fiables et sûres par la mise en place d'un protocole HTTPS.

Engageante

Elles proposent une expérience utilisateur immersive en plein écran et un réengagement facilité grâce à l'envoi de notifications push web.

Caractéristiques d'une PWA

Installable

L'utilisation d'un fichier manifeste permet aux PWA de proposer, à l'instar d'une application mobile native, l'installation d'un raccourci sur l'écran d'accueil du terminal mobile.

Rapidité

Navigation rapide et fluide.

Optimisation pour le référencement

Utilisant les technologies du web, les progressive web app peuvent être référencées sur les moteurs de recherche de la même manière que n'importe quel site web classique.

Indépendante de la connexion

Grâce à la gestion du cache via l'utilisation d'un Service Worker, une fois le contenu chargé une première fois, il est possible de le consulter même dans les zones de faible connexion réseau.

Manifeste

*Le **manifeste** d'une application web fournit des informations concernant celle-ci (comme son nom, son auteur, une icône et une description) dans un document texte JSON. **Le but du manifeste est d'installer des applications sur l'écran d'accueil d'un appareil, offrant aux utilisateurs un accès plus rapide et une expérience plus riche.***

```
{
  "name": "HackerWeb",
  "short_name": "HackerWeb",
  "start_url": ".",
  "display": "standalone",
  "background_color": "#fff",
  "description": "A readable Hacker News app.",
  "icons": [{
    "src": "images/touch/homescreen48.png",
    "sizes": "48x48",
    "type": "image/png"
  }, ...],
}
```

Service Workers

*Les **service workers** jouent essentiellement le rôle de **cache** entre une application web, et le navigateur ou le réseau. Ils sont destinés à permettre la création d'expériences de navigation déconnectées efficaces, en interceptant les requêtes réseau et en effectuant des actions appropriées selon que le réseau est disponible et que des ressources mises à jour sont à disposition sur le serveur. Ils permettent aussi d'accéder aux APIs de **notifications du serveur** (push) et de **synchronisation en arrière-plan**.*

Un service worker est un worker enregistré auprès d'une origine et d'un chemin. Il prend la forme d'un fichier JavaScript.

Angular

<https://angular.io/guide/service-worker-getting-started>

