

Addendum Service worker

Intégration – Projet

Service Worker API

- Déplacer le fichier workerCacheFetched.js de src/lib/ vers src/
- A cause du fonctionnement de Parcel JS, l'intégration d'un service worker est un peu différente de la syntaxe de base (à ajouter en fin de fichier par ex.).

```
navigator.serviceWorker.register(  
  new URL( 'workerCacheFetched.js' , import.meta.url )  
)
```

Notifications Push

Qu'est-ce que c'est ?

Notifications Push

- Les notifications Push sont des messages utilisés par les applications pour vous avertir spontanément d'une information précise
- Elles s'appellent "Push", car elles servent à "Pusher" du contenu depuis un serveur, vers l'utilisateur final
- Permet de relancer très facilement l'interaction avec l'application

Avantages

Notifications Push

- Pour les utilisateurs, les notifications push sont un moyen de recevoir des informations opportunes, pertinentes et précises
- Pour vous (propriétaire de site web), les notifications push sont un moyen d'accroître l'engagement des utilisateurs
- Typiquement utilisé pour informer d'un événement _précis_ dans le temps, comme un rabais sur une durée limitée ou "teaser" du contenu



Trop de notifications peuvent énerver vos utilisateurs

Comment ça marche ?

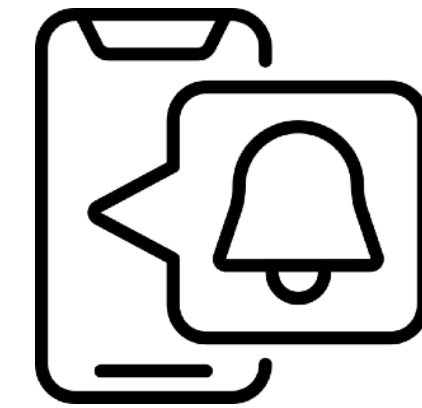
Notifications Push



Serveur Backend
de l'application



Push Service
(propre au browser)



Device

Comment ça marche ? – Subscription

Notifications Push

1. Demander l'autorisation au device (sécurité)
2. Créer une Subscription
3. La transmettre au serveur Backend et la stocker pour l'utiliser

Comment ça marche ? – Envoi

Notifications Push

1. Le serveur envoie une requête Push à un Push Service via un Web Push Protocol
2. Le Push Service l'authentifie et vérifie la validité de la requête
3. Le Push Service envoie la requête au device et l'application l'affiche, grâce à sa logique interne

Comment ça marche ?

Notifications Push

- Il s'agit ici d'une explication simplifiée du mode de fonctionnement
- Il y a plusieurs mécanismes de sécurité à chaque étape qui assure la validité et la sécurité des messages transmis
- Excellent article sur le fonctionnement des notifications Push sur: <https://web.dev/push-notifications-overview/>

Web Push

Notifications Push

- Les notifications push pour le web sont gérées par les browsers, pour la partie device
- Chaque browser choisi lui-même son Push service, pas d'incidence sur le code (implémentation interne)
- Les browser implémentent 2 Web APIs (les fameuses!) pour les intégrer et cela est ***très*** fortement lié aux Services Workers

Web Push

Notifications Push

- En gros:
 - Push API pour le “transport” des notifications depuis le service worker
 - Notification API pour les afficher du côté browser

Web Push – Autorisation

Notifications Push

- Avant d'utiliser les webpush, il faut d'abord ***demander l'autorisation*** à l'utilisateur, via le browser
- Cela se fait via:
`Notification.requestPermission()`
- Ou presque...

Web Push – Autorisation

Notifications Push

- Le browser peut dans certains cas simplement **ignorer** la demande de permission
- Pour des questions de sécurité, il peut ignorer cette demande, tant que l'utilisateur n'a pas fait un certain nombre d'opérations
- On appelle cela le "Media Engagement Index" - tant qu'une certaine valeur n'est pas atteinte, pas possible de les utiliser



Egalement valable pour l'installation d'une PWA

Web Push – A l'aide...

Notifications Push

- Tout cela devient compliqué...
- Heureusement, il existe des services de push notification qui s'occupe de faire cela pour vous !
- Exemple : onesignal.com

Web Push – OneSignal

Notifications Push

- Ce type de service vous permet d'écrire des notifications à la main comme des emails
- Il gère l'affichage des demandes d'autorisation, selon la logique du navigateur
- Il vous fournit un service worker clé en main pour afficher les notifications !
- Permet plein d'autres choses...! (Plugin wordpress, par exemple !!)

Web Push – OneSignal

Notifications Push

- Ce type de service vous permet d'écrire des notifications à la main comme des emails
- Il gère l'affichage des demandes d'autorisation, selon la logique du navigateur
- Permet plein d'autres choses...! (Plugin wordpress, par exemple !!)

Web Push – OneSignal

Notifications Push

1. Déployer sur netlify
2. Créer un compte OneSignal
3. Créer une app OneSignal avec des Web Push
4. Suivre les instructions

Web Push – OneSignal

Notifications Push

5. NB: Placez OneSignalSDKWorker.js dans src/
6. Modifiez votre package.json pour que les deux scripts ressemblent à cela:

```
"scripts": {  
  "start": "parcel index.html OneSignalSDKWorker.js",  
  "build": "parcel build src/index.html src/OneSignalSDKWorker.js"  
},
```
7. Supprimez votre serviceWorker actuel (dans index.js + via la Tab "Application" dans les devs tool)