

# PWA – Progressive Web Applications

Des bases solides pour développer des progressive web applications

2journs / 14h

## Public cible

- Développeurs, intégrateurs, architectes logiciel, chefs de projet technique

## Programme

- Ce cours suppose que vous ayez déjà les bases de de la programmation d'application web (ES6).
- Connaissances de base en Git
- Développement front-end avec NPM
- Installer nodejs (LTS) : <https://nodejs.org/>
- Installer git : <http://git-scm.com/>
- Votre éditeur favori (nous recommandons Visual Studio Code)
- Installer Chrome et au moins un autre browser à jour (Edge, Safari, Firefox)

## Programme

1. Introduction
  - Origines du concept
  - Définition du terme F.I.R.E.
  - Environnement de développement et de débogage

- Point sur les exercices à venir
- Point sur la sécurité dans les applications web
- Atelier: Mise en place d'un environnement de développement PWA

## 2. Manipulation des Services Workers

- Fonctionnement des services workers
- Installation d'un service worker
- Compréhension du cycle de vie d'un service worker. Comment agir sur le cycle de vie
- Manipulation des events du service worker: event 'install', 'activate', 'fetch'
- Ateliers:
  - Installer un premier service worker
  - Débogage d'une PWA
  - Prendre le contrôle des événements
  - Découvrir les événements 'install', 'activate' et 'fetch'

## 3. Gestion des stratégies de cache

- Présentation du concept d'App-shell
- Découvrons l'API de cache du service worker
- Découverte des stratégies de cache
- Découverte du mode offline
- Présentation de Workbox
- Ateliers:
  - Mettons en place un app-shell
  - Rendons notre application disponible hors ligne
  - Manipulons le cache d'un service worker
  - Utilisation de workbox pour gérer son cache

## 4. Aller plus loin avec le service worker

- Découverte de l'événement 'message'
- Découverte de l'événement 'sync'
- Ateliers:
  - Mise en place d'une communication cross onglets basé sur l'événement 'message'
  - Mise en place du background-sync pour l'envoi de messages

## 5. Rendre l'application installable

- Présentation du manifest
- Présentation des enjeux de l'installation d'une PWA
- Comment rendre une PWA disponible sur les Stores
- Atelier: Création d'un manifest et installation de l'application

## 6. Engageons notre utilisateur

- Présentation des concepts d'UX de l'engagement utilisateur

- Utilisation de l'événement 'push' du service worker
- Découverte des notifications web
- Comment déboguer une notification
- Ateliers:
  - Mise en place d'une notification avec Action
  - Mise en place d'un système de Push notification

## 7. Intégration à la plateforme

- Découverte des possibilités d'intégration d'API "Natives" : Geolocalisation, FullScreen
- Authentification simplifiée avec webAuthN
- PaymentRequest API
- Liste des APIs disponibles
- Design Responsive
- Ateliers:
  - Utilisation de l'api de geolocalisation, fullscreen
  - Mise en place d'une authentification WebAuthN

## 8. Améliorer la performance de l'application

- Retour sur les formats d'images à utiliser et les techniques associées
- Bonne pratiques de la performance web
- Présentation du pattern P.R.P.L.
- Atelier: Mise en place du P.R.P.L pattern

## 9. Tour des outils pouvant nous aider

- Liste d'outils pouvant aider à la réalisation d'une PWA
- CLI Angular, Vue, React, ...
- Présentation de Workbox
- Atelier: Audit avec LightHouse CI