

Progressive Web Apps (PWA)

Arthur MILLET : Arthur.Millet@etu.univ-grenoble-alpes.fr

Pierre BERNARD : Pierre.Bernard@etu.univ-grenoble-alpes.fr

Résumé :

Les Progressive Web Apps (PWA) constituent une avancée majeure dans le domaine du web moderne. Elles ont été pensées pour proposer une expérience se rapprochant des applications mobiles natives et hybrides, avec la possibilité d'intégrer des fonctionnalités et les avantages du web. Les PWA reposent sur trois points techniques : le service worker permettant le fonctionnement hors-ligne de l'application grâce à la gestion du cache, le manifeste permettant l'installation et l'apparence visuelle de l'application, et le dernier point concerne l'utilisation obligatoire du protocole HTTPS permettant des échanges sécurisés. Les PWA sont de plus en plus utilisées dans différents domaines d'application. On les retrouve donc dans le e-commerce, les médias et les applications internes des entreprises. L'avantage majeur des PWA concerne directement les entreprises et les développeurs, puisque grâce aux PWA, ils n'ont plus besoin de produire une application différente par plateforme (Android, IOS, desktop, ...). Cela permet donc de réduire le temps et les coûts de développement et de maintenance.

Abstract :

Progressive Web Apps (PWA) represent a major advance in the field of modern web technology. They are designed to offer an experience similar to native and hybrid mobile applications, with the ability to integrate web features and benefits. PWA are based on three technical points: the service worker, which allows the application to function offline thanks to cache management; the manifest, which allows for the installation and visual appearance of the application; and the mandatory use of the HTTPS protocol, which enables secure exchanges. PWA are increasingly used in various fields of application. They can be found in e-commerce, media, and internal company applications. The major advantage of PWA directly concerns companies and developers, since thanks to PWA, they no longer need to produce a different application for each platform (Android, iOS, desktop, etc.). This reduces development and maintenance time and costs.

Synthèse :

En 2015, le terme "progressive web apps" est proposé par Frances Berriman et Alex Russell (un ingénieur de Google). Ce terme désigne "les sites Internet profitant des nouveaux avantages donnés par les navigateurs web".

Elles apparaissent suite à plusieurs constats :

- les application mobiles sont de plus en plus nombreuses
- les application natives sont limités
- les standards web évoluent, proposant de nouvelles fonctionnalités
- la volonté de diminuer les coûts de développements
- améliorer l'engagement des utilisateurs

1. Contexte et motivations

L'usage massif des applications mobiles a transformé nos pratiques numériques. De plus en plus de personnes utilisent un appareil mobile pour naviguer sur le web. Les utilisateurs sont de plus en plus exigeants concernant les attentes : temps de chargement immédiat, navigation fluide, accessibilité sans accès à Internet ...

Les applications natives répondent partiellement à ces attentes, tandis que les applications web répondent à d'autres attentes. C'est donc dans ce contexte que les progressive web apps ont émergé essayant de trouver le meilleur compromis.

2. Marché et acteurs

Les PWA sont soutenues dès leur apparition par de grands acteurs du web. Parmi eux, on retrouve Google qui a su tirer profit de cette technologie en l'intégrant dans leur navigateur Chrome, ce qui a permis la démocratisation des PWA. Microsoft fait également partie des grandes entreprises qui soutiennent et qui mettent en avant les PWA sur Windows. De l'autre côté, on retrouve de nombreuses entreprises qui utilisent les PWA pour améliorer leurs performances, parmi elles, on retrouve : X (anciennement Twitter), Starbucks, Pinterest et bien d'autres...

3. Principes de fonctionnement et architecture

Les PWA se reposent sur ces trois piliers :

- Service Worker :

Il s'agit d'un script en JavaScript qui fonctionne en arrière-plan. Il permet le fonctionnement hors ligne de l'application grâce à une gestion avancée du cache. Il permet aussi de recevoir des notifications push, et notamment la synchronisation en arrière-plan utile pour les mises à jour.

- Web App Manifest :

Ce fichier JSON contient toutes les informations nécessaires à l'installation de la PWA, on y retrouve donc le nom et l'icône de l'application mais aussi le thème graphique, la page de démarrage et l'écran de chargement.

- Le protocole HTTPS :

Il n'est pas possible d'installer une PWA depuis un site web utilisant le protocole HTTP. Le protocole HTTPS est obligatoire pour des raisons de sécurité afin de protéger les échanges de données sur le réseau.

4. Technologies et standards associés

Les PWA se basent sur les technologies et standards du web moderne. Il est donc possible de créer une PWA à partir d'un site web utilisant les technologies natives comme HTML, CSS, JS. Mais les PWA sont également compatibles avec des frameworks comme TS, SCSS, TailwindCSS, et bien d'autres. Pour son bon fonctionnement les PWA se reposent entièrement sur ces API :

- Service Worker API
- Cache API
- Fetch API
- Notification API
- Background Sync

5. Solutions et cas d'usage

Les PWA sont désormais présentes dans divers domaines d'applications.

Dans le e-commerce avec Starbucks ou bien Aliexpress, où le grain de performance se traduit par une amélioration des conversions. Cela est notamment dû à la fonctionnalité hors-ligne, permettant de consulter les catalogues sans accès à Internet. Les articles des médias peuvent désormais être préchargés automatiquement et l'utilisation des notifications push permet d'augmenter la fidélisation des utilisateurs. Les réseaux sociaux comme X utilisent principalement la fonctionnalité de notification pour augmenter l'engagement des utilisateurs. Enfin, les entreprises elles-mêmes utilisent les PWA pour simplifier le déploiement et l'utilisation des applications en interne, plus besoin de passer par des stores et l'application peut s'adapter à tout type d'appareil.

6. Comparaison avec les alternatives

Les applications natives sont développées pour un système d'exploitation spécifique : Safari et garage band sont des applications natives IOS, il n'est pas possible d'obtenir ces applications sur d'autres systèmes d'exploitation.

Ces applications natives peuvent fonctionner sans connexion Internet si elles ne nécessitent pas d'échanges de données via des requêtes vers un serveur. Les applications natives sont connues pour avoir de très bonnes performances, car elles sont optimisées pour la plateforme sur laquelle elle va être utilisée.

Les applications hybrides sont développées avec des technologies du web, donc elle n'utilisent pas les technologies spécifiques à chaque système d'exploitation, comme le font les applications natives. L'inconvénient reste le même, pour accéder aux fonctionnalités et capteurs de l'appareil mobile : les méthodes d'accès dépendent du système d'exploitation. De plus, la publication de l'application passe par un store, donc on est toujours obligé de créer une couche spécifique à chaque système d'exploitation. Mais les applications hybrides restent mieux que les applications natives permettant un développement plus rapide, car le noyau de l'application reste toujours le même par application. En revanche, les performances sont limitées.

Les PWA restent dans la même lignée, permettant de déployer facilement l'application via un navigateur, plus besoin de passer par un store. Comme l'application se base uniquement sur les technologies du web, on développe une unique version, réduisant les coûts de développements et de maintenance car on a une application commune pour tout système d'exploitation. L'expérience utilisateur est proche de celle proposée par les applications natives.

7. Avantages et inconvénients

Les PWA proposent de nombreux avantages.

On peut les diviser en deux parties, les avantages pour les utilisateurs :

- une meilleure expérience mobile, plus fluide
- une installation facile, plus besoin de passer par un store
- de nouvelles fonctionnalités avec la possibilité d'intégrer des notifications push
- la possibilité d'accéder à l'application en mode hors-ligne
- les mises à jour sont désormais instantanées

Voici les principaux avantages pour les développeurs :

- la diminution du temps et donc du coût de développement : plus besoin de développer une version par système d'exploitation
- les maintenances sont donc également plus simples

Concernant les inconvénients, ils sont moins nombreux, mais concernent les accès aux fonctionnalités natives sur smartphone comme le bluetooth ou le NFC, certains navigateurs ne prennent pas en charge ces technologies. Et de manière plus générale, le support sur IOS est incomplet.

8. Conclusion

Les PWA répondent pleinement aux attentes numériques actuelles en offrant un compromis intéressant entre applications natives et applications web. Leur adoption ne cesse d'augmenter dans les différents secteurs que nous avons vus précédemment.

Malgré leur fort potentiel, les PWA restent encore limitées en raison de l'accès restreint à certaines fonctionnalités natives des appareils mobiles.

Cependant, comme les technologies web évoluent rapidement, on peut imaginer que ces limitations seront progressivement levées dans un avenir proche.

Démonstration :

<https://github.com/milletarthur/PWA>

Références :

web.dev (Google), « What are PWAs ? », Article,

<https://web.dev/articles/what-are-pwas?hl=fr>

- Ce guide officiel de Google explique ce qu'est une PWA, les principes fondamentaux (Service Worker, manifest, HTTPS), et son importance.

Mozilla, « Progressive Web Apps », Documentation MDN,

https://developer.mozilla.org/fr/docs/Web/Progressive_web_apps

- La page MDN offre une définition technique complète des PWA, décrit les API clés nécessaires et donne des exemples d'utilisation et de déploiement.

Progressive-Web-Apps.fr, « Définition des Progressive Web Apps (PWA) », Article,

<https://www.progressive-web-apps.fr/definition-progressive-web-apps-pwa>

- Ce site fournit une définition claire et accessible des PWA, avec des cas d'usage et des explications sur les bénéfices pour les entreprises et les développeurs.