

Introduction au développement mobile

1. Présentation

Il y a presque 7 milliards de personnes sur la planète, dont 5,1 milliards possèdent un téléphone portable. Le nombre d'utilisateurs accédant à Internet sur leurs smartphones a plus que doublé au cours des 5 dernières années, tout comme le nombre d'utilisateurs téléchargeant et utilisant des applications mobiles.

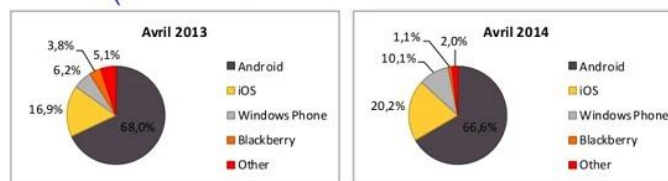
Parmi ceux qui utilisent Internet ou le courrier électronique sur leur téléphone, plus d'un tiers se connectent principalement via leurs appareils portables.

- **Utilisation des mobiles** (D'après l'union mondiale des télécommunications)
 - 6,9 milliards de terminaux vendus fin 2014
 - Taux de couverture atteint environ 95% de la population mondiale
- **Smartphones** (D'après l'union mondiale des télécommunications)
 - 1,76 milliards d'utilisateurs de smartphones
 - Augmentation de 23% par rapport à 2013
 - 395 millions de Go de données transférées
 - Augmentation de 48% par rapport à 2013
- **En Tunisie**
 - Plus de 12,63 millions de cartes SIM vendues (115% de taux de pénétration) (D'après l'Instance Nationale des Télécommunications (INT))
 - 12% des Tunisiens abonnés à la téléphonie mobile ont des smartphones (D'après Pew Global Attitude)

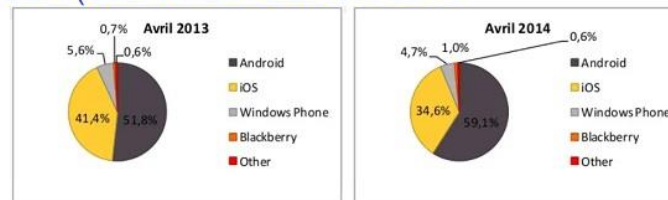
2. Systèmes d'exploitation pour mobile (OS mobile)

Système d'Exploitation Mobiles : Parts de Marché

France (de Avril 2013 à Avril 2014)



USA (de Avril 2013 à Avril 2014)



- **OS** est une abréviation pour **Operating System**. En français, le terme est traduit par "**système d'exploitation**".

Un système d'exploitation mobile est un système d'exploitation conçu pour fonctionner sur un appareil mobile. Ce type de système d'exploitation se concentre entre autres sur la gestion de la connectivité sans fil et celle des différents types d'interface.

Une application mobile ou simplement app mobile, est un programme informatique ou une application logicielle conçue pour s'exécuter sur un appareil mobile tel qu'un téléphone, une tablette, téléviseur, une montre, etc.

Les applications mobiles contrastent souvent avec les applications de bureau conçues pour s'exécuter sur des ordinateurs de bureau et les applications Web qui s'exécutent dans les navigateurs Web mobiles plutôt que directement sur l'appareil mobile.

Un système d'exploitation mobile permet à l'utilisateur d'exécuter de différents logiciels d'application sur le mobile, les tablettes, les montres intelligentes, etc. De plus, on peut dire qu'il s'agit d'un type de système d'exploitation spécialement conçu pour les mobiles. Ils sont un mélange de système d'exploitation informatique avec quelques fonctionnalités supplémentaires. En outre, ils sont relativement légers et simples.

- **Exemples d'OS pour Mobile**

- **Android : l'OS mobile de Google**

Android de Google fut développé par une petite startup ensuite achetée par Google qui poursuit activement son développement. Android distribué sous licence open source est une variante de Linux.

Le système d'exploitation Android est le plus utilisé parmi tous les systèmes d'exploitation mobiles qui existent. Il est gratuit et open source basé sur le noyau Linux développé par Google.

- **Système d'exploitation Blackberry**

Le développeur de ce système d'exploitation est Research In Motion (RIM). Il a été spécialement conçu pour les appareils BlackBerry.

- **iOS : l'OS mobile d'Apple**

C'est l'un des OS les plus populaires après Android. Il est conçu pour fonctionner sur les appareils Apple tels que les iPhones, les tablettes iPad, etc. De plus, comme dans le cas du système Android, IOS permet le téléchargement de ses applications via son AppStore.

C'est le concurrent le plus important de l'OS Android, le système d'exploitation développé par l'entreprise Apple.

Si la firme américaine équipe bien moins de smartphones puisqu'iOS est réservé aux produits appels (iPhone et iPad), le système d'exploitation iOS est apprécié, car il est considéré comme l'OS proposant la meilleure expérience utilisateur.

- **Harmony OS**

C'est un OS conçu par la société chinoise Huawei. Il est spécialement conçu pour être utilisé dans les appareils IoT en incluant les téléphones portables et tous ce qui est mobile.

- **Bada, l'OS de l'entreprise Samsung**

qui a équipé pour la première fois en 2010 un smartphone (Samsung Wave);

- **Meego, l'OS de Nokia et Intel**

Conçu pour fonctionner sur une majorité d'appareils;

3. Fonctions d'un système d'exploitation pour mobile

- **Gestion de la mémoire**

Quel que soit le programme exécuté, il doit être présent dans la mémoire principale. Par conséquent, il peut y avoir plus d'un programme présent à la fois ce qui rend nécessaire de gérer la mémoire. Le système d'exploitation alloue et désalloue la mémoire pendant le multitraitement.

- **Gestion du processeur**

Lorsque plus d'un processus s'exécute, le système d'exploitation décide comment et quand un processus utilisera le processeur. Le système d'exploitation Alloue et désalloue le processeur aux processus.

- **Gestion des dispositifs**

Les processus peuvent nécessiter des dispositifs pour leur fonctionnement tels que les capteurs et les périphériques d'entrées sorties. Cette gestion est effectuée par l'OS. Le système d'exploitation alloue et désalloue des dispositifs à différents processus.

- **Gestion des fichiers**

Les fichiers d'un système sont stockés dans différents répertoires. Le système d'exploitation :

Gère des registres de l'état et de l'emplacement des fichiers.

Alloue et désalloue des ressources.

- **Sécurité**

Le système d'exploitation assure la sécurité du système et des programmes grâce à l'authentification d'une part, et la gestion des permissions de l'utilisation des ressources d'une autre part

Les contraintes liées au développement mobile

1. Présentation

- Lorsque que l'on cherche à concevoir une application mobile, le positionnement de l'application ou le choix de la technologie ne sont pas les seules contraintes.
- En développant des applications pour mobiles, il faut prendre en considération certaines contraintes des appareils cibles
- Le développement sur smartphones doit prendre en compte les spécificités de ce type de terminaux comme la gestion des ressources matérielles et réseaux limitées, la fragmentation importante des systèmes et des marketplaces, et l'hétérogénéité des environnements de développement.

2. Des ressources limitées

- **Mémoire limitée / processeur lent**
 - Bien que les nouveaux modèles haut de gamme arrivent à 3Go de RAM (même 4 prévu pour 2015), la plupart des smartphones souffrent d'un temps de chargement long
 - Bien faire attention au type de public ciblé
 - Éviter les traitements complexes, gourmands en mémoire
- **Capacité de stockage**
 - Quelques smartphones très haut de gamme (iPhone6) ont une mémoire de 128Go
 - Mais plusieurs modèles d'entrée de gamme n'ont que 4Go (majorité de 16Go)
 - Penser à s'orienter plus vers le stockage sur le cloud, par exemple.
- **Automobile**
 - Éviter les applications gourmandes en énergie (caméra vidéo par exemple, nécessaire pour la réalité augmentée)
 - Application doit supporter
- **Taille d'écran réduite et variable**
 - Réduction du contexte de l'application par rapport à un écran
 - Tailles varient d'un appareil à un autre (smartphone, tablette, phablette,...)
 - Exploiter tout l'espace fourni de manière optimale
 - S'orienter dans le développement vers le responsive-design (adapté à la taille de l'écran)
 - Penser à des choix d'IHM qui facilitent la navigation (un header fixe, utilisation des icônes à la place des mots...)
- **Problèmes de connectivité**
 - Problèmes de connectivité dus à la mobilité
 - Utilisation des réseaux 3G/4G, donc payants
 - Penser à un mode offline pour vos applications/sites

- Attention aux mises à jour automatiques
- **Téléchargement plus lent**
 - Connexions internet plus lentes, latence réseau et mémoire et processeur limités
 - Vitesse de téléchargement assez faible par rapport aux ordinateurs
 - 80% des utilisateurs ne veulent pas utiliser leur téléphone pour surfer sur le web
 - 64% s'attendent à ce qu'une page soit disponible en 4s, alors qu'en moyenne, un smartphone la télécharge en 6s
 - Utiliser des technologies qui facilitent le chargement des pages, comme l'exploitation de la cache, ou l'utilisation de CSS3
 - Réduire la taille des images, le nombre de fichiers et les traitements côté client
- **Manipulation plus délicate**
 - Utilisation délicate du clavier tactile, sujette à beaucoup d'erreur de frappe et désagréable pour l'utilisateur
 - Minimiser le nombre de champs de texte
 - Favoriser les champs préremplis
 - Fonctionnalités d'auto-complete, correction d'orthographe...
 - Clics invalides, à cause de l'utilisation des doigts (problème appelé du FAT FINGER)
 - Attention à la taille et proximité des éléments cliquables
 - Penser à une fonctionnalité "Annuler" disponible un peu partout
 - Absence de l'effet de survol (hover)
 - Besoin de plus de liens et de boutons

3. Le budget de la création d'application

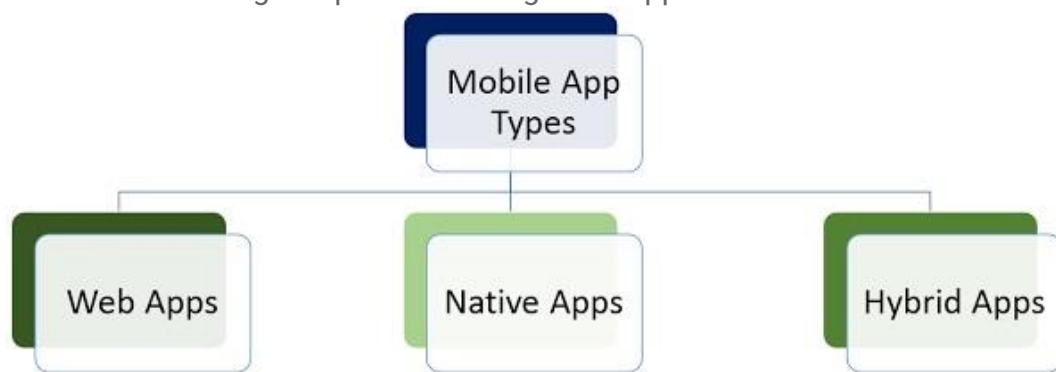
- Créer une application pour smartphone nécessite un budget pouvant être bien plus élevé que prévu surtout :
- Si l'application doit être disponible sur tous les systèmes d'exploitation.
- Si elle doit être développée en utilisant des technologies natives.
- Si l'on a besoin d'une infrastructure serveur pour répondre aux besoins fonctionnels le nécessitant.
- Cet investissement économique peut être une contrainte dans la création d'application mobile.

Types d'application mobile

1. Présentation

Il existe trois types de base d'applications mobiles si nous les catégorisons selon la technologie utilisée pour les coder :

- **Les applications natives** sont créées pour une plate-forme ou un système d'exploitation spécifique.
- **Les applications Web** sont des versions réactives de sites Web qui peuvent fonctionner sur n'importe quel appareil mobile ou système d'exploitation, car elles sont fournies à l'aide d'un navigateur mobile.
- **Les applications hybrides** sont des combinaisons d'applications natives et Web, mais enveloppées dans une application native, ce qui lui donne la possibilité d'avoir sa propre icône ou d'être téléchargée à partir d'un magasin d'applications.



2. Applications natives

- Les applications natives sont conçues spécifiquement pour le système d'exploitation (OS) d'un appareil mobile. Ainsi, vous pouvez avoir des applications mobiles Android natives ou des applications iOS natives, sans parler de toutes les autres plates-formes et appareils. Parce qu'ils sont conçus pour une seule plate-forme, vous ne pouvez pas mélanger et assortir – par exemple, utilisez une application Blackberry sur un téléphone Android ou utilisez une application iOS sur un téléphone Windows.
- Ces applications ne fonctionnent que sur les appareils pour lesquels elles sont conçues.
- Si une application est développée pour les mobiles Apple, c'est-à-dire macOS, cette application ne fonctionnera sur aucun autre appareil doté d'un système d'exploitation autre que macOS.
- Si une application est développée pour le système d'exploitation Android, elle ne fonctionnera que sur les appareils dotés du système d'exploitation Android, et sur aucun autre appareil.
- Les avantages des applications natives sont qu'elles communiquent facilement avec l'appareil sur lequel elles s'exécutent,
- Il communique facilement avec le matériel de l'appareil comme l'appareil photo, les haut-parleurs, le moteur graphique, les boutons, le toucher, les capteurs de localisation et les accéléromètres, etc.
- Le rendu de la page, c'est-à-dire le rendu de l'interface utilisateur, se poursuit sans aucun problème de suspension ou de blocage.

- Le flux de mise en page est impeccable et facile à utiliser.
- Ces applications sont plus conviviales, plus faciles à utiliser et peuvent facilement tolérer une charge d'activité rapide des utilisateurs.
- Ces applications peuvent réagir à la même vitesse avec ce que l'utilisateur les utilise.
- Ces applications en termes d'interactivité remplacent tous les autres types d'applications.

▪ Technologie utilisée

Les applications natives sont codées à l'aide de divers langages de programmation. Voici quelques exemples : Java, Kotlin, Python, Swift, Objective-C, C++ et React.

▪ Avantages

En raison de leur orientation singulière, les applications natives ont l'avantage d'être plus rapides et plus fiables en termes de performances. Ils sont généralement plus efficaces avec les ressources de l'appareil que d'autres types d'applications mobiles. Les applications natives utilisent l'interface utilisateur native de l'appareil, offrant aux utilisateurs une expérience client plus optimisée .

Et comme les applications natives se connectent directement au matériel de l'appareil, elles ont accès à un large choix de fonctionnalités de l'appareil telles que Bluetooth, les contacts du répertoire, la pellicule, le NFC, etc.

▪ Inconvénients

Cependant, le problème avec les applications natives réside dans le fait que si vous commencez à les développer, vous devez dupliquer les efforts pour chacune des différentes plateformes. Le code que vous créez pour une plateforme ne peut pas être réutilisé sur une autre. Cela fait grimper les coûts. Sans parler de l'effort nécessaire pour maintenir et mettre à jour la base de code pour chaque version.

Et puis, chaque fois qu'il y a une mise à jour de l'application, l'utilisateur doit télécharger le nouveau fichier et le réinstaller. Cela signifie également que les applications natives occupent un espace précieux dans le stockage de l'appareil.

3. Applications Web

Les applications Web se comportent de la même manière que les applications natives, mais sont accessibles via un navigateur Web sur votre appareil mobile. Ce ne sont pas des applications autonomes dans le sens où vous devez télécharger et installer du code sur votre appareil. Ce sont en fait des sites Web réactifs qui adaptent leur interface utilisateur à l'appareil sur lequel l'utilisateur se trouve. En fait, lorsque vous rencontrez l'option "d'installer" une application Web, elle marque souvent simplement l'URL du site Web sur votre appareil.

Un type d'application Web est l'application Web progressive (PWA), qui est essentiellement une application native exécutée dans un navigateur.

▪ Technologie utilisée

Les applications Web sont conçues à l'aide de **HTML5**, **CSS**, **JavaScript**, **Ruby** et des langages de programmation similaires utilisés pour le travail Web.

▪ Avantages

- Parce qu'il est basé sur le Web, il n'est pas nécessaire de le personnaliser en fonction d'une plate-forme ou d'un système d'exploitation. Cela réduit les coûts de développement.
- De plus, il n'y a rien à télécharger. Ils n'occuperont pas d'espace sur la mémoire de votre appareil comme une application native, ce qui facilite la maintenance – il suffit de pousser la mise à jour en direct sur le Web. Les utilisateurs n'ont pas besoin de télécharger la mise à jour sur l'App Store.
- Le coût de développement de ces applications est faible.
- Disponibilité facile des développeurs.
- Entretien facile et moins cher.
- Les données ou l'interface utilisateur qu'ils contiennent peuvent être mises à jour immédiatement car elles se trouvent sur les serveurs plutôt que sur l'appareil utilisateur.

▪ Inconvénients

Mais cela est également pertinent : les applications Web dépendent entièrement du navigateur utilisé sur l'appareil. Il y aura des fonctionnalités disponibles dans un navigateur et non disponibles dans un autre, offrant éventuellement aux utilisateurs des expériences variées.

4. Applications hybrides

- Le terme hybride signifie "mélange". Ainsi, une application hybride fait référence à des applications natives et basées sur le Web.
- Les applications hybrides sont une intégration d'applications natives et Web. Ces applications utilisent à la fois des technologies Web et des API natives.
- Ces applications peuvent utiliser la plupart des fonctionnalités de l'appareil, telles que les applications natives. Semblables aux applications Web, les applications hybrides sont constituées de technologies Web telles que HTML, CSS, Javascript, jquery, AJAX, JSON, etc.
- Ceux-ci utilisent également des navigateurs Web pour afficher le contenu et les interfaces utilisateur, mais ils utilisent les éléments natifs ainsi que les éléments Web pour le développement.
- Le code de ces applications doit juste être modifié pour le rendre compatible sur plusieurs plates-formes.

▪ Technologie utilisée

Les applications hybrides utilisent un mélange de technologies Web et d'API natives. Ils sont développés en utilisant : Ionic, Objective C, Swift, HTML5 et autres.

▪ Avantages

- Construire une application hybride est beaucoup plus rapide et plus économique qu'une application native.
- En tant que telle, une application hybride peut être le produit viable minimum – un moyen de prouver la viabilité de la création d'une application native.
- Ils se chargent également rapidement, sont idéaux pour une utilisation dans les pays où les connexions Internet sont plus lentes et offrent aux utilisateurs une expérience

utilisateur cohérente. Enfin, comme ils utilisent une base de code unique, il y a beaucoup moins de code à maintenir.

- Facile à développer.
- Une base de code d'application est compatible avec plusieurs plates-formes.
- Faible entretien.
- Plus facile à développer, mettre à jour, tester et ajuster.

▪ Inconvénients

Les applications hybrides peuvent manquer de puissance et de vitesse, caractéristiques des applications natives.

5. Conclusion

- Bien qu'il existe des avantages et des inconvénients pour chaque type d'applications, on peut néanmoins conclure que les applications natives sont les meilleures parmi toutes sortes d'applications sur tous les facteurs tels que les performances, la stabilité, le comportement et la sécurité.
- En deuxième priorité, les applications hybrides peuvent être envisagées, puis les applications Web peuvent être envisagées pour des applications mobiles rapides, plus rapides et légères.

Outils et Framework de développement des applications mobiles

1. Présentation

- Beaucoup de Framework et d'outils ont été développés pour répondre aux besoins spécifiques des constructeurs d'appareils mobiles.
- Elles offrent des mécanismes puissants, facile à mettre en œuvre et sécurisés pour le développement efficace et rapide d'applications mobiles.
- Cela est assuré tout en respectant le mieux les standards adoptés par les OS mobile.
- La création d'une application mobile est une tâche complexe. En tant que développeur, il faut trouver des moyens créatifs pour combiner toutes les fonctionnalités suivantes dans un seul projet.
- **Hautes performances** : utile pour garder les utilisateurs connectés et augmenter la base de données.
- **Rentabilité** : développer une application prend du temps, et le temps coûte cher. Un projet bien fait aura toujours à l'esprit le coût de production et la portée du marché potentiel.
- **Évolutivité** : le monde des applications mobiles est en constante évolution. De nouvelles mises à jour des systèmes d'exploitation sortent chaque jour ce qui nécessite que les applications à développer doivent être évolutives.
- De nouveaux frameworks ont été créés pour répondre à toutes ces considérations. Des Framework multiplateformes ont connus un succès vu la possibilité de déployer leurs produits sur plusieurs types de mobile comme le cas de Flutter, Ionic, et React native.

2. Flutter



- Flutter est le plus récent des Framework multiplateformes. Il a été créé par Google en 2013.
- Son langage de programmation DART a été inventé par Google en 2011. Flutter est présenté comme étant le successeur de JavaScript en mieux. Depuis sa 1ère version en 2018, Flutter connaît une croissance d'utilisation de 279 %. Comme exemple d'applications conçues avec Flutter, on peut citer Alibaba.
- Flutter est un kit de développement logiciel (SDK) d'application mobile créé par Google. Il a été publié en 2017 et depuis lors, il a pris un essor considérable parmi les organisations de développement de logiciels et les développeurs.
- Il utilise le langage de programmation Dart de Google, qui est calqué sur l'approche conventionnelle orientée objet, qui offre non seulement une pléthore de fonctionnalités, mais est facilement reconnaissable par les développeurs C++ ou Java.

▪ Avantages de Flutter

- **Simplicité** : Flutter est facile à apprendre car il est basé sur Dart qui ressemble un peu à Javascript. Il produit des applications compatibles avec les deux plateformes Android et IOS à partir d'une base de code unique.
- **Outils et fonctions avancés** : Flutter permet de bénéficier de nombreux outils de développement comme la fonction de rechargement à chaud intégrée qui permet de voir instantanément les modifications apportées à l'application sans avoir à la recompiler.
- Utilisable pour autres plateformes non mobiles: en mai 2019, Google a annoncé une nouvelle prise en charge de Flutter pour les applications de bureau, intégrées, mobiles et Web à partir de la même base de code.

▪ Inconvénients du flutter

- **Des mise à jours fréquentes** : cela rend plus difficile le suivi de toutes les mises à jour du framework.
- **Dart** : ce langage est encore très nouveau. Il peut donc être difficile de trouver un personnel suffisamment qualifiées pour créer une application avec.
- Les applications Flutter sont volumineuses : en raison de leurs widgets intégrés, les applications Flutter sont plus volumineuses que leurs homologues natives. Étant donné que les utilisateurs ont une mémoire limitée sur leurs appareils, il faut toujours en tenir compte de cette contrainte.

○ Applications développées avec Flutter

- Google Ads
- Reflectly
- Google Pay
- Stadia
- eBay Motors

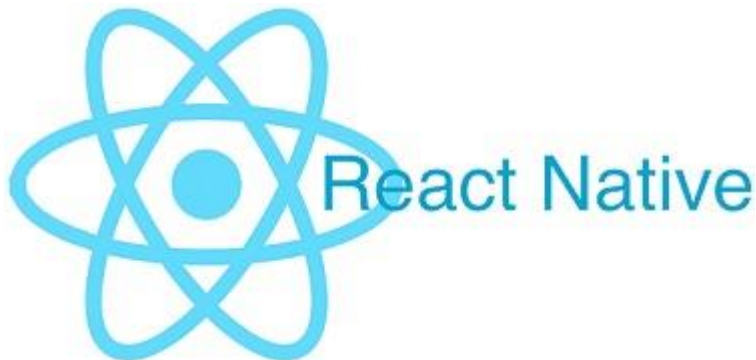
○ Ionic



- **Ionic** est disponible gratuitement, open-source et un Framework SDK frontal qui vous permet de créer des applications mobiles pour les téléphones iOS, Windows et Android en utilisant la même base de code. Il s'avère être un outil multiplateforme pour le développement mobile.
- **Ionic** est un Framework gratuit créé en 2013 qui utilise les technologies web comme HTML, CSS ou JavaScript.
- Il s'appuie sur AngularJS, Cordova et Node JS. IONIC est utilisé pour développer des applications mobiles et aussi des applications web comme le cas de Flutter.

- Cet outil permet de créer des applications mobiles hybrides. Cette plate-forme vous permet de créer des applications mobiles à l'aide d'applications Web et de langages tels que HTML , CSS , Javascript , Angular et Typescript . . Ionic possède une collection de composants qui fournit les fonctionnalités d'une plate-forme mobile. À l'exception de la fonctionnalité, Ionic propose différents comportements. Ionic est livré avec des performances efficaces avec un minimum de manipulation DOM.
- **Avantages d'Ionic**
 - Prise en charge de différentes technologies: à partir d'Ionic 4, il est possible d'utiliser n'importe quel Framework frontend pour créer des applications mobiles.
 - **Facile** : il se base sur HTML, CSS et JavaScript, donc l'utilisation de **ionic** ne nécessite pas beaucoup d'effort pour une personne qui a été déjà formé au développement Web.
- **Inconvénients d'Ionic**
 - Ne convient pas aux applications volumineuses : Malheureusement, pour les applications volumineuses, l'utilisation de la composante WebView qui est très utile pour Ionic peut entraîner un manque de performances important.
 - Peut manquer de prise en charge des plugins natifs : certains plugins natifs ne sont pas stables et peuvent entrer en conflit les uns avec les autres. De plus, ils peuvent être complètement absents dans la liste des plugins d'une distribution.

○ React Native



- **React Native** a été créé par Facebook. Sa première version date de 2015. Il est basé sur React qui est une librairie JavaScript, langage habituellement utilisé pour le Web. Facebook, Instagram, Skype ou Airbnb ont été réalisés en partie ou entièrement grâce à cette technologie.
- **React Native** est un Framework d'application mobile de Facebook. Ce Framework open-source peut être utilisé pour développer des applications Android, web, iOS et de bureau.
- **React Native** est très populaire en tant que cadre basé sur JavaScript utilisé pour le rendu natif des applications Android et iOS.
- Les développeurs peuvent utiliser React en conjonction avec les fonctionnalités des plateformes natives.
- **React Native** est basé sur la bibliothèque JavaScript de Facebook, axée sur l'interface utilisateur, et convient au développement de plateformes mobiles.
- Les développeurs peuvent utiliser la bibliothèque JavaScript pour écrire des applications natives et partager du code entre les plates-formes. React Native facilite également un meilleur développement simultané d'applications iOS et Android.

- **Avantages de React Native**

- **Apprentissage facile** : comme il est basé sur JavaScript ou React, l'utilisation de ce framework est simple. Il n'y a rien à apprendre de nouveau à part gérer les composants personnalisés offerts par cet outil.
- **Idéal pour la productivité** : React Native est livré avec des fonctionnalités utiles telles que le rechargement à chaud et les outils de développement Chrome pour le débogage des applications.
- **Performances** : les performances des applications créées à l'aide de React Native sont proches de celles d'un code Natif.
- **Fort soutien de la communauté** : Le framework React Native est soutenu par une large communauté de développeurs, ce qui facilite la tâche des nouveaux développeurs qui choisissent de l'utiliser. Le soutien communautaire facilite le développement d'applications mobiles.

- **Inconvénients de React Native**

- Peu de composants tiers : ce projet étant encore nouveau, il n'y a pas autant de composants tiers que parfois, on est obligé de mettre en œuvre une solution personnalisée à partir de zéro.

- **Applications développées avec React Native**

- **Facebook**

- Facebook a d'abord été un site web, et il a été utilisé comme une plateforme de réseaux sociaux, mais au fur et à mesure que la technologie des ordinateurs a progressé, ils ont développé une application. Non seulement il a émergé d'un type d'ordinateur à un autre, mais il a également couvert le monde avec beaucoup de ses caractéristiques étonnantes.

- **Instagram**

- Instagram est une application qui n'a pas beaucoup de fonctionnalités complexes et de choses qui se passent. Il permet à leurs développeurs de gérer très facilement les choses sur toutes les plateformes.

- **Bloomberg**

- Étant une plateforme qui fournit des informations commerciales à tous les utilisateurs, elle a gagné de nombreux utilisateurs, mais cela a été très difficile pour les développeurs. C'est parce que la gestion de deux applications différentes pour des plateformes différentes a pris beaucoup de temps.

- **Artsy**

- Il existait de nombreux marchés en ligne dans le monde, mais aucun n'était spécifiquement destiné aux artistes. De plus, les places de marché déjà disponibles n'étaient pas adaptées aux artistes. C'est parce que les œuvres d'art sont vendues principalement par enchères, et Artsy a été la plate-forme qui a offert une opportunité à plus de 100 000 artistes.

- **Coinbase Pro**

- Cette application fournit l'interface commerciale aux utilisateurs, mais il a été difficile pour l'équipe de conserver son interface phare et ses fonctionnalités standard pour les différentes plateformes. C'est la raison pour laquelle ils sont passés des applications

natives aux applications React Native. Il a permis une gestion plus facile et de nouvelles fonctionnalités qui n'existaient pas auparavant.

- **Airbnb**
- C'était une autre excellente application en raison d'une idée novatrice. Il a apporté de l'innovation sur le marché, de nombreuses personnes ont adoré leur application et leur base d'utilisateurs s'est rapidement développée.
- **Tesla**
- C'est l'un des plus grands noms en matière de voitures électriques. L'une des raisons est que leurs voitures sont très efficaces sur le plan énergétique et qu'elles sont très riches en fonctionnalités pour les utilisateurs.

3. Xamarin



- Xamarin est un autre Framework cross Platform pour le développement d'applications iOS et Android.
- Puisqu'il utilise le langage de programmation C#, il nécessite relativement peu de lignes de codes. Le résultat est donc un processus de codage beaucoup plus rapide.
- **Avantages d'Xamarin**
- Xamarin vous permet également un transfert facile de vos scripts entre les différentes plateformes, à l'image de Windows et macOS.
- Puisque la création d'applications mobiles est très rapide de nos jours, vous pourriez penser que cela insinue une perte en performances ou d'esthétique.
- Pourtant, les applications développées avec Xamarin présentent des performances natives irréprochables.
- Par ailleurs, Microsoft a racheté Xamarin permettant une connexion beaucoup plus simple à Microsoft Visual Studio, ce qui est un avantage pour les développeurs. Xamarin est apprécié pour :
 - le fait qu'il cible toutes les plateformes,
 - intègre plus facilement les API,
 - permet un développement plus rapide et plus rentable,
 - son support UI et l'accès aux API natives.
- **Inconvénients de Xamarin**
- Frais généraux de l'application
- La surcharge de l'application intégrée laisse une grande empreinte. Cela pourrait affecter le temps de téléchargement et l'espace de stockage requis pour l'application sur les appareils des utilisateurs.

- Bien sûr, l'équipe Xamarin fait de son mieux pour travailler sur ces problèmes, mais les utilisateurs de l'application doivent toujours en profiter au maximum.
- Manque de soutien communautaire
C'était l'un des inconvénients les plus importants de Xamarin. Les développeurs Android, iOS et NET disposent d'énormes forums, groupes et communautés de support communautaire avec une mine d'informations. Par rapport à tous ceux-ci, Xamarin est relativement nouveau, donc la communauté doit encore se développer. Si vous rencontrez des obstacles lors du développement de l'application, rien ne garantit que vous pourrez trouver une solution en ligne. Vous devrez peut-être embaucher un partenaire certifié Xamarin, un programme de partenariat qui vous aidera à résoudre vos problèmes car il a davantage accès aux ressources d'assistance.
- Impossible de partager des codes en dehors de Xamarin
- Les développeurs ne peuvent pas partager de codes avec d'autres développeurs ou équipes s'ils utilisent uniquement du code Swift, Java et Objective C écrit à l'aide de **C#** et **Xamarin**, ne peuvent pas être partagés tels quels. veulent travailler sur ce cadre.
- Certains codage manuel requis
- L'expression scénario « écrivez un, codez n'importe où » n'est pas tout à fait vraie car certains des codes devront être écrits individuellement pour chaque plate-forme. Le développement de l'interface utilisateur pour chaque plate-forme prendra du temps, d'autant plus qu'il doit être effectué manuellement.

○ JQuery Mobile



- **jQuery** est un Framework qui permet de développer dans une relative simplicité des applications et des sites internet adaptables à tous les supports, principalement les tablettes tactiles et les téléphones portables. Notons que HTML5 et CSS3 fonctionnent également par ce biais.
- **jQuery Mobile** est un framework crossplatform pour la création d'applications mobiles ou web pour différents appareils, et pour lesquels il est possible d'utiliser une seule version du code.
- Ce framework, basé sur HTML5, inclut des plugins comme un slider d'images, un slider de contenu, des pop-up boxes et bien d'autres afin de pouvoir réaliser un design dynamique et riche en fonctionnalités.

○ Avantages de JQuery Mobile

- Les avantages de jQuery sont nombreux et on ne va pas s'en plaindre. Tout d'abord, il est doté d'une ergonomie particulièrement aboutie qui permet de le prendre en main rapidement sans nécessairement disposer de compétences astronomiques en matière de développement. Aussi, une documentation particulièrement dense existe pour réaliser de nombreuses tâches différentes, un réel plus pour qualifier son application.

- Une fois l'item développé, le rendu est vraiment intéressant, et ce, sur tous les supports (Android, iOS, BlackBerry, Symbian, Windows Phone, etc...), clairement le mode « Responsive » est une des clés du succès de jQuery. Enfin, même en matière d'accessibilité, tout a été pensé puisque ARIA est pleinement intégré dans le Framework.
- Aussi, JQuery Mobile est interopérable avec un certain nombre de frameworks de développement mobile, à l'image de PhoneGap.
- Enfin, JQuery Mobile est convivial, a une base de code unique et permet aux développeurs de concevoir des applications et sites hautement personnalisés