

Cours **Apps mobiles**

Credit : 5

Coefficient : 3

Mode d'évaluation :

- Examen (60%) - contrôle continu (40%)

Author:

Aimad Eddine Debbi

Introduction (développement pour les mobiles)

1. Développement d'applications mobiles natif.
2. Développement d'applications mobiles web et hybride.
3. Pros et cons des # types de développement.

Développement natif pour mobile

- ❑ Développement pour une plateforme spécifique (Android, iOS, windowsphone, blackberry,...)
- ❑ Le développement réalisé moyennant des frameworks liés étroitement à l'API OS cible (Donc moins de Bridges et accès directe aux services système).
- ❑ Une similitude avec les applications de bureau traditionnelles sur le plan développement et installation.

Développement natif pour mobile

- ❑ Développement avec les langages prévus au sein des plateformes (Java et Kotlin sur la plateforme android, swift et objective C sur iOs, C# sur windowsphone).
- ❑ Accès quasi direct et total aux services système et aux périphériques (camera, GPS, Bluetooth, AR,...). Chemin le plus raccourci pour " la génération de l'exécutif " et donc meilleurs performances.

Développement natif pour mobile

- ❑ Le paramètre expérience utilisateur est aussi beaucoup plus opportun ici avec les apps natives. Sont généralement plus réactives et offrent une expérience utilisateur plus fluide, car elles respectent les lignes directrices de conception des plateformes respectives (Material Design pour Android, Human Interface Guidelines pour iOS).

Développement web pour mobile

- ❑ Les apps tournent en se chargeant dans le navigateur (Chrome, Safari et Firefox).
- ❑ Typiquement le développement repose sur l'utilisation des langages du web javascript, typescript, ...
- ❑ Le web et les pages web en générale ont une vocation informative, donc à contenu vague. Les apps doivent être condensées et optimisées au regard de l'interactivité.

Développement web pour mobile

- ❑ Les apps tournent en se chargeant dans le navigateur (Chrome, Safari et Firefox).
- ❑ Typiquement le développement repose sur l'utilisation des langages du web javascript, CSS et HTML5.
- ❑ Le web et les pages web en générale ont une vocation informative, donc à contenu vague. Les apps doivent être condensées et optimisées au regard de l'interactivité.

Développement web pour mobile

- ❑ Accès limité ou restrictif aux services système et aux périphériques (capteurs, réseaux, gestures, GPS, storage,
- ❑ Les progressives web apps viennent en guise d'atténuer ces limitations en poussant davantage la technologie des browsers pour satisfaire (le offline, l'accès aux périphériques, le storage, ...).

Développement web pour mobile

Check :

- ❑ BuildFire (No code)
- ❑ Mobile Angular [link_to_home_page](#)
- ❑ jQuery mobile its [home](#) and [sample](#)
- ❑ React Native this is a link to [learn basics](#) and [home_page](#)

Développement hybride pour mobile

- ❑ En partie développées en langages du web (javascript, CSS et HTML5,...) à l'instar des apps web. Alors qu'elles tournent dans des "web containers", font accès aux services système à travers des "run-time" ou manœuvres shell.
- ❑ Certaine nuance avec les cross-plateforme du fait qu'on est mesure de produire des apps tournant sur # plateformes.

Développement hybride pour mobile

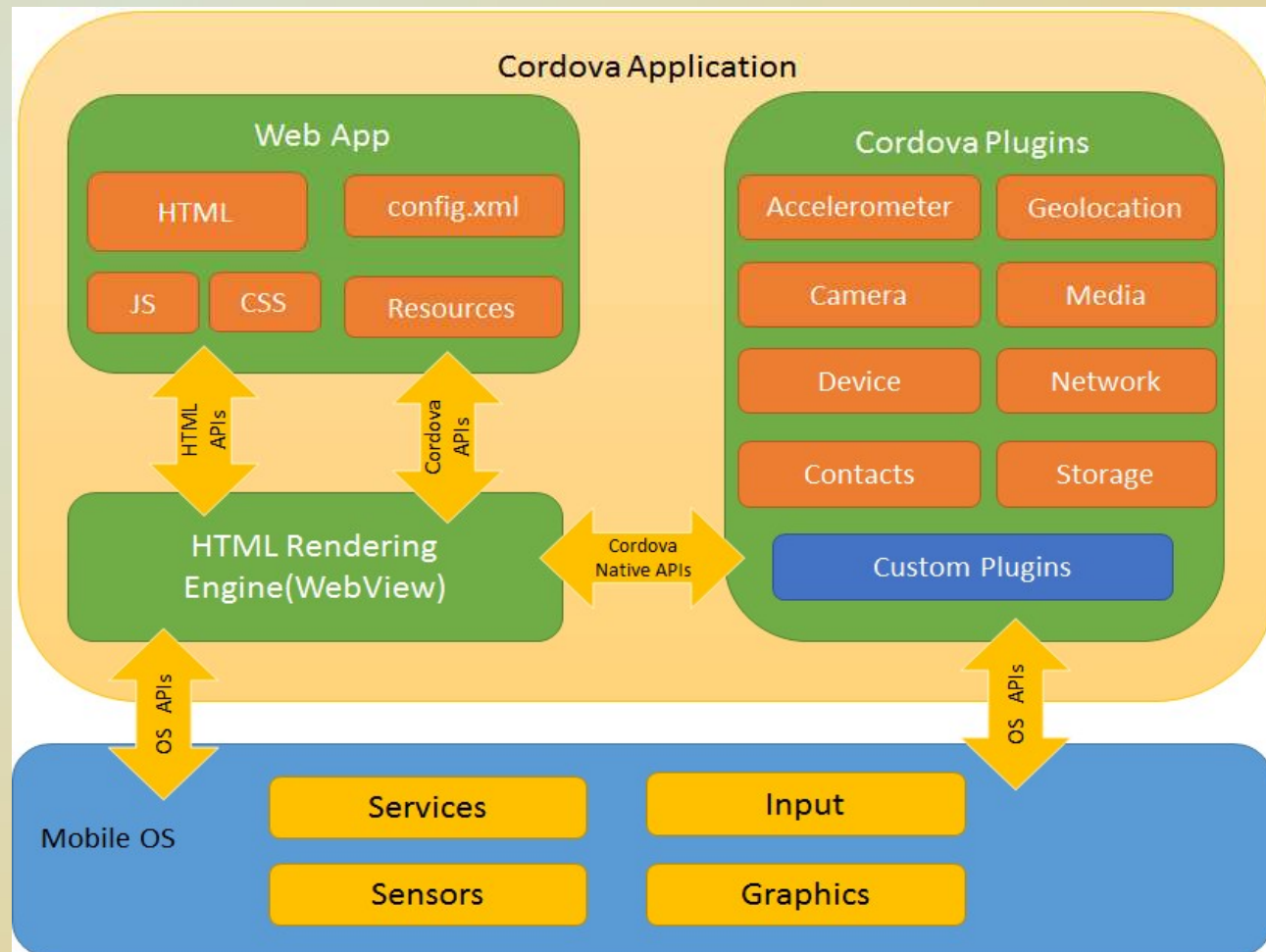
- ❑ On a moins de limites et restrictions en matière d'accès système.
- ❑ Phonegap, cordova, Sencha touch en sont des exemples.
- ❑ Apache cordova : un framework de développement mobile open source. Permet d'utiliser les technologies Web standard - HTML5, CSS3 et JavaScript pour le développement multiplateforme

Développement hybride pour mobile

- ❑ Apache cordova : Les applications s'exécutent dans des wrappers dédiés chacune à une plate-forme différente et s'appuient sur des liaisons d'API conformes à l'os cible pour accéder aux fonctionnalités de chaque appareil telles que les capteurs, les données, l'état du réseau, etc.
- ❑ Link [home_page](#)

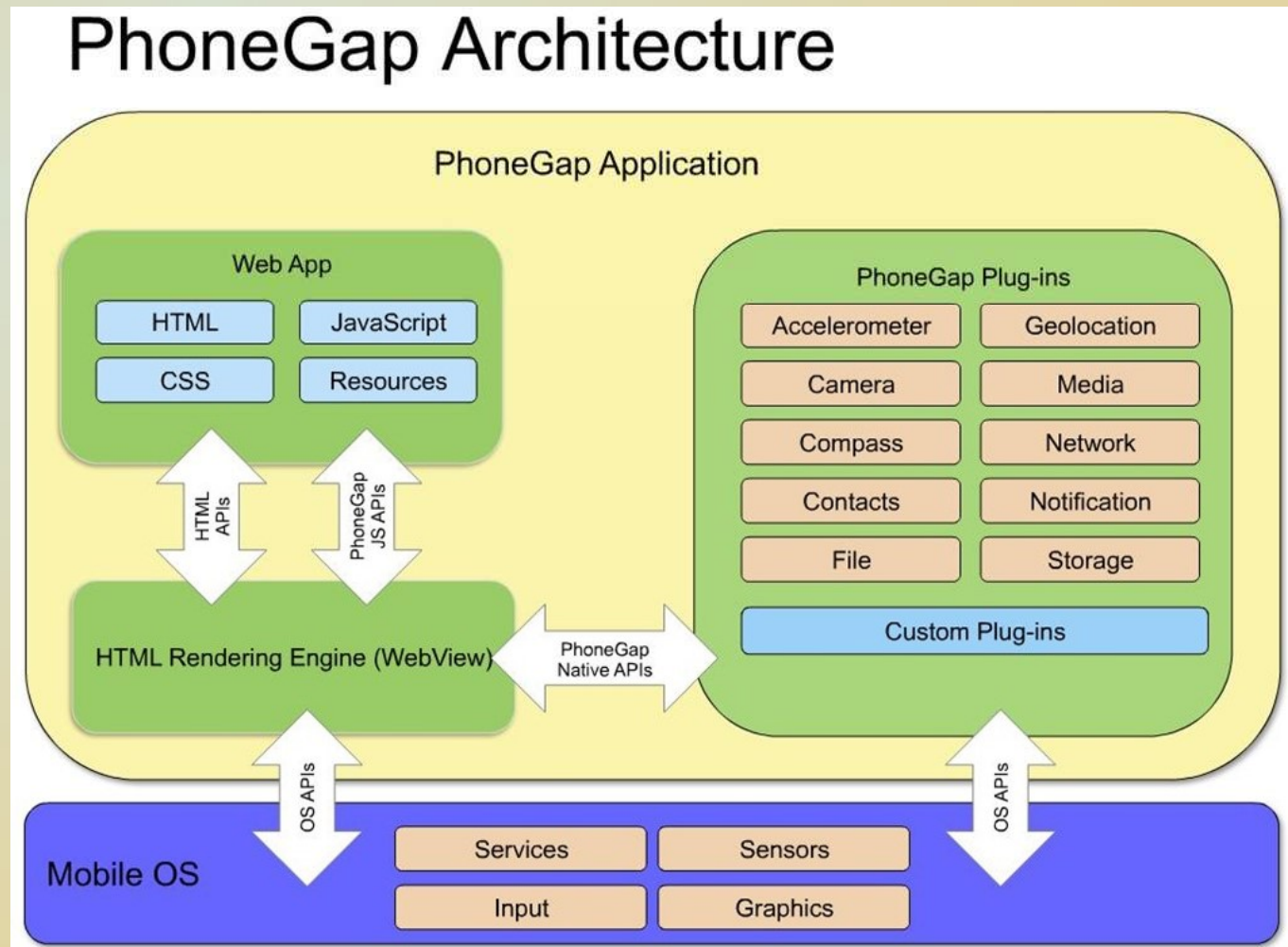
Développement hybride pour mobile

❑ Apache cordova : (architecture)



Développement hybride pour mobile

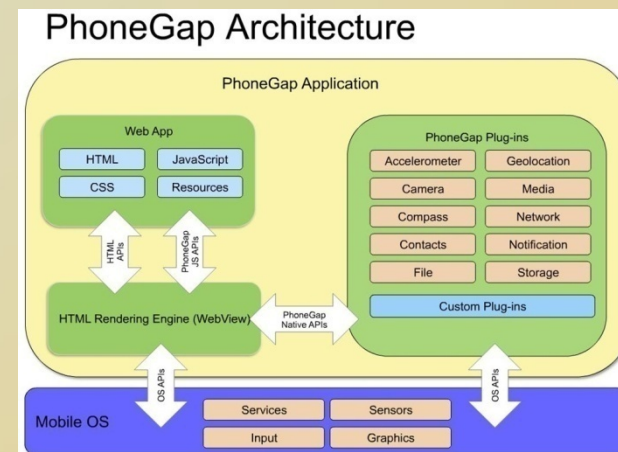
❑ Phonegap : (architecture)



Développement hybride pour mobile

❑ Phonegap : (architecture)

- ❑ L'entreprise Nitobi Software était à l'origine du projet PhoneGap. Ils seront par la suite racheté par Adobe.
 - ❑ En 2011, le projet PhoneGap est offert à la Foundation Apache.
- Il deviendra ce qui est aujourd'hui [Apache Cordova](#).



Développement hybride pour mobile

❑ Phonegap : (architecture)

- ▣ PhoneGap devient ensuite une distribution de Cordova et offre des services additionnels tels que PhoneGap Build qui permet la compilation de l'application mobile.
- ▣ Depuis le 1er octobre 2020, Adobe a interrompu le service de PhoneGap, PhoneGap Build et Apache Cordova. PhoneGap demeure gratuit et open source, mais n'est donc plus développé et l'outil est aujourd'hui obsolète en raison des mises à jour des systèmes d'exploitation d'Android, iOS et Windows Phone. PhoneGap Build n'étant pas open source, la distribution du logiciel a pris fin le 1er octobre 2020.

Développement pour mobile

exemples de frameworks

- ❑ **React native : est un framework open source créé par Meta Platforms (facebook).**

Extraits

du site React Native (Variante de code)

```
import React from 'react';
import {Text, View} from 'react-native';
import {Header} from './Header';
import {heading} from './Typography';
```

```
const WelcomeScreen = () => (
  <View>
    <Header title="Welcome to React Native"/>
    <Text style={heading}>Step One</Text>
    <Text>
      Edit App.js to change this screen and turn it
      into your app.
    </Text>
    <Text style={heading}>See Your Changes</Text>
    <Text>
      Press Cmd + R inside the simulator to reload
      your app's code.
    </Text>
```

Développement pour mobile

exemples de frameworks

- ❑ **React native** : est un framework open source créé par Meta Platforms (facebook).

Extraits
du site React Native
(affirmations)

Written in JavaScript – rendered with native code:

React primitives render to native platform UI, meaning your app uses the same native platform APIs other apps do.

Aussi : remarque concernant
l'installation suivre le [lien](#)

Développement pour mobile

exemples de frameworks

- ❑ Ionic : est un framework open source créé en 2013 par Max Lynch, Ben Sperry, et Adam Bradley. [Link...](#)

[Extraits](#)
du site
ionicframework.com

Ionic is an open source UI toolkit for building performant, high-quality mobile and desktop apps using web technologies — HTML, CSS, and JavaScript — with integrations for popular frameworks like [Angular](#), [React](#), and [Vue](#).

Développement pour mobile

exemples de frameworks

- ❑ Sencha touch : est une bibliothèque JavaScript d'interface utilisateur, ou framework Web, spécialement conçue pour le Web mobile.

Extraits
du site
sencha.com

Sencha Touch is no longer supported. It's core functionality for building mobile apps has been merged with Ext JS, which provides everything developers need to build data-intensive, cross-platform web and mobile applications.

Développement pour mobile

exemples de frameworks

- ❑ Flutter : est un kit de développement logiciel d'interface utilisateur open-source créé par Google. Il est utilisé pour développer des applications (cross-plateformes) : Android, iOS, Linux, Mac, Windows.
- ❑ Link .. [home_page](#)

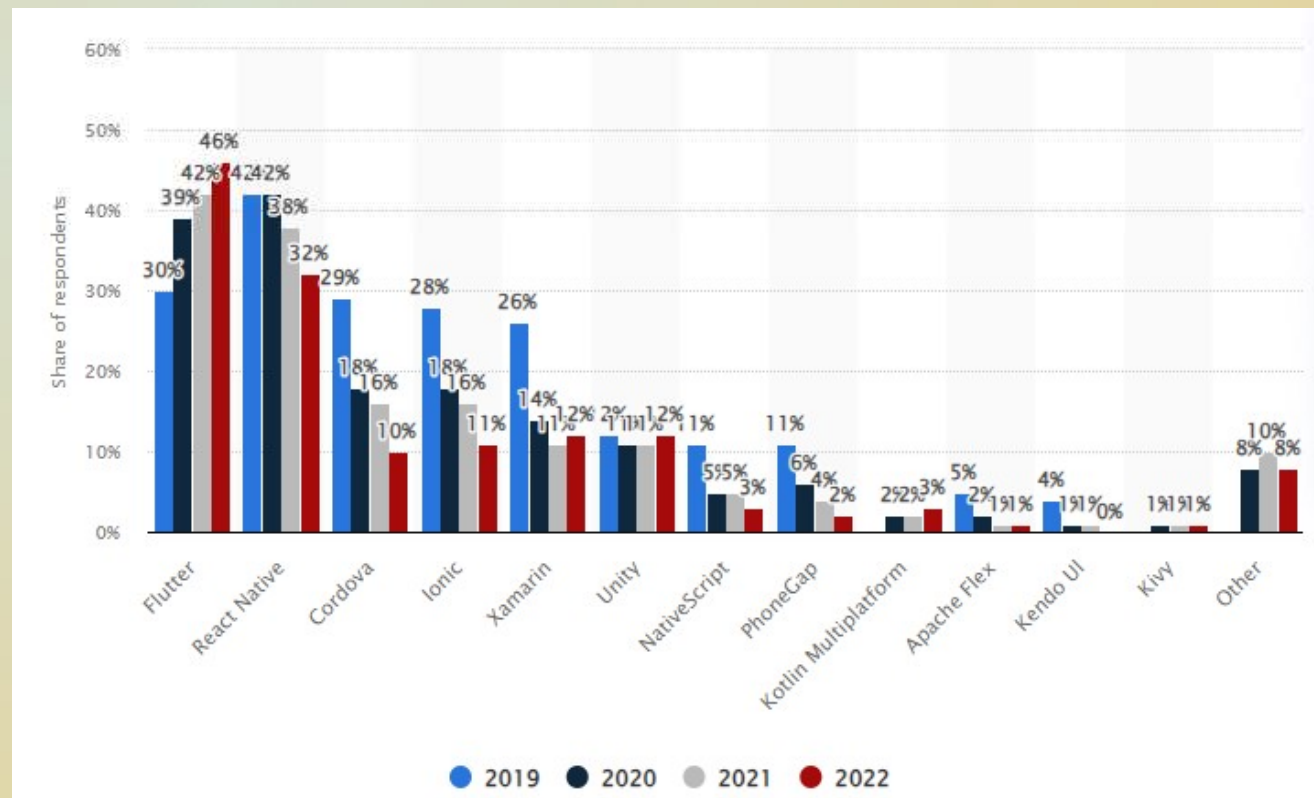
Développement pour mobile

exemples de frameworks

- ❑ Link to app_sample
- ❑ Link to touch in Dart

Développement pour mobile charts

▣ Cross-platform mobile frameworks usage



Développement pour mobile : critiques

- ❑ Les paramètres à considérer lors de la critique:
 - L'accès aux périphériques et les propriétés système.
 - Vitesse et rapidité
 - La dépendance de la plateforme.
 - La maintenance.
 - Installation
 - Expérience utilisateur.
 - Coût de développement.

Développement pour mobile : critiques

❑ Les "pros" et "cons" :

Avantages des applications natives:

- Sont considérées les meilleures du point de vue performance.
- Le support est aussi assez adéquat surtout si le kit de développement et la plateforme sont issues du même fournisseur.
- Plus intuitives et plus interactives.
- Accès non restreint aux périphériques et services système (capteurs, accéléromètre, camera, GPS, réseaux, gestes, storage, contacts,.....).

Développement pour mobile : critiques

❑ Les "pros" et "cons" :

Avantages des applications natives:

- L'expérience utilisateur est aussi meilleur et interface utilisateur native.
- La Discoverabilité est assuré via les stores. (Apple store et google play) et un niveau plus élevé d'exposition.
- L'approbation liée à la plateforme cible.
- Fonctionnement offline et inline.

Développement pour mobile : critiques

❑ Les "pros" et "cons" :

Controverses des applications natives:

- Question de coût.
- Question des langages.
- Le cas des applications simples (rentabilités).

Développement pour mobile : critiques

❑ Les "pros" et "cons" :

Avantages des applications hybrides/web:

- Portabilité sur #plateformes, ou bien on fait appel à des outils de développement web ou si bien on prévoit des bridges pour des solutions cross-plateforme.
- Sont (dites) à coût restreint.
- Discovrabilité sur #stores.

Développement pour mobile : critiques

❑ Les "pros" et "cons" :

Controverses des applications hybrides/web:

- Performances limités.
- Plus de dépendance dans le cas du développement hybride du fait de l'intégration des couches middleware tierces.
- Le coût peut augmenter plus drastiquement si on désire pousser davantage les optimisations.