

## **TP #3 : Création d'une application**

Nous avons vu jusqu'à présent comment créer et/ou adapter un site internet aux terminaux nomades. Nous allons voir maintenant qu'il est possible de développer une application mobile et/ou tablette à l'aide des technologies du web.

### **1. Préambule**

Si vous disposez d'un téléphone sous Android (avec une version assez récente), vous pourrez sans problème vous en servir pour tester tout au long de ce TP (dans vos paramètres, activez le mode développement).

Si vous avez un iPhone, il faudra utiliser une tablette pour effectuer vos tests. En effet, normalement on doit utiliser un Mac pour le développement (via Xcode) et une licence Apple pour pouvoir tester sur son appareil. On pourrait trouver des solutions alternatives, mais ce n'est pas le but du TP.

Si vous êtes sous Windows Phone, vous n'avez pas de chance :) Plus sérieusement je n'ai jamais eu l'occasion de tester.

### **2. Cordova**

Pour pouvoir créer une application mobile/tablette grâce à du HTML/CSS/JS, il faut utiliser un framework, en effet par défaut, les applications natives sur Android sont développées en Java, en Objective-C sur IOS...

Il existe plusieurs framework, comme Titanium ou Ionic, on utilisera ici Cordova qui est une solution libre (licence Apache).

Grâce à Cordova nous pourrions accéder en Javascript à diverses fonctionnalités de votre terminal (contacts, batterie, géolocalisation, vibreur, ...)

Les avantages d'utiliser une telle solution sont nombreux :

- Une seule version du code pour plusieurs plateformes : le temps de développement est ainsi moins long
- La maintenance est plus facile
- On a pas besoin d'apprendre de nouveaux langages
- Ou utilise des technologies du web, c'est cool le web :)

Par contre, on note trouve quelques inconvénients :

- Même si elles sont en progrès, les performances restent moins bonnes
- Certaines fonctionnalités du smartphone ne sont pas accessibles

### 3. C'est parti

Le but de ce TP est de réutiliser le site ce que vous avez réalisé à l'exercice 5 du TP précédent et grâce à Cordova d'en faire une application fonctionnelle sur le système d'exploitation de votre choix. Dans un 2ème temps, vous enrichirez votre application de fonctionnalités diverses et variées.

#### Sur Windows :

1. Téléchargez et installez NodeJs <https://nodejs.org/en/download/>
2. Ouvrez le terminal et entrez les commandes suivantes avec vos infos :
  - npm config set proxy http://nom.prenom:[motdepasse@10.10.28.5:3128/](#)
  - npm config set https\_proxy  
http://nom.prenom:[motdepasse@10.10.28.5:3128/](#)
3. Entrez la commande : npm install -g cordova

4. Enfin testez la commande : `cordova -v`
5. Si la commande précédente retourne la version, vous pouvez passer à la suite
6. Créez le fichier `~/.gradle/gradle.properties` et mettre à l'intérieur les lignes suivantes (en renseignant vos infos):
7. 

```
systemProp.http.proxyHost=10.10.28.5
systemProp.http.proxyPort=3128
systemProp.http.proxyUser=nom.prenom
systemProp.http.proxyPassword=motdepasse
systemProp.https.proxyHost=10.10.28.5
systemProp.https.proxyPort=3128
systemProp.https.proxyUser=nom.prenom
systemProp.https.proxyPassword=motdepasse
```

### **Sur Linux :**

1. Ouvrez le terminal et entrez : `sudo su`
2. entrez la commande suivante avec vos infos :
  - `export http_proxy=http://nom.prenom:motdepasse@10.10.28.5:3128/`
3. entrez la commande : `apt-get update`
4. puis la commande : `apt-get install npm`
5. ensuite la commande `npm install -g cordova`
6. enfin : `ln -s /usr/bin/nodejs /usr/bin/node`
7. testez la commande : `cordova -v`
8. Si la commande précédente retourne la version, vous pouvez passer à la suite
9. Créez le fichier `~/.gradle/gradle.properties` et mettre à l'intérieur les lignes suivantes (en renseignant vos infos):

```
systemProp.http.proxyHost=10.10.28.5
systemProp.http.proxyPort=3128
systemProp.http.proxyUser=nom.prenom
systemProp.http.proxyPassword=motdepasse
systemProp.https.proxyHost=10.10.28.5
systemProp.https.proxyPort=3128
systemProp.https.proxyUser=nom.prenom
systemProp.https.proxyPassword=motdepasse
```

**ATTENTION : Il faut absolument que le chemin absolu vers votre futur projet ne contienne aucun caractère spécial, accentué et aucun espace sinon vous serez confronté à des problématiques lors de l'installation.**

Je ne vais pas vous détailler ici la création de votre première application sous Cordova car celle-ci dépend de votre système d'exploitation. La documentation est très bien faite, je vous invite à la suivre minutieusement (appelez votre projet tp3 au lieu de hello) et pensez à vous placer à un endroit judicieux car la création de l'application va engendrer la création d'un répertoire parent :

<https://cordova.apache.org/docs/en/latest/guide/cli/index.html>

Je vous conseille de faire une application sous android car vous devez déjà disposer de tout le nécessaire (Android studio, SDK ...) étant donné que vous avez fait du développement Android au cours de l'année. Pour vérifier si vous disposez des dépendances nécessaires, entrez la commande :

```
Cordova requirements
```

Si ce n'est pas le cas, il faut les installer, pour Android :

<https://cordova.apache.org/docs/en/latest/guide/platforms/android/index.html>

Enfin vous pouvez compiler votre projet et le tester grâce au simulateur en lançant la commande

```
cordova run
```

Vous avez créé votre première application sous Cordova :)

Si vous avez l'erreur suivante :

**Error: Could not find gradle wrapper within Android SDK. Might need to update your Android SDK.**

**Looked here: /Users/olivier/android/sdk/tools/templates/gradle/wrapper**

Appelez moi.

## 4. Transformation de votre site en application

Importez maintenant votre exercice du TP précédent dans le dossier `www/` et faites le nécessaire pour qu'il devienne une application.

N'oubliez pas de compiler après chaque modification pour pouvoir visualiser le résultat.

```
cd /votre_repertoire_de_travail  
cordova build  
cordova run
```

Si vous utilisez Android studio, vous pouvez ouvrir directement le dossier `/votre_repertoire_de_travail/platforms/android` les fichiers compilés viendront se placer ici.

Astuce : dans votre fichier `index.html`, importez le script `cordova.js` de la manière suivante (avant la fermeture de la balise `</body>`) pour disposer des fonctionnalités de Cordova. En effet même si vous ne le visualisez pas dans votre arborescence, ce fichier est présent une fois que vous avez compilé votre projet.

```
<script type="text/javascript" src="cordova.js"></script>
```

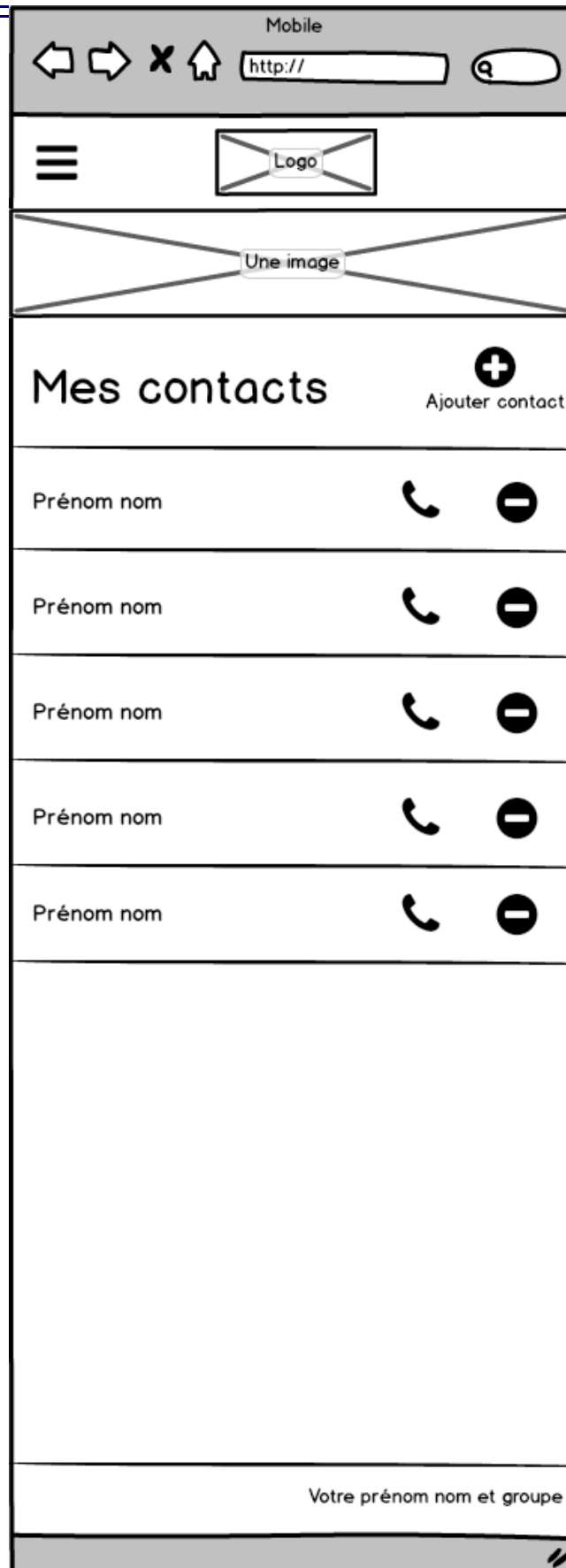
## 5. Ajout de fonctionnalités

Cordova dispose de très peu de fonctionnalités de bases afin d'éviter de charger des choses inutiles. Grâce aux différents plugins, vous devrez ajouter les fonctionnalités suivantes, une partie de la note prendra en compte le style de votre application ainsi que sa compatibilité avec d'autres terminaux (pensez au responsive design).

- Mettre une icône pour votre application.
- Ajouter une image qui va apparaître pendant le chargement de votre application.
- Tester si votre appareil dispose d'une connexion internet, si c'est le cas, afficher le type de connexion, sinon, affichez un message d'erreur.
- Afficher les informations suivantes de votre terminal:
  - Le modèle
  - la langue
  - le système d'exploitation et sa version
  - l'UUID
  - Afficher l'état (sous forme de pourcentage) de votre batterie, prévoir un message d'alerte si celle-ci est critique.
- Changer l'apparence de la status bar de votre terminal une fois votre application lancée.

## 6. Répertoire

Créer une nouvelle page sur votre application (n'oubliez pas de l'ajouter à votre menu). Pour la première maquette ci-dessous, il faut que vous récupériez la liste des contacts de votre terminal, pour chaque contact vous devez avoir la possibilité de l'appeler directement ou de le supprimer (pour la suppression, une alerte doit être affichée avant la suppression définitive).



*Astuce : Si vous disposez de Chrome, quand votre terminal est connecté, vous pouvez entrer l'adresse suivante : <chrome://inspect/> cela vous permettra d'utiliser l'inspecteur d'éléments directement sur votre terminal. Pour pouvoir utiliser des alertes dans la console, il faut au préalable avoir installé le plugin cordova « console » : <https://cordova.apache.org/docs/en/latest/reference/cordova-plugin-console/> On doit également pouvoir ajouter un contact avec les informations suivantes : Nom, prénom, numéro de téléphone, adresse et email.*

Dans un second temps, lorsque vous sélectionnez un contact, sa fiche apparaît, celle-ci doit contenir les informations et les fonctionnalités présentes dans la maquette ci-dessous :

