

Informatique Mobile : Fiche 11

Objectifs

- Créer une interface mobile native qui utilise un serveur Web dans son fonctionnement

Projet PAE

Dans le cadre du projet de PAE, vous avez dû développer un serveur Web de type Single Page Application. En particulier vous avez dû restreindre le travail actif avec le serveur à des requêtes AJAX utilisant du JSON comme manière d'échanger les données. Du coup il devient relativement aisé d'utiliser le serveur à partir d'autre chose qu'un navigateur. C'est ce que nous allons faire ici.

HTTPUtils

Les classes fournies nativement par Java pour effectuer des requêtes Web ne sont pas des plus simples à utiliser. Vous trouverez sur l'eCampus la classe HTTPUtils. Vous y trouverez notamment :

`String utils.HTTPUtils.performPostCall(String url, Map<String,String> params)`

- Envoie l'équivalent d'un appel Ajax à l'URL url.
- Les paramètres sont spécifiés dans la map params (null si aucun).
- Renvoie soit :
 - La réponse si le code HTTP est OK (200)
 - Un `utils.HTTPUtils.HTTPException` si le code HTTP \neq 200 ; `getMessage()` donne la réponse renvoyée et `getStatusCode()` le code HTTP.
 - Un `utils.HTTPUtils.HTTPNetworkException` si l'URL est mal formée ou qu'il y a un problème avec le réseau ; `getCause()` renvoie l'exception qui a réellement été jetée.

Il y a aussi `performGetCall` qui est similaire.

Attention : ces méthodes sont synchrones, elles bloquent en attendant la réponse. A vous de rendre ces appels asynchrones à l'aide d'une `AsyncTask`.

Authentification

Jetty utilise le cookie `JSESSIONID` pour gérer les sessions avec le front-end. C'est sur base de cette session que l'utilisateur sera identifié, il faudra donc retenir ce cookie. De plus vous utilisez vous-même un cookie qui signe avec JWT un jeton d'authentification. Ce cookie devra donc être aussi retenu. Par défaut, `URLConnection` ne gère pas les cookies, mais ceci s'ajoute très simplement :

```
CookieManager cookieManager = new CookieManager();  
CookieHandler.setDefault(cookieManager);
```

Remarque : Jetty ne crée un cookie de session que si le back-end appelle `getSession()` sur son `HttpServletRequest`.

Mini-projet en collaboration avec PAE

Lancez votre serveur. Regardez l'adresse IP de la machine Windows qui l'exécute : c'est l'adresse de votre serveur pour l'appareil Android.

Créez une application Android. La première activité permettra à un utilisateur de s'authentifier.

La seconde activité utilisera le principe du Master Details. La ListView contiendra l'ensemble des journées d'entreprises, triées de la plus récente à la plus ancienne. Le détail de l'entreprise sélectionnée affichera sa date ainsi que les entreprises qui sont confirmées pour cette journée.

S'il vous reste du temps, enrichissez cette solution : par exemple sélectionner une entreprise listera les contacts de cette dernière.

Si vous n'avez pas le cours de PAE, nous allons reprendre l'exercice de Master Details de la semaine passée : à la place d'avoir l'unique Details contenant la WebView de l'artiste, on aura deux onglets. Le premier onglet contiendra les informations « bio » de l'artiste, tandis que le second onglet contiendra ses albums. De plus, cliquer sur un album ouvrira une troisième activité qui liste les chansons de cet album.