

Document d'architecture technique (DAT) JCertif Mob

1. Principe de base

L'application devra être conçue sur deux principes :

- C'est à destination de la plateforme Android, donc devra répondre aux contraintes de l'embarqué (cache et sécu).
- Générique. L'objectif étant de faire de JCertif App, l'application pour l'organisation de conf, qui tout le monde pourra télécharger, alors le client mobile devra être suffisamment générique et paramétrable pour être utilisée par d'autres instances de JCertif App.

2. Fonctionnalités à implémenter

2.1. Affichage des informations de base

L'application devra être pensée générique afin de permettre une utilisation par d'autres installations de JCertif App. On permettra aux utilisateurs de JCertif Mob de fournir l'url de base des services. On pourra ainsi, afficher des informations basiques d'une conf : nom de la conf, lieu, dates, texte de présentation

Prérequis :

1. Avoir un service rest qui permet de récupérer un objet conférence (nom, dates, description, lieu)

2.2. Lister des speakers

Cette fonctionnalité permet d'afficher la liste des speakers avec une petite description (sur une ligne). En cliquant sur un speaker, on affiche une description plus complète. La liste des speakers est à stocker dans la base de données locale (sqlite). On pourra mettre en place un mécanisme qui périodiquement, 1 fois par jour par exemple, va mettre à jour la liste de speakers locale avec celle remote.

Prérequis :

1. Avoir un service rest qui permet de récupérer l'ensemble des speakers

2.3. Calendrier global avec détails par session

Comme dans l'application web, avoir un calendrier global de toutes les sessions. On doit avoir la possibilité d'afficher le détail.

Prérequis :

1. Avoir un service rest qui permet de récupérer la liste des sessions (date, short description, long description, speaker)

2.4. Inscription et connexion

Permettre à un utilisateur de s'inscrire puis se connecter. On stocke les informations de connexion en local (SharedPreferences)

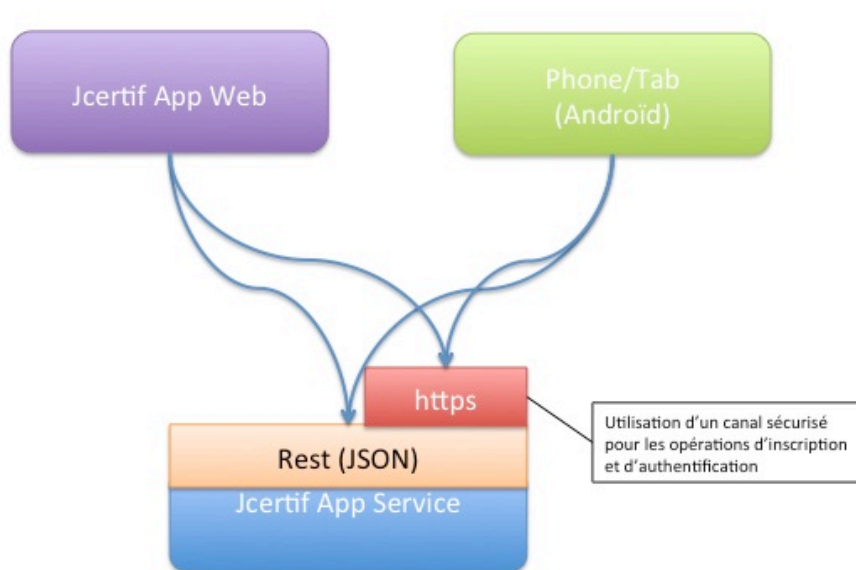
2.5. Inscription à des sessions

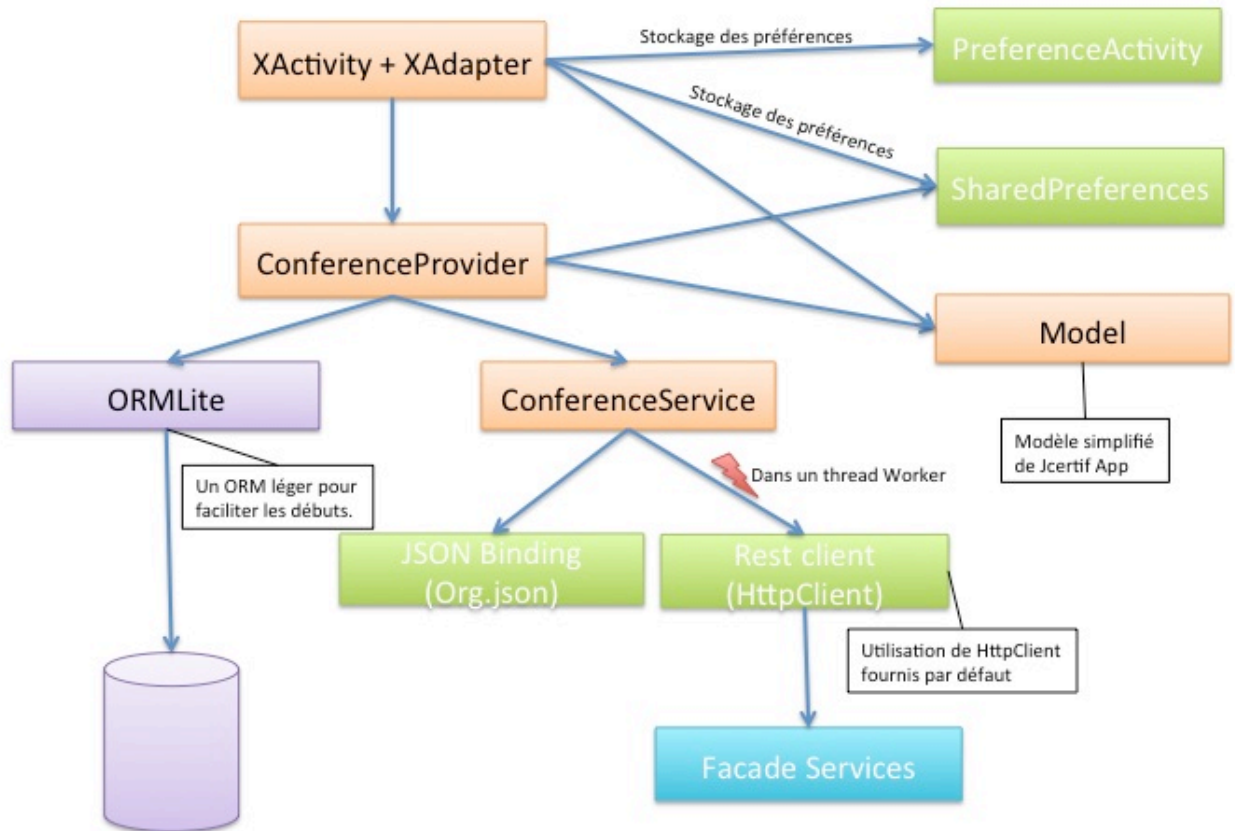
Permettre à un utilisateur de s'inscrire à une session.

2.6. Affichage d'un calendrier personnel

Idem que 2.3 sauf qu'ici ceux sont uniquement les sessions auxquelles l'utilisateur participe qui sont affichées.

Architecture technique





Environnement de développement

2. EDI & Settings

On va utiliser Eclipse comme IDE avec le plugin ADT (Android Development Tools). Il est conseillé qu'on ait tous la même version d'Eclipse. Je préconise la dernière Helios. Pour ceux qui encore leur install utilisée pour JCertif App, ils peuvent rester dessus et juste rajouter le plugin ADT

Eclipse : <http://www.eclipse.org/downloads/packages/eclipse-ide-java-developers/heliossr2>

Plugin ADT, utiliser le Repository : <https://dl-ssl.google.com/android/eclipse/>

Il faudra également installer un **plugin SVN** pour eclipse : je conseillerai Subclipse

Une fois l'installation terminée, il faut créer un émulateur Android. La version cible sera la 2.3.

Pour plus de détails sur l'environnement Android : <https://dl-ssl.google.com/android/eclipse/>

Plugin maven :

- installer **M2Eclipse** : <http://m2eclipse.sonatype.org/>
- installer le plugin **Maven Android Plugin** (obligatoire pour fonctionner <http://code.google.com/p/maven-android-plugin/wiki/EclipseIntegration>)

3. Découpage de l'application (maven et package)

Pour l'instant, nous avons un projet maven avec 2 modules :

- project-jcertifmobile-app
- project-jcertifmobile-test
- avec un pom parent à la racine

I usually separate stuff into sub-packages in my apps.

- com.jcertif.android.app : contient les classes de base et les fonctionnalités globales de l'application
- com.jcertif.android.net - network stuff, network related utils
- com.jcertif.android.data - db helpers, providers, etc
- com.jcertif.android.model - object model

4. Repo

<https://project-jcertif-mobile.googlecode.com/svn/trunk/>

5. Plateforme d'intégration continue

(en cours)

6. Quelques liens utiles

<http://code.google.com/p/maven-android-plugin/wiki/GettingStarted>

<http://code.google.com/p/maven-android-plugin/wiki/EclipseIntegration>

<http://developer.android.com/guide/index.html>

<http://developer.android.com/guide/practices/design/performance.html>

http://ormlite.com/sqlite_java_android_orm.shtml