

## Développement d'applications Android

Durée: 4 jours

2040 €

12 au 15 février  
14 au 17 mai

27 au 30 août  
19 au 22 novembre

### Public:

Développeurs et intégrateurs d'applications sous Android.

### Objectifs:

Connaître les principes de fonctionnement et savoir développer des applications sous Android

### Connaissances préalables nécessaires:

Connaissance de la programmation en java.

### Programme:

- Introduction** : Présentation du système d'exploitation Android, des plateformes matérielles, des outils : SDK, android market, etc .. Architecture : la couche noyau Linux, l'environnement de développement, machines virtuelles, base de données SQLite  
Les bibliothèques de base.
- Applicatif** : Les principales applications existantes.  
Installation/désinstallation d'une application.  
Arborescence des fichiers.
- Développement** : Présentation du SDK pour le développement en java, installation. Utilitaires : émulateurs, simulateur de carte.  
Développement d'une application de base.  
Structure générale des applications. Les quatre types de composants applicatifs : Activity, Services, Broadcast receivers, Content providers.  
Cycle de vie des composants.  
Les versions d'android et du SDK : propriétés, adaptation aux matériels (smartphones, tablettes, télévisions, etc ...)  
Les outils de développement : Android Studio IntelliJ, AVD : Android Virtual Device, ADB : Android Debug Bridge, DDMS : Dalvik Debug Monitor Server  
Développement d'une première application.

## Développement d'applications Android

- Interface utilisateur** : Définitions. Présentation des layouts.  
Récupération du contexte applicatif. Les widgets.  
Gestion des menus. Boîtes de dialogue. Thèmes. Notifications (Toast, Status Bar, Dialog).
- Interaction d'applications** : Présentation des Intents.  
Le fonctionnement des services. L'interrogation de WebServices RESTfull (client http, json)  
Les bonnes pratiques.
- Persistance des données** : Stockage des préférences utilisateur. Le système de fichiers.  
Mise en oeuvre de SQLite.
- Graphique et multimédia** : Développement 2D. Les APIs. Utilisation du MediaPlayer.
- API** : Accès réseau, accès au système de fichiers.  
Capteurs internes. Gestion des périphériques : carte son, écran, caméra, clavier,..  
Mise en oeuvre de la classe Sensor.