

Stage pratique de 4 jour(s)
Réf : IOD

Participants

Développeurs Android.

Pré-requis

Très bonnes connaissances du langage Java. Expérience demandée dans le développement Android.

Modalités d'évaluation

L'évaluation des acquis se fait tout au long de la session au travers des multiples exercices à réaliser (50 à 70% du temps).

Compétences du formateur

Les experts qui animent la formation sont des spécialistes des matières abordées. Ils ont été validés par nos équipes pédagogiques tant sur le plan des connaissances métiers que sur celui de la pédagogie, et ce pour chaque cours qu'ils enseignent. Ils ont au minimum cinq à dix années d'expérience dans leur domaine et occupent ou ont occupé des postes à responsabilité en entreprise.

Moyens pédagogiques et techniques

- Les moyens pédagogiques et les méthodes d'enseignement utilisés sont principalement : aides audiovisuelles, documentation et support de cours, exercices pratiques d'application et corrigés des exercices pour les stages pratiques, études de cas ou présentation de cas réels pour les séminaires de formation.

- A l'issue de chaque stage ou séminaire, ORSYS fournit aux participants un questionnaire d'évaluation du cours qui est ensuite analysé par nos équipes pédagogiques.

- Une feuille d'émargement par demi-journée de présence est fournie en fin de formation ainsi qu'une attestation de fin de formation si le stagiaire a bien assisté à la totalité de la session.

Android, perfectionnement

Vous aborderez dans ce cours les aspects avancés du développement d'applications Android portant sur la conception d'IHM ou l'utilisation de capteurs. Vous apprendrez à contrôler les fonctionnalités avancées d'Android Studio (build Gradle, test) et à utiliser les bibliothèques et services tiers utiles au développement.

OBJECTIFS PEDAGOGIQUES

Découvrir les outils de développement liés à Android Studio

Appréhender le développement des fonctionnalités avancées liées aux IHM, capteurs, services et threads

Intégrer des bibliothèques tierces à un projet Android

Mettre au point, tester et optimiser une application Android

1) Introduction

2) Outils avancés de développement

3) Création d'IHM avancées

4) Utilisation des capteurs

5) ContentProvider et Services

6) Tester une application Android

7) Bibliothèques et services utiles pour le développement Android

Travaux pratiques

Les travaux pratiques seront réalisés en Java.

1) Introduction

- Rappels des principes de base Android. Présentation du développement natif avec NDK. JNI.
- Présentation des fonctionnalités avancées Android Studio (Instant Run, debug, profiling...).
- Apports du dernier SDK. Les permissions à la demande.

2) Outils avancés de développement

- Paramétrer le build avec Gradle. Comprendre les Build Variants.
- Améliorer son code-source avec Lint.
- Mettre au point et profiler/monitorer une application.
- Optimisation de l'APK avec ProGuard.

Travaux pratiques

Utilisation des différents outils.

3) Création d'IHM avancées

- Construction d'IHM avancées suivant les préconisations Material Design. Utilisation des styles.
- Mécanismes des widgets.
- Présentation OpenGL/ES.

Travaux pratiques

Réalisation d'une activité complexe avec des fragments. Réalisation d'une application graphique.

4) Utilisation des capteurs

- Mise en œuvre de capteurs. Accéléromètre, proximité, température, luminosité, baromètre, GPS...
- Paramétrage dans le simulateur des capteurs.
- Utilisation d'outils pour la simulation d'événements.

Travaux pratiques

Réalisation d'une application intégrant différents capteurs.

5) ContentProvider et Services

- Créer un ContentProvider. Cycle de vie des services. Arrière-plan et premier plan. Lier services et activités.
- Utiliser des threads depuis un service. Utiliser des Intents. Définir des alarmes.

Travaux pratiques

Réalisation d'un service et contrôle depuis une activité.

6) Tester une application Android

- Présentation des outils adaptés aux différents types de tests. Rappels sur JUnit.
- Simulation d'interactions utilisateur avec Espresso.
- Paramétrage d'Android Studio pour les tests. Utilisation de Cloud Test Lab.
- Présentation de Firebase Test Lab for Android.

Travaux pratiques

Mise en œuvre de tests avec Espresso.

7) Bibliothèques et services utiles pour le développement Android

- Utiliser les Google Play Services.
- Intégrer des bibliothèques tierces à un projet Android.
- Simplifier l'accès à des ressources REST avec Retrofit.
- Maîtriser le chargement des images avec Picasso.
- L'injection de dépendances (Dagger).

Travaux pratiques

Sélection et mise en place de librairies.