

Mis à jour le 28/07/2023

S'inscrire

# Formation Android Studio avec Android 12

5 jours (35 heures)

## Présentation

Android Studio est un environnement de développement qui permet développer de façon professionnelle des applications mobiles Android natives. Il est à l'origine basé sur IntelliJ IDEA et utilise Gradle, un moteur de production codé en Java. Cet IDE/EDI est compatible avec les systèmes d'exploitation Windows, macOS et Linux. Depuis la sortie d'Android Studio, l'initiative Project Marble s'est étendue sur plusieurs versions qui se concentrent sur l'amélioration de trois domaines principaux de l'IDE : la stabilité du système, l'amélioration et l'uniformité des fonctionnalités et bien évidemment la correction des bugs.

Cette formation s'adresse à tous les développeurs qui souhaitent amorcer un projet mobile de qualité professionnel sous Android. Vous découvrirez tout au long de cette formation pratique comment exploiter les toutes dernières fonctionnalités mises en oeuvre par les équipes de Google à destination des Tablets, Smartphone ou tout autre système embarqué compatible. Créez des applications Android en utilisant la suite d'outils Android Studio, un environnement de développement intégré (IDE) pour les développeurs Android utilisant les API Java. Nous créerons, avec toutes les bonnes pratiques de développement, une application mobile native, from scratch jusqu'à son déploiement en production avec sa publication dans la marketplace Play Store.

Avec cette formation Android, vous apprendrez les outils les plus récents et les plus productifs de l'écosystème Android, vous assurant un développement rapide d'applications mobile avec un effort minimal de votre part. Nous couvrirons les sujets principaux d'Android tels que les notifications et les toasts, intents & broadcast receivers, ainsi que les services associés. Ensuite, vous apprendrez comment publier vos applications et les vendre en ligne dans la boutique Google Play.

Vous connaîtrez ainsi grâce aux retours d'expérience de notre expert, les principaux outils, les best practices, les ressources indispensables pour toujours continuer à devenir un expert !

Comme pour toutes nos formations, celle-ci vous présentera la toute dernière version d'Android SDK (à la date de rédaction de l'article : [Android 12 API level 31](#)).

## Objectifs

- Installer et configurer son environnement de développement
- Créer en mode natif des interfaces graphiques mobiles modernes
- Découvrir les nouvelles fonctionnalités de l'IDE Android Studio
- Exploiter la puissance de l'API Level 31 d'Android 12
- Déployer votre application sur la marketplace de Google Play Store
- Utiliser une base de données offline et connecter votre mobile App à des Web Services

## Public visé

- Développeurs, Lead Developer.

## Pré-requis

- Expériences et Connaissances préalables en Java recommandées.

## Pour aller plus loin

- Découvrez notre nouvelle [formation sur le nouveau langage Kotlin](#)
- Plus de secret pour vous avec notre [formation avancée sur Android Embarqué](#) ou notre [formation Android](#)
- Si vous souhaitez passer au développement mobile hybride, nous vous conseillons :
  - [Flutter](#) de Google
  - [Formation Ionic](#)
  - [Formation Xamarin](#) de Microsoft

## Programme de Notre formation Android Studio avec Android 12

### Installation d'Android Studio

- Installation
- Installer un SDK
- App Designer
- App Preview
- Workspace Structure
- Editor & Tool windows
- The Android API level/NDK
- Virtual devices
- Instant Run
- Importer un projet dans Android Studio

### Android SDK

- Introduction
- Présentation des différentes versions
- Cycle de vie d'une application Android
- Éditeur graphique d'interface
- Les différents composants graphiques
- Interaction composant/code
- Utilisation des Assets dynamique/Resourçes Statique
- Les permissions
- Les layouts
- Navigation entre différents écrans
- Les listes
- Les custom views
- Menu
- Préférence

## Choisir son langage : Language Support

- Kotlin support
  - Avantages de Kotlin
  - Extending Kotlin
- C/C++ support
  - The NDK
- Kits de développement
- Créer un projet Things

## Premiers Pas en Android

- Création et organisation d'un projet
- Introduction au développement d'une vue
- Styles et ressources
- Lancer l'application sur votre téléphone (ou émulateur)
- Mobile vs Tablette
- Internationalisation d'une application

## Accéder aux APIs matérielles

- Géolocalisation
- Accéléromètre
- Système de fichiers
- Caméra
- Média
- Bluetooth

## Device Development

- Screen orientation
- Virtual devices
  - Layout and image qualification
  - Scale and platform
  - Alternative emulators
- Android Wear
  - Connecting to a wearable AVD
  - Wearable layouts
  - The WatchViewStub class
  - Shape - aware layouts
- Accessing sensors
  - Sensor emulation
- Device monitoring
  - Project templates
  - Monitoring and profiling

## Ressources

- Assets & Resources
- Asset Studio
  - Image Asset Studio
  - Layered icons
  - Launcher icon tools
  - Vector Asset Studio
  - Vector drawables
- Animation d'icônes
  - Autre type animations
- General drawables
- Palette library

## UI Design & Development

- Lifecycle, ViewModel, LiveData, Room
- The Layout Editor
- Linear & relative layout classes
- The constraint layout
- Création ConstraintLayout
  - Applying constraints
- Graphic Properties tool
- Alignement : Baseline alignment
- Controlling position
- Toolbar
- Multiple screen previewing
  - Hardware profiles
  - Virtual storage
- Material design

- Android styles
  - Material themes
  - The Theme Editor
  - XML fonts
- The design library
  - Coordinator layout
  - Snackbars & Floating Action Buttons
- Collapsing app bars
  - Raw text resources
  - The percent library

## Accès aux données

- Parseur de données JSON
- Sauvegarder des données en local / offline
- Synchronisation
- Accès à des données distantes
  - Détecter et utiliser le réseau
  - Créer une connexion HTTP
  - Stratégie pour mettre en cache des données distantes

## Accès aux données

- Templates and Plugins
  - Project templates
  - Navigation Drawer template
  - Structure Explorer
  - Class inspection plugins
- The Master/Detail Flow template
- Personnalisation : Custom templates
- Third-party templates
- Project samples
- Exemple de plugins tiers
  - ADB Wi-Fi
  - Codota

## Bonnes pratiques

- Utilisation de bibliothèques / composants externes (Dagger, Glide, Rx....)
- Design
- Architecture MVP
- Éviter les fuites de mémoire

## Outils

- Genymotion
- Leak canary
- Fabric
- PostMan

## Debugging & test

- Débugger le code Android à l'aide des outils de Studio
- Débugger les apps Mobile Web, et via l'API Android
- Monitorer les performances
- Logcat filters
- JUnit testing
  - Local unit tests
- Testing a UI
  - Testing views
  - Testing lists and data
  - Recording tests
  - A distance : Remote testing
  - Stress testing
- Performance monitoring
  - CPU profiling
  - Memory profiler
  - Network profiler

## Publication

- Déployer et Publier sur la marketplace de Google Play Store
- Gradle build configurations
- Command-line options
- Product flavors
- APK analysis
- Publishing applications
- Signer son App
- Gérer le keystores
- Google Play app signing
- Gérer l'automatisation : Automatic signing

## Sociétés concernées

Cette formation s'adresse à la fois aux particuliers ainsi qu'aux entreprises, petites ou grandes, souhaitant former ses équipes à une nouvelle technologie informatique avancée ou bien à acquérir des connaissances métiers spécifiques ou des méthodes modernes.

## Méthodes pédagogiques

Stage Pratique : 60% Pratique, 40% Théorie. Support de la formation distribué au format numérique à tous les participants.

## Organisation

Le cours alterne les apports théoriques du formateur soutenus par des exemples et des séances de réflexions, et de travail en groupe.

## Validation

À la fin de la session, un questionnaire à choix multiples permet de vérifier l'acquisition correcte des compétences.

## Sanction

Une attestation sera remise à chaque stagiaire qui aura suivi la totalité de la formation.