

G2 - Programmation Android

Programmation d'applications pour la plateforme Android

Objectifs

- Découvrir l'architecture système d'Android
- Comprendre le SDK et le NDK Android
- Maîtriser l'architecture des applications Android
- Maîtriser les tâches de programmation principales avec Android
 - conception d'interfaces utilisateur
 - stockage et récupération de données
 - accès aux réseaux de communication
 - ...
- Intégration de votre application dans un système Android
 - appel des composants système
 - être appelé par d'autres composants

Les exercices sont réalisés sur carte cible à base de processeurs NXP i.MX6 ou i.MX8
Nous utilisons la dernière version libre d'Android, telle que supportée par la carte.

A qui s'adresse ce cours

- Ingénieurs qui doivent créer des programmes pour des plateformes Android

Prérequis

- Bonne connaissance de la programmation Java et de ses mécanismes avancés (voire notre cours [L4G - Java pour Android](#))

Matériel

- Un PC Linux ou Windows et un système cible sous Android par binôme
- Support de cours imprimé (en anglais)
- Documentation et solutions des exercices

Environnement du cours

- Cours théorique
 - Support de cours au format PDF (en anglais) et une version imprimée lors des sessions en présentiel
 - Cours dispensé via le système de visioconférence Teams (si à distance)
 - Le formateur répond aux questions des stagiaires en direct pendant la formation et fournit une assistance technique et pédagogique
- Activités pratiques
 - Les activités pratiques représentent de 40% à 50% de la durée du cours
 - Elles permettent de valider ou compléter les connaissances acquises pendant le cours théorique.
 - Exemples de code, exercices et solutions
 - Pour les formations à distance:
 - ▶ Un PC Linux en ligne par stagiaire pour les activités pratiques, avec tous les logiciels nécessaires préinstallés.
 - ▶ Le formateur a accès aux PC en ligne des stagiaires pour l'assistance technique et pédagogique
 - ▶ Certains travaux pratiques peuvent être réalisés entre les sessions et sont vérifiés par le formateur lors de la session suivante.
 - Pour les formations en présentiel:
 - ▶ Un PC (Linux ou Windows) pour les activités pratiques avec, si approprié, une carte cible embarquée.
 - ▶ Un PC par binôme de stagiaires s'il y a plus de 6 stagiaires.

- Pour les formations sur site:
 - ▶ Un manuel d'installation est fourni pour permettre de préinstaller les logiciels nécessaires.
 - ▶ Le formateur vient avec les cartes cible nécessaires (et les ramène à la fin de la formation).
- Une machine virtuelle préconfigurée téléchargeable pour refaire les activités pratiques après le cours
- Au début de chaque session (demi-journée en présentiel) une période est réservée à une interaction avec les stagiaires pour s'assurer que le cours répond à leurs attentes et l'adapter si nécessaire

Audience visée

- Tout ingénieur ou technicien en systèmes embarqués possédant les prérequis ci-dessus.

Plan du cours

Premier jour

Introduction

- Histoire
- Vue d'ensemble
- L'architecture système d'Android
 - Les services Android
 - Les frameworks Android

Architecture des applications Android

- Structure d'une application Android
- Composants des applications Android
 - Activité
 - Service
 - Récepteur de diffusion
 - Fournisseur de contenu
- Le fichier Manifest
 - Déclaration des composants applicatifs
 - Autorisations

Exercice : Application Hello world

Activités et interface utilisateur

- Le cycle de vie des Activités
- Callbacks des activités
 - onCreate
 - onStart...
- Intentions et filtre d'intentions
- Invocation d'activité avec et sans résultat
 - startActivity
 - startActivityForResult
- Tâches (piles d'activités) et navigation entre activités
- Ressources
 - Chaînes de caractères
 - Images
 - Mise en page...
- Vues
 - boutons, étiquettes et champs d'édition
 - Instanciation d'une vue depuis une ressource
- Mise en page

- Types de mise en page
- Propriétés des composants liées à la mise en page
- Vues spécialisées
 - ListView
 - Accès aux données (classe et sous-classes d'adaptateur)

Exercice : Application Notepad simplifiée

Deuxième jour

Interaction avec l'utilisateur

- Saisies utilisateur
 - écran tactile et clavier
 - gestion du clavier logiciel
- Notifications à l'utilisateur
 - Boîte de dialogue
 - Barre d'état
 - Toast
 - Dialogue de recherche (SearchManager)
- Notifications et le NotificationManager
 - Notifications dans la barre d'état
 - Vibrations et flashes
- Notifications "push"

Exercice : Amélioration de l'application Notepad

- Adaptation de l'interface utilisateur
 - En fonction de la langue
 - En fonction des caractéristiques de l'écran (dimensions, orientation ...)

Exercice : Application Hello World bilingue (Anglais-Français)

Test et debug

- Utilisation du debugger depuis Eclipse
- Logs
- Test unitaire

Éléments avancés d'interface utilisateur

- Interface utilisateur et le multitâche
 - Accès aux vues depuis une autre tâche

Exercice : Création d'une interface multi-tâche avec des boutons et des barres de progression

- Création de contrôle personnalisé
 - En dérivant la classe View
 - En dérivant une vue existante
- Dessin 2D
 - Canevas et formes
 - Dessin à partir de la tâche principale
 - Dessin depuis une autre tâche
- Animations

Exercice : Déplacement d'une image à l'écran

Troisième jour

Les Fragments

- Qu'est-ce qu'un fragment
- Le cycle de vie des fragments
 - Interaction avec le cycle de vie de l'activité

- Les callbacks spécifiques du cycle de vie des fragments
- Fragments et orientation de l'écran
 - Utilisation de fragments pour s'adapter aux différentes tailles d'écran
 - Réaction à des changements d'orientation
- Les différents fragments spécialisés
 - Fragments listes
 - Fragments de dialogue

Exercice : Utilisation de fragments pour adapter l'application Notepad à des tailles d'écrans variables

Le NDK Android

- Le NDK Android
 - Définir des méthodes Java en C++
 - JNI pour Android
 - Utilisation de SWIG
- Intégration de code natif dans un package
 - Utilisation du NDK depuis Eclipse
 - Debug de code natif

OpenGL/ES

- OpenGL et OpenGL/ES
 - Les différentes versions d'OpenGL/ES
 - Utilisation d'OpenGL/ES en Java
 - Utilisation depuis le NDK
- Concepts de base
 - Vertex et Triangles
 - Transformations
 - Tracé
- OpenGL/ES 1.1
 - Le Pipeline OpenGL/ES 1.1
 - Projection et point de vue de la Camera
- OpenGL/ES 2.0
 - Le pipeline programmable OpenGL/ES 2.0
 - Les shaders de Vertex
 - Les shaders de Fragment
 - Application de transformations
- Le setup d'OpenGL dans une application Android

Exercice : Création d'une vue 3D animée en OpenGL/ES (depuis Java et à travers le NDK)

RenderScript

- La couche RenderScript dans Android
 - L'API du framework renderscript
 - La couche de réflexivité donnant accès au code renderscript depuis Java
- Le code RenderScript
 - Le langage C renderScript
 - Le moteur de calcul renderScript
 - Le moteur graphique (obsolete)

Exercice : Utilisation de Renderscript pour transformer une image couleur en noir et blanc

Quatrième Jour

Services

- Déclaration de service
- Démarrage et arrêt d'un service

- Les RPC
 - Définition and implémentation d'une interface AIDL
 - Lien à un service et invocation de RPC
- Le Binder d'Android

Exercice : Création d'un service applicatif

- Les services système
 - Qu'est-ce qu'un service système
 - Services statiques ou dépendants du contexte
 - Structure d'un service système
 - Le process ServiceManager
- Gestion d'alimentation par les applications

Exercice : Utilisation du PowerManager pour empêcher la mise en veille

Multimedia

- Playback audio et video (La classe MediaPlayer)
- Capture Audio et video (La classe MediaRecorder)

Exercice : Implementation d'un service de playback mp3

Les BroadcastReceivers

- Installation d'un récepteur d'événements
 - Création statique
 - Instanciation dynamique et enregistrement
- Diffusion d'Intents
 - Diffusion normale
 - Diffusion ordonnée
- Utilisation de PendingIntent avec des BroadcastReceivers
- Les événements système

Exercice : Implementation d'un BroadcastReceiver applicatif

Réseau

- Gestion des Connexions
- Sockets
- Requêtes HTTP
- Le contrôle WebView
- Les services Web

Exercice : Communications sur socket

Exercice : Affichage d'une page web avec WebView

Exercice : Utilisation d'un service web

Cinquième jour

Gestion de données

- Stockage
 - Préférences partagées
 - Stockage interne
 - Stockage externe
 - SQLite
- Fournisseur de contenu
 - Communication avec un fournisseur de contenu
 - Création d'un fournisseur de contenu

Exercice : Stockage de fichiers MP3 et des informations associées dans une table SQLite

Exercice : Implémentation d'un fournisseur de contenu et d'un client pour accéder aux fichiers MP3

Interaction avec la plateforme

- Gestion des contacts
- Envoi et réception d'e-mails
- Emission d'un appel
- Prise de photos et de vidéos
- Les principaux événements du système
- Récepteur de diffusion

Exercice : Application pour prendre une photo et l'envoyer à un contact

Les capteurs Android

- Capteurs Android
 - Les types de capteurs
 - Le SensorManager
 - Accès aux capteurs
- Architecture du framework
 - Découverte des capteurs
 - Calibration des capteurs

Exercice : Lecture et affichage des valeurs d'un capteur

Bluetooth

- Configuration
- Device search
- Services management (SDP)
- RFCOMM sockets

Exercice : Bluetooth chat

Services de localisation et API Google Map

- Localisation
 - Le framework de localisation GSM, WIFI et GPS
 - La classe LocationManager
- Intégration de l'API Google Map dans une application
 - La classe MapView

Exercice : Utilisation de MapView et affichage d'images sur la carte pour des points d'intérêt