

Advanced Android App Development

(MD101)

48 Hours

Outline

This course provides Android developers with the skills required to build android applications, services and other components all over the android stack. The curriculum includes writing a variety of android applications using java and native (C/C++) code

Objectives

Upon completion of this course, students will be able to:

- Build android applications
- Manage activities
- Create and work with services
- Add native code

Target Audience

The course has been designed for developers who want to master android application development.

Prerequisites

Students should have a working knowledge with at least one of the following programming language: C/C++/Java/C#.

Contents

Android Overview

- History
- Android vs. Linux
- Android Stack
- Development tools
- Writing Application
- Using Log and Toast
- Using ADB
- Lab: write a simple application
- Application Components - overview
 - Activities
 - Services
 - Broadcast receivers Content providers
- Intents
- Building GUI
- Managing resources
- Manifest file
- Lab: writing application with activities and intents

GUI elements

- Building UI
- Layouts and common widgets
- Menus
- Dialogs
- Building dynamic UI using xml and code
- Handling events
- Lab: building Application with GUI

Processes and threads

- Creating threads
- UI threads
- AsyncTask
- Looper
- Handler
- Lab: working with threads

Android Services

- Android Services
- Binder and IPC
- Using System services
- Permissions
- Examples from system services (Location, Bluetooth, etc.)
- Lab: using system services
- AIDL
- AIDL types
- Custom types
- Custom permissions
- Building a service
- Building a client application
- Async-IPC
- Lab: creating a service

Data storage

- Data storage overview
- Shared preferences
- Files
- Databases
- Accessing web services
- Lab: using data

- Content providers
- Contents resolver
- URI
- Managing data: query, insert, update and delete
- Lab: content provider

Advanced features

- App Widgets
 - Overview
 - Implementation
- Android Media
 - Graphics
 - Audio
 - Video
 - External libraries and tools
- Android framework
- JNI and NDK
 - NDK tools
 - JNI overview
 - Types
 - Local and global references
 - Reflection
 - Exceptions
 - Using C++
 - Best practices
 - Lab: adding native code to android application