

Associate Android Developer Certification

Overview

Earn an Android Developer Certification from Google. For a long time, there was no way for developers to demonstrate their understanding of the Android platform. You can now officially get recognized for your Android development skills by earning the Associate Android Developer Certification directly from Google. The program is developed for developers with extensive programming experience in developing Android mobile applications with the Java programming language.

Duration

The program is full time and runs over 5 days. The training is highly hands on and less time is devoted to theory. In the course we will develop a realtime mobile chat application. The mobile app will also be able to interact with the web based chat system. You will be required to use the skills from the mobile chat application and integrate them into your own project. By the end of the course you should developed your own Android app and publish it to the Google Play store. You will go from working with the sketches, before the class begins, all the way to developing and publishing the app on the Google Play store. You will get 6 months after training support, telephone, Skype and Teamviewer remote access.

Programming Experience

The course is designed for programmers with a strong background in Java programming. A strong computer science background is a must. Alternatively, students who have passed the Oracle Java Programming Certification exam are also eligible to enroll in this course. Delegates who have a strong object oriented programming experience in other languages can take the course but they will need to brush up on their skills of the Java programming language. For .Net Developers we recommend taking the Developing Android Mobile Applications with Xamarin, which is offered using the C# programming language.

Technical Skill

To be successful in this course you should be proficient in Git and Github, integrating external libraries into an application, Maven and Gradle, a strong understanding of XML, a good grasp of object oriented programming language concepts, create and work with Classes, implement anonymous and abstract classes, define and implement interface, a good command of Visual Studio

Private Training

The course is primarily offered privately to a group, team or company. The training will be run onsite or you can come to us. We have offices in Johannesburg and Cape Town and can also conduct the training in Durban at Regus Business Centre. We can train anywhere in South Africa and the rest of the African continent.

Public Training

We also conduct the training publicly. This is suitable for individuals or delegates that cannot book the private training. The class will be comprised of delegates and individuals from different companies. The training content and duration is the same and training will run at our premises in Johannesburg or Cape Town, South Africa. A minimum of 4 delegates is

required to run the course. A preliminary date is scheduled and will be confirmed once we have 4 delegates to run the course.

Course Curriculum

Android Getting Started

The goal of this module is to have you write a simple Hello, World application. By the end of this module, you should be able to identify main parts of an Android app. You should also be able to verify that your tools are setup properly and that you can run the application on an Android device, either physical or emulated.

- Exploring the Structure of an Android Application
- Fundamental Components
- Application Life Cycle

Building Blocks and User Interfaces

- Activity
- Intent
- Content Provider
- Broadcast Receiver
- Services
- Understanding Android resources
- Understanding Android Intents
- Text Controls
- Button Controls
- The ImageView Control
- Date and Time Controls
- List Control
- GridView Control
- Spinner Control
- Gallery Control
- Understanding Adapters
- Styles and Themes
- Understanding Layout Managers

Application (User Interfaces)UI and UX(User Experience)

- Mock up the main screens and navigation flow of the application
- Describe interactions between UI, background task, and data persistence
- Construct a layout using XML or Java code
- Create a custom view class and add it to a layout
- Implement a custom application theme
- Apply a custom style to a group of common widgets
- Define a RecyclerView item list
- Bind local data to a RecyclerView list
- Implement menu-based or drawer navigation
- Localize the application's UI text into one other language
- Apply content descriptions to views for accessibility
- Add accessibility hooks to a custom view

Fundamental Application Components

- Describe an application's key functional and nonfunctional requirements
- Create an Activity that displays a layout resource
- Fetch local data from disk using a Loader on a background thread
- Propagate data changes through a Loader to the UI
- Schedule a time-sensitive task using alarms
- Schedule a background task using JobScheduler
- Execute a background task inside of a Service
- Implement non-standard task stack navigation (deep links)
- Integrate code from an external support library

Persistent Data Storage

- Define a database schema; include tables, fields, and indices
- Create an application-private database file
- Construct database queries returning single results
- Construct database queries returning multiple results
- Insert new items into a database
- Update or delete existing items in a database
- Expose a database to other applications via Content Provider
- Read and parse raw resources or asset files
- Create persistent preference data from user input
- Toggle application logic based on preference values

Enhanced System Integration

- Create an app widget that displays on the device home screen
- Implement a task to update the app widget periodically
- Create and display a notification to the user

Working With Menus

- Structure of Android menu
- Working with Menu and Events
- Submenus, Context Menus,Pop-up Menus

Dialogs and Toolbar

- What Is a Fragment
- Fragment Manager
- Using Dialogs in Android

- Working with Toast
- Working with Toolbar and Tabs

Working With Preferences and Saving State

- Exploring the Preferences Framework
- ListPreference
- CheckBoxPreference
- EditTextPreference
- Organizing Preferences

Persisting Data With SQLite

- Introduction to SQLite Database
- SQLiteOpenHelper and working with DB
- Working with Cursor

Sharing Data with ContentProviders

- Android's Built-in Providers
- Architecture of Content Providers

Backgrounding and Consuming Network Services

- Consuming HTTP Services
- Using the AndroidHttpClient
- Addressing Multithreading Issues
- Handler, message and message queue relationship
- Using Background Threads (AsyncTask)

Working With Web Services

- JSON parsing
- Making HTTP API Calls

Security in Android

- Understanding the Android Security Model
- Signing Applications for Deployment
- Self-Signed Certificate Using the Keytool
- Installing Updates to an Application and Signing
- Understanding Security at the Process Boundary
- Declaring and Using Permissions
- Library Projects StrictMode

Testing and Debugging

- Write and execute a local JVM unit test
- Write and execute a device UI test
- Given a problem description, replicate the failure
- Use the system log to output debug information
- Debug and fix an application crash (uncaught exception)
- Debug and fix an activity lifecycle issue
- Debug and fix an issue binding data to views

Working With Maps and Location Services

- Understanding the Mapping Package Obtaining a Maps API Key from Google

Preparing Your Application For Market

- Develop Market ready application