Android™ Application Development

Duration: 35 hours

Prerequisites: Prior experience with a scripting or programming language is required.

Course Description: This hands-on course conveys the fundamental skills necessary to deploy Android Apps on mobile devices such as phones and tablets. Attendees will design and build a variety of Android Apps throughout the course using Android Studio. Previous Java programming knowledge is not essential, but basic programming experience is required. Java code used in the exercises is fully explained.

The course emphasizes proper layout of the user interface (UI), including how to add buttons, labels, textboxes, checkboxes, images and other widgets to the UI. Students will learn how to utilize Android's XML-based layout system, which builds the UI with containers and widgets, as well as how to set wallpapers and add menus to the UI. Students practice with dialog techniques including the display of popup messages.

Students also learn how to handle screen rotation, and how to define UIs so they can adjust for different screen sizes. The course teaches students how to accept user input from keyboards (either externally attached or from the built-in keyboard), how to use the date/time picker, and how to present users with choices using Selection Lists. Students will learn how to add tabs to the UI, as well as how to display HTML content using the built-in WebKit browser.

Students will learn how to program control of state changes in the Activity Lifecycle: active, paused, stopped or dead. For instance, since Apps often run on phones, taking a call may send an App from the active state to the paused state. Or, if battery life is low the App may be forced into the dead state. Student will learn how to manage Activities moving between these states so that Apps can, for example, save inputted data before transitioning into the dead or paused state. Also related to this is the ability for an Activity to launch Sub-Activities, which allows Apps to run processes in the background (such as downloading files).

Coverage of data storage includes best practices for storing images and files. Student will also learn how to embed SQLite databases in Apps, and then use these databases to store and retrieve any kind of data.

The course emphasizes Service Oriented Architecture (SOA), and students learn how to connect Apps to web services such as Google Maps and UPS's package tracking API. In addition to using outside web services, the course also demonstrates how to use many of a phone or tablet's built-in features such as the camera, location service (which determines where the device is physically located), and the Accelerometer (which determines if the device is being tilted and how fast it's being tilted). Additional topics include how to play audio and video, and standard techniques for storage and retrieval.

Students employ Android Studio to perform comprehensive hands on exercises throughout the course to reinforce learning and develop genuine competency.

Students may bring in their own Android device and USB cable, if they prefer to test on physical Android hardware. Students who do not provide their own Android devices will use Android emulators during class lab work.

Students Will Learn

- Background Threads and Services

Introduction to Android

- Installing the Android SDK
- Creating Android Projects in the IDE
- Java Language Highlights
- Activities and User Interfaces
- Widgets and Layouts
- Accepting User Input
- Developing Layouts
- Spinners and Lists
- Passing Control With Intents
- User Feedback - Toasts, Dialogs, Notifications and Alarms
- Menus and the Action Bar
- Broadcast Receivers
- SQLite Databases
- Content Providers
- Cameras, Audio and Video
- Telephony and Message Services
- Sensors
- Networking and Web Services
- Packaging Apps for the Google Play Store
- Mapping and Location Services
- Advanced UI Techniques
- Bluetooth Networking and NFC

Overview

Getting Started with Android App Development

- Fundamental Components of Android Phones and Tablets
- Installing and Configuring Android Studio to Create Apps
- Testing Apps Using the Simulator
- Understanding Service Oriented Architecture (SOA)

Creating the User Interface (UI)

- Core Components of Android User Interfaces (UIs)
- Using Android's XML-Based Layout System to Define UIs
- Common Widgets
  - Buttons
  - Labels
  - Checkboxes
- UI Layout Best Practices

Mastering User Interface Layouts

- Pros And Cons of Different Layout Models
- Using Containers to Control the Location of Widgets
- Defining a Widget's Position Relative to Other Widgets
- Overlapping Widgets
- Using Table-Based Layouts
- Enabling Scrolling

Accepting User Input

- The Android Input Method Framework
- Getting Input from Attached and/or Internal Keyboards
- Processing Choices with Selection Lists
- Adding Icons to Selection Lists
- Using Sliders and Checkboxes
- Using Date and Time Pickers
- Responding to Taps And Swipes

Menus, ActionBars and Popups

- Adding Traditional Application Menus to Apps
- Displaying Context Menus When Users Tap and Hold
- Using Actionbars (Toolbars For Android Apps)
- Displaying Popup Messages

Advanced User Interfaces and the WebKit Browser

- Embedding HTML Content in Apps with WebKit
- Displaying Analog and Digital Clocks
- Using Tabs to Display Multiple UIs
- Applying Standard Styles and Fonts
- Creating Reusable UI Components with Fragments

Working with App Activities and Intent Filters

- The Application Activity Lifecycle
- Saving Data when Apps are Paused Or Killed
- Using Threads to Run Background Services
- Launching Sub-Activities
- Handling Phone Calls
- Optimizing Device Resources

Saving Files and Images

- Storing Images, Strings and Arrays
- Playing Audio and Video Files
- Best Practices for Saving Files
- Creating and Storing Wallpapers

Handling Rotation and Different Screen Sizes

- Different Strategies for Handing Screen Rotations
- Controlling UI Changes when Screens Rotate
- Issues Related to Tablets and Larger Screens
- Creating UIs that are Compatible with All Screen Sizes (Right-Sizing)

Deploying to Android Devices

- Checking for and Requiring Phone or Tablet Features
- Deploying to the Android Store
- Deploying Directly to Devices

Data Management

- Embedding SQLite Databases in Apps
- Permanently Storing Data in Local Databases
- Retrieving Database Data when Apps Start
- Best Practices for Data Management

Using Java Libraries and Web Services

- Using Pre-Compiled Java Libraries
- Connecting to Web Services (such as Google Maps)
- Downloading Internet Content

Working with Built-in Android Features

- Accessing and Using MapView
- Getting the Location of the Phone Or Tablet
- Working with the Camera
- Working with Permissions and the Security Framework

Related Bootcamp

Track	Duration	Price
Mobile Application Developer	2-course track 3-course track	$2,400 $3,600

Android is a trademark of Google Inc.

Contact Us

Address: 6 Hemlock Drive Chelmsford, MA 01824

Phone: 978.250.4983

Mon - Thur: 9 am - 5 pm EST
Fri: 9 am - 4 pm EST

E-mail: info@developer-bootcamp.com

Copyright© 2025 Developer Bootcamp