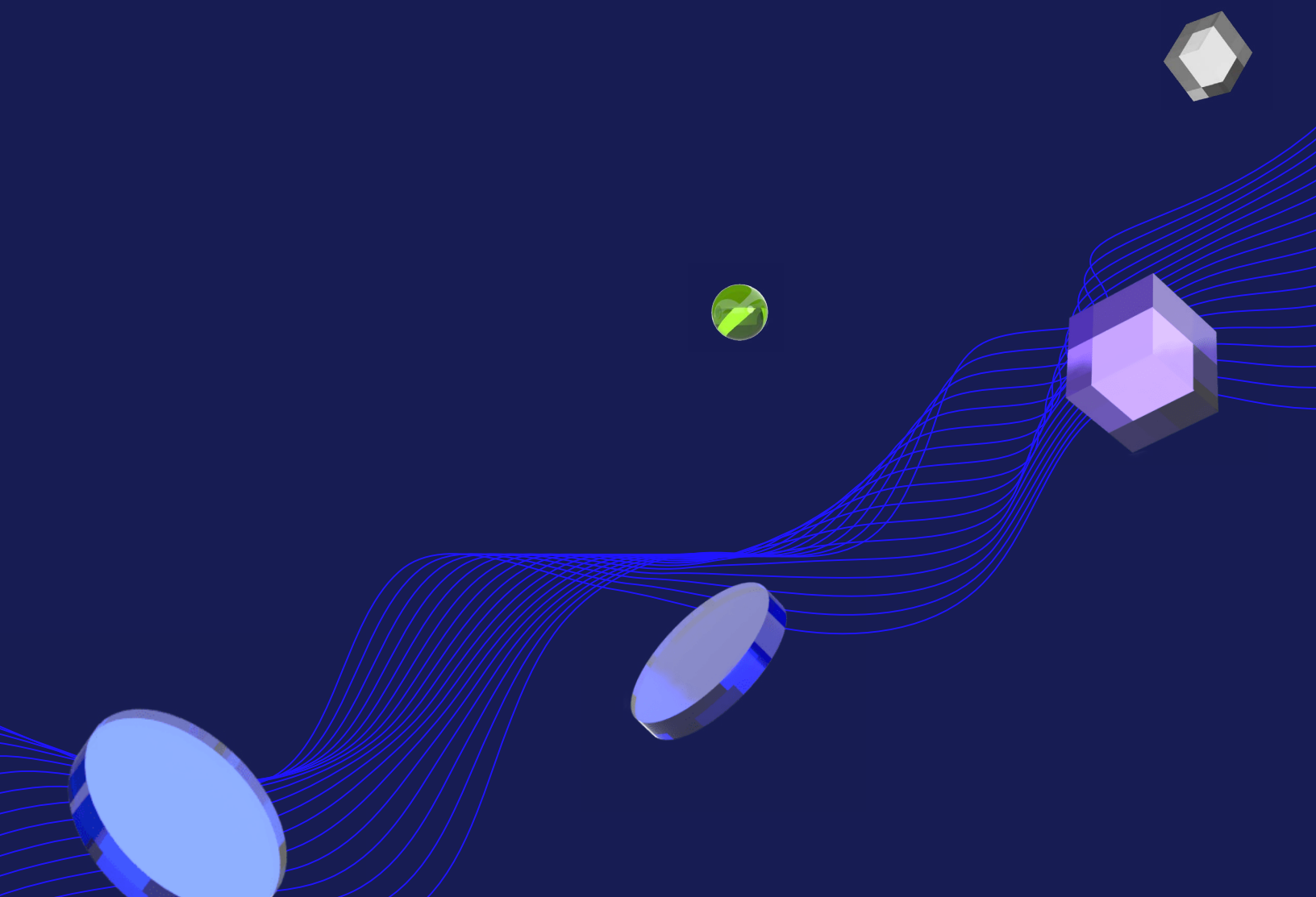
INDIVIDUAL LEARNERS

UDACITY

SCHOOL OF PROGRAMMING AND DEVELOPMENT

Android Kotlin Developer

Nanodegree Program Syllabus



Overview

Built in collaboration with Google, this program will prepare learners to become professional Android developers, and allow them to create a diverse portfolio of projects to show employers. By the end of this program learners will be able to use Android development platform best practices, Android Studio, Android Jetpack, and Kotlin to build their own applications for the world's most used mobile platform.

Learning Objectives

A graduate of this program will be able to:

- Design engaging interfaces that implement modern Android components to effectively build compelling features for the end user.
- Incorporate remote data into an app by utilizing RESTful interfaces and web APIs.
- Organize, store, retrieve, and display content on an Android device to provide users with a more consistent, performant, and accessible experience, even while offline.
- Integrate hardware capabilities such as location to provide users with mobile enriched features within an Android app.
- Architect an Android app using the established MVVM pattern for scalability and performance.

Built in collaboration with:

Google

Program information

Estimated Time

4 months at 10hrs/week*

Skill Level

Intermediate

Prerequisites

A well-prepared learners should:

- Be comfortable with object-oriented programming and the Android platform.
- Have experience navigating GitHub and be comfortable using a Modern IDE.
- Be familiar with threads and concurrency, and with modular app architectures.

Required Hardware/Software

Learners need access to:

- A personal computer that is capable of running Android Studio.
- An Android device (helpful, but not necessary).

*The length of this program is an estimation of total hours the average student may take to complete all required coursework, including lecture and project time. If you spend about 5-10 hours per week working through the program, you should finish within the time provided. Actual hours may vary.

Course 1

Developing Android Apps, Part 1

Use common Android UI components to create a basic user interface, handle user input and Android lifecycle events, and create dynamic and navigable interfaces using constraint-based layouts. Learners will also find out how to use the Gradle build process to declare library dependencies and establish app parameters, and integrate app functionality with other apps or components with Android.

Course Project

Build a Shoe Store Inventory App

Build an Android app with Kotlin! Learners will build an Android app with multiple screens and create a navigation graph to take the user through the app. They will use fundamental Android development skills to set up a development environment for an Android app, use Android Studio's layout editor, and implement best practices for navigation and user interface in Android. Learners will also follow recommended Android app architecture guidance with ViewModel and LiveData lifecycle classes.

Lesson 1

Build Your First App

- Explore the basics of Android, such as creating text, images, and interactive buttons.
- Set up the development environment and create a dice roller Android app.
- Navigate the main map anatomy of an Android app.

Lesson 2

Layouts

- Learn different kinds of views and resources.
- Explore arranging elements with the Android Studio's layout editor.
- Connect views with data through data binding.

Lesson 3

App Navigation

- Learn how to build apps that contain multiple screens known as destinations.
- Use Android Studio tools to create and visualize a map, or graph, of destinations that show navigation paths in your app.
- Learn the navigation patterns and user interface that Android users expect to see, so that your app will be intuitive and familiar.

Lesson 4

Activity & Fragment Lifecycle

- Learn all about the Android activity lifecycle.
- Create a one-screen app called “Dessert Pusher.”
- Debug common issues through an understanding of lifecycles.

Lesson 5

App Architecture (UI Layer)

- Learn one way to structure an Android app and the benefits that come with this design.
- Learn about two classes in the lifecycle library: ViewModel and LiveData.

Course 2

Developing Android Apps, Part 2

Learn how to implement data persistence in an app, display collections of data to users using RecyclerView, and use APIs to connect to, store, and retrieve data. Learn best practices of material design to create a quality user experience and boost accessibility to as many users as possible.

Course Project

Build an Asteroid Radar App

Build an app using a free, open source API provided by the NASA JPL Asteroid team. Learners will create an app that connects to the internet to retrieve and display live data, implement networking best practices to fetch and display data and images, and create a database to store and access user data over time. They will also learn to use RecyclerView to create a clear and compelling UI to display the data. Finally learners will test their app with Talkback enabled and make their app more accessible for as many users as possible.

Lesson 1

Recycler View

- Implement the ViewHolder pattern to optimize performance when displaying large sets of data with RecyclerView.
- Display large collections of data in a user consumable and navigable format.
- Optimize application performance when updating data collections that affect the UI.

Lesson 2

Connect to the Internet

- Build an application that connects to an internet server to retrieve and display live data.
- Simplify fetching data and images, to make sure the app reasonably conforms to networking and image loading best practices.

Lesson 3

Behind the Scenes

- Learn how to implement offline caching by building an app that lets users watch DevByte videos.
- Take an online-only app and transform it to work offline by adding offline caching.

Lesson 4

Designing for Everyone

- Improve your app design to support multiple languages as well as support multiple device sizes and orientations.
- Learn how to make your app accessible for users who might need assistance navigating, like supporting talkback and pushbutton navigation.

Course 3

Advanced Android Apps with Kotlin, Part 1

Learn how to enhance your app's functionality and drive user engagement using Android's robust notification system, build custom views, and use canvas drawing to allow for the update of a display based on user interactions. Learners will also be able to create simple animations to enhance the presentation of content and overall usability of the app.

Course Project

Design an App with Application Loading Status

Create an Android app that will download a file from the internet and create notifications, custom views and animations to build a status bar in the app. Learners will be able to create a notification to send messages to a user within an Android app, and design and style the notifications. They will also build custom views using canvas and paint, animate UI elements with property animations, and use MotionLayout to enhance the user experience of the app.

Lesson 1

Using Notifications

- Send messages to users using notifications.
- Design and style notifications.
- Add buttons and actions to notifications.
- Send push messages using Firebase Cloud Messaging.

Lesson 2

Creating Custom Views

- Create custom views for your app.

Lesson 3

Drawing on Canvas Objects

- Build an app that allows users to paint directly on the screen.

Lesson 4

Clipping Canvas Objects

- Create and display transformed and clipped regions to the screen.
- Translate the origin of a drawing surface of a region.
- Draw multiple shapes on a canvas.

Lesson 5

Android Property Animations

- Use animations to draw attention to important UI elements and beautiful designs.
- Animate UI elements with property animations.

Lesson 6

Using Motion Layout to Animate Android Apps

- Use declarative XML with MotionLayout to coordinate animations across multiple views.

Course 4

Advanced Android Apps with Kotlin, Part 2

Learn how to build an app with the location awareness and Google Maps. Find out best practices and techniques for testing to enable quick scalability, while mitigating any negative effects. Lastly, learn to use Firebase for authentication and remote storage.

Course Project

Build a Location Reminder App

Build a to-do list app that includes Google Maps and location services. Learn how to add Google Maps and style map views in an Android application, and enable location services and tracking. With location services and reminders, the app will remind users to perform an action when the user is at a specific location.

Capstone Project

Design and Build an Android App

In this final project, learners will have the opportunity to design and build either a custom Android app inspired by an original idea or a political preparedness app that will deliver civic data to end users via the app. Learners will apply skills acquired throughout the program to design an engaging user interface that incorporates data from RESTful interfaces and web APIs, and utilizes mobile hardware to enhance app functionality and provide an engaging user experience. The project will allow learners to showcase recommended Android app architecture patterns, delivering a highly functional and scalable app that takes full advantage of the Android platform.

Lesson 1

Wandering in Google Maps with Kotlin

- Add Google Maps functionality to an Android app.
- Style Google Maps views in multiple ways in an Android app.

Lesson 2

Virtual Treasure Hunt with Geofences

- Enable location services and tracking.

Lesson 3

Testing: Basics

- Learn how to test your app before distribution to avoid crashes or unpredictable behavior.

Lesson 4

Introduction to Test Doubles & Dependency Injection

- Use test doubles and dependency injection to test an app.
- Write tests with Espresso for UI testing.
- Use Mockito to create an integration test.

Lesson 5

Survey of Advanced Testing Topics

- Implement end to end testing using navigation, coroutines, room, and databinding.

Lesson 6

Implementing Login on Android with FirebaseUI

- Implement user login and identity management for your app using the open-source library FirebaseUI.
- Enable login and logout for your app's users.
- Control navigation in your app based on whether a user is logged in.

Meet your instructors.

Dan Galpin

Android Developer Advocate at Google

Dan has been on the Android team for over 10 years, working on app design, architecture, performance, and development best practices. He's passionate about teaching, and has been inspired by stories from students over the years about how these courses have helped to change the trajectory of their lives and careers.

Aleks Haecky

Developer Advocate

Aleks is a writer and developer advocate with over 20 years of experience developing media and tools that bring technologies and programming to developers. They believe in the power of education, and Android development as a skill that can change lives.

Sean McQuillan

Developer Advocate

Sean has a decade of experience as a startup engineer in San Francisco where he learned how to build successful apps. Sean is passionate about building high quality products—quickly. When he is not working on Android you can find him fiddling on the piano or crocheting hats.

Murat Yener

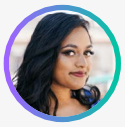
Android Developer Advocate

Murat has been an Android Developer back to Froyo, worked on wearable and other form factor Android devices. He is a code geek, open source committer, Java Champion, and the author of Expert Android Studio and Professional Java EE Design Patterns books.

Chet Hasse

Chief Android Advocate at Google

After being on, and leading, the UI Toolkit team on Android for several years, Chet joined the developer relations team. His focus and passion has always been UI, graphics, animation, performance, and anything that puts the pixels on the screen, in addition to helping developers write great apps.



Meghan Mehta

Android Developer Advocate at Google

Meghan is a developer advocate on the Android team. She has been a mobile developer for many years at Disney, Foursquare, Yelp, and now Google. She loves sharing her knowledge and experience with other developers. When she is not working you can find her singing, dancing, or baking!

Caren Chang

Developer Programs Engineer

Caren is a developer programs engineer for the Android Frameworks team at Google.

Lyla Fujiwara

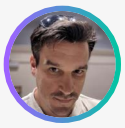
Android Developer Advocate at Google

Lyla authored many of the fundamental Android samples and trainings for Android Jetpack, Kotlin, and testing. She's also had the honor of teaching everyone from highschool students to senior developers how to make Android apps. These days, she's part of the team bringing you the Google News Android app.

Asser Samak

Android Developer Advocate at Google

Aser has been building educational Android apps with Udacity & Google for the past 4 years, he enjoys teaching with a focus on best-practices and building a solid foundation at an early stage. Aser loves solving the Rubik's cube which is featured in many of his videos—try to find them all.



Joshua Donlan

Senior Android Developer at Halogen TV

Joshua has 20 years experience as a web and mobile application developer helping launch multiple startups and grow established companies alike. His client portfolio includes Fortune 100 companies Audi, Disney, Mitsubishi, American Express, BD Pharmaceuticals, and more.

Kevin Moore

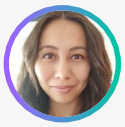
Staff Software Engineer at Affirm

Kevin has been doing Android development for over 9 years, developing many different types of apps. In addition, Kevin has been writing articles and tech editing books, as well as creating videos for raywenderlich.com and LinkedIn Learning.

Jesus Valdez

Senior Android Developer at Handy

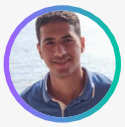
Jesus is a mechatronics engineer with an MS in machine learning. He works as a mobile developer, and is proficient in: mobile development, image processing, machine learning, electronics, and automation.



Aida Issayeva

Android Engineer at Clarity Money

Aida is an Android engineer at Clarity Money, a personal finance management app. Previously, she has built android applications for various industries, ranging from cloud gaming services to satellite data communications. When she's not coding, she's chasing great food experiences all over the world.

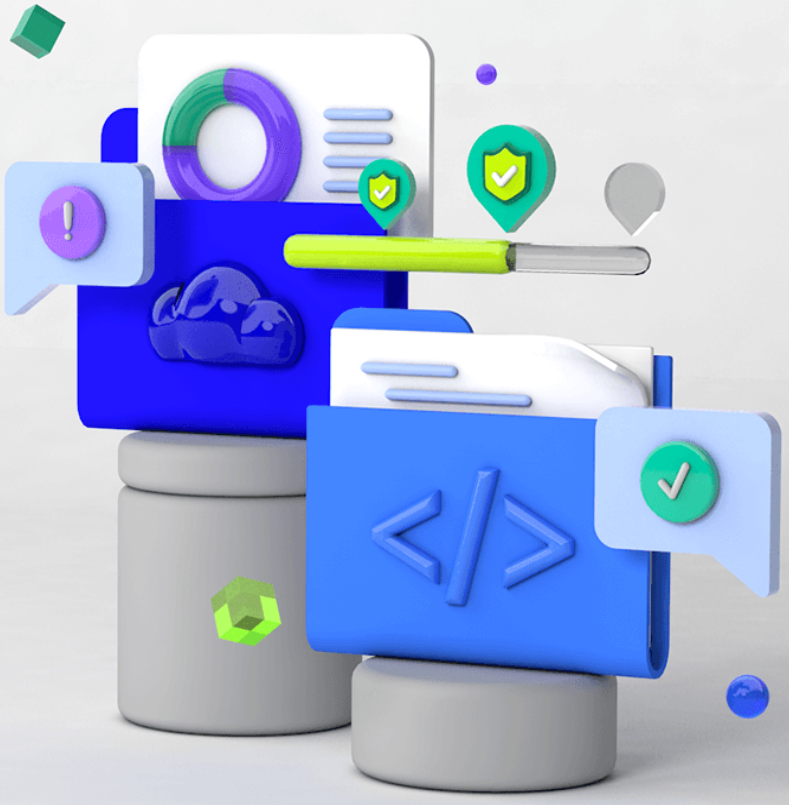


Mohamed Habib

Android Engineer at Andela

Mohamed is an experienced Android Engineer with 5 years of experience, passionate about teaching and mentoring, he has a strong engineering professional with a bachelor's degree in computer science from Ain Shams University.

Udacity's learning experience



Hands-on Projects

Open-ended, experiential projects are designed to reflect actual workplace challenges. They aren't just multiple choice questions or step-by-step guides, but instead require critical thinking.

Quizzes

Auto-graded quizzes strengthen comprehension. Learners can return to lessons at any time during the course to refresh concepts.

Knowledge

Find answers to your questions with Knowledge, our proprietary wiki. Search questions asked by other students, connect with technical mentors, and discover how to solve the challenges that you encounter.

Custom Study Plans

Create a personalized study plan that fits your individual needs. Utilize this plan to keep track of movement toward your overall goal.

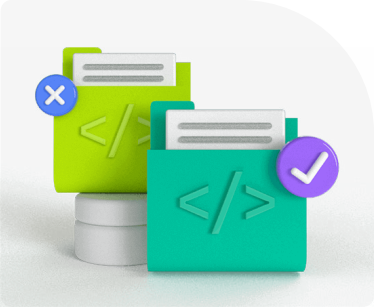
Workspaces

See your code in action. Check the output and quality of your code by running it on interactive workspaces that are integrated into the platform.

Progress Tracker

Take advantage of milestone reminders to stay on schedule and complete your program.

Our proven approach for building job-ready digital skills.



Experienced Project Reviewers

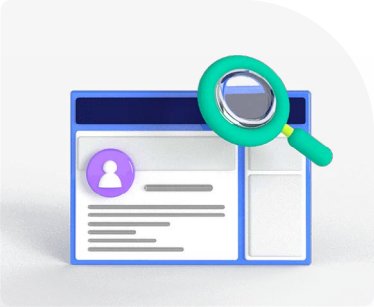
Verify skills mastery.

- Personalized project feedback and critique includes line-by-line code review from skilled practitioners with an average turnaround time of 1.1 hours.
- Project review cycle creates a feedback loop with multiple opportunities for improvement—until the concept is mastered.
- Project reviewers leverage industry best practices and provide pro tips.

Technical Mentor Support

24/7 support unblocks learning.

- Learning accelerates as skilled mentors identify areas of achievement and potential for growth.
- Unlimited access to mentors means help arrives when it's needed most.
- 2 hr or less average question response time assures that skills development stays on track.



Personal Career Services

Empower job-readiness.

- Access to a Github portfolio review that can give you an edge by highlighting your strengths, and demonstrating your value to employers.*
- Get help optimizing your LinkedIn and establishing your personal brand so your profile ranks higher in searches by recruiters and hiring managers.

Mentor Network

Highly vetted for effectiveness.

- Mentors must complete a 5-step hiring process to join Udacity's selective network.
- After passing an objective and situational assessment, mentors must demonstrate communication and behavioral fit for a mentorship role.
- Mentors work across more than 30 different industries and often complete a Nanodegree program themselves.

*Applies to select Nanodegree programs only.

UDACITY

Learn more at

www.udacity.com/online-learning-for-individuals →

12.29.22 | V1.0