UDACITY

NANODEGREE PROGRAM SYLLABUS

Android Developer

Overview

We built this Nanodegree program in partnership with Google for students with intermediate programming skills who want to become professional Android developers. By the end of this program, you'll have a diverse portfolio of projects to show employers, including your own app on Google Play.

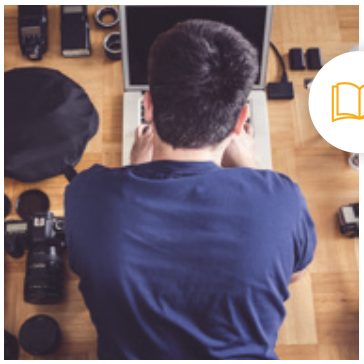
This program is aimed at developers with 1-2 years of experience programming in Java or another object-oriented programming language like Python or C#. If you don't have this experience, we recommend you first complete our Android Basics Nanodegree program or its individual classes prior to starting this program.

Experience with git and GitHub is highly recommended. If you don't have this experience, please take our free course on using git and GitHub prior to starting this Nanodegree program. In addition, taking our free course "GitHub & Collaboration" would also be beneficial to you.

IN COLLABORATION WITH

Google

Estimated Time: 6 Months at 10hrs/week



Prerequisites: Java, Git, GitHub

Flexible Learning: Self-paced, so you can learn on the schedule that works best for you.

Need Help? udacity.com/advisor Discuss this program with an enrollment advisor.

Course 1: Developing Android Apps

In this project, you will complete an app by building a layout and populating its fields from data received as JSON.

Course Project
Sandwich Club

In this project, you will complete an app by building a layout and populating its fields from data received as JSON.

Course Project
Popular Movies App, Stage 1

Build a simple movies app that communicates with the Internet and provides a responsive user experience.

In this project, you will:

- Fetch data from the Internet using the Movie Database API.
- Use adapters and custom list layouts to populate list views.
- Incorporate libraries to simplify the amount of code you need to write

Course Project
Popular Movies App, Stage 2

Build off of your existing movies app to create a fully featured application that looks and feels natural on the latest stable Android operating system (Android version Oreo, as of August 2017).
In this project, you will:

- Allow users to view and play trailers (either in the YouTube app or a web browser).
- Allow users to read reviews of a selected movie.
- Allow users to mark a movie as a favorite in the details view by tapping a button(star).
- Create a database and content provider to store the names and IDs of the user's favorite movies (and optionally, the rest of the information needed to display their favorites collection while offline).
- Modify the existing sorting criteria for the main view to include an additional pivot to show their favorites collection.

	LEARNING OUTCOMES	
LESSON ONE	**Create Project Sunshine**	• Create and configure a new project in Android Studio using mock data
LESSON TWO	**Connect to the Internet**	• Connect your project to the cloud and replace mock data with data from a weather service, using Android permissions and network I/O
LESSON THREE	**Recyclerview**	• Display and interact with scrolling information using a RecyclerView, and update your app to display information in individual views
LESSON FOUR	**Intents**	• Create structure of your app and navigate between screen, using activities from other apps within your own app
LESSON FIVE	**Lifecycle**	• Leverage the Android framework to handle the Android lifecycle
LESSON SIX	**Preferences**	• Enable users to set their preferences for customized views, and save those preferences in app settings and configurations
LESSON SEVEN	**Storing Data in Sqlite**	• Build an SQLite database for your app
LESSON EIGHT	**Content Providers**	• Use a pre-existing content provider in your app to grab user data from your user's phone

LESSON NINE	**Android Architecture Components**	• Learn how Android Architecture components classes can help manage your application's lifecycle and its data persistence needs
LESSON TEN	**Background Tasks**	• Learn how to effectively run jobs in the background, create notifications, and periodically schedule long running background processes
LESSON ELEVEN	**Completing the UI**	• Try different views, viewgroups and alternative layouts, perform data binding, make your app accessible
LESSON TWELVE	**Polishing the UI**	• Add visual polish and styling to your app, including custom colors, fonts and styles, accounting for multiple devices

Course 2: Advanced Android App Development

Make your apps more responsive, and create a total user experience with home screen widgets, third-party libraries, and more. Also, learn to deeply integrate rich media, test user interfaces, and publish to Google Play.

Course Project
Baking App

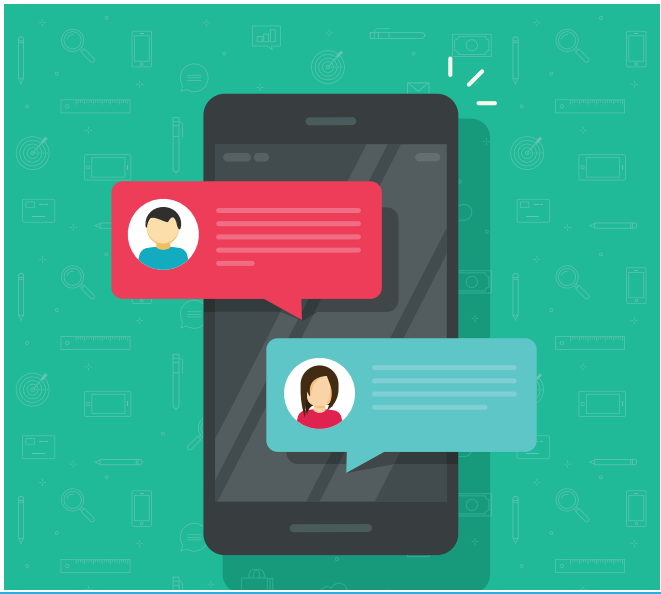
In this project, you will create an app to view cooking recipes. You will handle media loading, verify your user interfaces with UI tests, integrate third party libraries and provide a complete UX with home screen widget.

In this project, you will:

- Use MediaPlayer/ExoPlayer to display videos.
- Handle error cases in Android.
- Add a widget to your app experience.
- Leverage a third-party library in your app.
- Use Fragments to create a responsive design that works on phones and tablets.

		LEARNING OUTCOMES
LESSON ONE	**Fragments**	• Make reusable components in your app to support different form factors
LESSON TWO	**Libraries**	• Use external libraries to personalize your app
LESSON THREE	**Firebase Cloud Messaging**	• Add messaging functionality and push notifications to your app using Firebase Cloud Messaging
LESSON FOUR	**Places**	• Use the Places API to add location to your app.

LESSON FIVE	**Media Playback**	• Integrate video and audio media in your app
LESSON SIX	**Widgets**	• Give users easy access to your app with a homescreen widget
LESSON SEVEN	**Espresso**	• Use User Interface testing to test and verify workflows in your app
LESSON EIGHT	**Publishing Your App**	• Publish your app on the Google Play Store



Course 3: Gradle for Android and Java

Learn how to customize your Gradle build, and explore advanced topics like app testing, configuring free vs. paid apps, and creating and integrating libraries.

Course Project
Build It Bigger

In this project, you will use Gradle to build a joke-telling app, factoring functionality into libraries and flavors to keep the build simple. You will configure a library to connect to a web service for jokes. The finished app will consist of four modules:

- A Java library that provides jokes
- A Google Cloud Endpoints (GCE) project that serves those jokes
- An Android Library containing an activity for displaying jokes
- An Android app that fetches jokes from a web service and passes them to the Android Library for display

	LEARNING OUTCOMES	
LESSON ONE	**Gradle Fundamentals**	• Learn the basics of Gradle and Gradle build scripts
LESSON TWO	**Gradle For Java**	• Use Gradle to build Java projects.
LESSON THREE	**Gradle For Android**	• Use Gradle with Android Studio, and create free and paid versions of an app.
LESSON FOUR	**Advanced Android Builds**	• Use Gradle to automate tasks, including dependency management and unit testing.
LESSON FIVE	**Special Topics**	• Update old projects and libraries, learn more about custom tasks

Course 4: Material Design for Android Developers

Apply the design principles that define Android's visual language to your apps, using material design elements, transitions and graphics, across multiple form factors.

Course Project
Make Your App Material

In this project, you'll update the look and feel of an app to meet Material Design specifications using design elements, surfaces, and transitions across multiple form factors.

		LEARNING OUTCOMES
LESSON ONE	**Android Design Fundamentals**	• Build a layout using Material Design principles.
LESSON TWO	**Surfaces**	• Implement paper surfaces in your design
LESSON THREE	**Bold Graphic Design**	• Use design elements of space, color, type and imagery to make your app more beautiful and impactful
LESSON FOUR	**Meaningful Motion**	• Apply real world principles of motion to make your interface more understandable and continuous
LESSON FIVE	**Adaptive Design**	• Implement Material Design for new form factors
LESSON SIX	**Constraint Layout**	• Learn and apply the new features of ConstraintLayout that will help you create performant user interfaces

Course 5: Capstone Project

Integrate all you've learned in this program to bring your own app idea to life, and publish your app to Google Play.

Course Project
Capstone, Stage 1 - Design

In this project, you will demonstrate the ability to communicate an app idea formally, using:

- An app description
- UI flow mocks, similar to what you have seen in other Nanodegree projects, like the Popular Movie overview
- A list of required tasks that you will complete to build the app

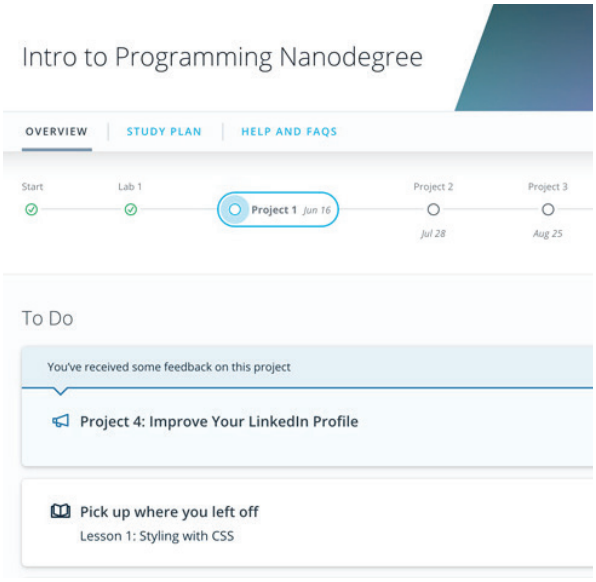
The Capstone project will give you the experience you need to own the full development of an app. This first stage replicates the design and planning experience that proficient Android Developers are expected to demonstrate.

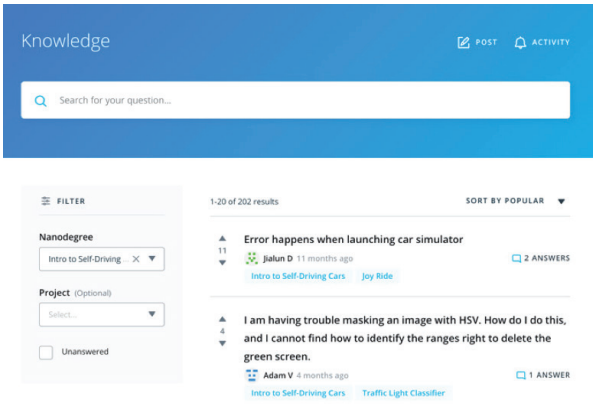
Course Project
Capstone, Stage 2 - Build

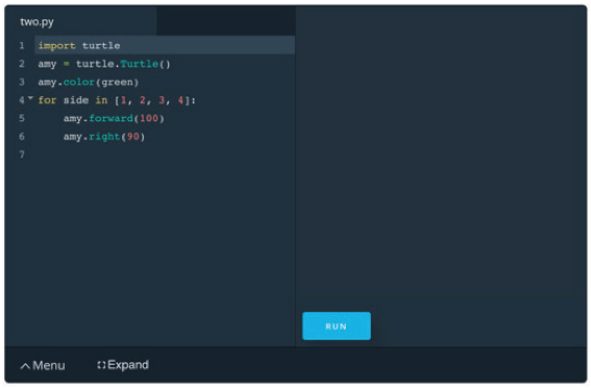
In this project, you will demonstrate the skills you have learned in your Nanodegree program journey and apply them to creating a unique app experience of your own. By the end of this project, you will have an app that you can submit to the Google Play Store for distribution.

The Capstone project will give you the experience you need to own the full development cycle of an app.

Our Classroom Experience







REAL-WORLD PROJECTS

Build your skills through industry-relevant projects. Get personalized feedback from our network of 900+ project reviewers. Our simple interface makes it easy to submit your projects as often as you need and receive unlimited feedback on your work.

KNOWLEDGE

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

STUDENT HUB

Leverage the power of community through a simple, yet powerful chat interface built within the classroom. Use Student Hub to connect with your fellow students in your Executive Program.

WORKSPACES

See your code in action. Check the output and quality of your code by running them on workspaces that are a part of our classroom.

QUIZZES

Check your understanding of concepts learned in the program by answering simple and auto-graded quizzes. Easily go back to the lessons to brush up on concepts anytime you get an answer wrong.

CUSTOM STUDY PLANS

Preschedule your study times and save them to your personal calendar to create a custom study plan. Program regular reminders to keep track of your progress toward your goals and completion of your program.

PROGRESS TRACKER

Stay on track to complete your Nanodegree program with useful milestone reminders.

Learn with the Best

James Williams

CURRICULUM LEAD

James Williams is the Android Curriculum Lead at Udacity, where he also creates Web development courses. He is the author of "HTML5 Game Programming," and runs obstacle course/adventure races in his spare time.

Reto Meier

INSTRUCTOR

Reto Meier has been part of Android Developer Relations at Google since 2009, and leads Google's Scalable Developer Advocacy team. He is the author of the "Professional Android Application Development" book series from Wrox.

Dan Galpin

INSTRUCTOR

Dan Galpin is a Developer Advocate for Android at Google, focusing on Android performance tuning, developer training, and games. He has over 10 years of experience in mobile, developing at almost every layer of the phone stack.

Jocelyn Becker

SENIOR PROGRAM MANAGER

Jocelyn Becker wrote the developer documentation for the first external Google API in 2004, and has been teaching developers to use Google APIs and technologies ever since. She has managed the creation of many of the Android courses built by Google and Udacity.

Learn with the Best

Katherine Kuan

INSTRUCTOR

Katherine Kuan, formerly a Developer Advocate at Google, was a software engineer on the Android Apps team for Google Keep, Google Play, and the People app.

Jessica Lin

INSTRUCTOR

Jessica Lin is the Android Basics Curriculum Lead at Udacity, teaching various aspects of the Android development ecosystem. When untethered from her devices, she can be found training for her next Muay Thai competition.

Jennie Kim Eldon

PRODUCT LEAD

Jennie Kim Eldon is the Product Lead for Android and iOS Nanodegree programs at Udacity, where she previously worked as a software engineer. Before Udacity, she served at the US State Department, leading programs for women and girls in Afghanistan.

Asser Samak

INSTRUCTOR

Asser Samak is a Content Developer at Udacity, with over 9 years experience in software engineering, and a great passion for teaching. He also teaches Udacity's Java course series.

All Our Nanodegree Programs Include:

EXPERIENCED PROJECT REVIEWERS

REVIEWER SERVICES

- Personalized feedback & line by line code reviews
- 1600+ Reviewers with a 4.85/5 average rating
- 3 hour average project review turnaround time
- Unlimited submissions and feedback loops
- Practical tips and industry best practices
- Additional suggested resources to improve

TECHNICAL MENTOR SUPPORT

MENTORSHIP SERVICES

- Questions answered quickly by our team of technical mentors
- 1000+ Mentors with a 4.7/5 average rating
- Support for all your technical questions

PERSONAL CAREER SERVICES

CAREER COACHING

- Personal assistance in your job search
- Monthly 1-on-1 calls
- Personalized feedback and career guidance
- Interview preparation
- Resume services
- Github portfolio review
- LinkedIn profile optimization

Frequently Asked Questions

PROGRAM OVERVIEW

WHY SHOULD I ENROLL?

Android dominates the market of mobile operating systems, with over 80 percent of the global market share. According to the **2017 Stack Overflow Job Trends Report**, Android Developer is one of the Top-3 most in-demand developer positions in the job market.

In this Nanodegree program, you'll learn best practices for Android and mobile development, build a portfolio of Android apps, and publish your own app to Google Play. By the end of the program, you will have the skills you need to become a professional Android Developer.

WHAT JOBS WILL THIS PROGRAM PREPARE ME FOR?

This Nanodegree program is designed to prepare you for a job as a professional, junior-level Android Developer within a wide range of organizations and environments: from large corporations where you'd likely be part of a development team, to entrepreneurial start-ups and contract projects where you could be working independently to deliver an application.

HOW DO I KNOW IF THIS PROGRAM IS RIGHT FOR ME?

If you are a Java programmer who is interested in mastering the Android platform and building top-rated Android apps, this is the program for you.

The best way to see if you are ready for this Nanodegree program is to check out our free Developing Android Apps course. This is the first course in the program. If you are comfortable taking this course, you should be ready to enroll in the Android Developer Nanodegree program.

If you are not ready for this intermediate-level program, or if you are new to programming, please check out our Android Basics Nanodegree program instead of the Android Developer Nanodegree program.

ENROLLMENT AND ADMISSION

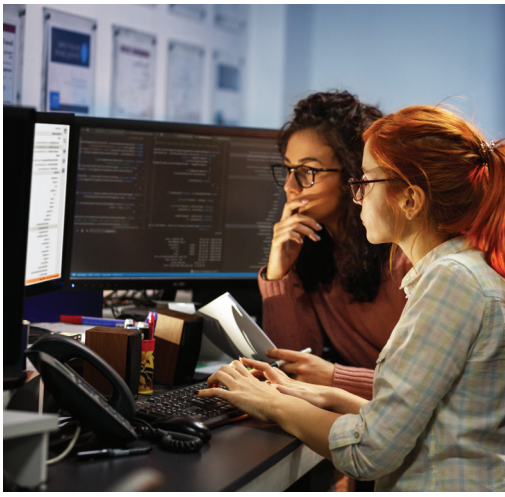
DO I NEED TO APPLY? WHAT ARE THE ADMISSION CRITERIA?

There is no application. This Nanodegree program accepts everyone, regardless of experience and specific background.

WHAT ARE THE PREREQUISITES FOR ENROLLMENT?

In order to succeed in this Nanodegree program, we strongly recommend that you are proficient in Java. We use Java exclusively for our example code and require you to submit your coding in Java.

FAQs Continued



If you are proficient in another object-oriented programming language like Python or C++, you should be able to complete the program successfully as long as you are comfortable learning Java throughout the Nanodegree program.

You also must have experience working with and sharing code using git and GitHub. If you lack this background, we recommend the following courses:

- Java Programming Basics
- How to Use Git and GitHub

If you are new to programming and want to get started developing Android apps, we recommend you check out our **Android Basics Nanodegree program**.

You will need to be able to communicate fluently and professionally in written and spoken English.

IF I DO NOT MEET THE REQUIREMENTS TO ENROLL, WHAT SHOULD I DO?

We have a number of Nanodegree programs and free courses that can help you prepare, including:

- **Android Basics Nanodegree program**
- **Java Programming Basics**
- How to Use Git and GitHub

TUITION AND TERM OF PROGRAM

HOW IS THIS NANODEGREE PROGRAM STRUCTURED?

The Android Developer Nanodegree program is comprised of content and curriculum to support eight (8) projects. We estimate that students can complete the program in six (6) months working 10 hours per week.
Each project will be reviewed by the Udacity reviewer network. Feedback will be provided and if you do not pass the project, you will be asked to resubmit the project until it passes.

HOW LONG IS THIS NANODEGREE PROGRAM?

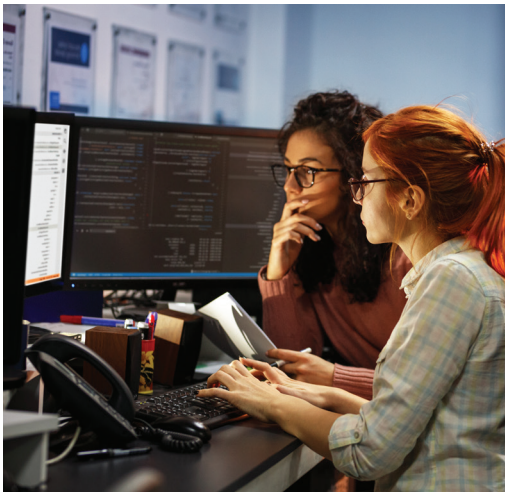
Access to this Nanodegree program runs for the length of time specified in the payment card above. If you do not graduate within that time period, you will continue learning with month to month payments. See the **Terms of Use** for other policies around the terms of access to our Nanodegree programs.

CAN I SWITCH MY START DATE? CAN I GET A REFUND?

Please see the Udacity Nanodegree program **FAQs** for policies on enrollment in our programs.

FAQs Continued

SOFTWARE AND HARDWARE



WHAT SOFTWARE AND VERSIONS WILL I NEED IN THIS PROGRAM?
All students will need a personal computer that is capable of running Android Studio. Please see the System Requirements listed on the Android Studio download page and ensure that your computer meets these minimum requirements.

Access to an Android device is helpful, but not necessary. You may use the emulator in Android Studio to run your apps if you do not have a physical Android device.