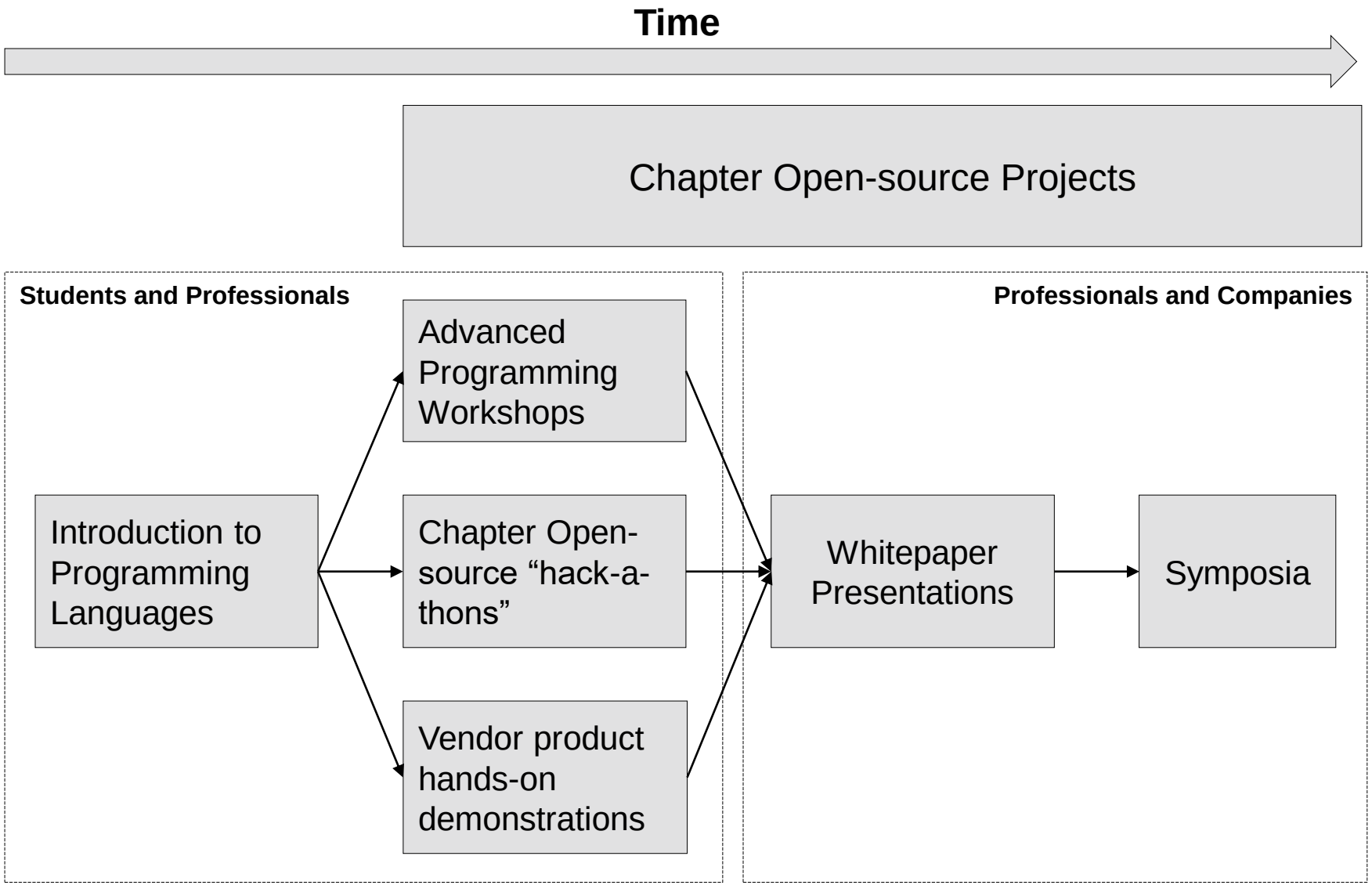
Android Programming Workshop I

Adam J. Cook, Chair of SME Chapter 112

Chapter “Digital” Initiative



Android at 10,000 Feet

- Free and **open-source** operating system and runtime platform.
- Apps written mostly in the Java (or Kotlin) programming language.
- Powerful and free development environment (Android Studio).
- Apps can be developed and simulated on any platform (Windows, MacOS or Linux).
- New Android devices sold were 81.7% globally in 2016 Q4. The rest is basically iOS. Many popular apps target both Android and iOS natively to give their user's the best experience on the devices that they prefer.
- Apps are deployed to a publicly available "app store" (Google Play).
- Apps can be free, paid or consist of various in-app payment schemes. **We will talk a little more about this.**
- Apps can be upgraded "in-place" on user devices when a new version is released.
- Android apps can run on watches, TVs, VR headsets, tablets, automobiles and phones.

What does “open-source” mean?

- The Open Source Definition - https://opensource.org/osd-annotated
- Free redistribution, royalty-free.
- Source code availability.
- Popular open-source licenses – MIT, BSD, Apache 2.0, GPL, LGPL.
- The use of open-source components in larger applications is very common and you can use them too!
- **Always** check with your legal representation if a non-standard license is encountered or if you are unsure of your legal rights and responsibilities with the standard open-source licenses.

Today's Agenda

- Explore GitHub and emulate an existing Android project to check if your computer is set up correctly.
- A quick review and then we will walk through an Android project from scratch (the best that we can).
- Discussion of some very useful resources for future study.

Caveats and Warnings

green COW coworking

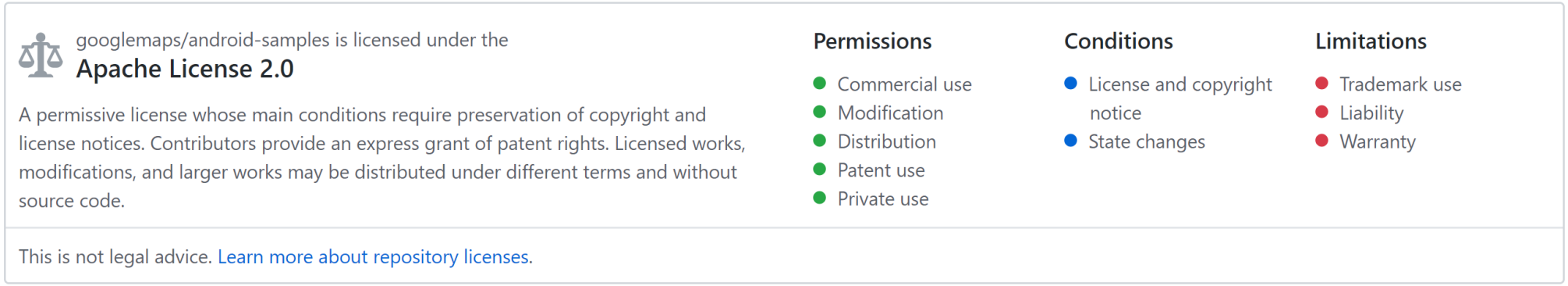
sme

- Programming is challenging – this event will not make you into an expert. **Practice, read code, read books, watch videos and ask questions! It will take time, but you can do it!**
- Android app development is a combination of visual programming and Java (or Kotlin) programming. These can be complex languages at times and will require some work to become comfortable with.
- Every programming project should use a version control system. Git is **highly** recommended. See the resources at end.
- “Premature optimization is the root of all evil.” – Donald Knuth

GitHub

- GitHub is a code hosting repository and source control platform (using Git). It holds public (open-source) and private code repositories.
- There are **tons** of open-source and valuable Android libraries, demos and “starter apps” available that can save you time. Let’s check out Awesome Python on GitHub.
- Let’s grab the ZIP file for a starter app that works with the Google Maps API to get started! Here is the link: http://bit.ly/2tr0Av3

Wait! Always check the license!



googlemaps/android-samples is licensed under the
Apache License 2.0

A permissive license whose main conditions require preservation of copyright and license notices. Contributors provide an express grant of patent rights. Licensed works, modifications, and larger works may be distributed under different terms and without source code.

Permissions
- Commercial use
- Modification
- Distribution
- Patent use
- Private use

Conditions
- License and copyright notice
- State changes

Limitations
- Trademark use
- Liability
- Warranty

This is not legal advice. Learn more about repository licenses.

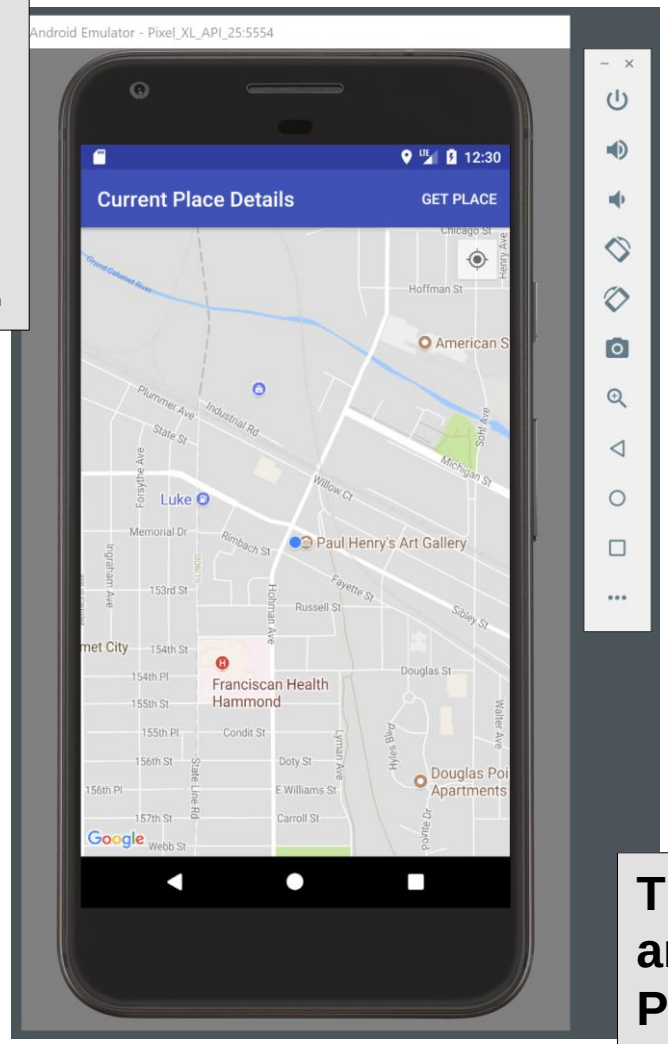
Tip! Sometimes this file is called LICENSE, LICENSE.md, LICENSE.txt, COPYING, COPYING.txt (those are the most common, but it can be called anything).

Current Place Details On Map Tutorial

- To use the Google Maps API, you need an **API Key**. If you already have one, use it. If you don't and you are in this workshop, you can use mine. If you want to see how to get your own key for later, see here: https://developers.google.com/maps/documentation/android-api/signup
- When problems crop up, Stack Overflow and Google are your friends (or contact me and I will try to help). Trust me. There will be problems from time-to-time. Some possibly right now!
- **Tip!** If you app times out prior to actually starting in the AVD, follow these instructions: https://stackoverflow.com/a/43187806. If that does not work, try deleting and re-creating your AVD.

Current Place Details On Map Tutorial

If everything went as planned, you should see something like this in your AVD.



This was run on an emulated Pixel XL at API Level 25.

Diving Deeper

What are the fundamental parts of an Android application?

- App components. These are entry points through which the system or the user can enter your app.
- There are four different types:
  - Activities.
  - Services.
  - Content providers.
  - Broadcast receivers.

Sources:
https://developer.android.com/guide/components/fundamentals.html#Components

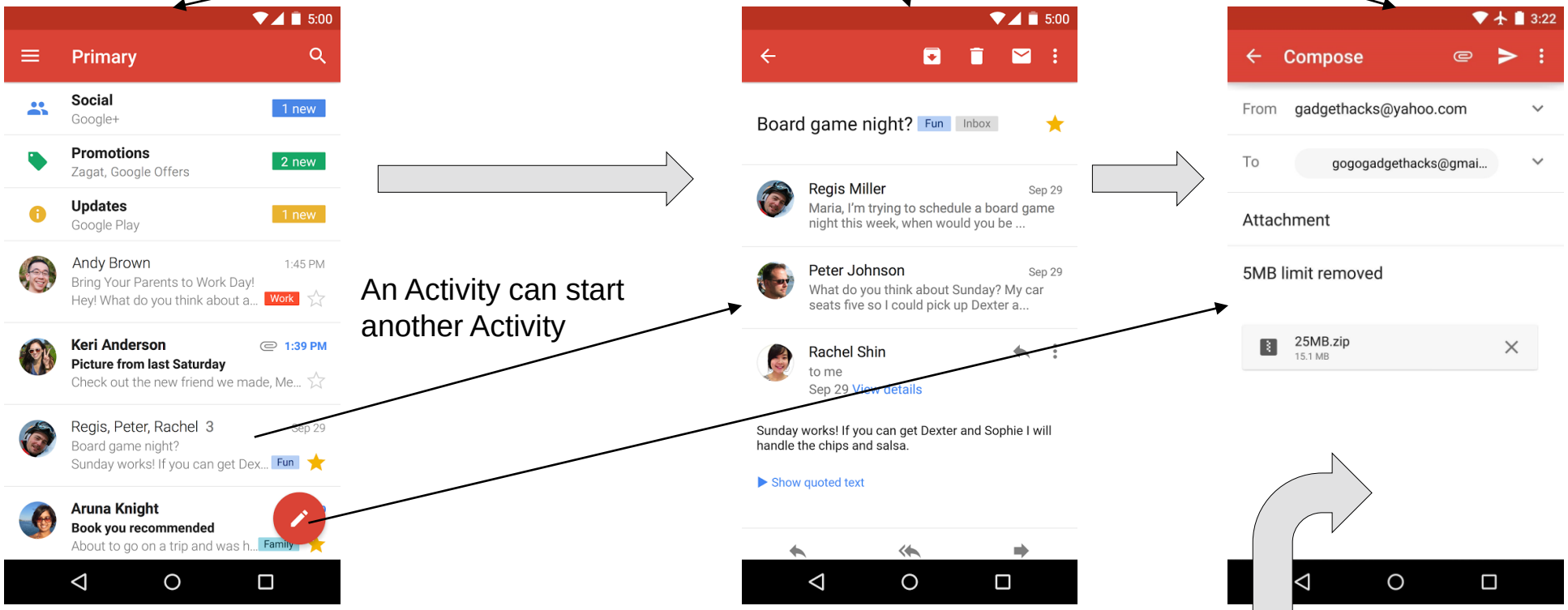
Diving Deeper

What is an Activity?

Each of these is a separate Activity – one app can have multiple Activities

An Activity is an entry point for interacting with the user. The UI.

Gmail Android app



An Activity can start another Activity

"Gallery" app can find entry into this Activity, for example

Diving Deeper

What is a Service?

- An entry point for keeping an app running in the background.
- Does not provide a user interface.
- Why would this even be useful? Several possible reasons:
  - Long running network requests.
  - Intensive computations.
  - Play music while in your pocket.
  - Hail a cab and keep tracking your location while in your pocket.

Diving Deeper

So that's it? Nope. There is so much more.

- **Intents – messaging object which can start an Activity or start a Service**
- **Fragments – a common, resuable portion of an Activity to use across multiple Activities**
- **App Resources – images and (possibly localized) text for your app**
- App Manifest – much of the actual content of the manifest is handled for you by Android Studio, but its still helpful to know what is going on in there.
- Activity Lifecycle
- Loaders – for "asynchronous" loading of remote resources.

Diving Deeper

Anything else? Yep! Get even more familiar with Java and Android and then look into these topics.

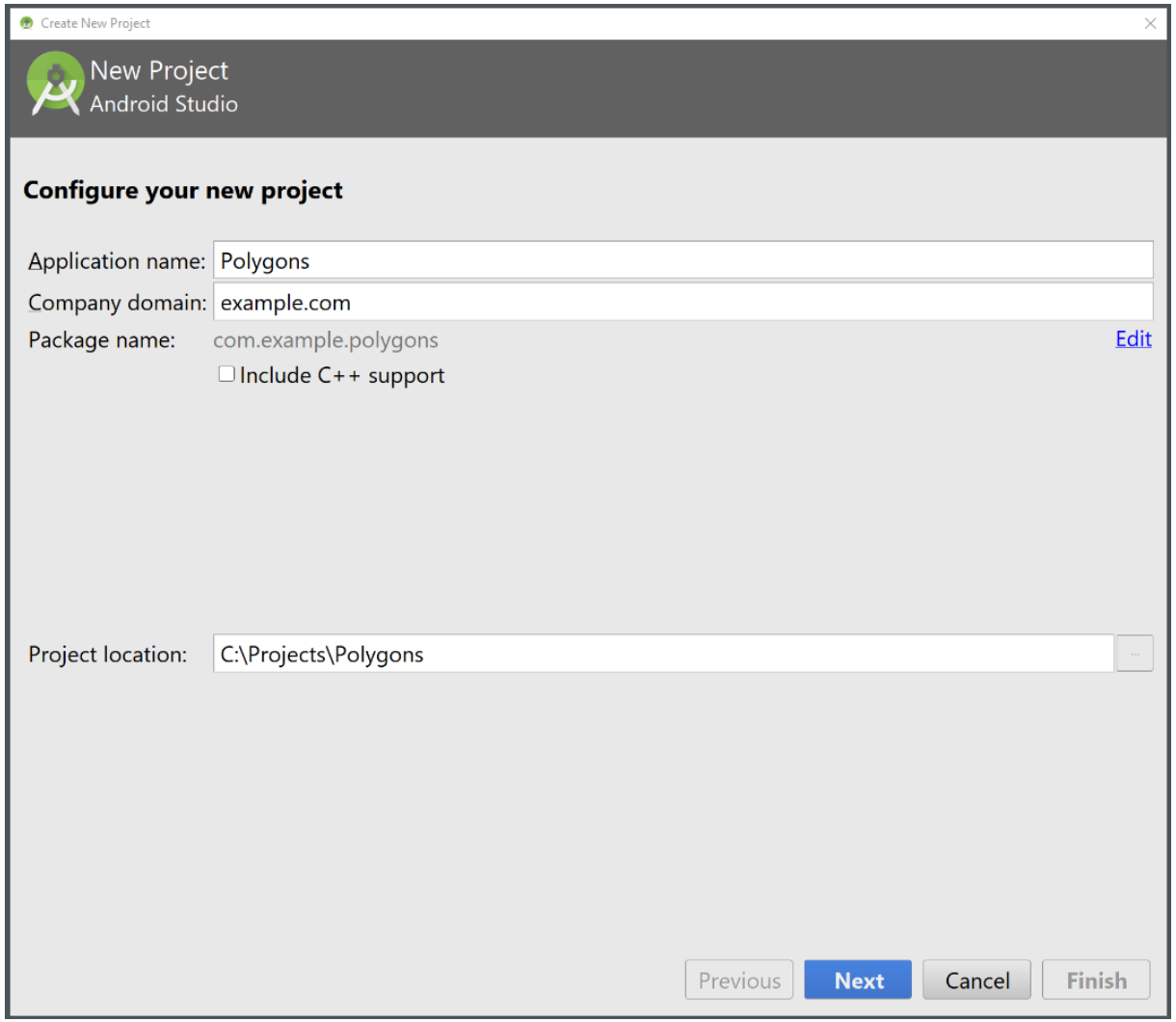
- Building your app with Gradle
- Debugging your app
- **Testing your app**
- Profiling your app
- Publishing your app
- Debugging, testing and profiling can be tough topics. **Practice!** In time, you will get better.

Creation is Messy

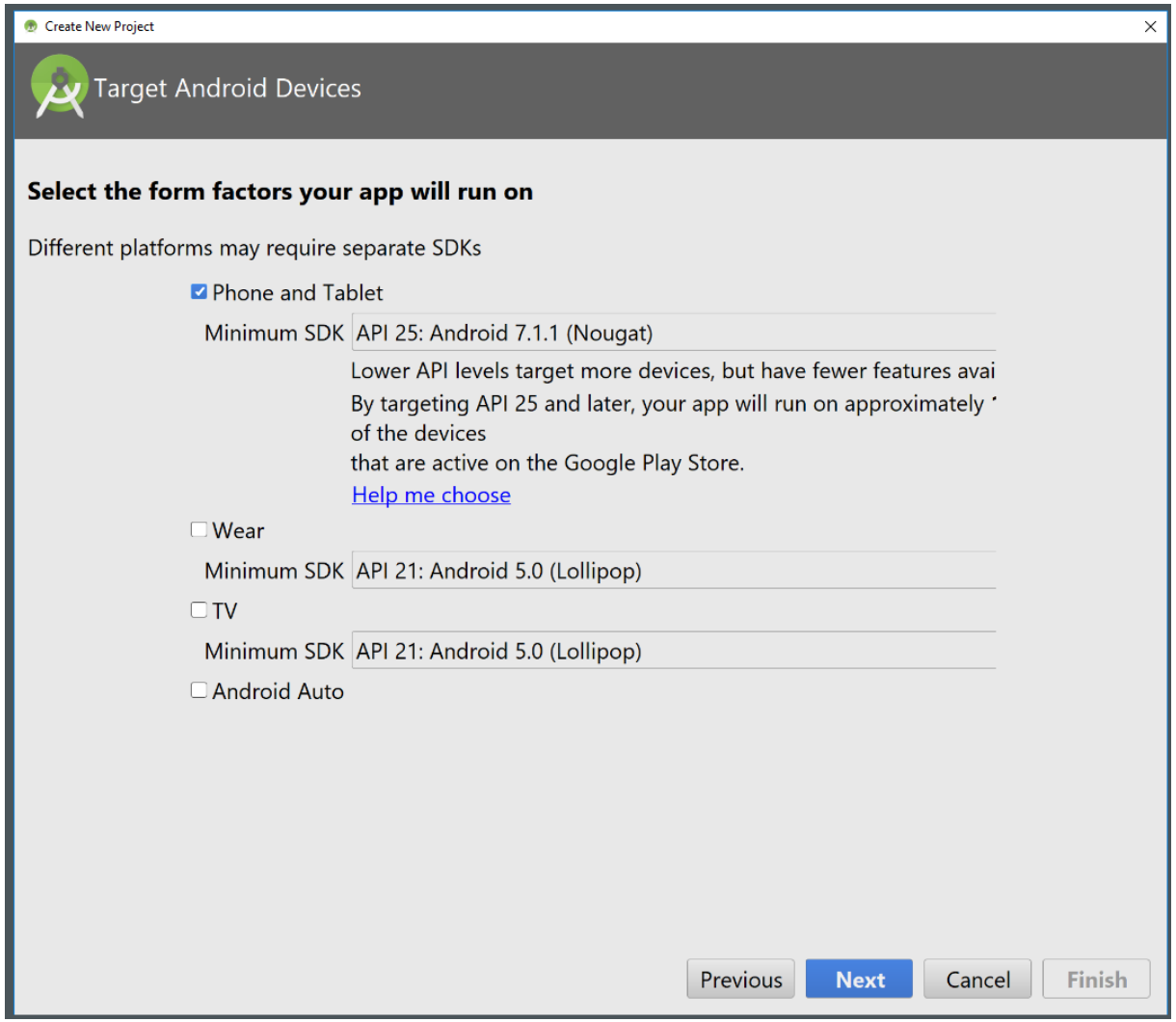
Let’s try and create an Android app project from scratch.

What we do here will very closely follow a more advanced map tutorial (called “Polygons”) which can be found here: http://bit.ly/2tr0Av3

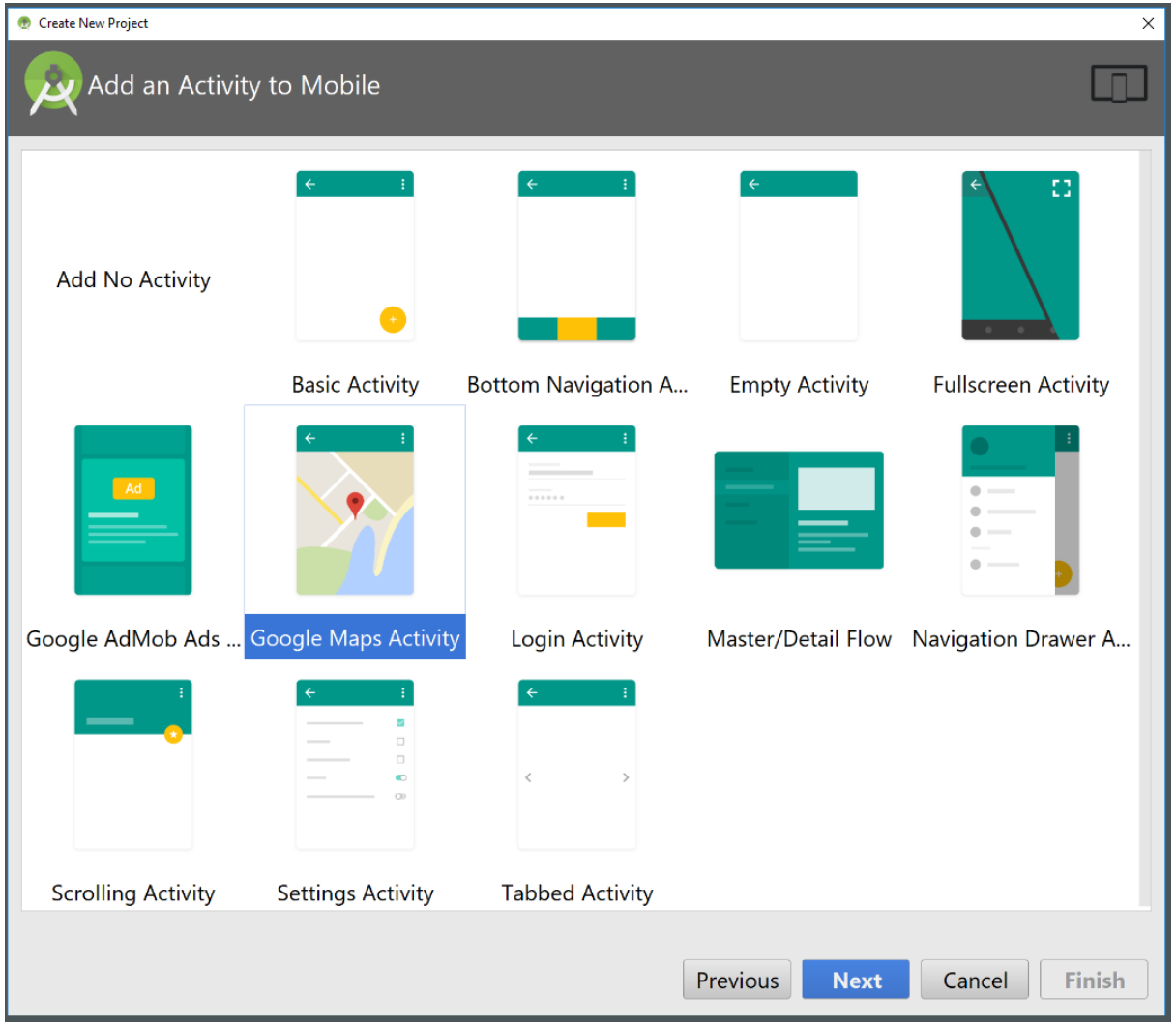
Polygon Overlay on Map App



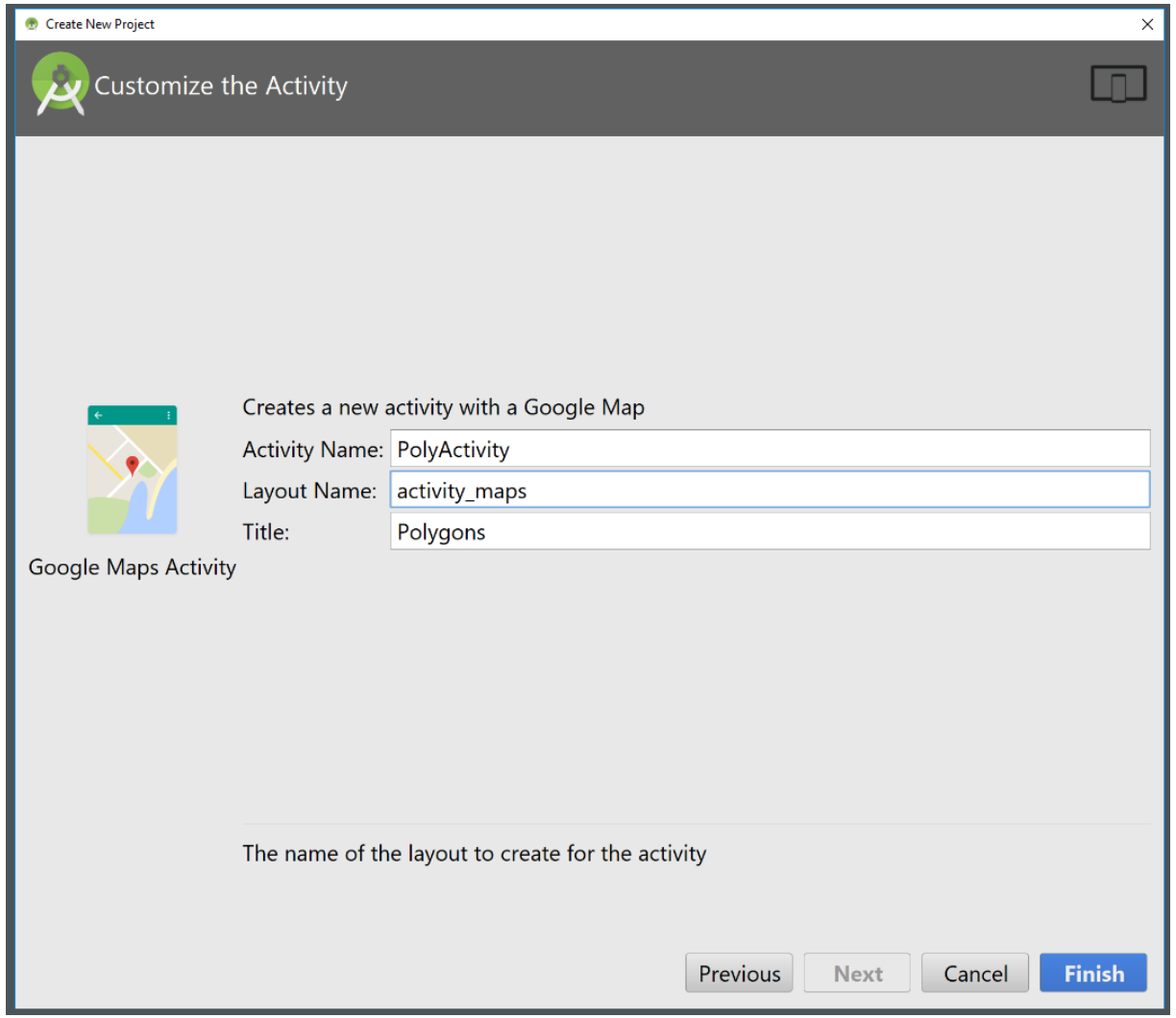
Polygon Overlay on Map App



Polygon Overlay on Map App



Polygon Overlay on Map App



What just happened? Android Studio created a bunch of starter files based on your input. Let's look at some of them right away.

- **manifests/AndroidManifest.xml**
- **res/values/google_maps_api.xml**
- **res/layout/activity_maps.xml**
- **java/com.example.polygons/PolyActivity.java (the "activity code behind" file)**

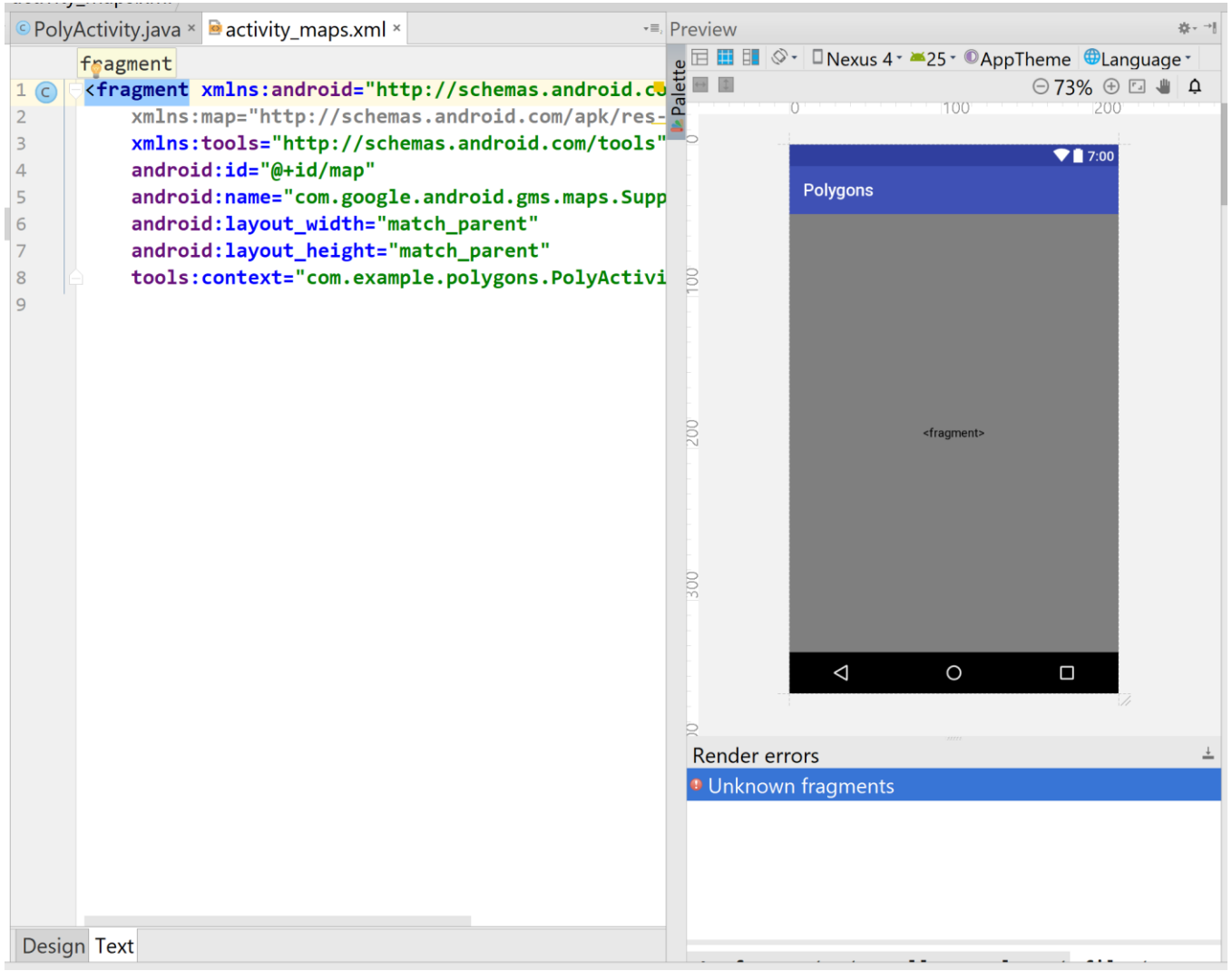
Polygon Overlay on Map App - Manifest

green COW coworking

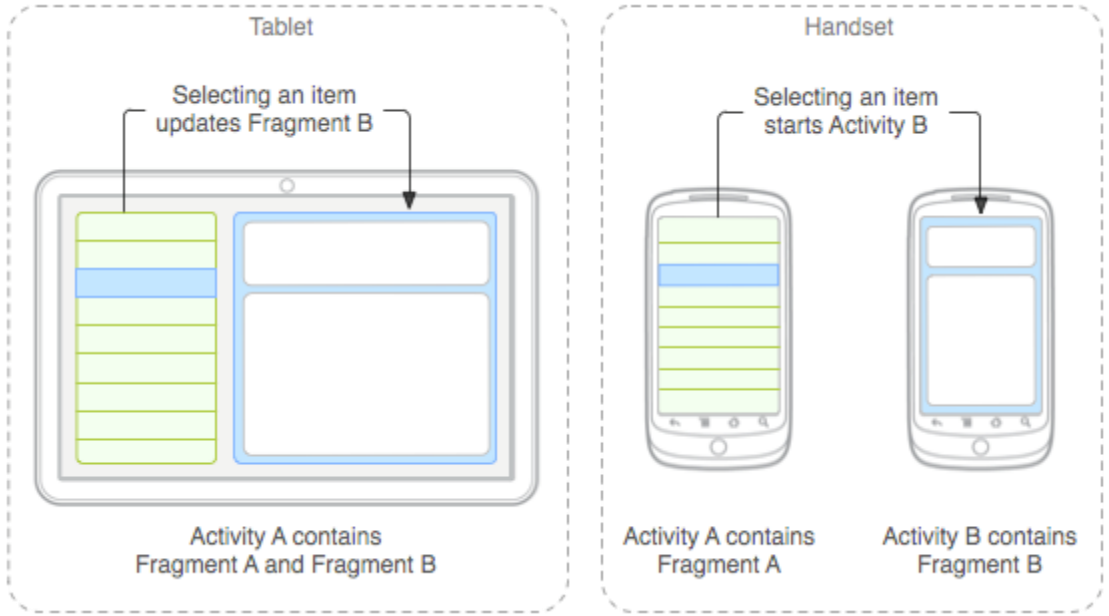
sme

Let’s look at the first two files (manifest and resource file).

Polygon Overlay on Map App - Layout



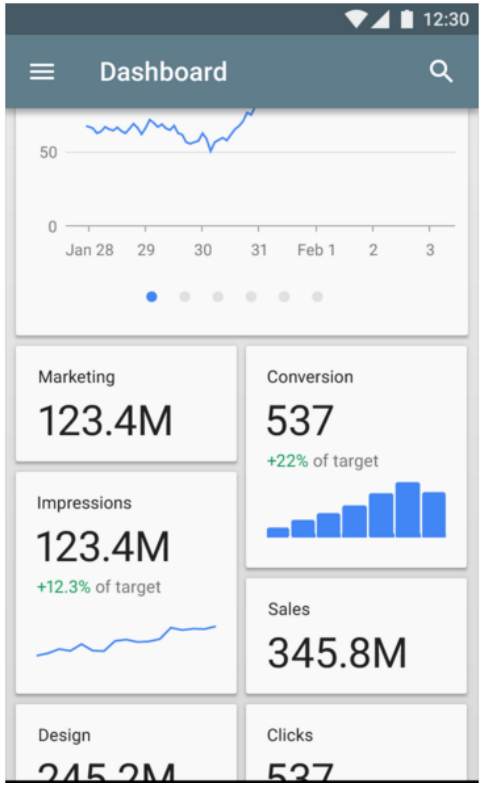
What is a fragment?



Want the full scoop? Check out the this page.

Polygon Overlay on Map App - Layout

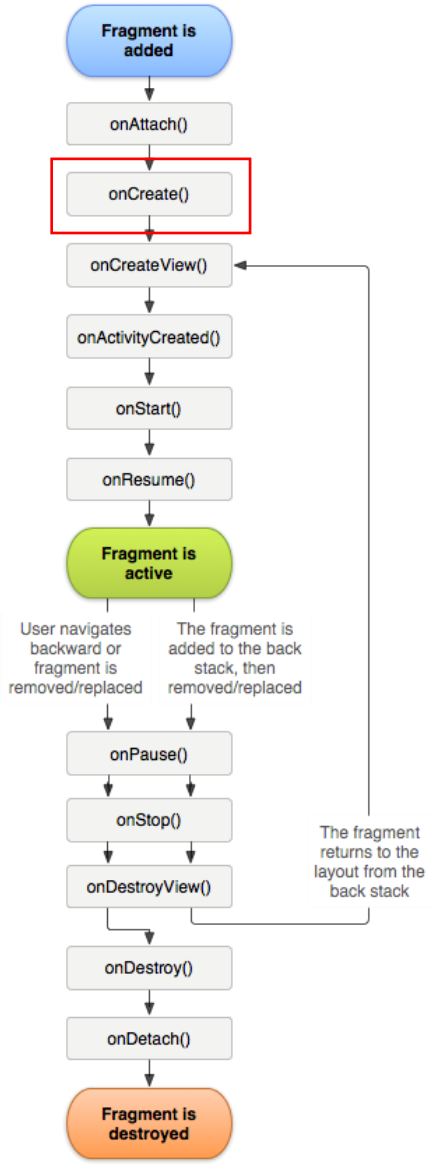
What are layout tags?



For more information, visit this page for information on Android Layouts.

Polygon Overlay on Map App - Layout

Let’s talk (a little) about lifecycle.



Polygons App - Layout

What's in a name? In this case, it is a quick, out-of-the-box way to drop a map into your app. See more here: http://bit.ly/2uO1G3i

Polygons App - Activity Code Behind

green COW coworking

sme

Great! So we now have a user interface. But how do you make it actually do something? That magic happens in the “activity code behind” file. Let’s dive into the auto-generated file.

Polygons App - Activity Code Behind

green COW coworking

sme

Check that the auto-generated code is sane by running it before moving on.

Polygons App - Activity Code Behind

green COW coworking

sme

The auto-generated code behind is boring. Let's make it more interesting.

The interesting replacement code is here: http://bit.ly/2uujlbG

We also have to copy in a new icon resource. This can be found here:

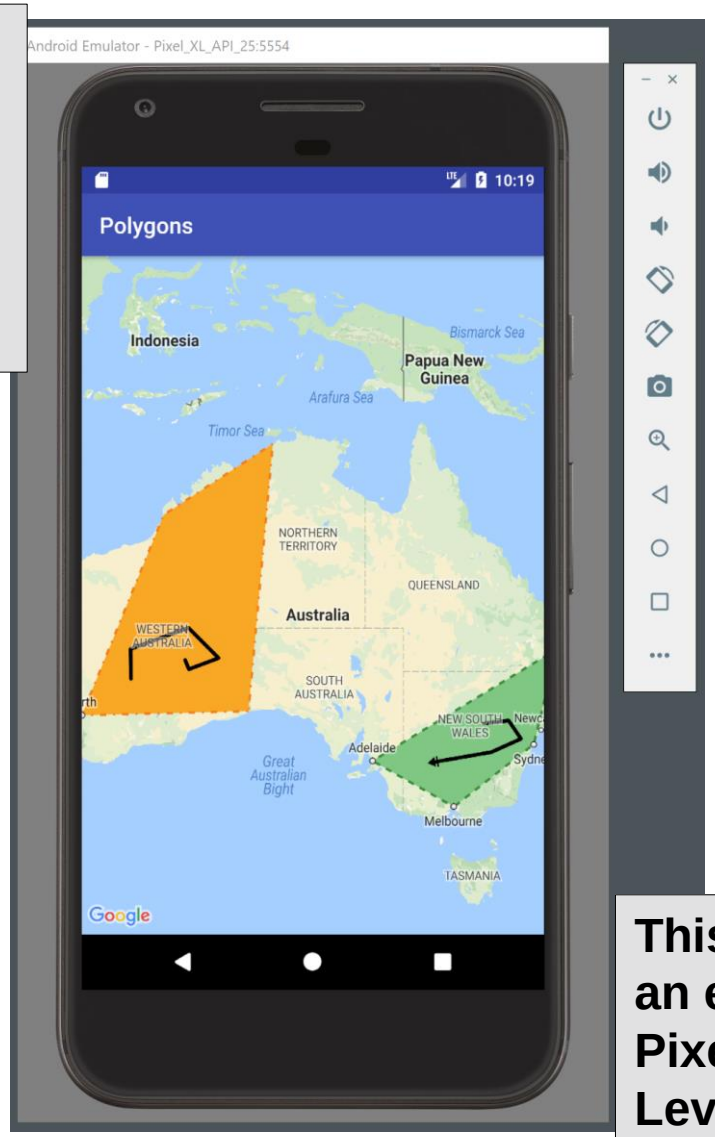
(If there is time, we will try to walk through this code a little.)

Polygon Overlay on Map App

It’s show time! Cross your fingers and press run.

Current Place Details On Map Tutorial

If everything went as planned, you should see something like this in your AVD.



This was run on an emulated Pixel XL at API Level 25.

Learning Resources

- **Book - Android Programming: The Big Nerd Ranch Guide (3rd Edition)**
- **Web – Guide to App Architecture**
- **Web – Android Best Practices**
- **Web – Google Samples**
- **Book - Java: A Beginner's Guide (6th Edition)**
- **Video Course – Developing Android Apps by Google (free)**
- Web - Stack Overflow Android
- Web - Pro Git (free online book)
- Web - Getting Started on Android Developer site
- **Web - Awesome Python on GitHub**

Popular Third-party Components

- Litho - a declarative UI framework for Android
- Unity 3D game engine (Android backend)
- React Native - a framework for building native apps with React
- MPAndroidChart - a visually-appealing and powerful charting/graphing library
- RxJava - reactive extensions for the JVM

Thank you!

green COW coworking

sme

Any questions?

Any suggestions for future events? Any interest in a Java class?

The slide deck for this event can be found here for future reference: http://bit.ly/2ttZ9vO

Thank you!

Thanks for attending!

Special thanks to our hosting partners – greenCOW Coworking.

Adam J. Cook

Chief Technical Officer of Alliedstrand

Chair of SME Chapter 112

adam.j.cook@alliedstrand.com

www.alliedstrand.com

https://twitter.com/AdamJosephCook

https://linkedin.com/in/adam-j-cook

https://github.com/adamjcook

SME

www.sme.org

https://facebook.com/SMEmfg

https://twitter.com/SME_MFG

https://linkedin.com/company/sme

SME Chapter 112

Serving Northwest Indiana and Chicagoland

https://twitter.com/sme112

https://facebook.com/sme112

https://github.com/sme112