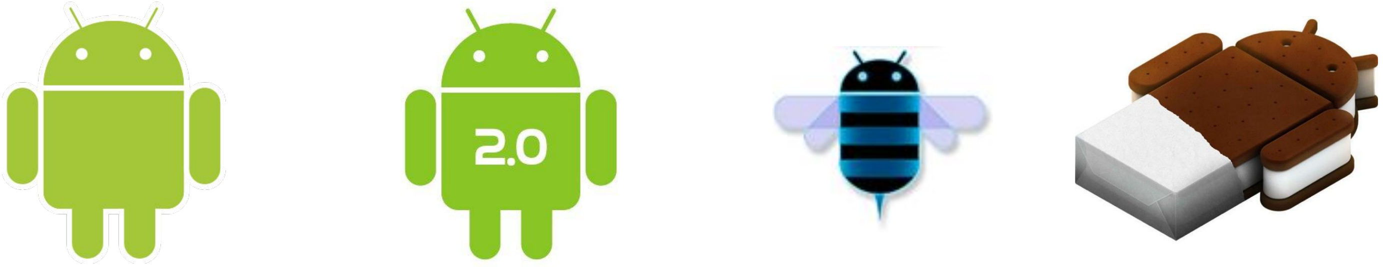


Programming with Android: SDK install and Initial setup

Federico Montori

Dipartimento di Informatica: Scienza e Ingegneria
Università di Bologna

SDK and initial setup: Outline

- Today:
  - How to setup a machine to start developing Android applications
  - An overview of an Android project
  - Some useful tools
  - Your first Android application
    - Maybe on a real device!

(Not really) two options

- Historically Android development was in Eclipse
  - Download the SDK
  - Install the Android plugin

- Now there is Android Studio: the official development platform
  - Customized environment

Which one?

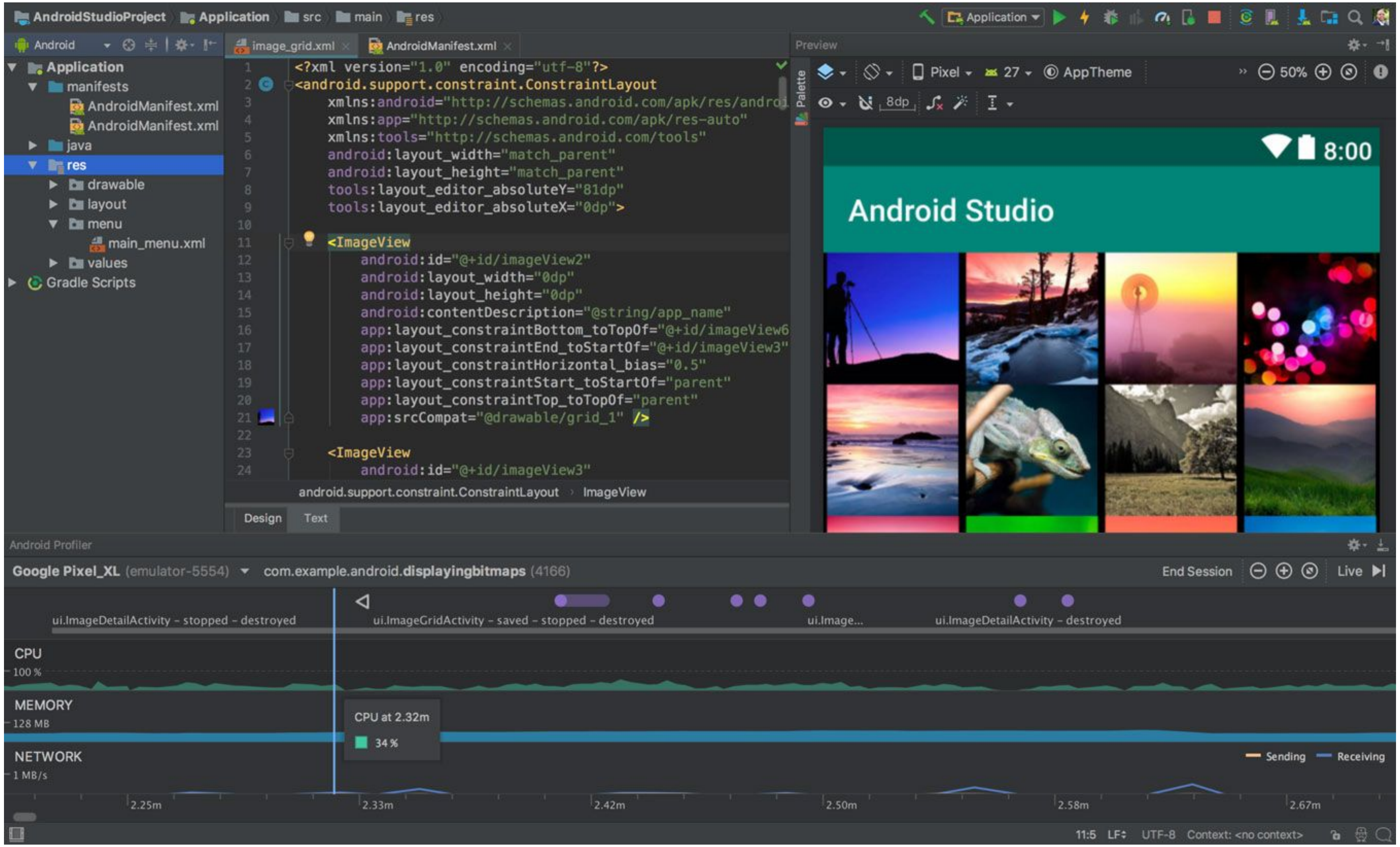
- For LAM, it’s the same: if you already use and love Eclipse, go with it
  - You might see examples done in both platforms
    - You can always import an Eclipse project into Android Studio
    - The inverse is difficult

BUT

- Eclipse SDK has been DEPRECATED since the end of 2015, which means newer versions of Android are no longer supported.
  - Want a better explanation?

https://android-developers.googleblog.com/2015/06/an-update-on-eclipse-android-developer.html

Android Studio



```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/androi
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:layout_editor_absoluteY="81dp"
    tools:layout_editor_absoluteX="0dp">

    <ImageView
        android:id="@+id/imageView2"
        android:layout_width="0dp"
        android:layout_height="0dp"
        android:contentDescription="@string/app_name"
        app:layout_constraintBottom_toTopOf="@+id/imageView6
        app:layout_constraintEnd_toStartOf="@+id/imageView3"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:srcCompat="@drawable/grid_1" />

    <ImageView
        android:id="@+id/imageView3"
```

android.support.constraint.ConstraintLayout › ImageView

Google Pixel_XL (emulator-5554) com.example.android.displayingbitmaps (4166)

ui.ImageDetailActivity - stopped - destroyed

ui.ImageGridActivity - saved - stopped - destroyed

ui.Image...

ui.ImageDetailActivity - destroyed

11:5 LF UTF-8 Context: <no context>

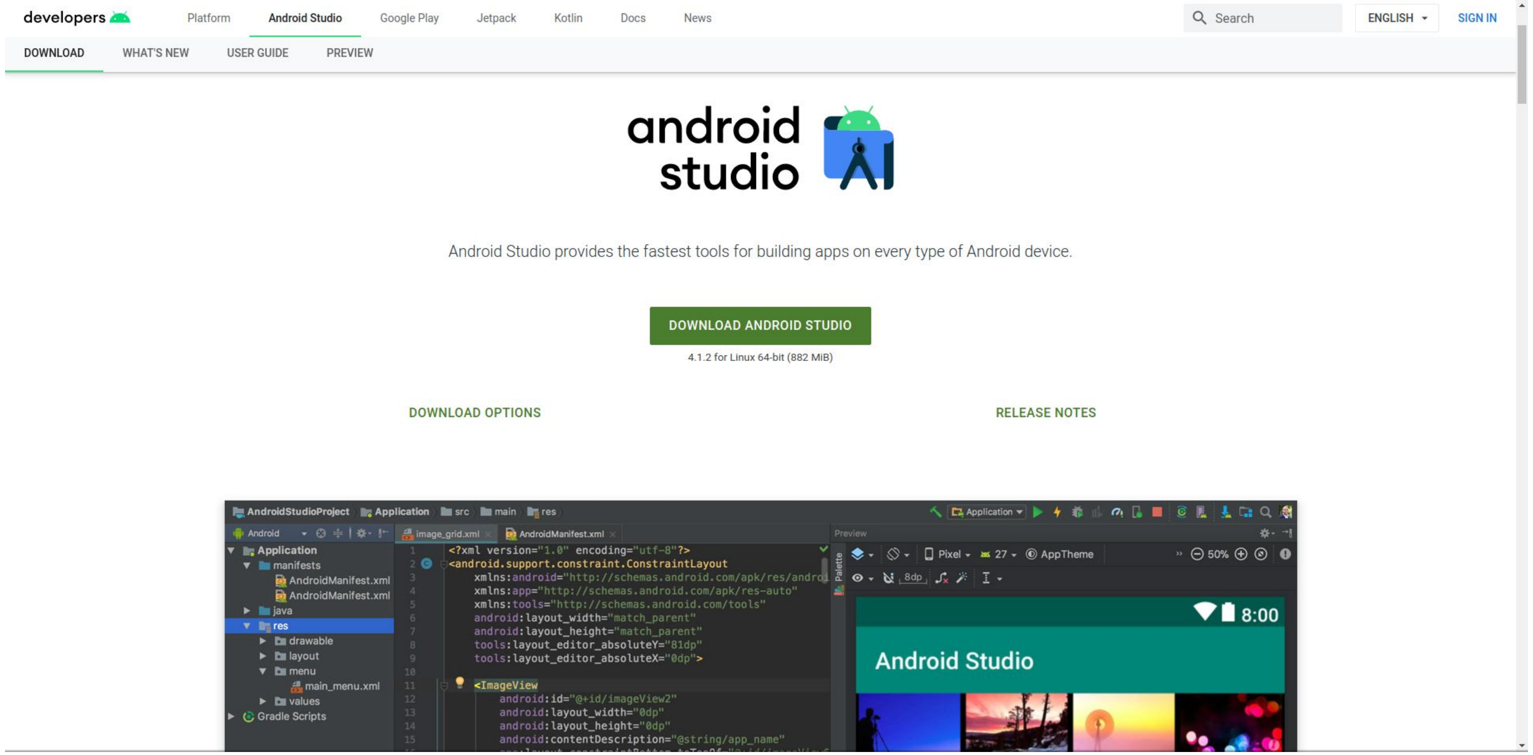
Where and how to get it

- Go to https://developer.android.com/studio
- Download Android Studio and the SDK
- Install it and you're done!

Version Bumblebee

2021.1.1 patch 1

at the time of writing



Android Studio provides the fastest tools for building apps on every type of Android device.

4.1.2 for Linux 64-bit (882 MiB)

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/androi
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:layout_editor_absoluteY="81dp"
    tools:layout_editor_absoluteX="0dp">

    <ImageView
        android:id="@+id/imageView2"
        android:layout_width="0dp"
        android:layout_height="0dp"
        android:contentDescription="@string/app_name"
```

What's new compared to past two years...

Changes from 3.5.3 to 4.1.2

- Database Inspector for querying application databases
- Studio templates now use Material Design Components and Themes
- Ability to run Android Emulator inside of Android Studio
- Support for Dagger Navigation library
- Support for TensorFlow Lite models
- All improvements from IntelliJ IDEA 2020.1.4
- Support for foldables in the Android emulator
- Faster Builds using Apply Changes for devices running Android 11 or higher
- Ability to export C/C++ dependencies from AARs
- Improvements to System Trace UI and Native Memory Profiling

What's new compared to past two years...

Changes from 4.1.2 to Bumblebee

- New Device Manager
  - The old AVD manager has been removed, now there is a new tab that lets you quickly get access to all the virtual and physical devices.
- ADB (android debug bridge) now works **over WiFi** on android phones that run Android 11 or later.
- Android Gradle Plugin for running instrumented tests
- Other stuff…

Basic requirements

- How to develop Android applications?
  - Linux / Mac OS / Windows? Doesn’t matter
  - A real device is not mandatory
    - although suggested...

- Code your application in the IDE
  - Test it with the emulator
  - Deploy it on a real device (if you can)

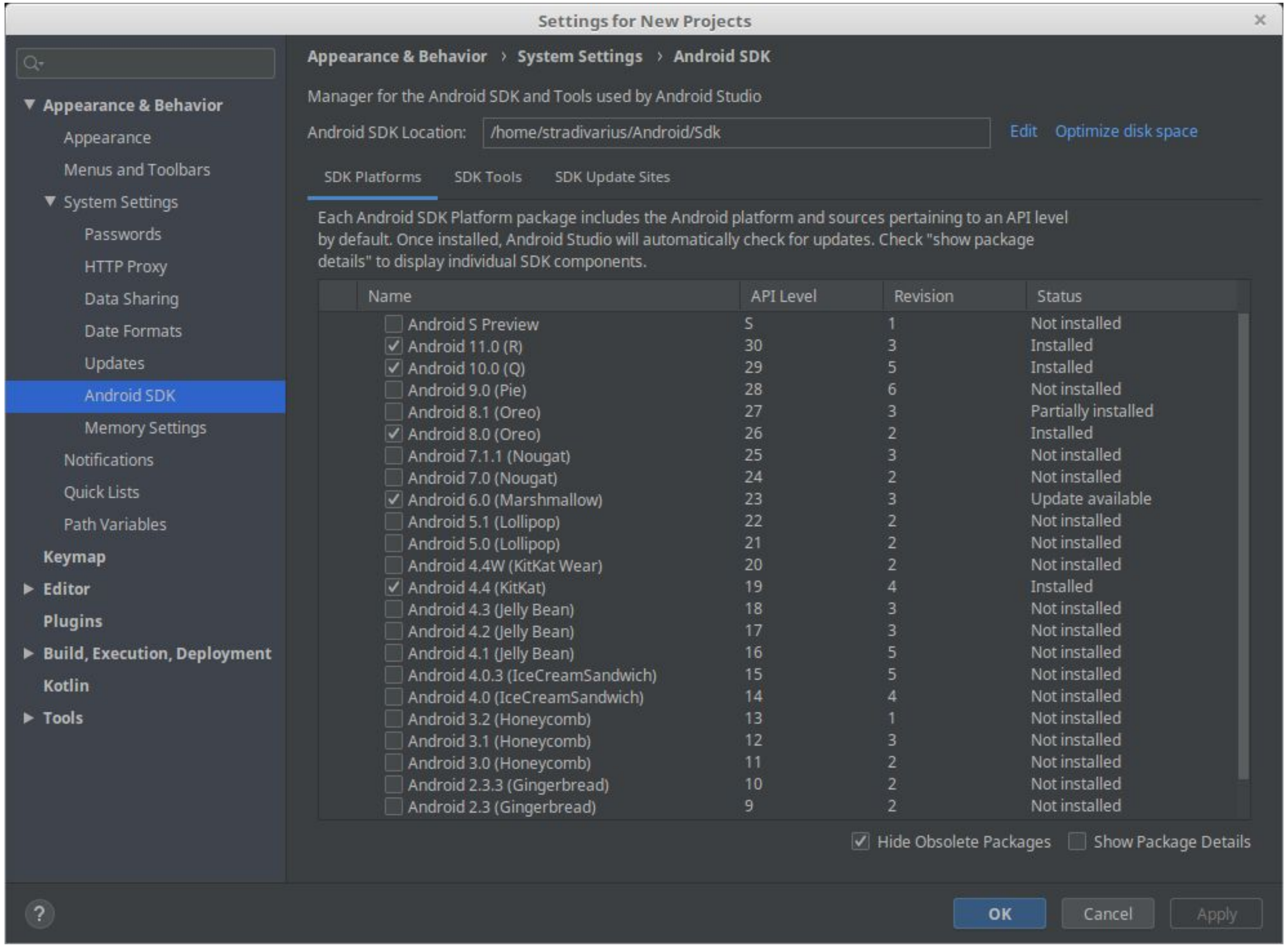
A bit of terminology

- **SDK (Software development kit)**
  - A set of tools that help you in creating software
  - Compiler, tester, debugger, **libraries**
- **IDE (Integrated Development Environment)**
  - Graphical environment in which all the tools are accessible.
- **API (Application Program Interface)**
  - A set of calls that the underlying world exposes to the developer for interaction.
  - It does <u>not</u> correspond to “libraries”.

SDK manager

Tools > SDK Manager

- Android tool
- Used to get APIs and add-ons
- You can also start it from Android Studio
- It'll affect the compileSdkVersion (details later)



Manager for the Android SDK and Tools used by Android Studio

Android SDK Location: /home/stradivarius/Android/Sdk

Each Android SDK Platform package includes the Android platform and sources pertaining to an API level by default. Once installed, Android Studio will automatically check for updates. Check "show package details" to display individual SDK components.

Name	API Level	Revision	Status
□ Android S Preview	S	1	Not installed
☑ Android 11.0 (R)	30	3	Installed
☑ Android 10.0 (Q)	29	5	Installed
□ Android 9.0 (Pie)	28	6	Not installed
□ Android 8.1 (Oreo)	27	3	Partially installed
☑ Android 8.0 (Oreo)	26	2	Installed
□ Android 7.1.1 (Nougat)	25	3	Not installed
□ Android 7.0 (Nougat)	24	2	Not installed
☑ Android 6.0 (Marshmallow)	23	3	Update available
□ Android 5.1 (Lollipop)	22	2	Not installed
□ Android 5.0 (Lollipop)	21	2	Not installed
□ Android 4.4W (KitKat Wear)	20	2	Not installed
☑ Android 4.4 (KitKat)	19	4	Installed
□ Android 4.3 (Jelly Bean)	18	3	Not installed
□ Android 4.2 (Jelly Bean)	17	3	Not installed
□ Android 4.1 (Jelly Bean)	16	5	Not installed
□ Android 4.0.3 (IceCreamSandwich)	15	5	Not installed
□ Android 4.0 (IceCreamSandwich)	14	4	Not installed
□ Android 3.2 (Honeycomb)	13	1	Not installed
□ Android 3.1 (Honeycomb)	12	3	Not installed
□ Android 3.0 (Honeycomb)	11	2	Not installed
□ Android 2.3.3 (Gingerbread)	10	2	Not installed
□ Android 2.3 (Gingerbread)	9	2	Not installed

☑ Hide Obsolete Packages □ Show Package Details

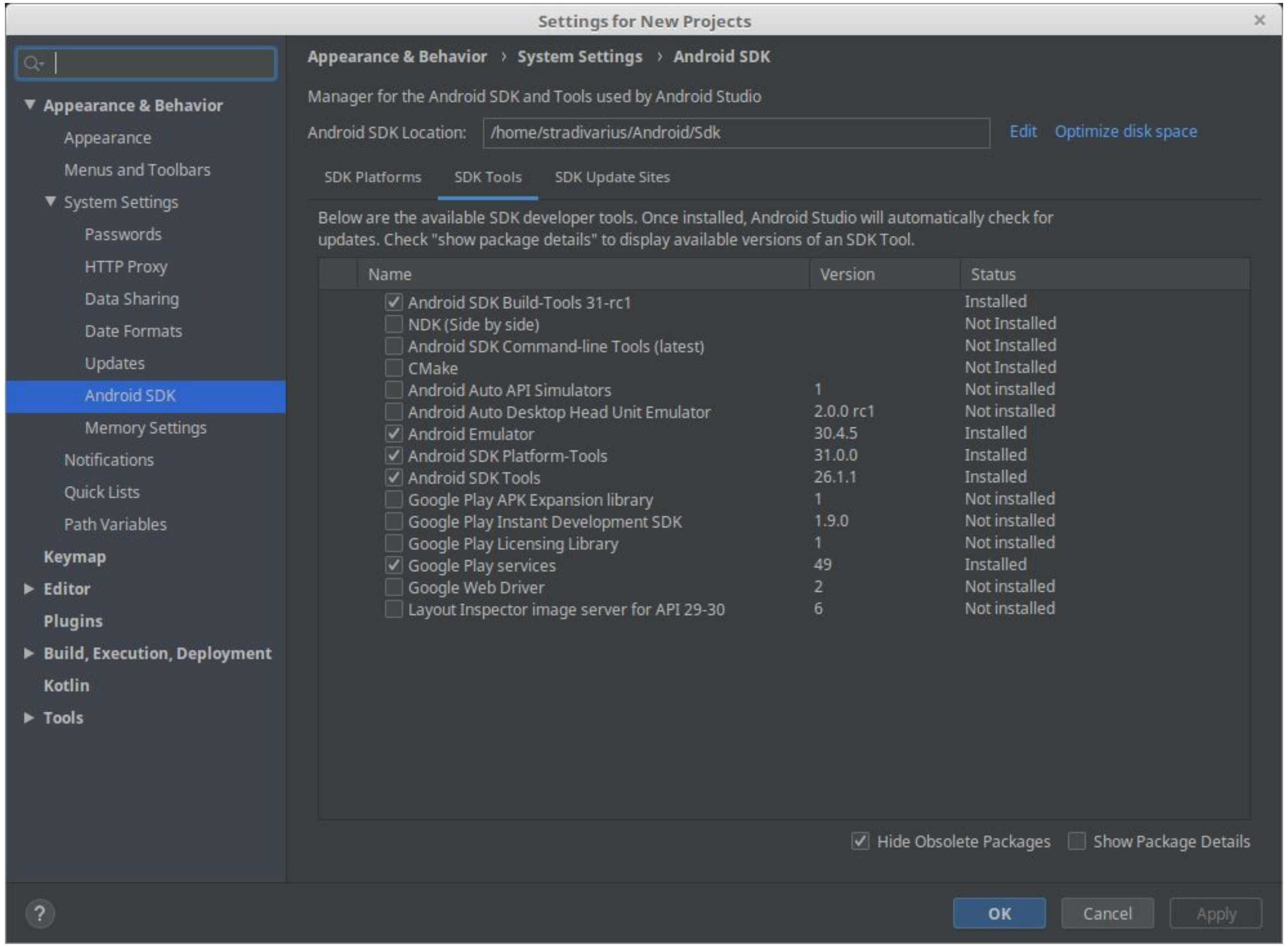
SDK manager

Tools > SDK Manager

- Android tool

- Used to get APIs and add-ons

- You can also start it from Android Studio



Appearance & Behavior › System Settings › Android SDK

Manager for the Android SDK and Tools used by Android Studio

Android SDK Location: /home/stradivarius/Android/Sdk

Below are the available SDK developer tools. Once installed, Android Studio will automatically check for updates. Check "show package details" to display available versions of an SDK Tool.

Name	Version	Status
☑ Android SDK Build-Tools 31-rc1		Installed
□ NDK (Side by side)		Not Installed
□ Android SDK Command-line Tools (latest)		Not Installed
□ CMake		Not Installed
□ Android Auto API Simulators	1	Not installed
□ Android Auto Desktop Head Unit Emulator	2.0.0 rc1	Not installed
☑ Android Emulator	30.4.5	Installed
☑ Android SDK Platform-Tools	31.0.0	Installed
☑ Android SDK Tools	26.1.1	Installed
□ Google Play APK Expansion library	1	Not installed
□ Google Play Instant Development SDK	1.9.0	Not installed
□ Google Play Licensing Library	1	Not installed
☑ Google Play services	49	Installed
□ Google Web Driver	2	Not installed
□ Layout Inspector image server for API 29-30	6	Not installed

☑ Hide Obsolete Packages □ Show Package Details

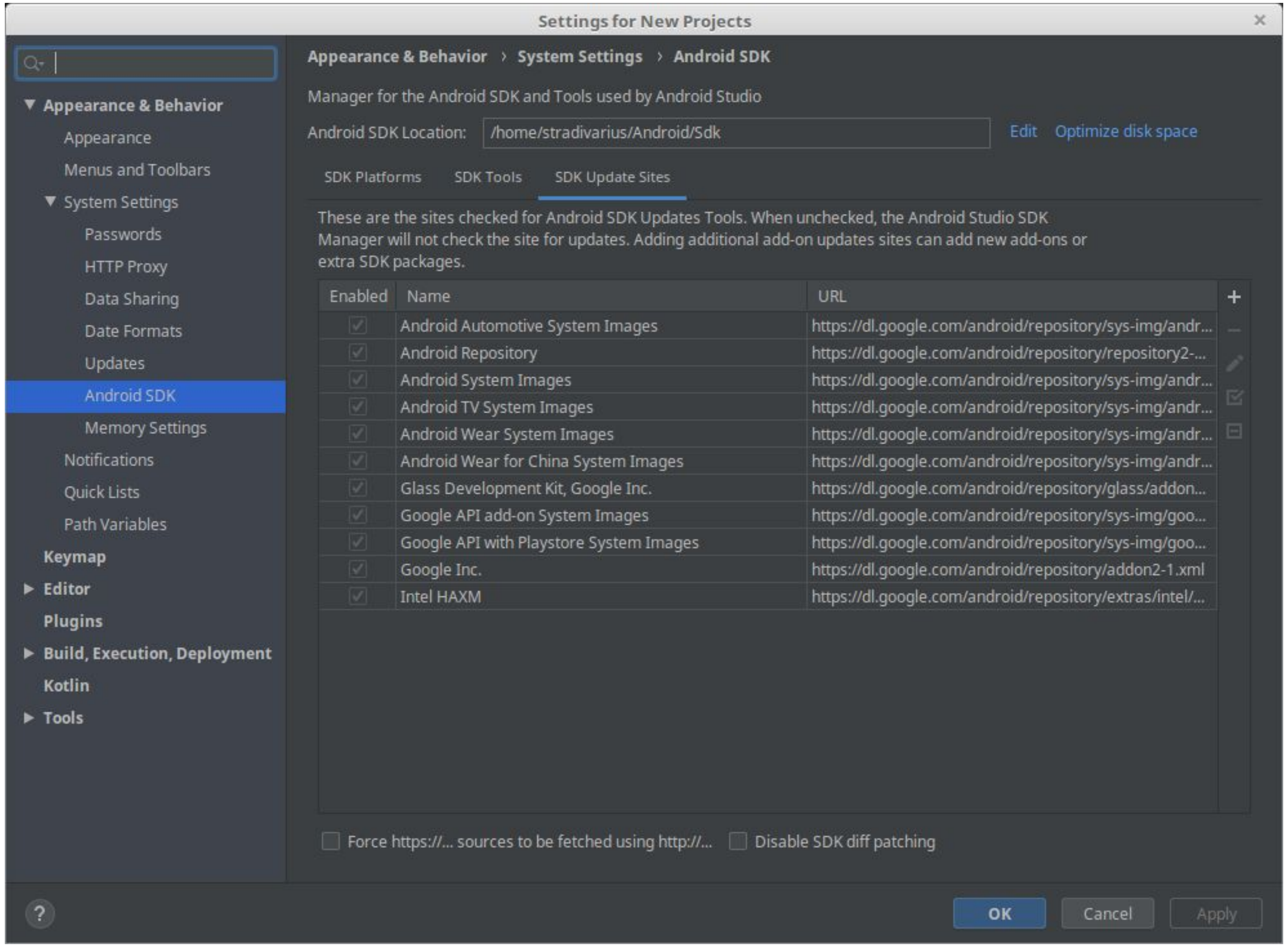
SDK manager

Tools > SDK Manager

- Android tool

- Used to get APIs and add-ons

- You can also start it from Android Studio



Appearance & Behavior › System Settings › Android SDK

Manager for the Android SDK and Tools used by Android Studio

Android SDK Location: /home/stradivarius/Android/Sdk

These are the sites checked for Android SDK Updates Tools. When unchecked, the Android Studio SDK Manager will not check the site for updates. Adding additional add-on updates sites can add new add-ons or extra SDK packages.

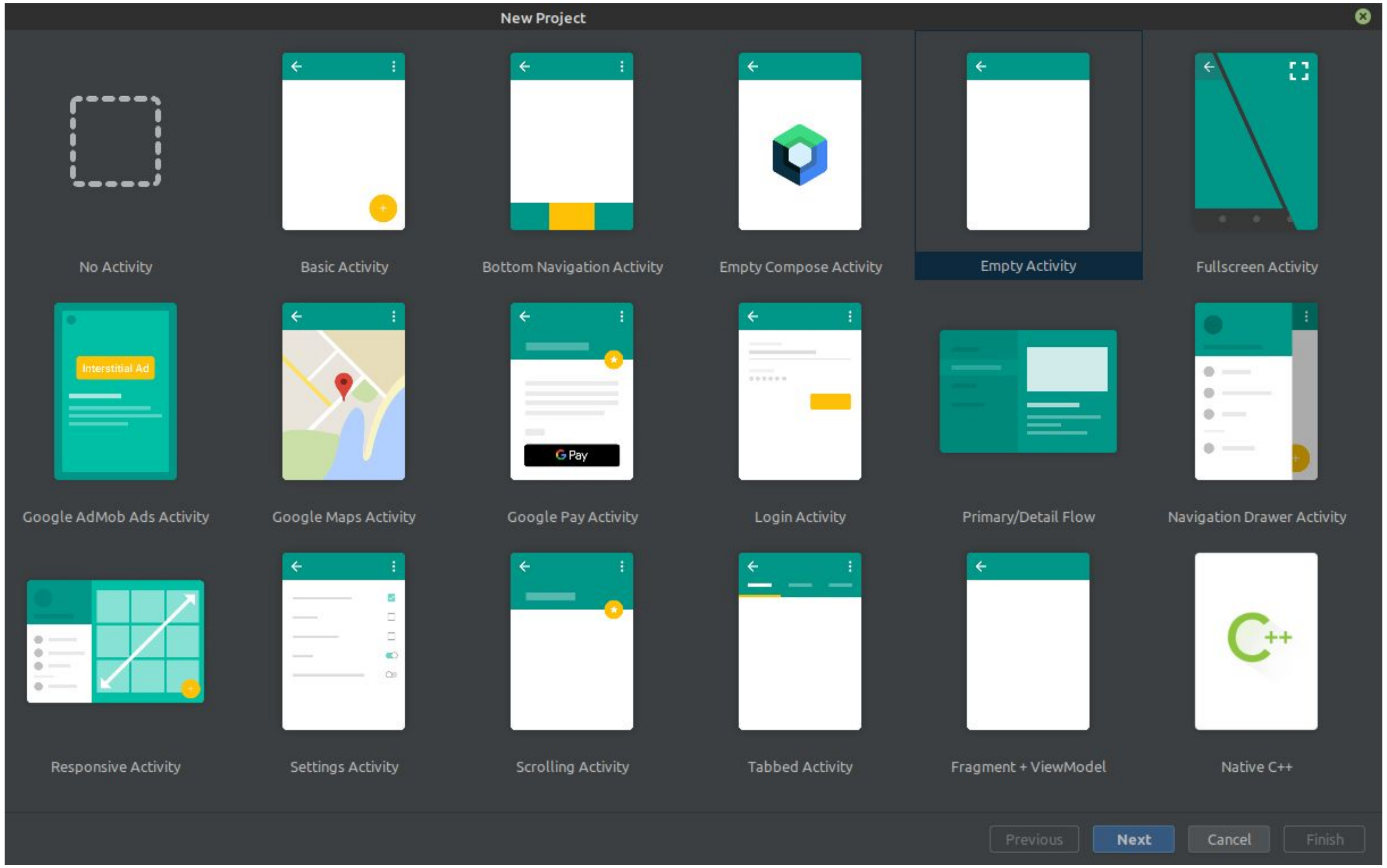
Enabled	Name	URL
☑	Android Automotive System Images	https://dl.google.com/android/repository/sys-img/andr...
☑	Android Repository	https://dl.google.com/android/repository/repository2-...
☑	Android System Images	https://dl.google.com/android/repository/sys-img/andr...
☑	Android TV System Images	https://dl.google.com/android/repository/sys-img/andr...
☑	Android Wear System Images	https://dl.google.com/android/repository/sys-img/andr...
☑	Android Wear for China System Images	https://dl.google.com/android/repository/sys-img/andr...
☑	Glass Development Kit, Google Inc.	https://dl.google.com/android/repository/glass/addon...
☑	Google API add-on System Images	https://dl.google.com/android/repository/sys-img/goo...
☑	Google API with Playstore System Images	https://dl.google.com/android/repository/sys-img/goo...
☑	Google Inc.	https://dl.google.com/android/repository/addon2-1.xml
☑	Intel HAXM	https://dl.google.com/android/repository/extras/intel/...

□ Force https://... sources to be fetched using http://... □ Disable SDK diff patching

Your first Android application

- Go to File > New Project

Newest version of Android Studio makes you choose first which kind of activity you want to start with.

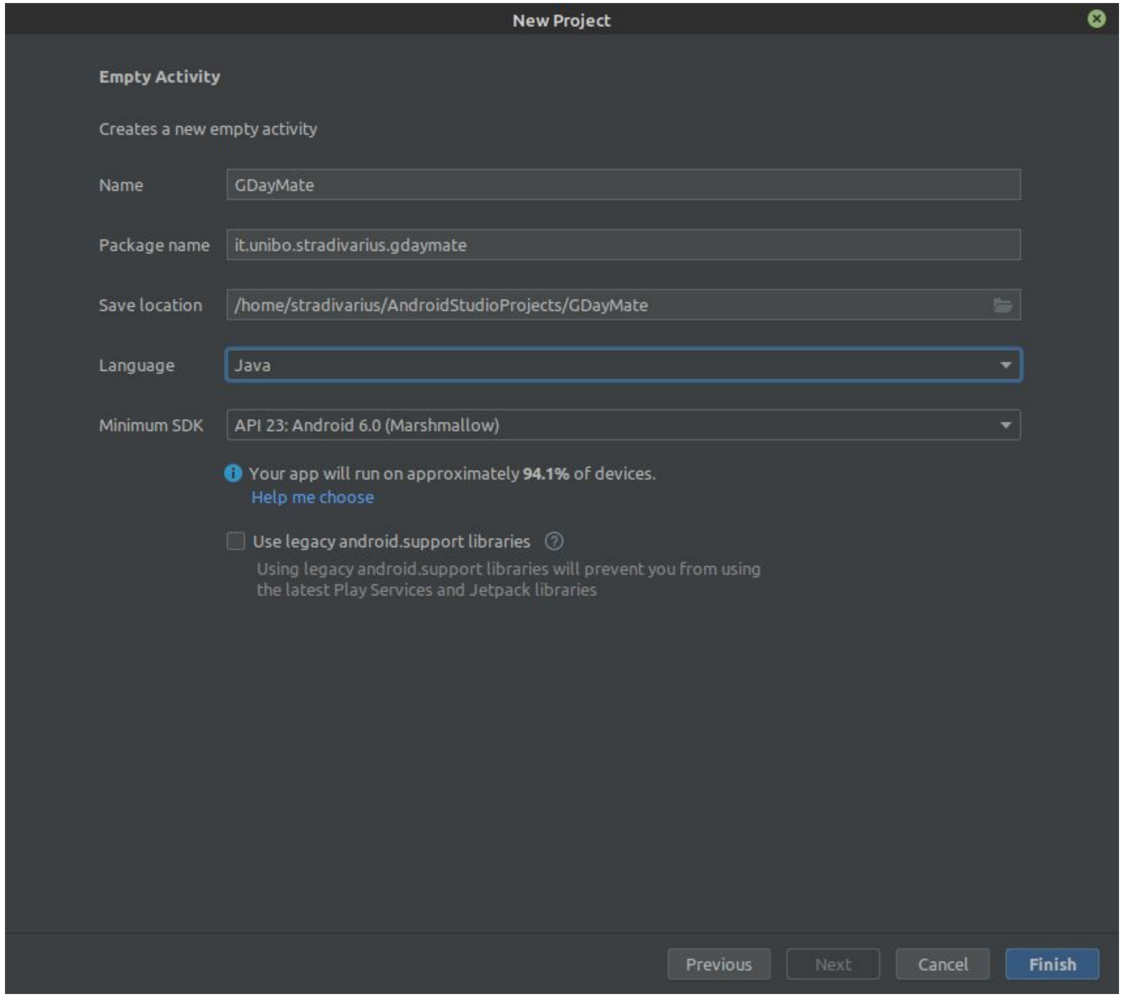


Your first Android application

- Go to File > New Project

Your activity will be named MainActivity by default (Java class).

Of course you can refactor it ...



New Project

Empty Activity

Creates a new empty activity

Name: GDayMate

Package name: it.unibo.stradivarius.gdaymate

Save location: /home/stradivarius/AndroidStudioProjects/GDayMate

Language: Java

Minimum SDK: API 23: Android 6.0 (Marshmallow)

Your app will run on approximately **94.1%** of devices.
Help me choose

□ Use legacy android.support libraries
Using legacy android.support libraries will prevent you from using the latest Play Services and Jetpack libraries

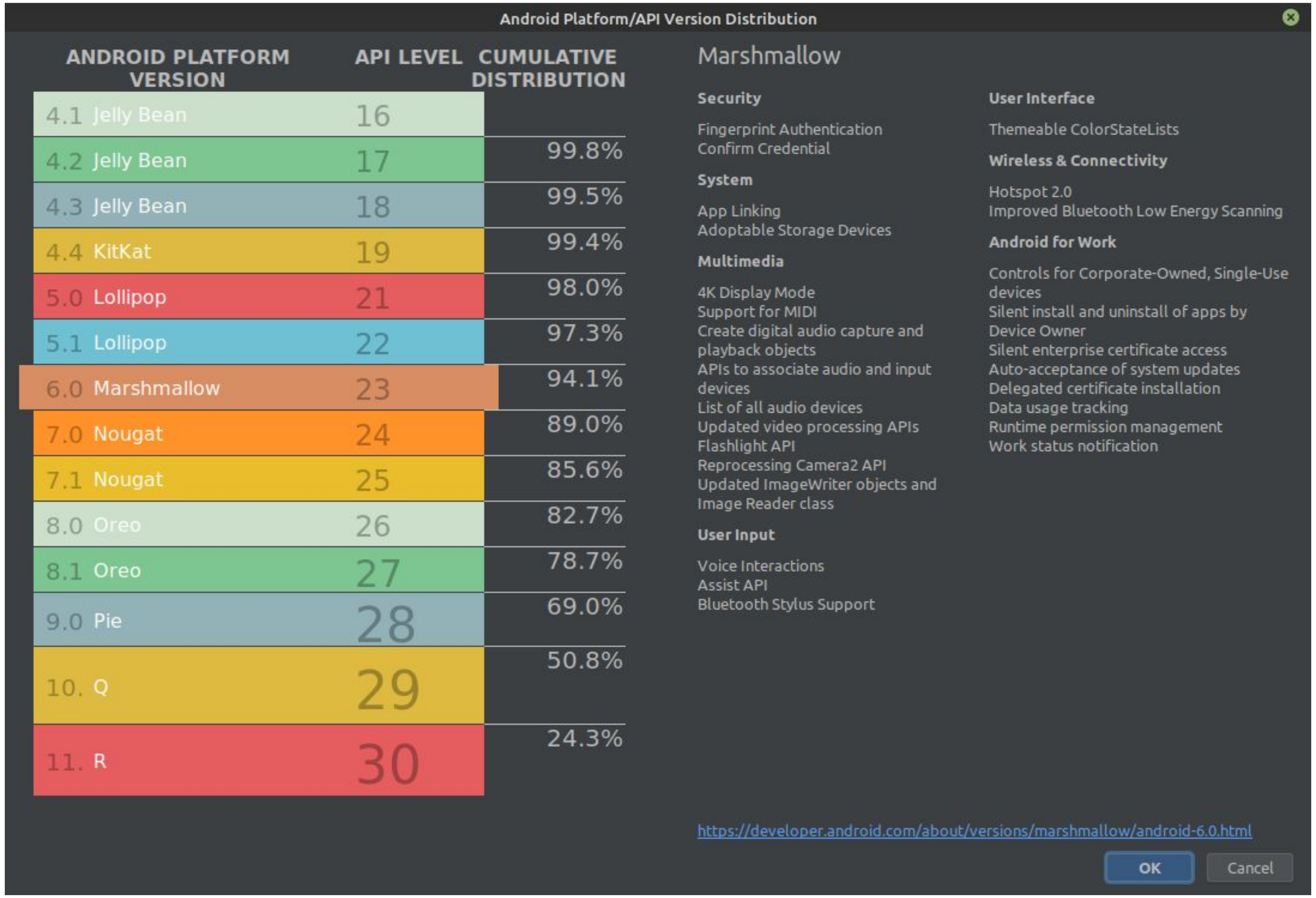
Your first Android application

- Go to File > New Project

Choose carefully which API version to use.

Low version = high compatibility

Low version = less features



Android Platform/API Version Distribution

ANDROID PLATFORM VERSION	API LEVEL	CUMULATIVE DISTRIBUTION
4.1 Jelly Bean	16	99.8%
4.2 Jelly Bean	17	99.5%
4.3 Jelly Bean	18	99.4%
4.4 KitKat	19	98.0%
5.0 Lollipop	21	97.3%
5.1 Lollipop	22	94.1%
6.0 Marshmallow	23	89.0%
7.0 Nougat	24	85.6%
7.1 Nougat	25	82.7%
8.0 Oreo	26	78.7%
8.1 Oreo	27	69.0%
9.0 Pie	28	50.8%
10. Q	29	24.3%
11. R	30	

Marshmallow

Security

Fingerprint Authentication
Confirm Credential

System

App Linking
Adoptable Storage Devices

Multimedia

4K Display Mode
Support for MIDI
Create digital audio capture and playback objects
APIs to associate audio and input devices
List of all audio devices
Updated video processing APIs
Flashlight API
Reprocessing Camera2 API
Updated ImageWriter objects and Image Reader class

User Input

Voice Interactions
Assist API
Bluetooth Stylus Support

User Interface

Themeable ColorStateLists

Wireless & Connectivity

Hotspot 2.0
Improved Bluetooth Low Energy Scanning

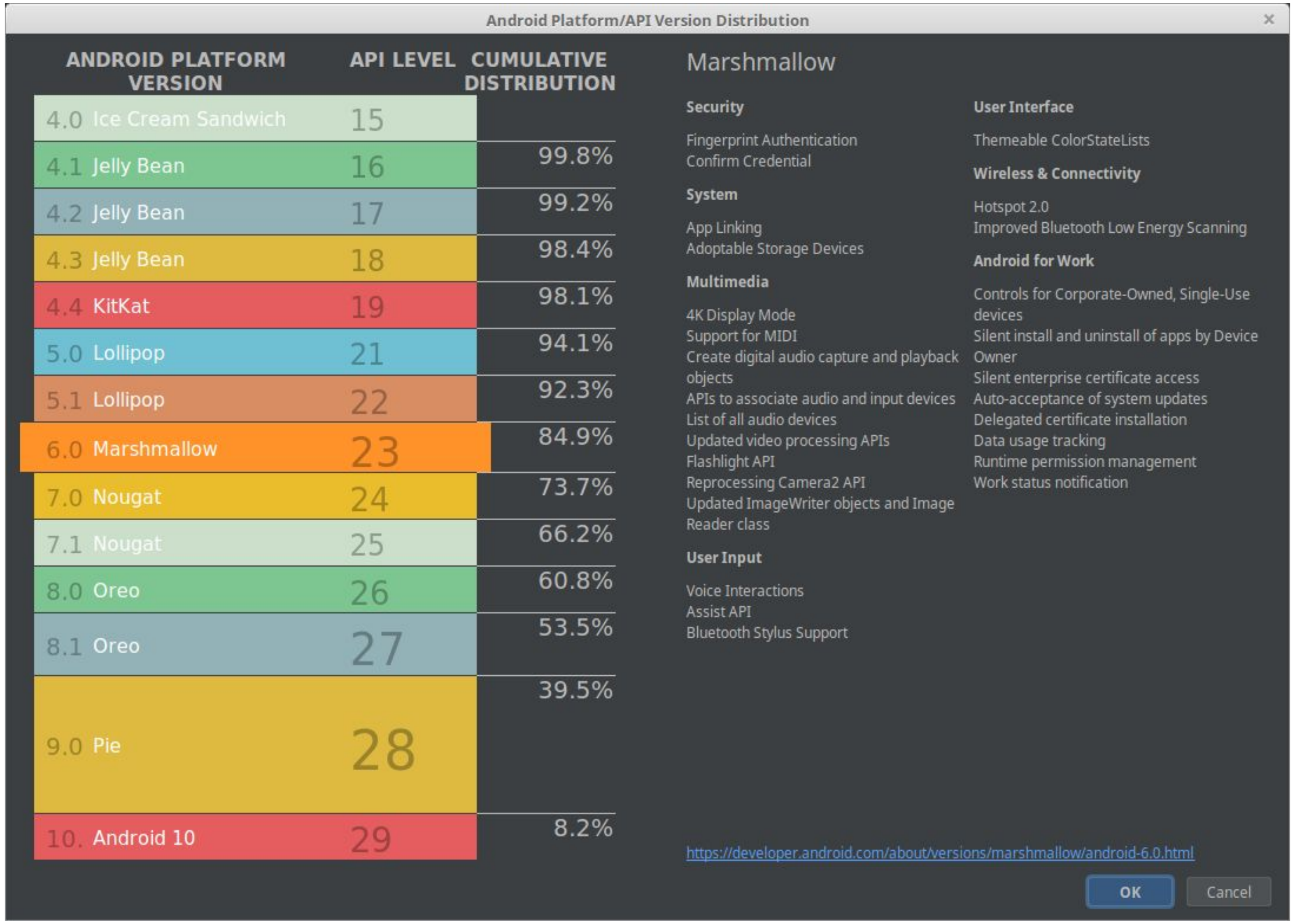
Android for Work

Controls for Corporate-Owned, Single-Use devices
Silent install and uninstall of apps by Device Owner
Silent enterprise certificate access
Auto-acceptance of system updates
Delegated certificate installation
Data usage tracking
Runtime permission management
Work status notification

https://developer.android.com/about/versions/marshmallow/android-6.0.html

Your first Android application

- This was from last year, just as a comparison!



Android Platform/API Version Distribution

ANDROID PLATFORM VERSION	API LEVEL	CUMULATIVE DISTRIBUTION
4.0 Ice Cream Sandwich	15	
4.1 Jelly Bean	16	99.8%
4.2 Jelly Bean	17	99.2%
4.3 Jelly Bean	18	98.4%
4.4 KitKat	19	98.1%
5.0 Lollipop	21	94.1%
5.1 Lollipop	22	92.3%
6.0 Marshmallow	23	84.9%
7.0 Nougat	24	73.7%
7.1 Nougat	25	66.2%
8.0 Oreo	26	60.8%
8.1 Oreo	27	53.5%
9.0 Pie	28	39.5%
10. Android 10	29	8.2%

Marshmallow

Security

Fingerprint Authentication
Confirm Credential

System

App Linking
Adoptable Storage Devices

Multimedia

4K Display Mode
Support for MIDI
Create digital audio capture and playback objects
APIs to associate audio and input devices
List of all audio devices
Updated video processing APIs
Flashlight API
Reprocessing Camera2 API
Updated ImageWriter objects and Image Reader class

User Input

Voice Interactions
Assist API
Bluetooth Stylus Support

User Interface

Themeable ColorStateLists

Wireless & Connectivity

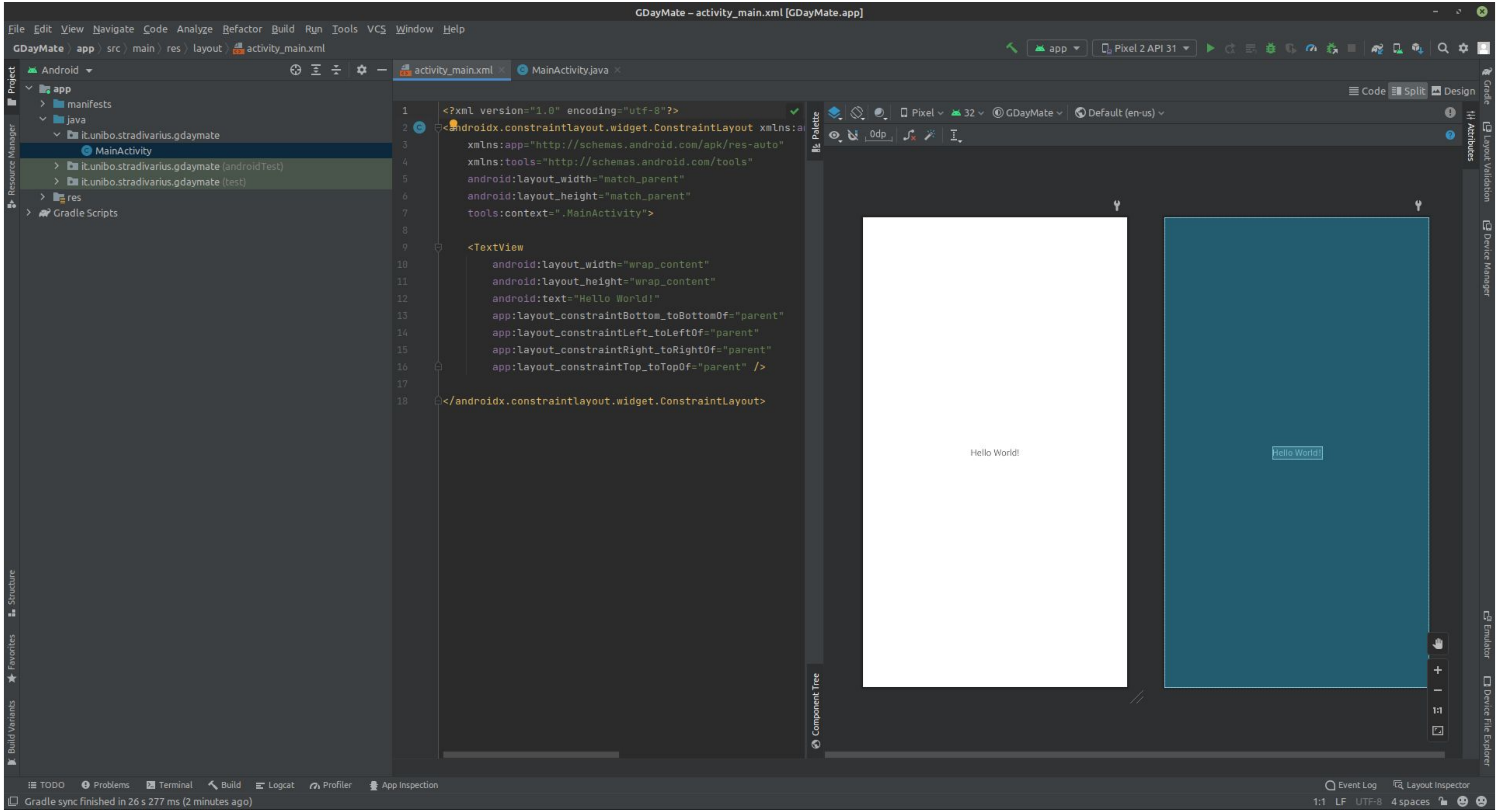
Hotspot 2.0
Improved Bluetooth Low Energy Scanning

Android for Work

Controls for Corporate-Owned, Single-Use devices
Silent install and uninstall of apps by Device Owner
Silent enterprise certificate access
Auto-acceptance of system updates
Delegated certificate installation
Data usage tracking
Runtime permission management
Work status notification

https://developer.android.com/about/versions/marshmallow/android-6.0.html

Your first Android application



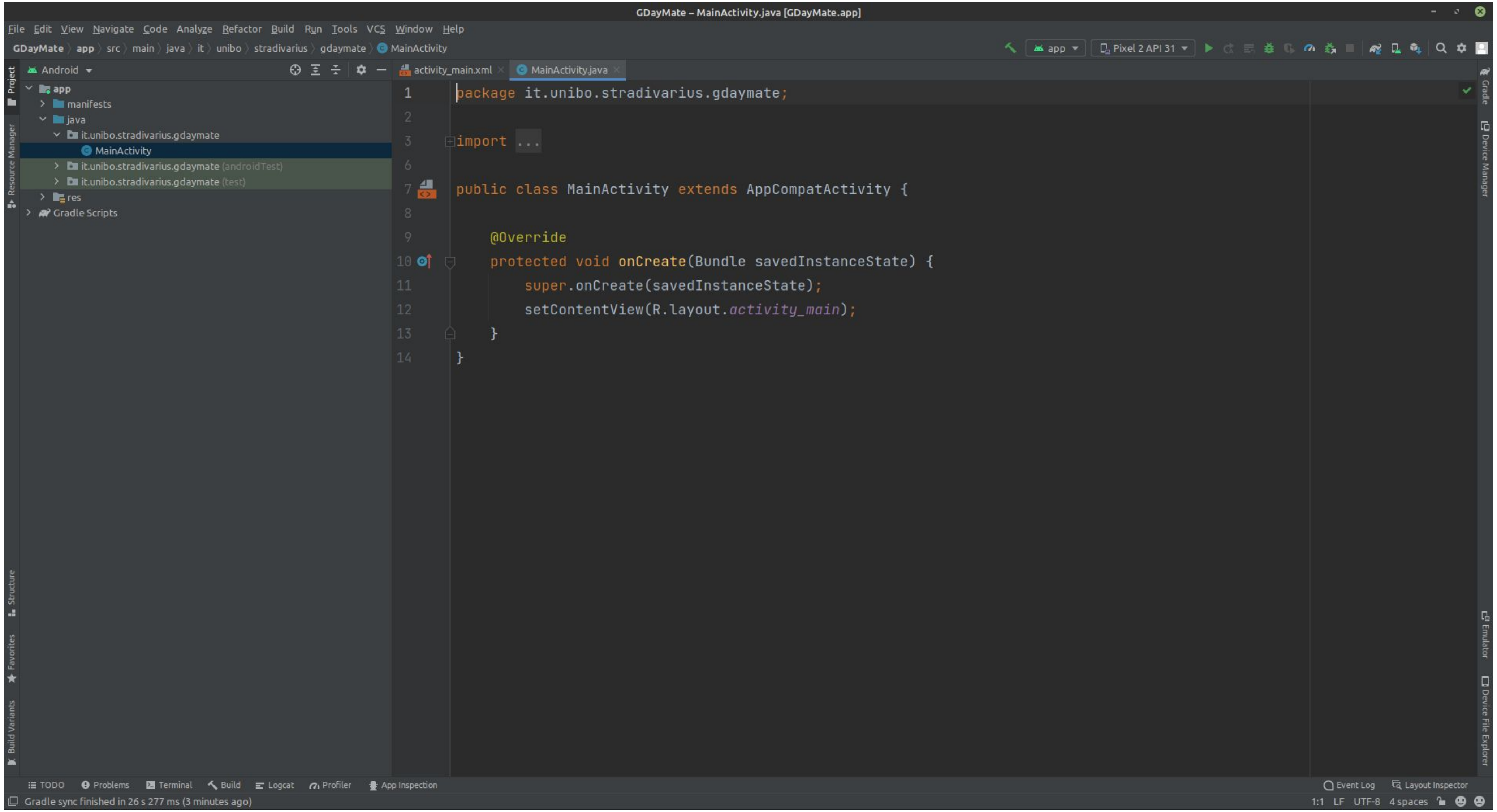
```xml
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:a
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World!"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

Gradle sync finished in 26 s 277 ms (2 minutes ago)

Your first Android application


GDayMate – MainActivity.java [GDayMate.app]
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
GDayMate 〉 app 〉 src 〉 main 〉 java 〉 it 〉 unibo 〉 stradivarius 〉 gdaymate 〉 MainActivity
app
Pixel 2 API 31
Android
app
manifests
java
it.unibo.stradivarius.gdaymate
MainActivity
it.unibo.stradivarius.gdaymate (androidTest)
it.unibo.stradivarius.gdaymate (test)
res
Gradle Scripts
activity_main.xml
MainActivity.java
Project
Resource Manager
Structure
Favorites
Build Variants
Gradle
Device Manager
Emulator
Device File Explorer
TODO
Problems
Terminal
Build
Logcat
Profiler
App Inspection
Event Log
Layout Inspector
1:1 LF UTF-8 4 spaces

```java
package it.unibo.stradivarius.gdaymate;

import ...

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

Gradle sync finished in 26 s 277 ms (3 minutes ago)

Creating an emulator

Hit Tools >

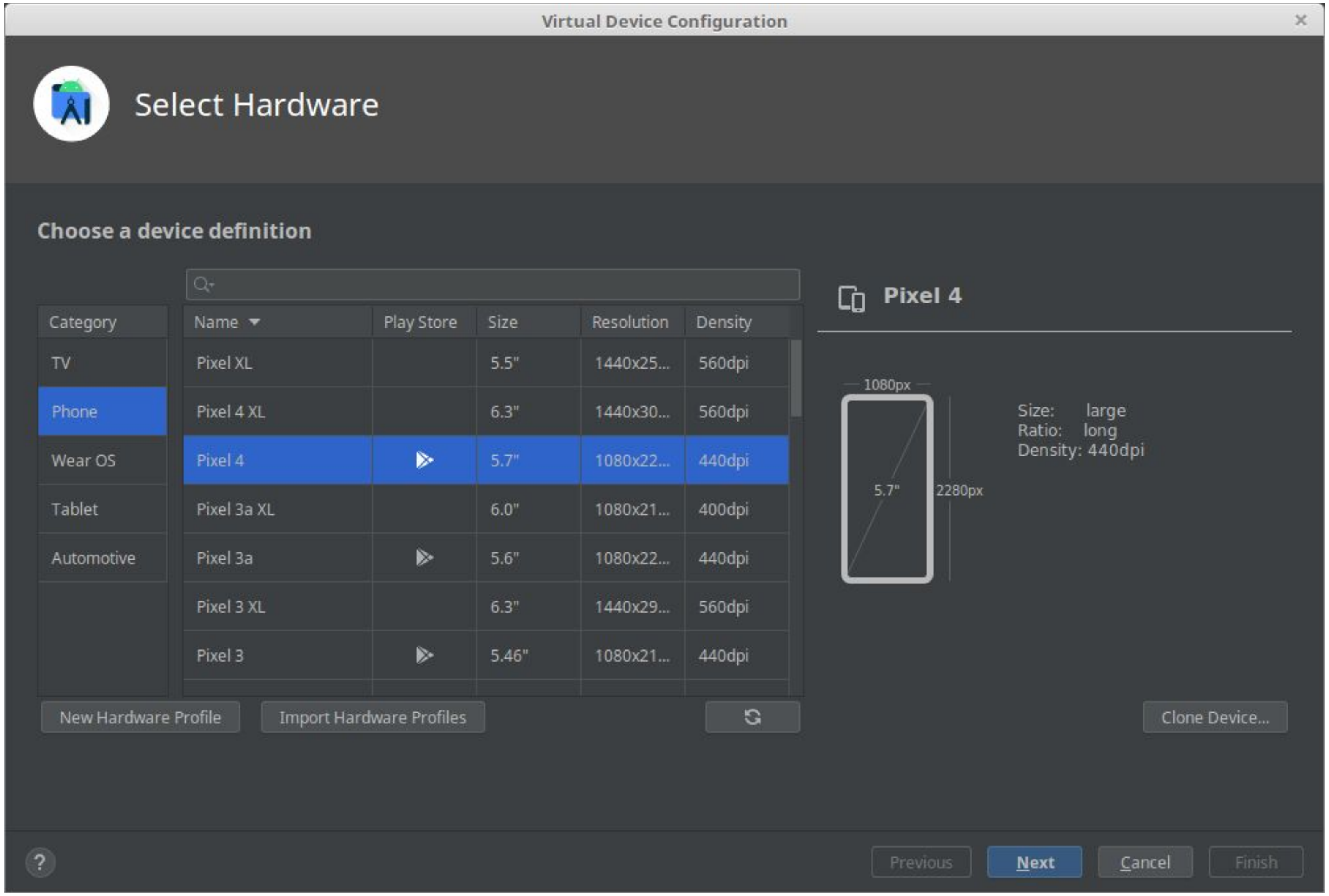
Device Manager

Previously AVD

AVD = Android Virtual Device

You can select options for the emulator

You can create as many as you want



Virtual Device Configuration

Select Hardware

Choose a device definition

Category
TV
Phone
Wear OS
Tablet
Automotive

Name	Play Store	Size	Resolution	Density
Pixel XL		5.5"	1440x25...	560dpi
Pixel 4 XL		6.3"	1440x30...	560dpi
Pixel 4	▶	5.7"	1080x22...	440dpi
Pixel 3a XL		6.0"	1080x21...	400dpi
Pixel 3a	▶	5.6"	1080x22...	440dpi
Pixel 3 XL		6.3"	1440x29...	560dpi
Pixel 3	▶	5.46"	1080x21...	440dpi

Pixel 4

Size: large
Ratio: long
Density: 440dpi

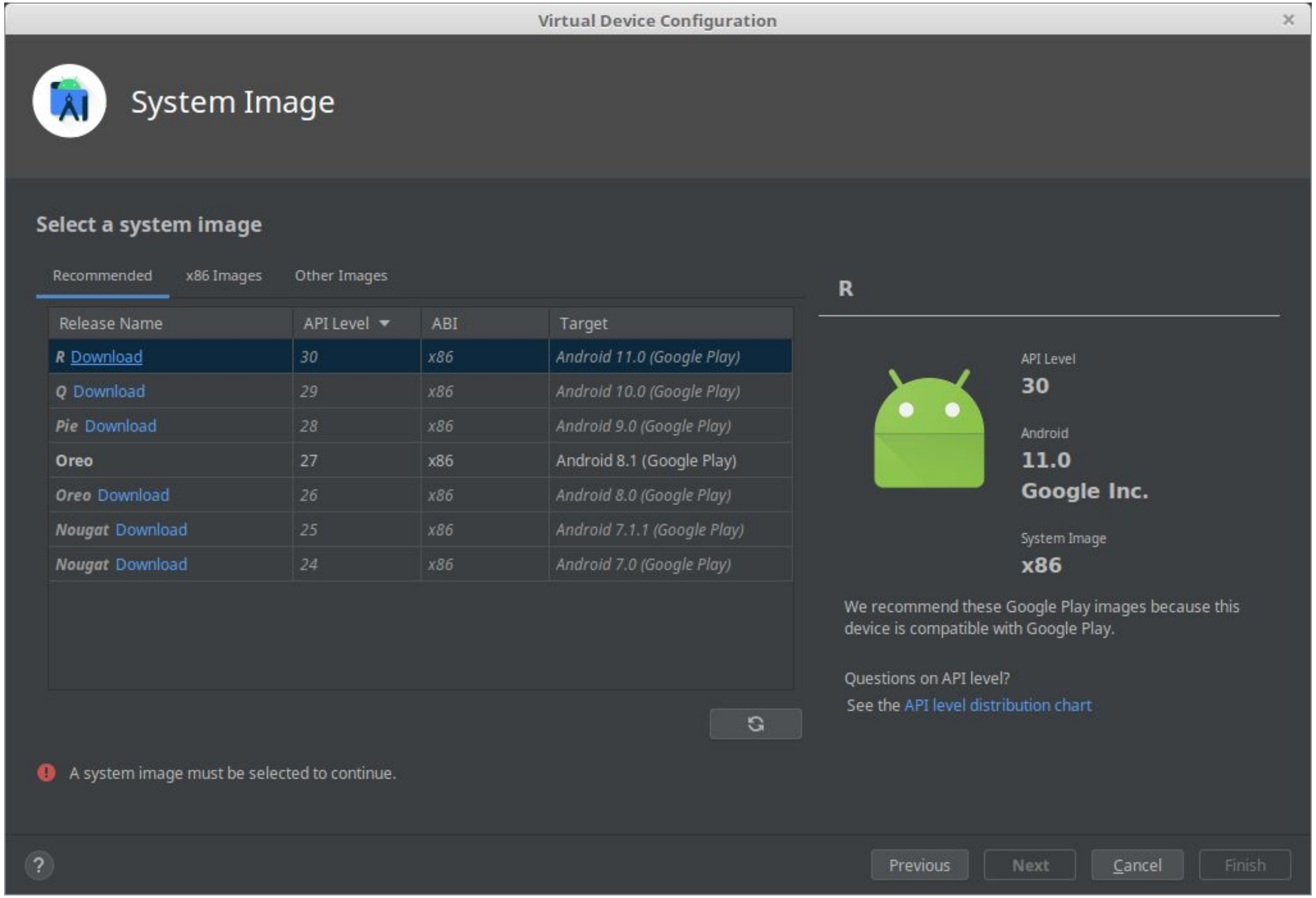
Creating an emulator

Hit Tools >

AVD Manager

AVD = Android Virtual Device

You obviously **need** to download the Android system image for the version you want.



Select a system image

Release Name	API Level ▾	ABI	Target
R Download	*30*	*x86*	*Android 11.0 (Google Play)*
Q Download	*29*	*x86*	*Android 10.0 (Google Play)*
Pie Download	*28*	*x86*	*Android 9.0 (Google Play)*
Oreo	27	x86	Android 8.1 (Google Play)
Oreo Download	*26*	*x86*	*Android 8.0 (Google Play)*
Nougat Download	*25*	*x86*	*Android 7.1.1 (Google Play)*
Nougat Download	*24*	*x86*	*Android 7.0 (Google Play)*

R

API Level
30

Android
11.0
Google Inc.

System Image
x86

We recommend these Google Play images because this device is compatible with Google Play.

Questions on API level?
See the API level distribution chart

A system image must be selected to continue.

Creating an emulator

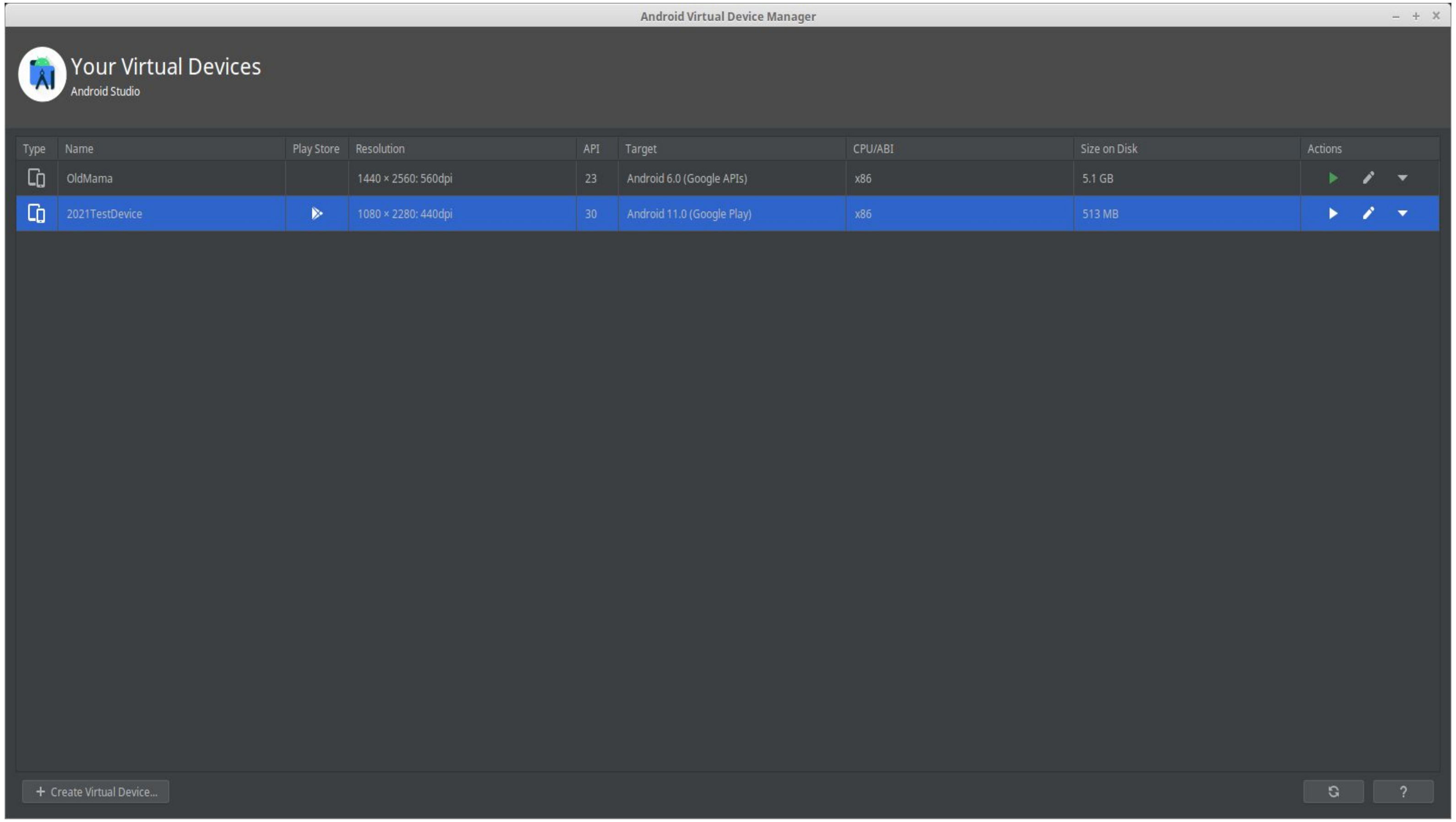
Hit Tools >

AVD Manager

AVD = Android Virtual Device

Wanna test it?

Hit the Play button



Android Virtual Device Manager

Your Virtual Devices

Android Studio

Type	Name	Play Store	Resolution	API	Target	CPU/ABI	Size on Disk	Actions
	OldMama		1440 × 2560: 560dpi	23	Android 6.0 (Google APIs)	x86	5.1 GB	▶ ✎ ▾
	2021TestDevice	▶	1080 × 2280: 440dpi	30	Android 11.0 (Google Play)	x86	513 MB	▶ ✎ ▾

Creating an emulator

Hit Tools >

AVD Manager

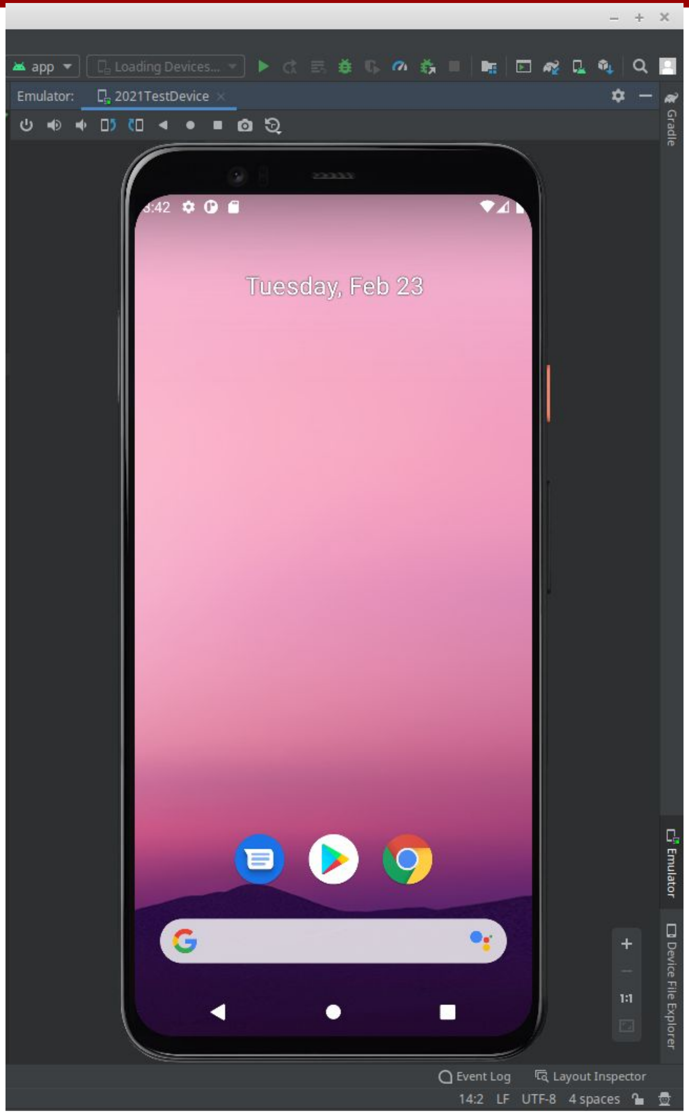
AVD = Android Virtual Device

If you run it, it will turn on and resemble pretty much a real device.

It is legit a virtual machine.

YES you can use the internet (by default each AVD is individually NATted).

NO you can't call (lol)

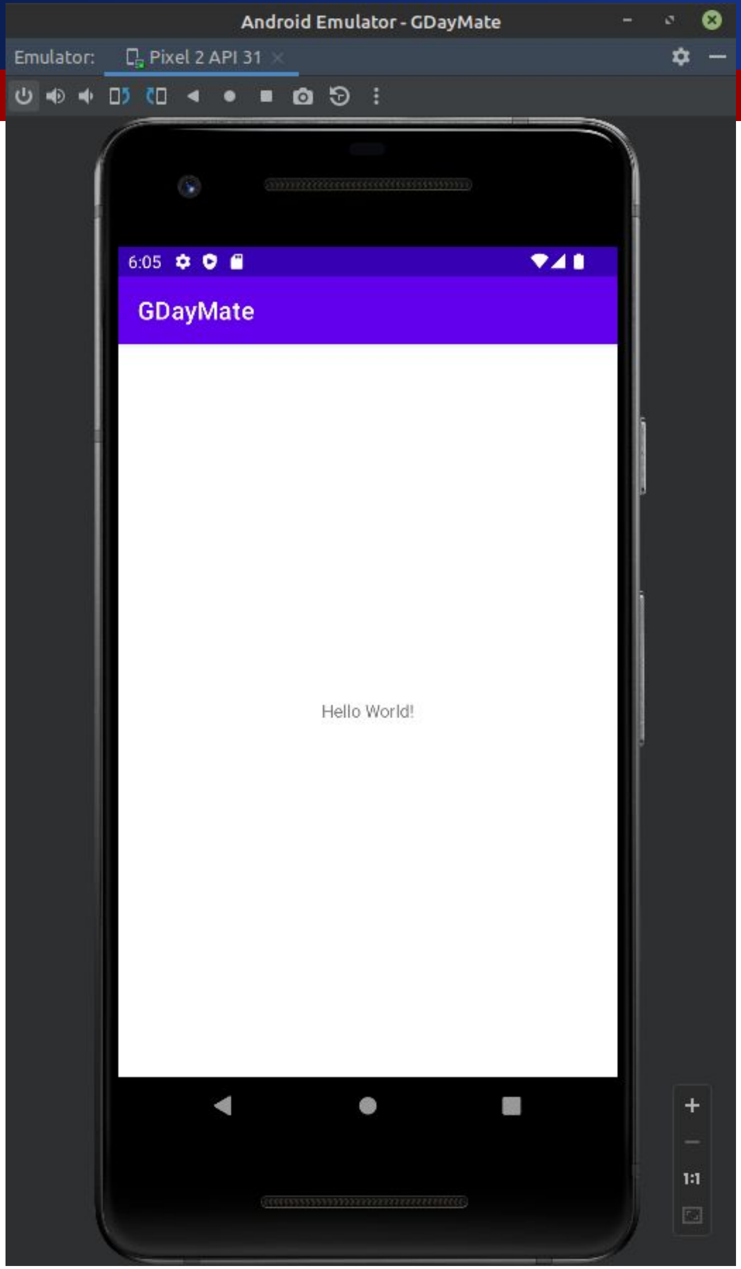


All good? Test it!

Hit Run > Run ‘app’

- Test in on the emulator

- You should see something similar to this



Delving into versions and building

Hit “File > Project Structure > Modules” and you’ll see two sections that’ll help you deal with versioning.

- Properties
  - versioning and other stuff at **Compile time**
  - Mostly having to do with Gradle
- Default Config
  - versioning and other stuff at **Run Time**

Android: versions

2008
API 1

2009
API 3

2009
API 4

2009
API 5

2010
API 8

2010
API 9

2011
API 11

2011
API 14

2012
API 16

2013
API 19

2014
API 21

2015
API 23

2015
API 24

2017
API 26

2018
API 28

2019 - API 29

Initially “Android Q”

No more desserts...

2020 - API 30

Android
“R”

2021 - API 31
2022 - API 32

Tiramisu in in preview!

Delving into versions and building

What is Gradle?

It’s the official build automation tool for Android, coming with a lot of optimizations.

It has config files that can be modified through the Android Studio GUI.

It manages build configurations that no longer are assigned to the Android Developer.

- New Gradle two weeks ago… https://gradle.org/whats-new/gradle-7/

Delving into versions and building

compileSdkVersion

- Used by Gradle to compile the project
- i.e. which set of classes and functions should I use?
- It’s the newest possible SDK theoretically supported by your app (watch out, NOT the API).
- Suggested to use the latest available (unless you haven’t learned it yet…)
- It’s COMPILED, therefore retro-compatibility is structurally ensured.

Delving into versions and building

minSdkVersion

- Indicates which is the oldest release of the SDK (but also API) your app is compatible with...
- … though it is compiled with another version.
- Obviously you cannot implement certain functionalities (e.g. channels).
- In practice if a customer has a phone that’s too old, then the app is neither installable nor visible.

Delving into versions and building

targetSdkVersion

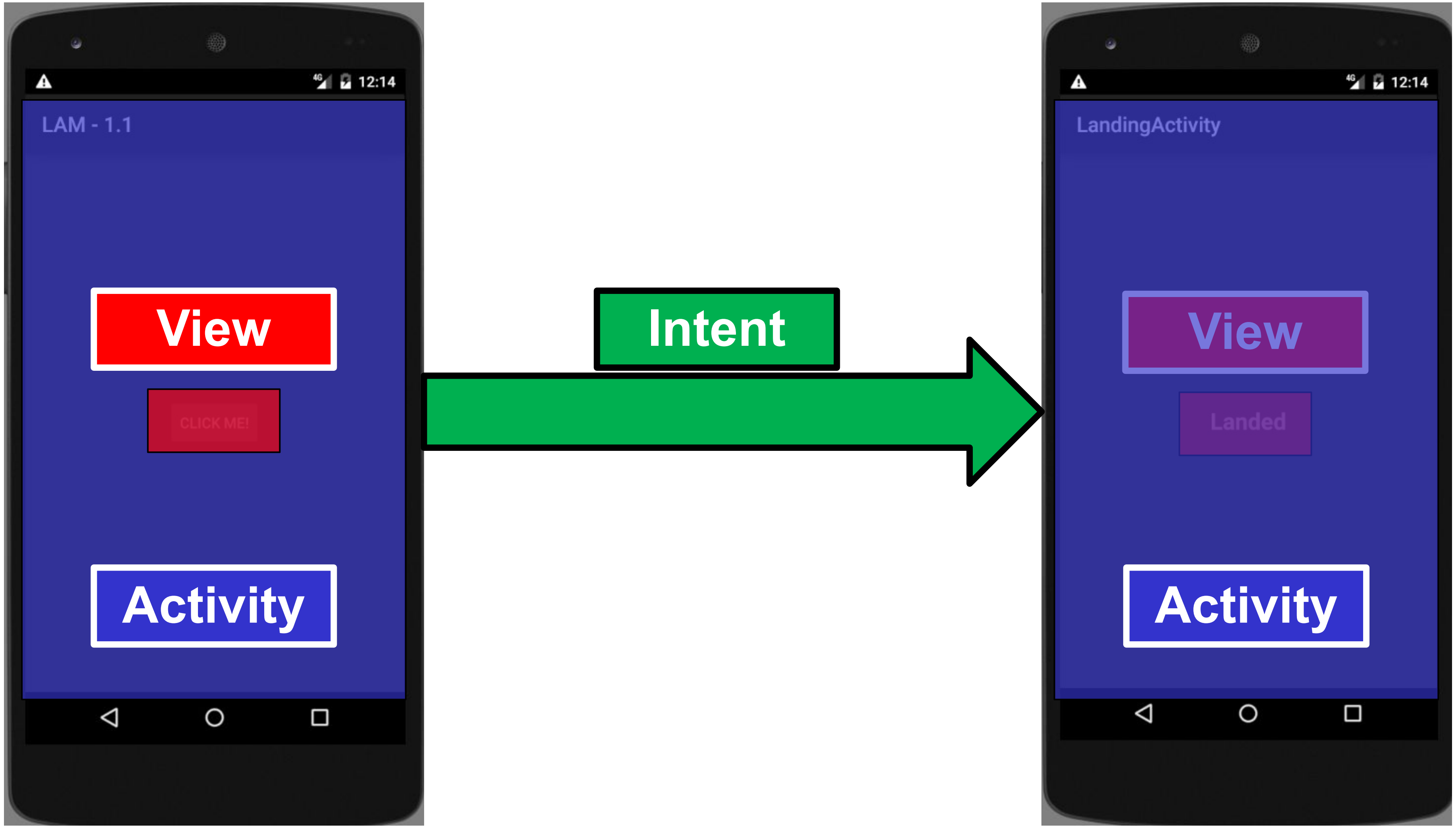
- Indicates which is the newest release of the SDK (but also API) your app is compatible with...
- … in practice it tells what is the expected version.
- It is ideally the same as the compileSdkVersion, however it can be older if newer versions had not been tested.
- In short:
  - minSdkVersion <= targetSdkVersion <= compileSdkVersion
- Even though it’s better:
  - minSdkVersion <= targetSdkVersion == compileSdkVersion

Hello world, Android!

- Anatomy of an application
  - Activity: what is started
  - View: what is seen
  - Intent: how to communicate with others

- Mix of XML and Java
  - Generally XML for layouts
  - Java for the app logic

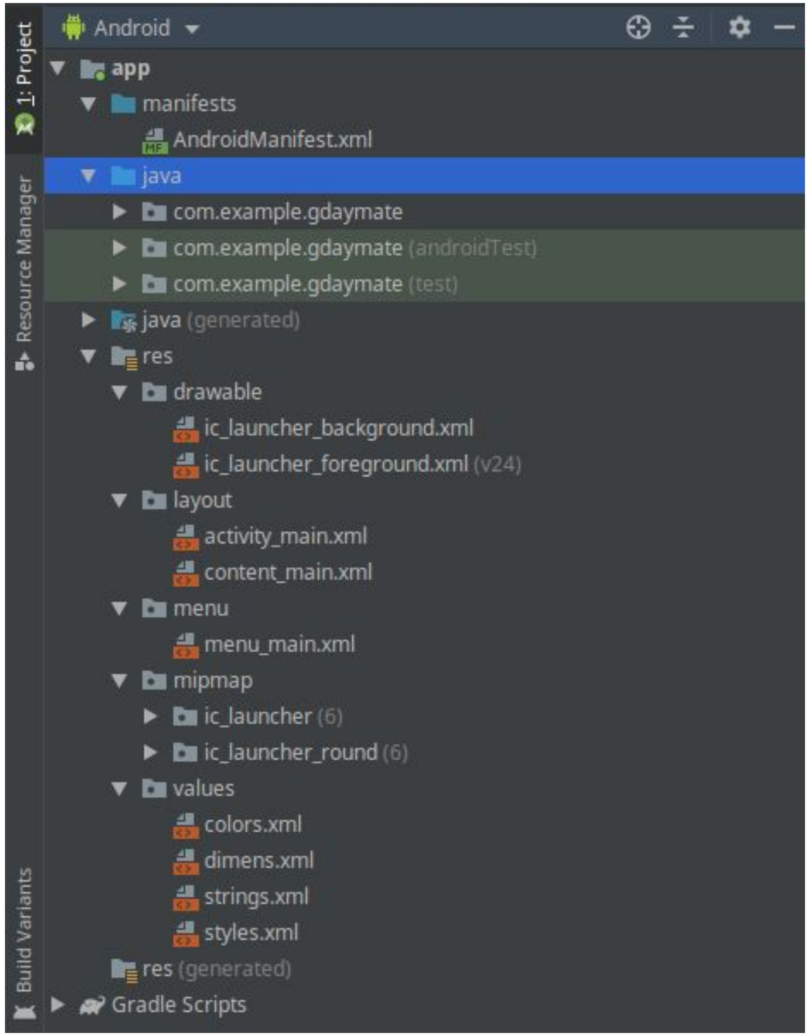
Activities, Intents and Views



The java and res folders

The code of your application:

- The java folder contains… the java code!
- Inside res there are a lot of resources
  - Images
  - Layouts
  - Xml files
  - Strings
- AndroidManifest.xml

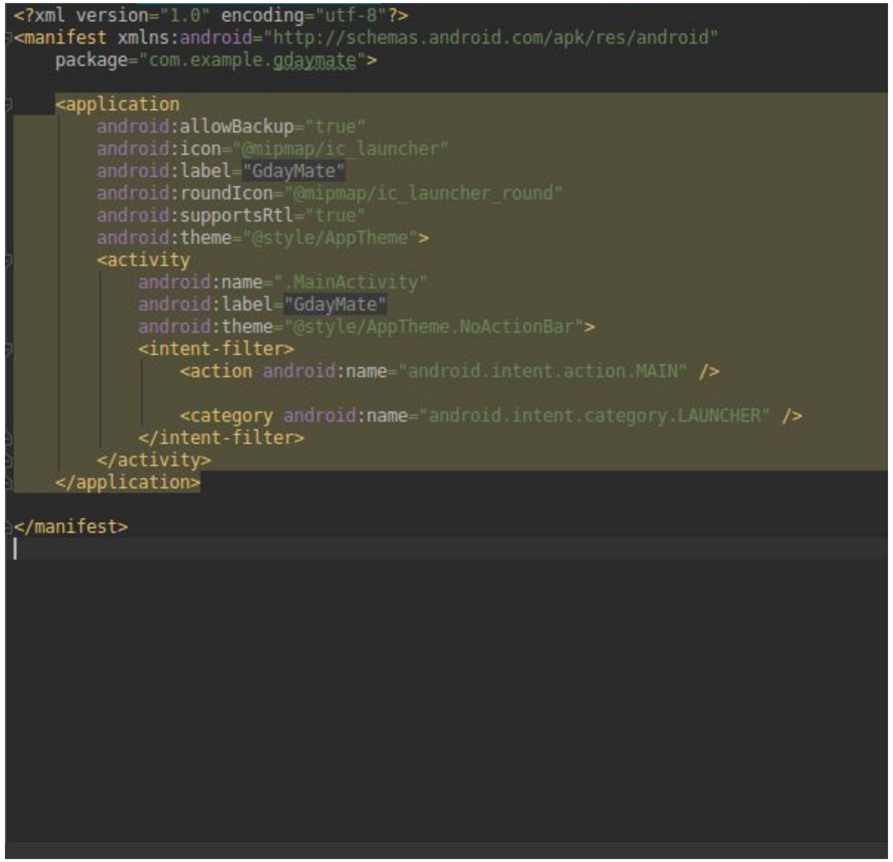


```
Android
▼ app
  ▼ manifests
      AndroidManifest.xml
  ▼ java
    ▶ com.example.gdaymate
    ▶ com.example.gdaymate (androidTest)
    ▶ com.example.gdaymate (test)
  ▶ java (generated)
  ▼ res
    ▼ drawable
        ic_launcher_background.xml
        ic_launcher_foreground.xml (v24)
    ▼ layout
        activity_main.xml
        content_main.xml
    ▼ menu
        menu_main.xml
    ▼ mipmap
      ▶ ic_launcher (6)
      ▶ ic_launcher_round (6)
    ▼ values
        colors.xml
        dimens.xml
        strings.xml
        styles.xml
    res (generated)
▶ Gradle Scripts
```

AndroidManifest.xml

Mandatory file for every application

- Contains:
  - Application declaration
  - Permissions
  - Intent filters
  - Targets



```xml
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.gdaymate">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="GdayMate"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity
            android:name=".MainActivity"
            android:label="GdayMate"
            android:theme="@style/AppTheme.NoActionBar">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

How to test

- Via an AVD
  - ✓ Fast, possible to have different resolutions, APIs
  - ✗ Not a real device
- On a real device
  - ✓ You get exactly what is deployed
  - ✗ Must own a real device
- So?
  - Test in on AVDs.
  - When you feel that the application is ready, go with a real device

Test on a Real Device

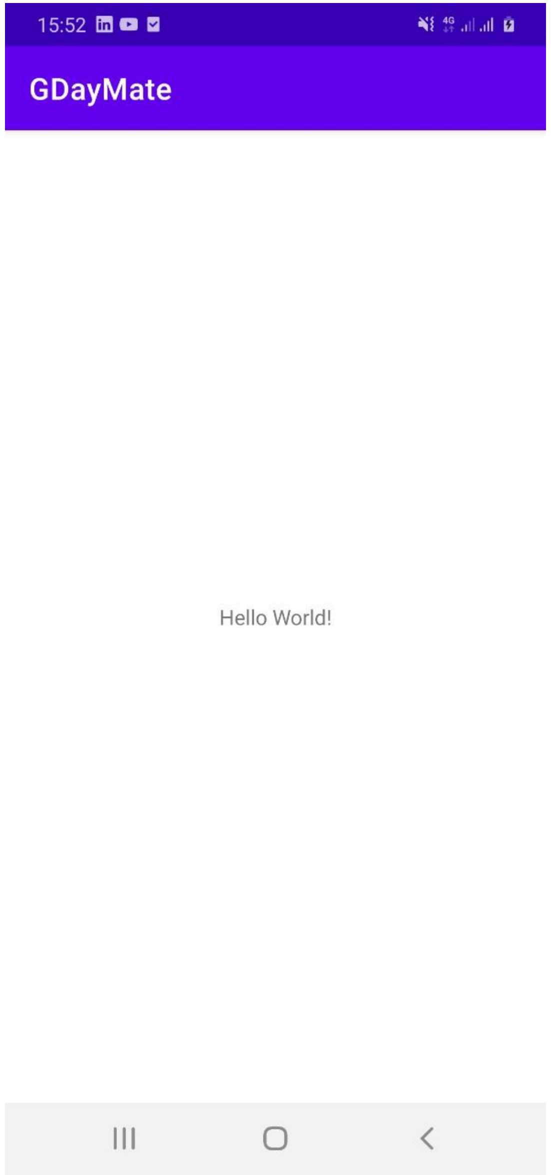
Mainly 2 ways of doing it:

- Via USB debugging (more for **debugging**)
  - Phone must have developer options and USB debugging enabled (howto: https://developer.android.com/studio/debug/dev-options.html#enable)
  - PC’s OS must have the correct driver/module (howto: https://developer.android.com/studio/run/device#setting-up)
  - Can then run apps just by hitting the Run > Run ‘app’
  - You can use the newest WiFi pairing!
- Create an *apk* (more for **releasing** and **sharing**)
  - Must be **signed**

Test on a Real Device

Here's the app running on my Samsung A40 with Android 10.

--------------------------------->



How to deploy through APK

- Android applications must be signed before installing them on a real device.

Hit: Build > Generate Signed Bundle / APK

You can generate a Bundle (ABB) alternatively:

a Bundle is Google Play’s new app serving model, called Dynamic Delivery, then uses your app bundle to generate and serve optimized APKs for each user’s device configuration, so they download only the code and resources they need to run your app. You no longer have to build, sign, and manage multiple APKs to support different devices, and users get smaller, more optimized downloads. It is a **publishing** format.

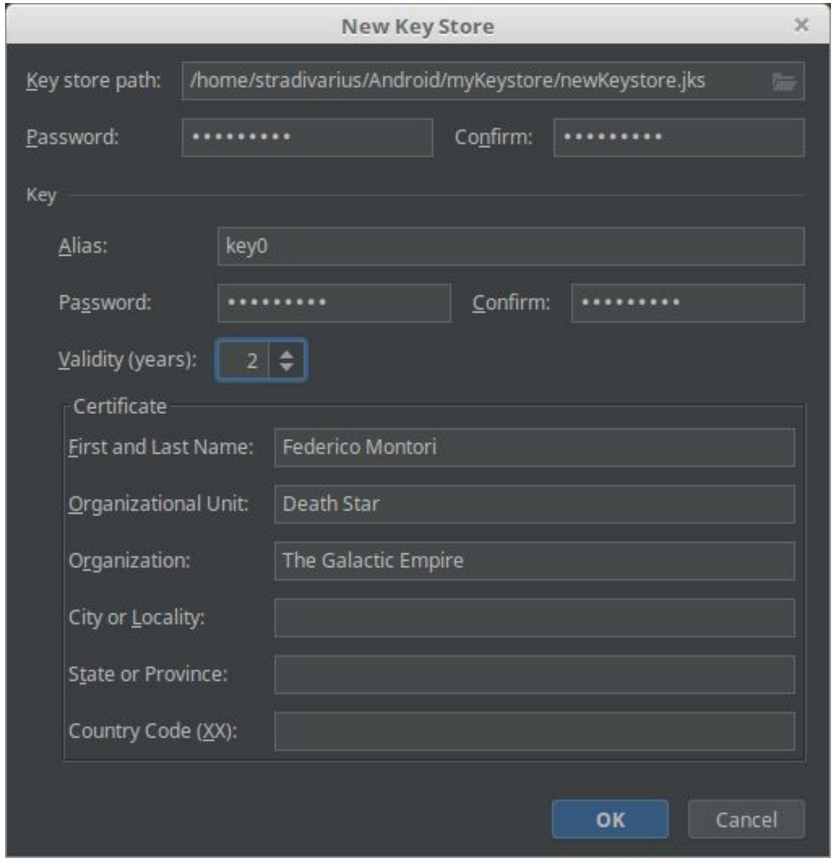
https://developer.android.com/guide/app-bundle

You need a key for this and you can generate one from the menu.

You can potentially use no key, but it will generate a debug version.

How to deploy through APK

- Using here V2 Signature (faster, since Android 7.0)
  https://developer.android.com/about/versions/nougat/android-7.0.html#apk_signature_v2



New Key Store

Key store path: /home/stradivarius/Android/myKeystore/newKeystore.jks

Password: ••••••••• Confirm: •••••••••

Key

Alias: key0

Password: ••••••••• Confirm: •••••••••

Validity (years): 2

Certificate

First and Last Name: Federico Montori

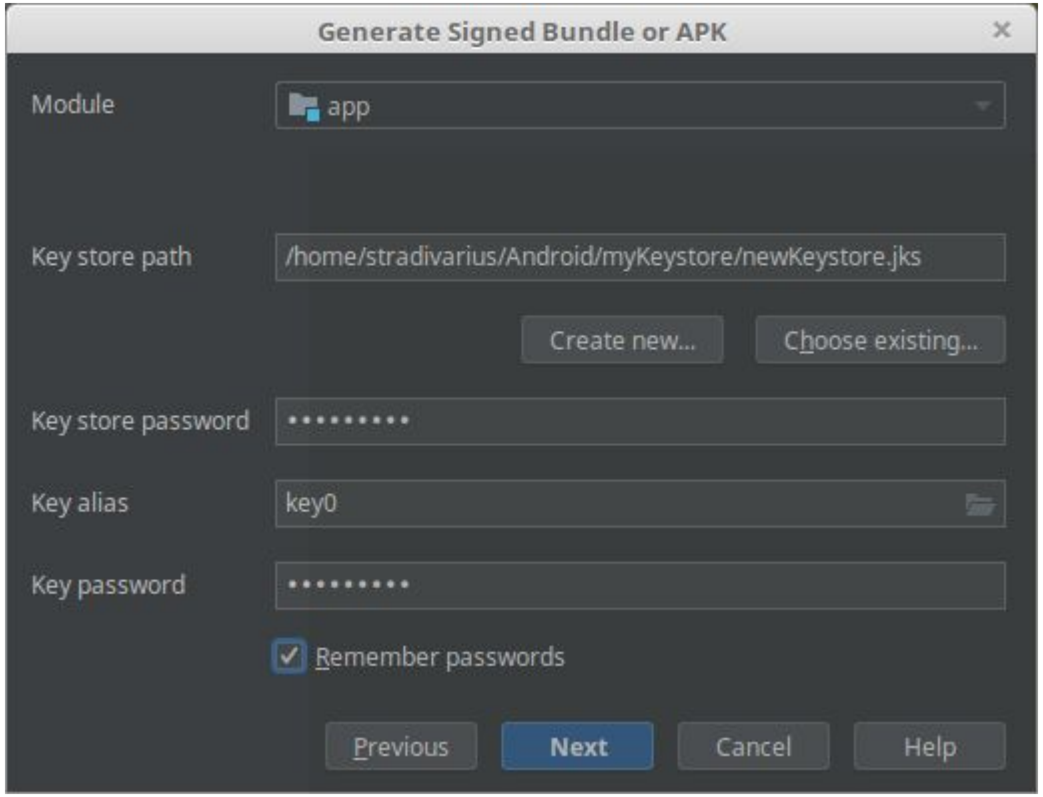
Organizational Unit: Death Star

Organization: The Galactic Empire

City or Locality:

State or Province:

Country Code (XX):



Generate Signed Bundle or APK

Module: app

Key store path: /home/stradivarius/Android/myKeystore/newKeystore.jks

Key store password: •••••••••

Key alias: key0

Key password: •••••••••

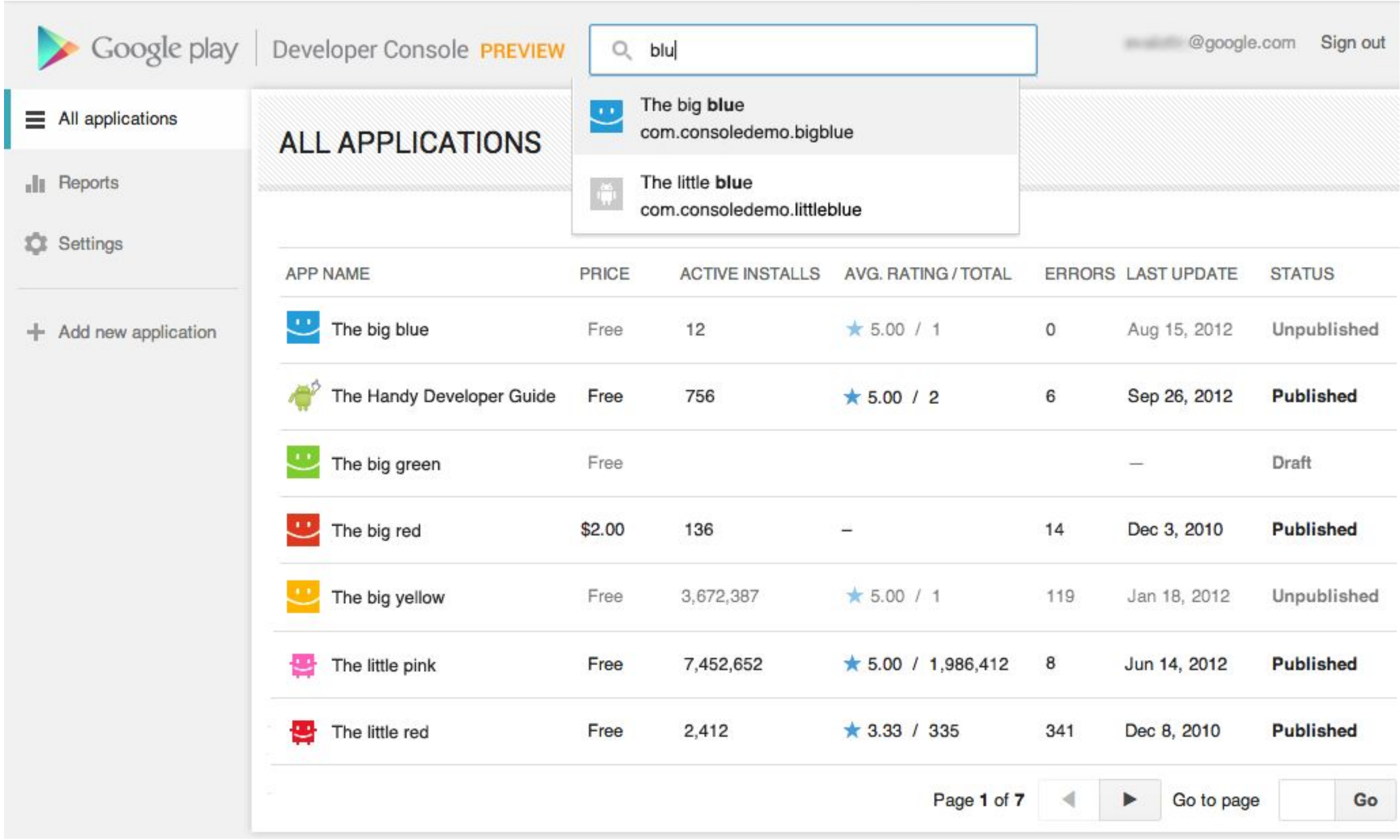
☑ Remember passwords

… transfer the .apk file to your phone and you’re done.

How to deploy

- To be published on the market, you have to pay 25 Euro
  - Lifetime fee, unlimited APPs
  - Not required for the LAM class
- Upload the APK or the ABB, and in few hours/days the APP is on the play store
  - Receive comments, improve, update
  - Smartphone specific bugs? AVDs

Android Developer Console



blu

The big **blue**
com.consoledemo.bigblue

The little **blue**
com.consoledemo.littleblue

[redacted]@google.com Sign out

ALL APPLICATIONS

APP NAME	PRICE	ACTIVE INSTALLS	AVG. RATING / TOTAL	ERRORS	LAST UPDATE	STATUS
The big blue	Free	12	★ 5.00 / 1	0	Aug 15, 2012	Unpublished
The Handy Developer Guide	Free	756	★ 5.00 / 2	6	Sep 26, 2012	Published
The big green	Free				–	Draft
The big red	$2.00	136	–	14	Dec 3, 2010	Published
The big yellow	Free	3,672,387	★ 5.00 / 1	119	Jan 18, 2012	Unpublished
The little pink	Free	7,452,652	★ 5.00 / 1,986,412	8	Jun 14, 2012	Published
The little red	Free	2,412	★ 3.33 / 335	341	Dec 8, 2010	Published

Page **1** of **7** ◀ ▶ Go to page Go