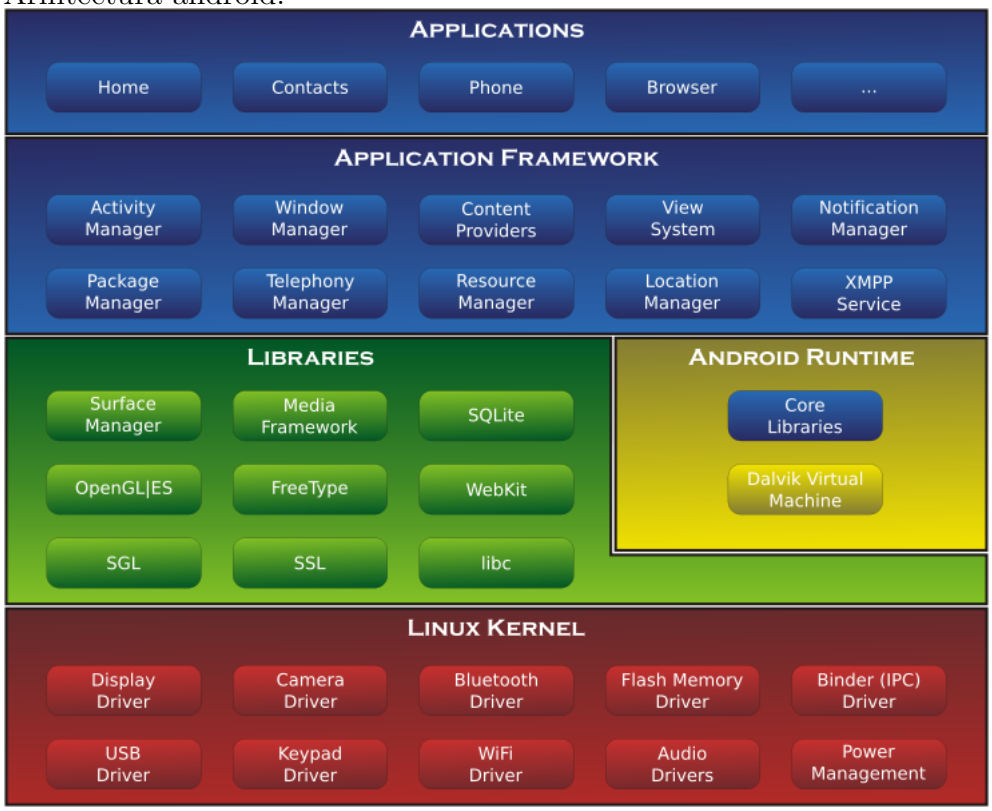
Tehnici de programare pe platforma Android
Laborator 1

Introducere

Android este o platforma software si un sistem de operare pentru dispozitive si telefoane mobile bazate pe nucleul **Linux**.
Aplicatiile sunt dezvoltate in limbajul Java si permite interactiunea cu utilizatorul prin intermediul bibliotecilor dezvoltate de **Google**.
`http://developer.android.com/index.html`
Arhitectura android:



Mediul de lucru

Pentru dezvoltarea aplicatiilor pe platforma Android trebuie configurat urmatorul mediu de lucru:

- Instalarea Java Development Kit:
  `http://www.oracle.com/technetwork/java/javase/downloads/index.html`
- Instalarea Android Studio ce contine:
  - IDE Android Studio (Arhitectura din IntelliJ)
  - Android SDK tools
  - Android platform tools

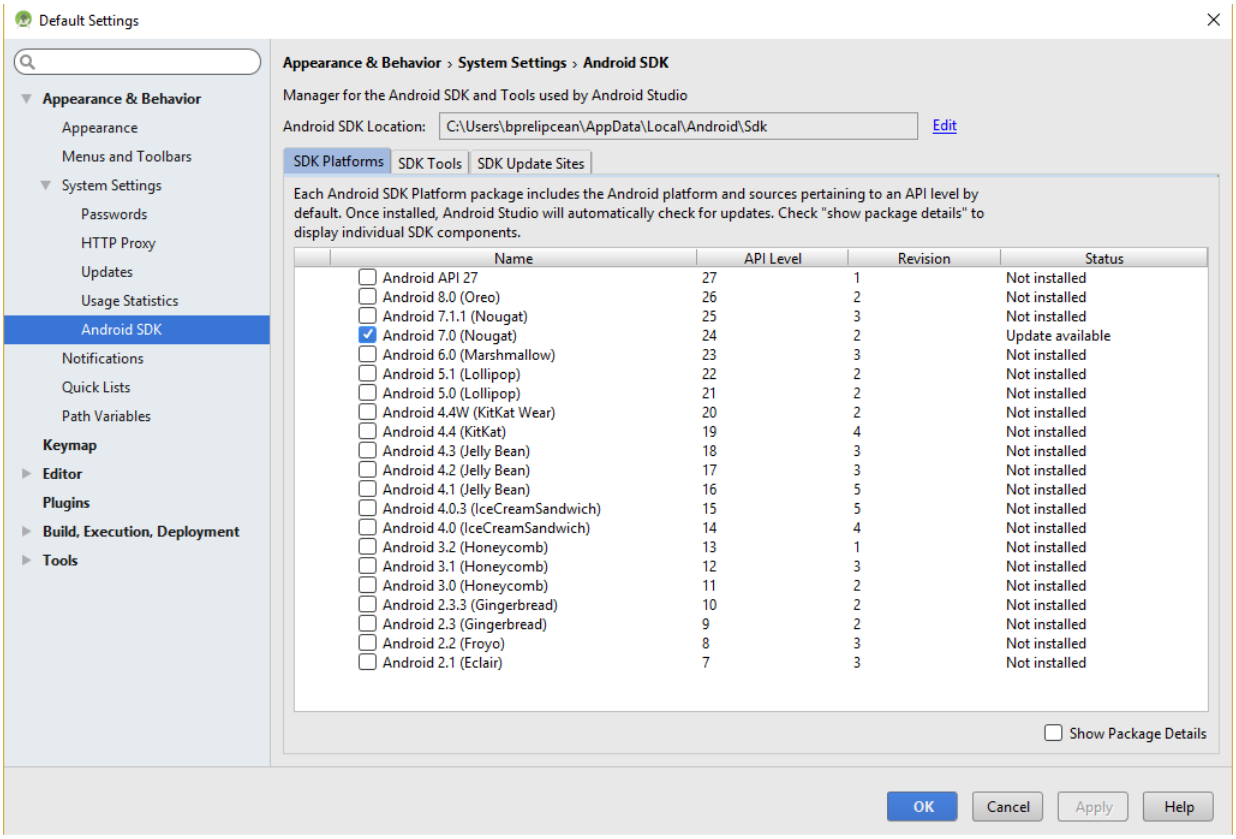
  `http://developer.android.com/sdk/index.html`
  Pentru folosirea unui IDE deja instalat este suficienta instalarea pluginurilor si a SDK-ului
- Exista varianta folosirii mediului **Eclipse** cu ADT plugin.

Utilitare

Android SDK Manager

Este folosit pentru instalarea componentelor si a actualizarilor.

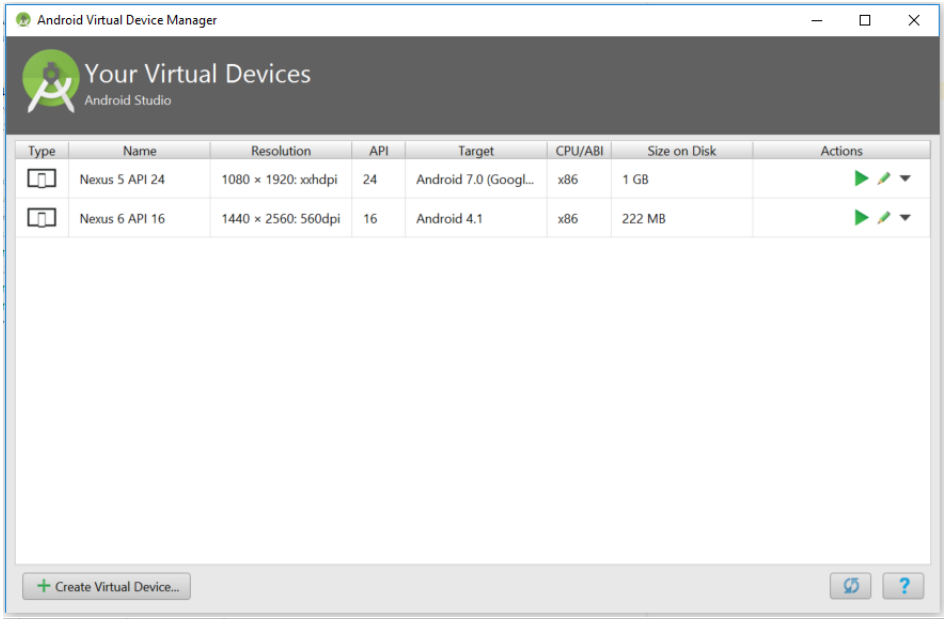
- Android Studio → Tools → Android → Android SDK Manager
- Eclipse → Window → Android SDK Manager
- *sdk-path*/SDK Manager.exe

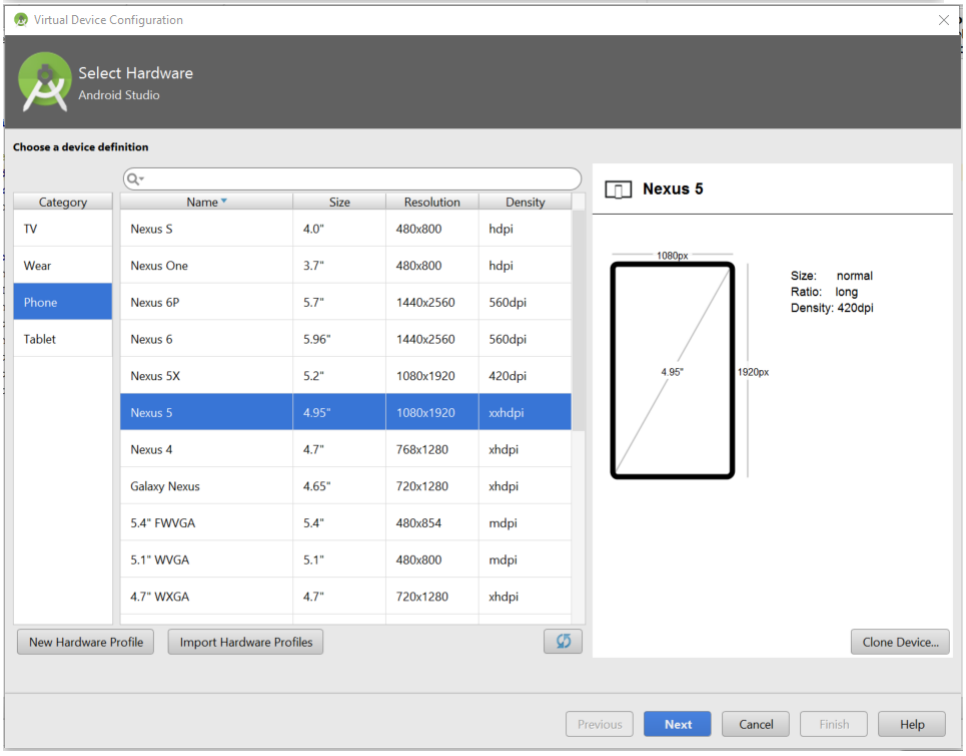


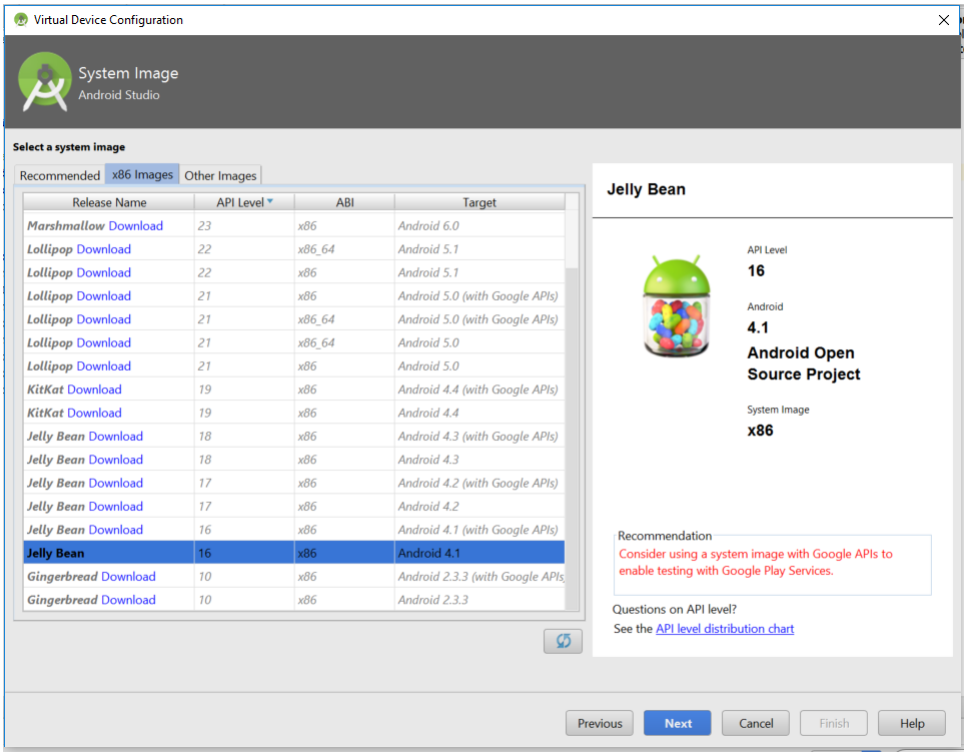
AVD Manager

Utilitar folosit pentru managementul dispozitivelor virtuale (Android Virtual Devices)

- Android Studio → Tools → Android → Android Virtual Devices Manager
- Eclipse → Window → Android Virtual Devices Manager
- *sdk-path*/tools/android.bat avd


Android Virtual Device Manager
Your Virtual Devices
Android Studio
Type
Name
Resolution
API
Target
CPU/ABI
Size on Disk
Actions
Nexus 5 API 24
1080 × 1920: xxhdpi
24
Android 7.0 (Googl...
x86
1 GB
Nexus 6 API 16
1440 × 2560: 560dpi
16
Android 4.1
x86
222 MB
Create Virtual Device...


Virtual Device Configuration
Select Hardware
Android Studio
Choose a device definition
Category
Name
Size
Resolution
Density
TV
Wear
Phone
Tablet
Nexus S
4.0"
480x800
hdpi
Nexus One
3.7"
480x800
hdpi
Nexus 6P
5.7"
1440x2560
560dpi
Nexus 6
5.96"
1440x2560
560dpi
Nexus 5X
5.2"
1080x1920
420dpi
Nexus 5
4.95"
1080x1920
xxhdpi
Nexus 4
4.7"
768x1280
xhdpi
Galaxy Nexus
4.65"
720x1280
xhdpi
5.4" FWVGA
5.4"
480x854
mdpi
5.1" WVGA
5.1"
480x800
mdpi
4.7" WXGA
4.7"
720x1280
xhdpi
Nexus 5
1080px
4.95"
1920px
Size: normal
Ratio: long
Density: 420dpi
New Hardware Profile
Import Hardware Profiles
Clone Device...
Previous
Next
Cancel
Finish
Help



Task: Creati un dispozitiv virtual.

Android Debug Bridge

Utilitar folosit pentru interactiunea cu dispozitivul(real sau virtual), necesar pentru debugging.
Se gaseste in:

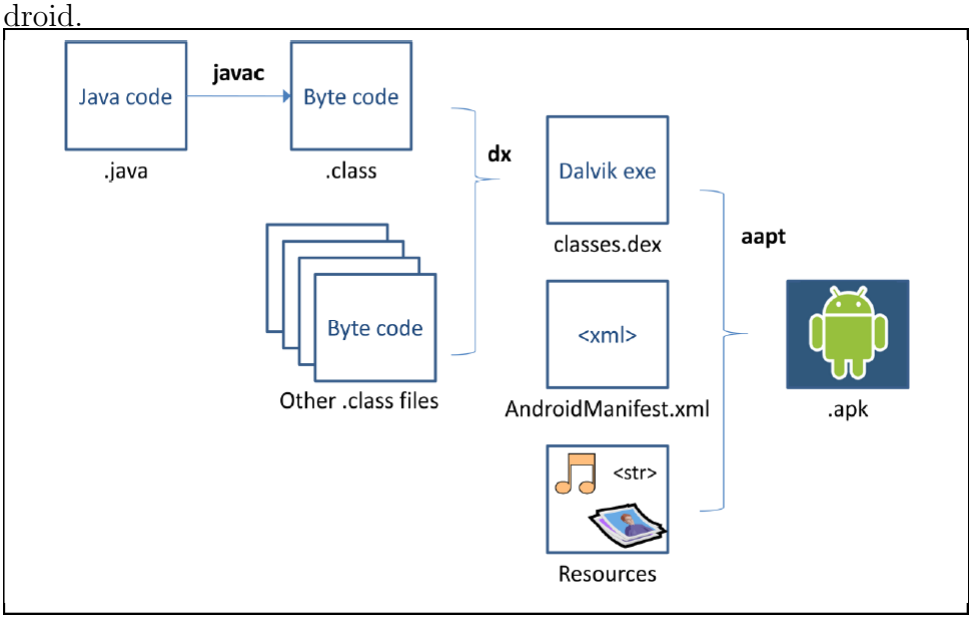
- *sdk-path*/platform-tools/adb.exe

Manualul complet: `http://developer.android.com/tools/help/adb.html`
Comenzi folosite frecvent:

- *devices* - afisarea dispozitivelor conectate
- *logcat* - afisarea logului
- *install* - instalarea unei aplicatii pe un dispozitiv
- *push* - copierea unui fisier pe un dispozitiv
- *pull* - copierea unui fisier de pe un dispozitiv
- *shell* - folosirea unei console remote de linux pe dispozitiv

Formatul unui fisier APK

Un fisier APK este folosit pentru distribuirea unei aplicatii pe un sistem Android.



Formatul este similar cu cel al unui fisier .JAR (arhiva ZIP cu meta-informatii) Componentele unui fisier APK:

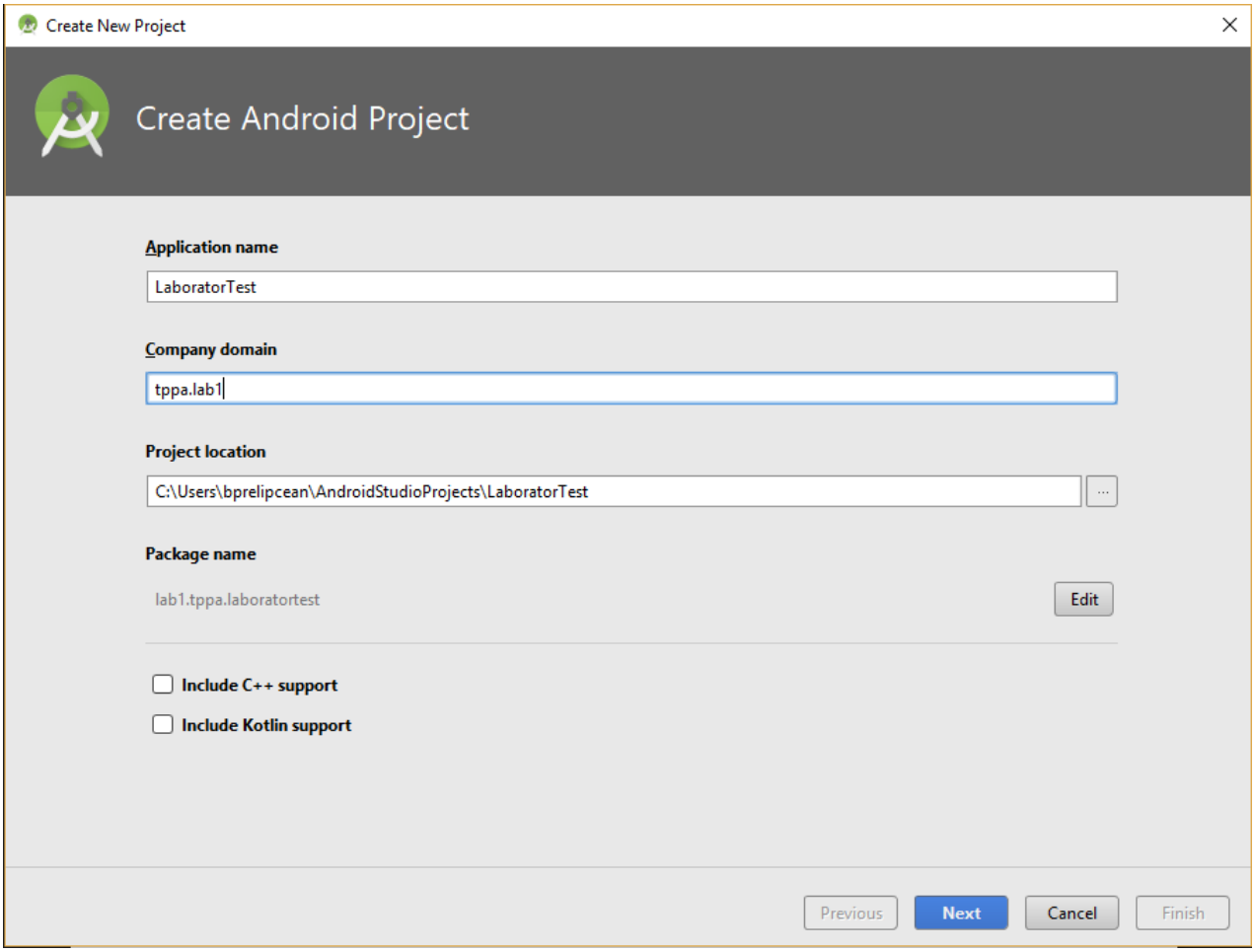
- Directorul **META-INF**:
  - MANIFEST.MF
  - CERT.RSA - certificatul applicatiei
  - CERT.SF - lista resurselor si rezumatul SHA-1 al liniilor din MANIFEST.MF
- Directorul **lib** - contine cod compilat nativ pentru diferite tipuri de procesor
  - armeabi - cod pentru toate procesoarele ARM
  - armeabi-v7a - cod pentru procesoarele ARMv7
  - x86 - cod pentru procesoarele x86
  - mips - cod pentru procesoarele MIPS
- Directorul **res** - contine resurse necompilate
- Directorul **assets** - resurse folosite de aplicatie

- **AndroidManifest.xml** - un fisier Android manifest ce specifica numele, versiunea, drepturile necesare, bibliotecile utilizate.
- **classes.dex** - clasele aplicatiei compilate in formatul dex, cel inteles de masina virtuala Dalvik
- **resources.arsc** - fisier ce contine resurse precompilate.

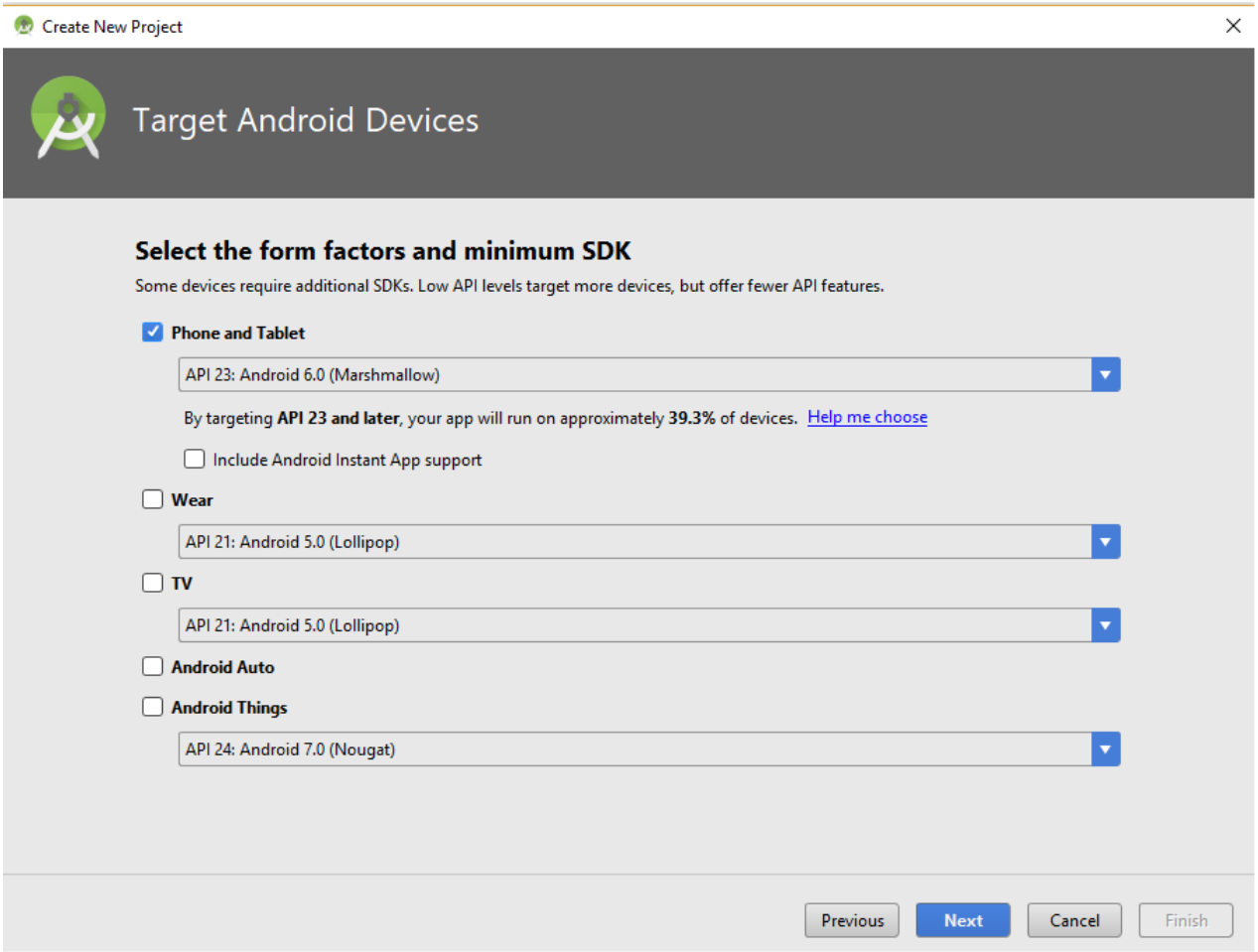
Prima aplicatie Android

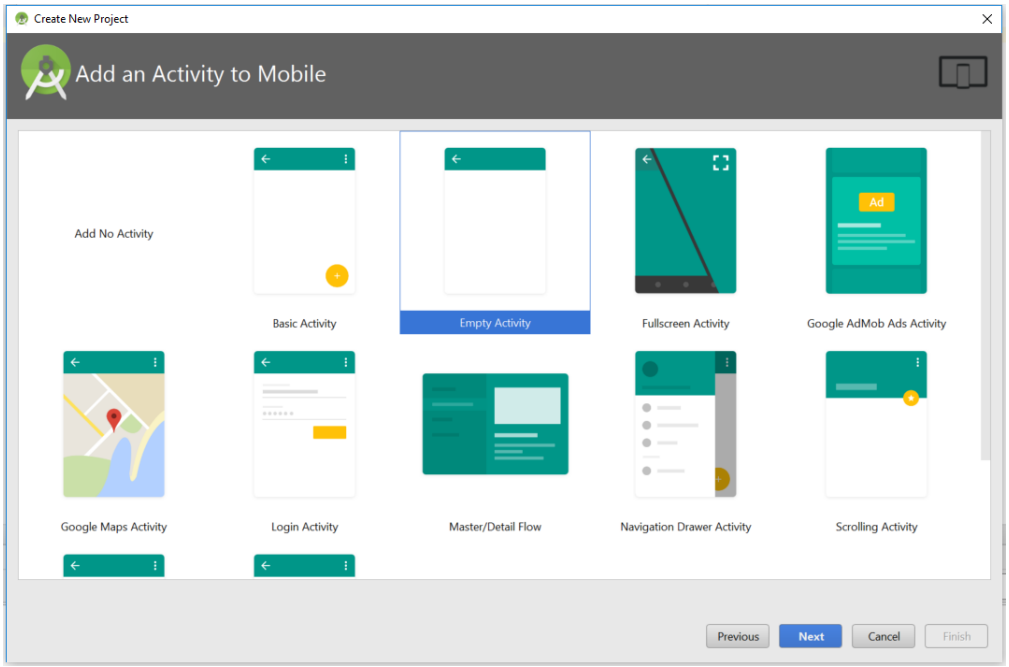
Crearea unui proiect nou:

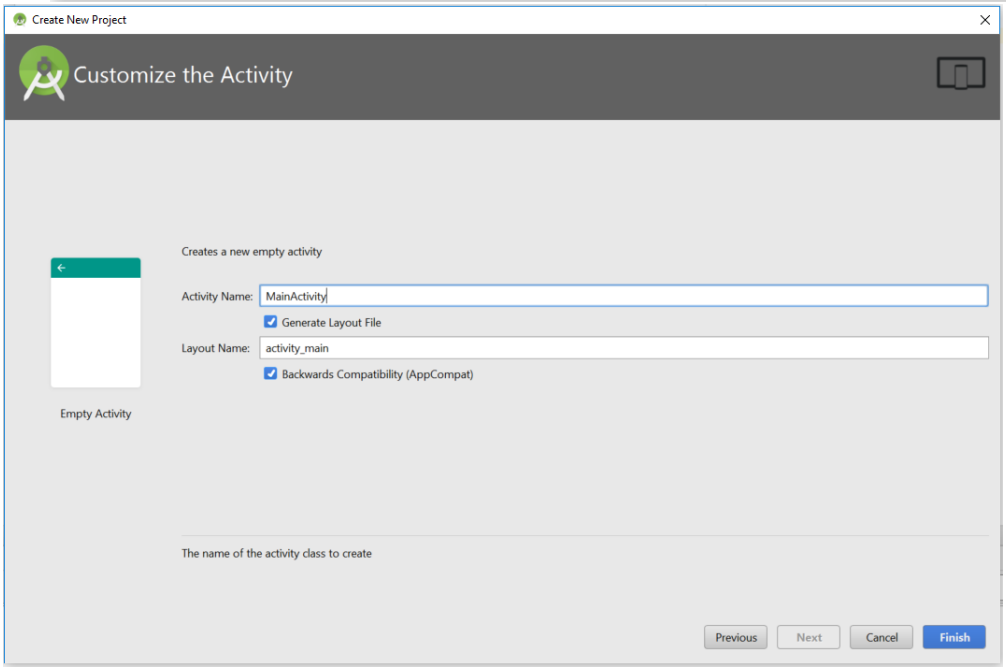
- File → New Project

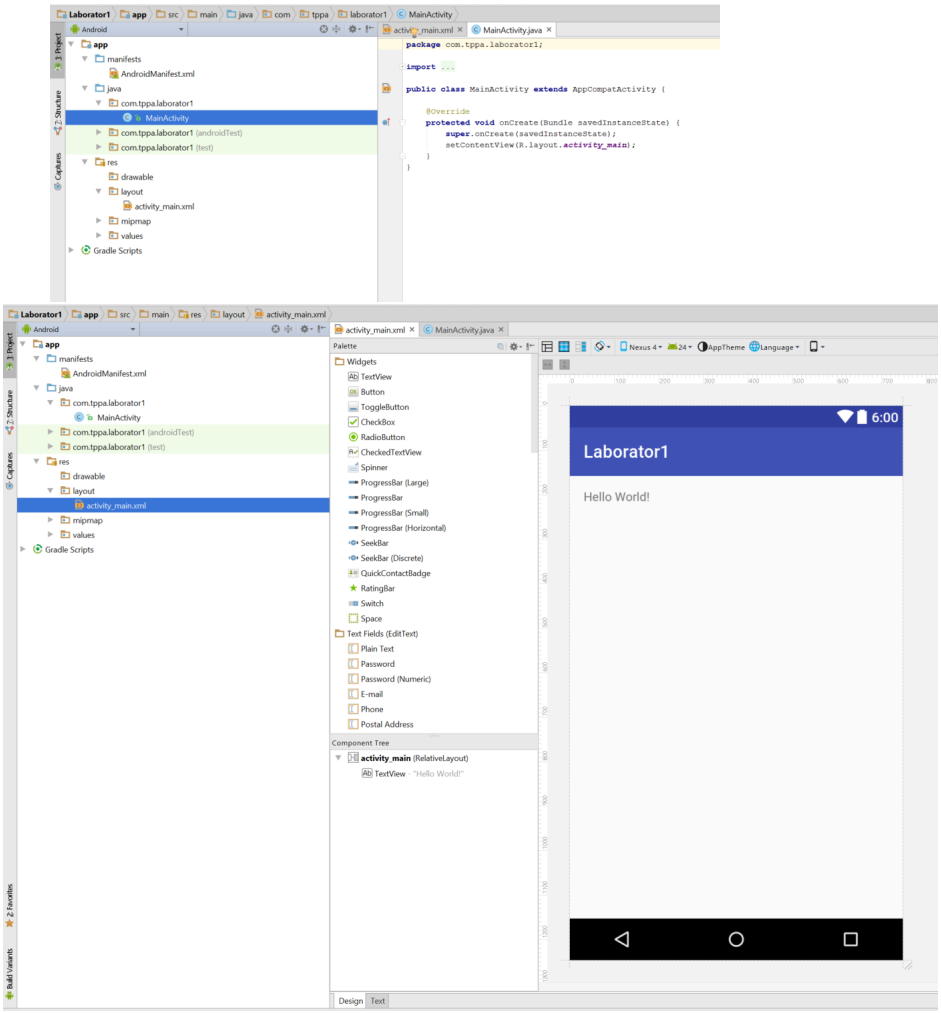


Configurarea aplicatiei:


Create New Project
Target Android Devices
Select the form factors and minimum SDK
Some devices require additional SDKs. Low API levels target more devices, but offer fewer API features.
Phone and Tablet
API 23: Android 6.0 (Marshmallow)
By targeting API 23 and later, your app will run on approximately 39.3% of devices. Help me choose
Include Android Instant App support
Wear
API 21: Android 5.0 (Lollipop)
TV
API 21: Android 5.0 (Lollipop)
Android Auto
Android Things
API 24: Android 7.0 (Nougat)
Previous
Next
Cancel
Finish


Create New Project
Add an Activity to Mobile
Add No Activity
Basic Activity
Empty Activity
Fullscreen Activity
Google AdMob Ads Activity
Google Maps Activity
Login Activity
Master/Detail Flow
Navigation Drawer Activity
Scrolling Activity
Previous
Next
Cancel
Finish


Create New Project
Customize the Activity
Creates a new empty activity
Activity Name: MainActivity
Generate Layout File
Layout Name: activity_main
Backwards Compatibility (AppCompat)
Empty Activity
The name of the activity class to create
Previous
Next
Cancel
Finish



Adaugati un control de tip **EditText** si un control de tip **Button**.Pentru butonul adaugat setati actiunea ce trebuie facuta(metoda apelata) in cazul evenimentului onClick

```xml
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/activity_main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="16dp"
    android:paddingLeft="16dp"
    android:paddingRight="16dp"
    android:paddingTop="16dp"
    tools:context="com.tppa.laborator1.MainActivity">

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World!"/>

    <EditText
        android:id="@+id/editText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:inputType="textPersonName"
        android:text="Name"
        android:ems="10"
        android:layout_below="@+id/textView"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true"
        android:layout_marginTop="42dp"/>

    <Button
        android:id="@+id/button"
        android:text="Button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignTop="@+id/editText"
        android:layout_alignParentRight="true"
        android:layout_alignParentEnd="true"
        android:onClick="clickTest"/>
</RelativeLayout>
```

In activitatea principala adaugati urmatoarea metoda, cu numele setat pentru butonul respectiv:

```java
public void clickTest(View view){
    EditText t1 = (EditText) findViewById(R.id.editText);
    TextView t2 = (TextView) findViewById(R.id.textView);
    t2.setText(t1.getText());
}
```

Porniti un dispozitiv virtual si lansati aplicatia.