

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 plugin-urilor si a SDK-ului

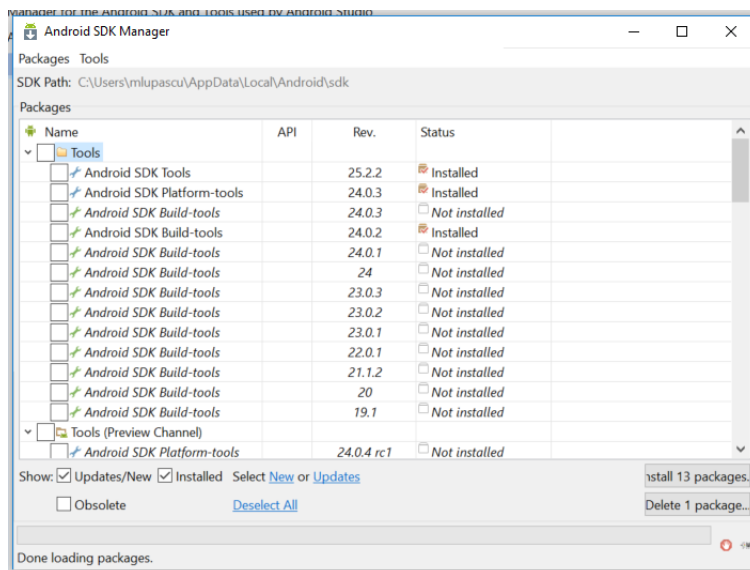
- Exista varianta folosirii mediului **Eclipse** cu ADT plugin.

Utilitare

Android SDK Manager

Este folosit pentru instalarea componentelor si a actualizarilor.

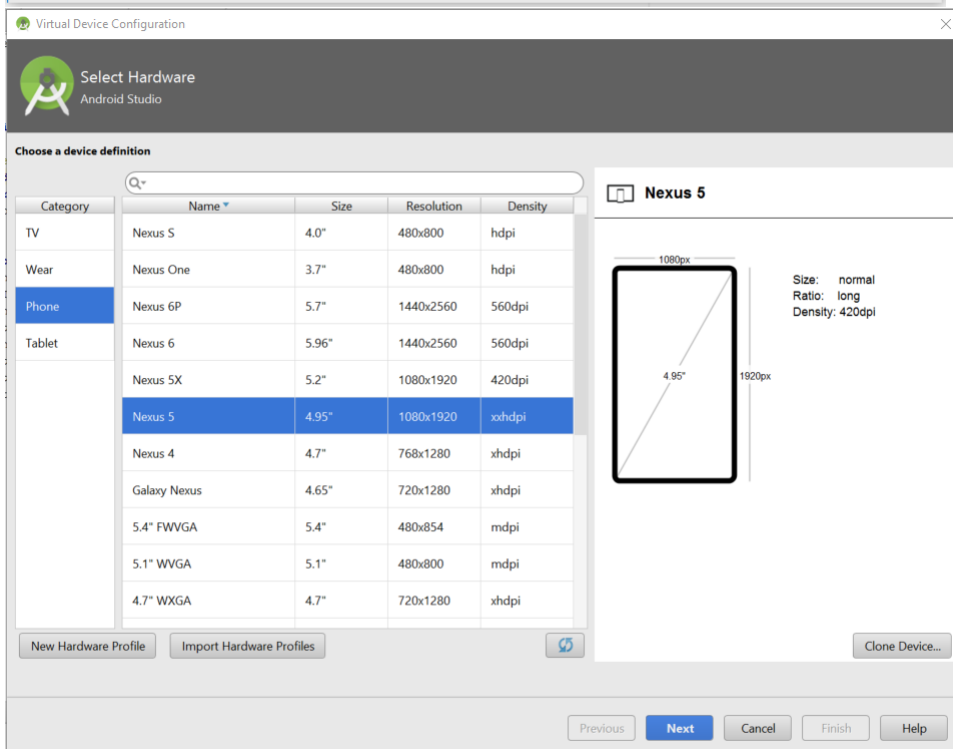
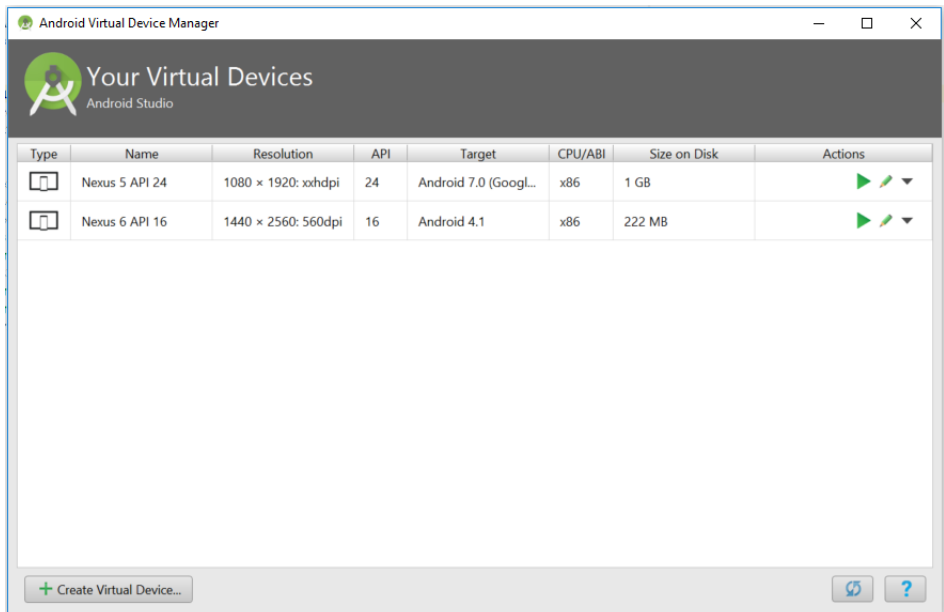
- Android Studio → Tools → Android → Android SDK Manager
- Eclipse → Window → Android SDK Manager
- *sdk-path*/SDK Manager.exe

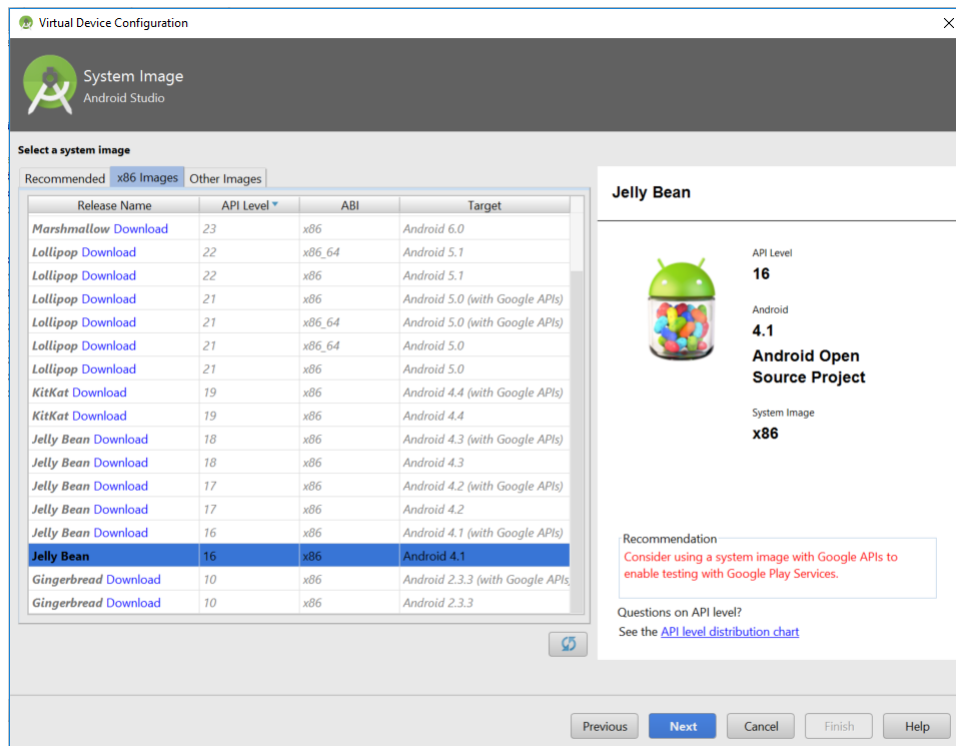


AVD Manager

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`





Task: Creați un dispozitiv virtual.

Android Debug Bridge

Utilitar folosit pentru interacțiunea cu dispozitivul (real sau virtual), necesar pentru debugging.

Se găsește în:

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

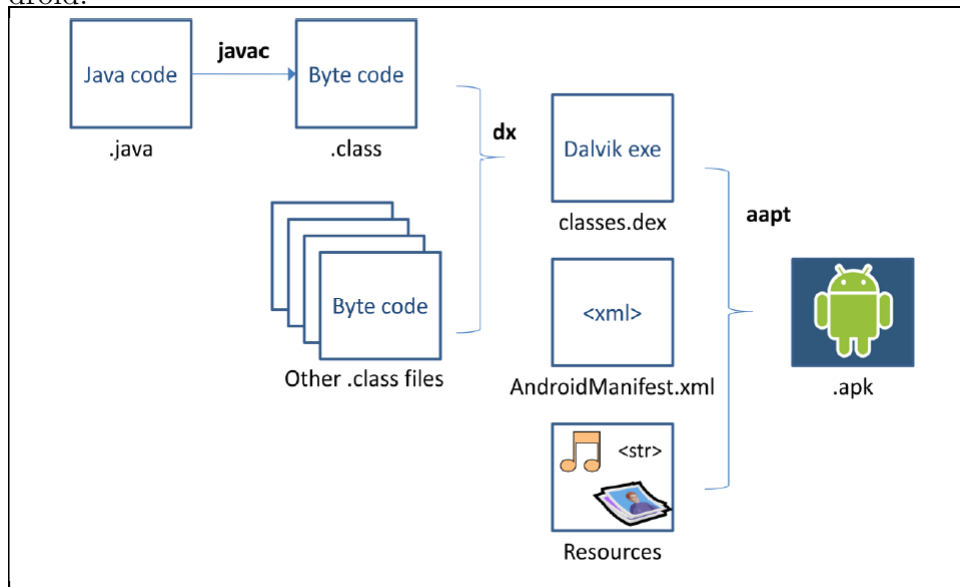
Manualul complet: <http://developer.android.com/tools/help/adb.html>

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.



For-

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

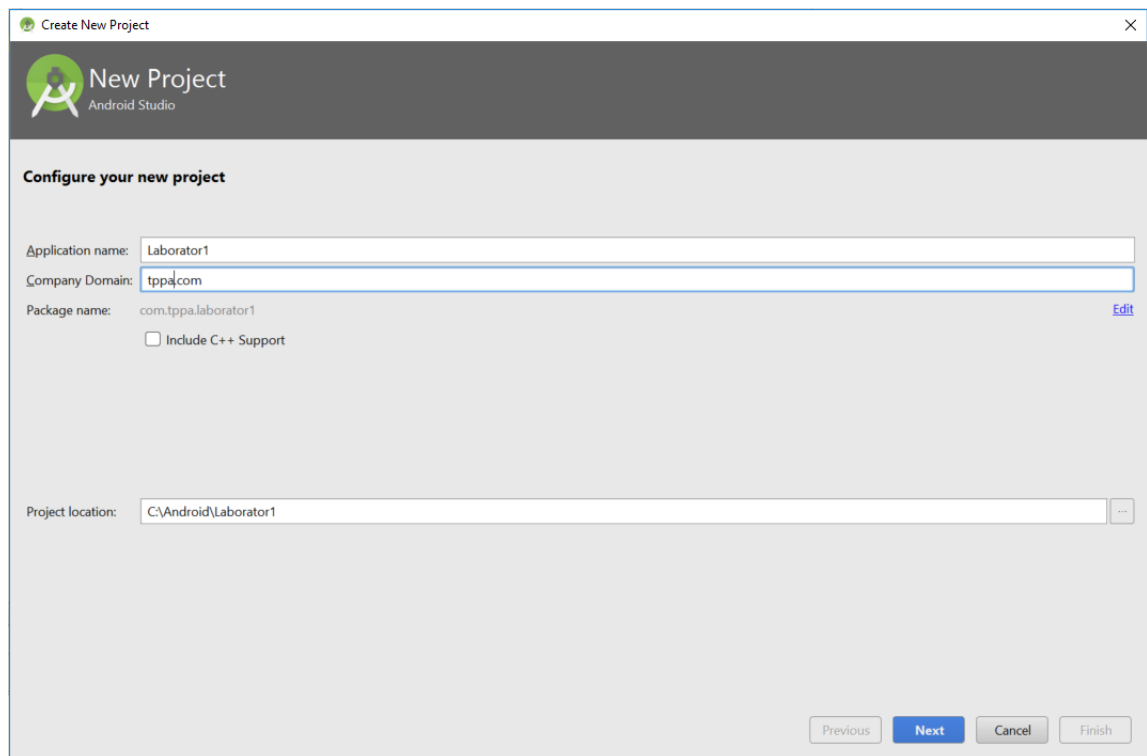
- Directorul **META-INF**:
 - MANIFEST.MF
 - CERT.RSA - certificatul aplicatiei
 - CERT.SF - lista resurselor si rezumatul SHA-1 al liniilor din MANIFEST.MF
- Directorul **lib** - contine cod compilat nativ pentru diferite tipuri de procesor
 - armeabi - cod pentru toate procesoarele ARM
 - armeabi-v7a - cod pentru procesoarele ARMeabi-v7
 - x86 - cod pentru procesoarele x86
 - mips - cod pentru procesoarele MIPS
- Directorul **res** - contine resurse necompilete
- 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



The screenshot shows the 'Create New Project' dialog box in Android Studio. The window title is 'Create New Project'. The header bar is dark gray with the Android Studio logo and the text 'New Project' and 'Android Studio'. The main area is light gray and titled 'Configure your new project'. It contains the following fields and options:

- Application name:** Laborator1
- Company Domain:** tppa.com
- Package name:** com.tppa.laborator1 (with an [Edit](#) link to its right)
- ☐ Include C++ Support
- Project location:** C:\Android\Laborator1 (with a browse button '...')

At the bottom right, there are four buttons: 'Previous' (disabled), 'Next' (active), 'Cancel' (disabled), and 'Finish' (disabled).

Configurarea aplicatiei:

Create New Project

Target Android Devices

Select the form factors your app will run on
Different platforms may require separate SDKs

☒ Phone and Tablet

Minimum SDK

API 16: Android 4.1 (Jelly Bean)

Lower API levels target more devices, but have fewer features available.
By targeting API 16 and later, your app will run on approximately **95.2%** of the devices that are active on the Google Play Store.
[Help me choose](#)

☐ Wear

Minimum SDK

API 21: Android 5.0 (Lollipop)

☐ TV

Minimum SDK

API 21: Android 5.0 (Lollipop)

☐ Android Auto

☐ Glass

Minimum SDK

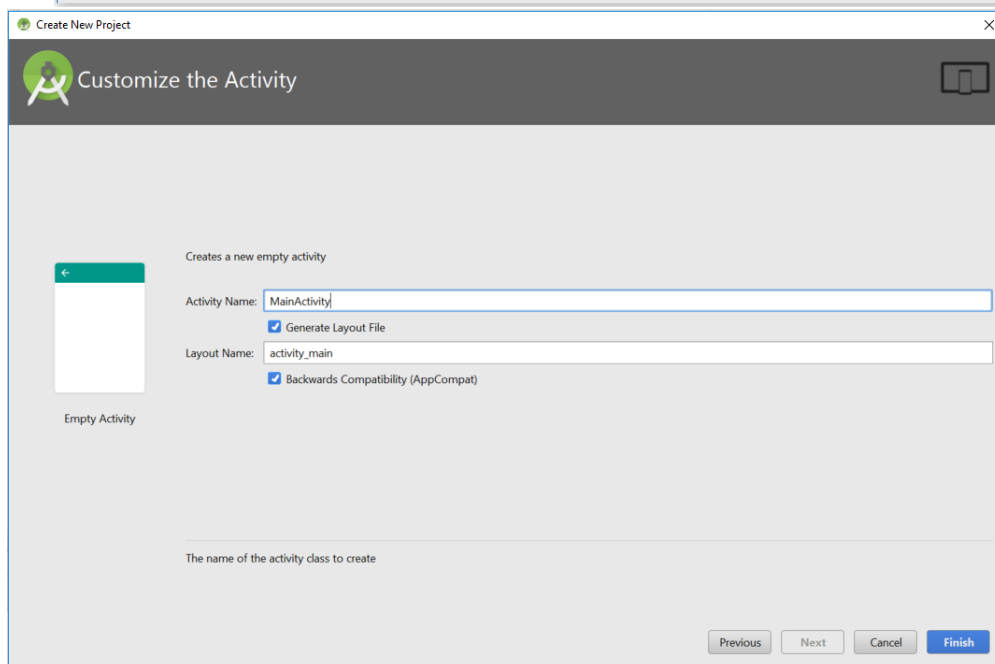
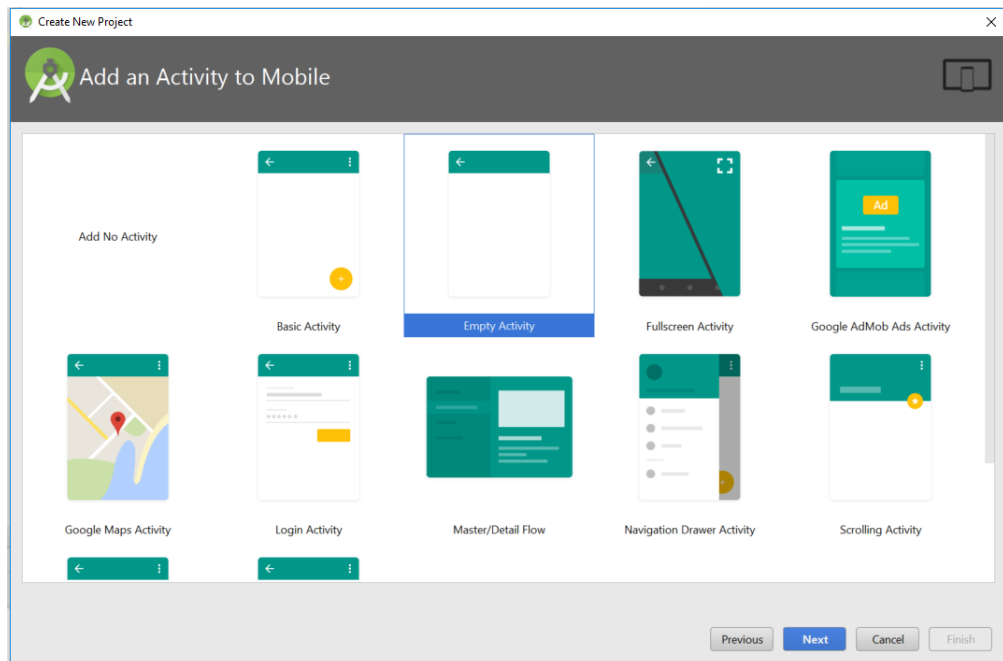
Glass Development Kit Preview (API 19)

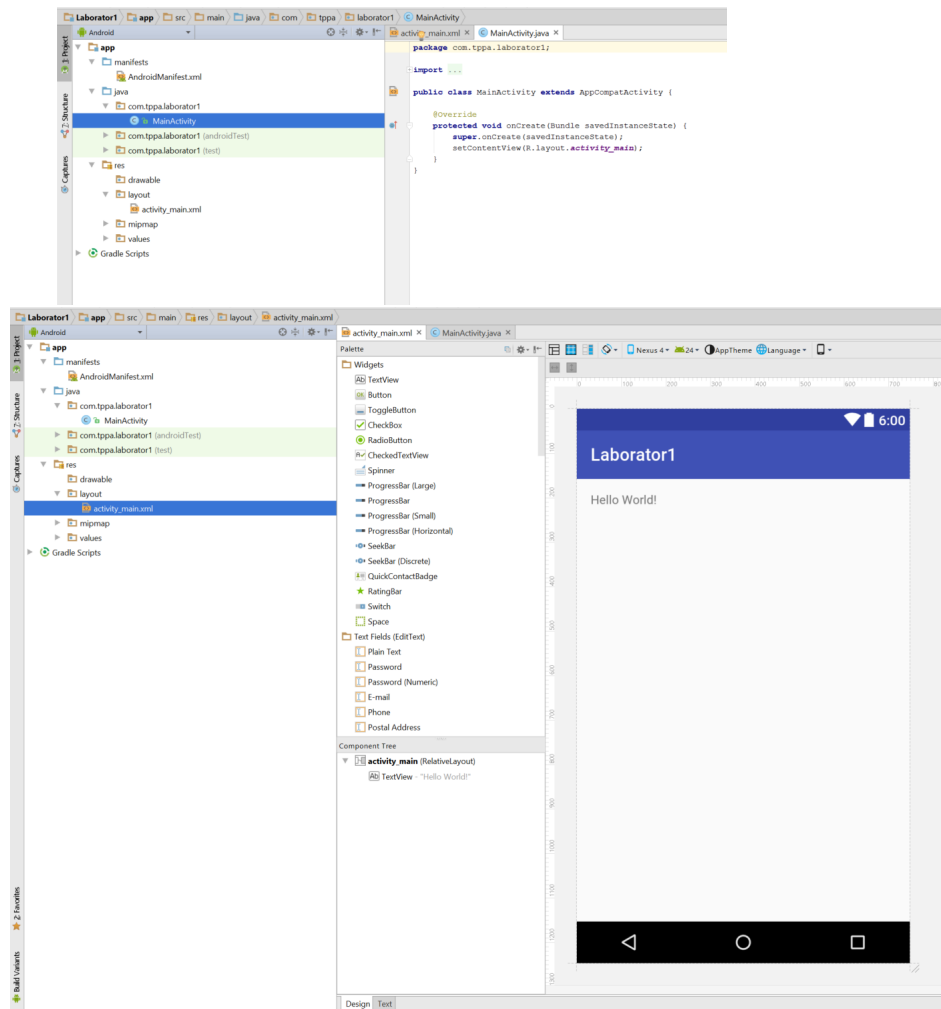
Previous

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 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:

```

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.