



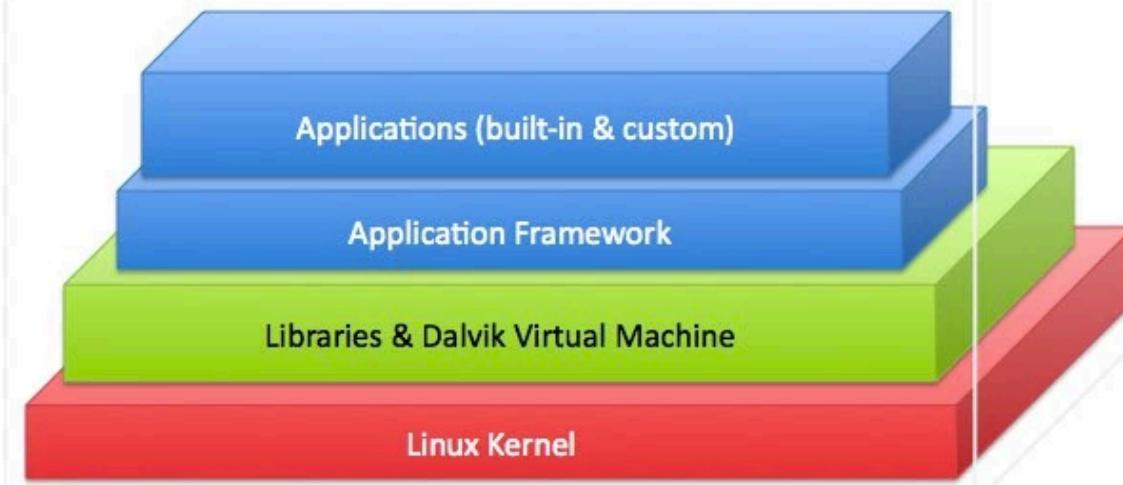
# Història

- A l'octubre de 2003 es va fundar Android Inc. i al juliol de 2005 Google va comprar la companyia.
- Presentació el 5 de novembre del 2007.
- Primer mòbil anunciat el 23 de setembre del 2008.
- Al 2015, 3 millions de dispositius activats cada dia.



# *Android*

*Software Environment*

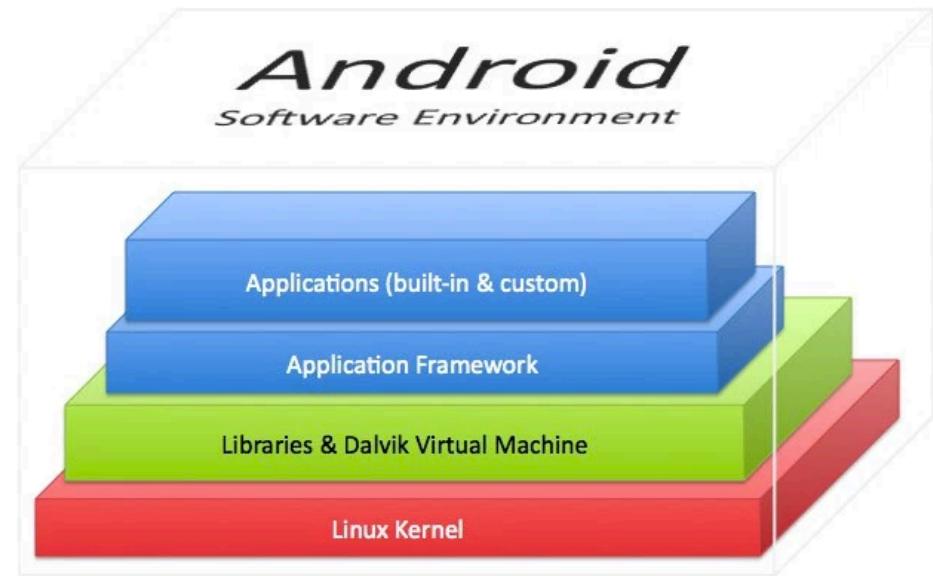


# Kernel



- Intermediari entre software i hardware.
- Controla totes les operacions que realitza la màquina.
- És independent del sistema operatiu.

# Dalvik Virtual Machine (DVM)

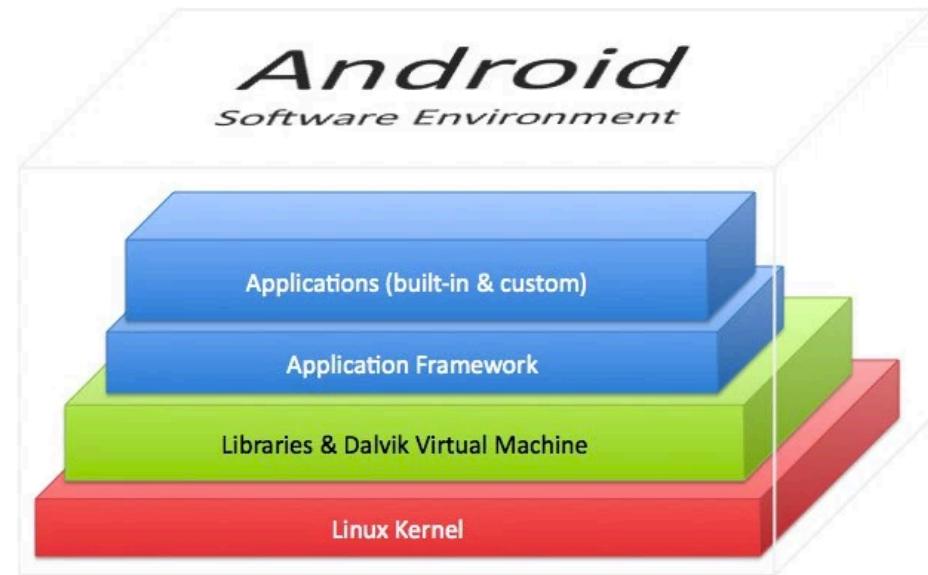


- Executa les aplicacions d'Android.
- Proporciona una execució independentment del hardware.
- Està dissenyat per a funcionar en dispositius amb pocs recursos.

# Dalvik

## ART

### Android Runtime



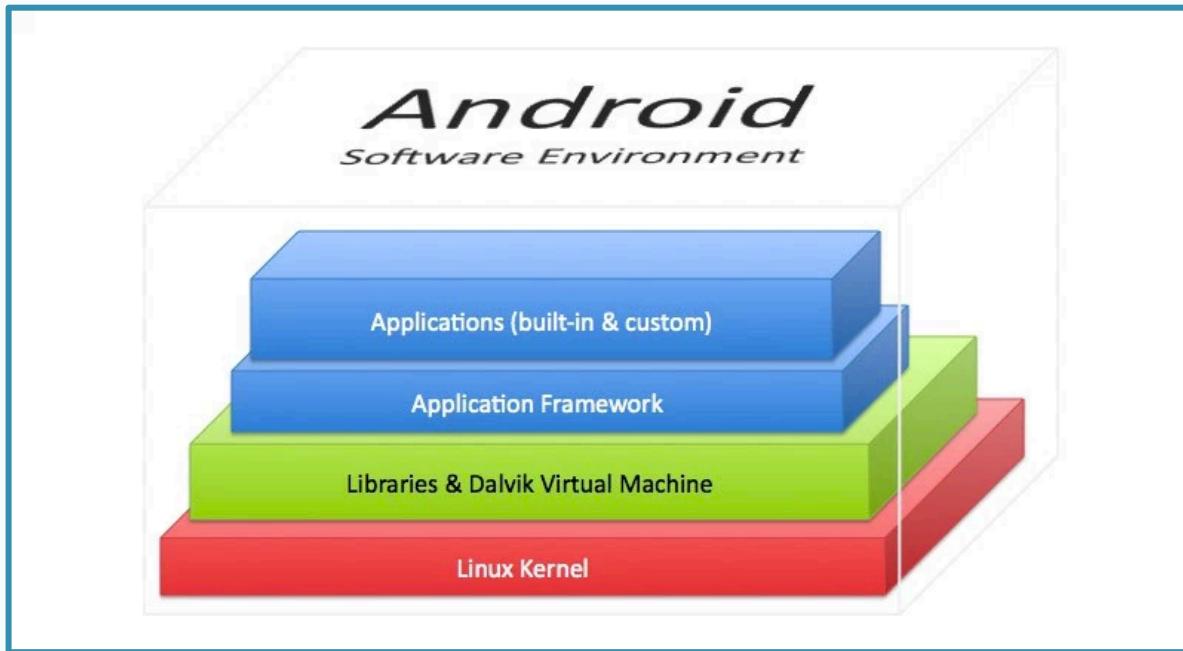
- Executa les aplicacions d'Android.
- Proporciona una execució independent del hardware.
- Està dissenyat per a funcionar en dispositius amb pocs recursos.
- A partir d'Android 4.4 (KitKat).
- Velocitat x4.

# Application Framework

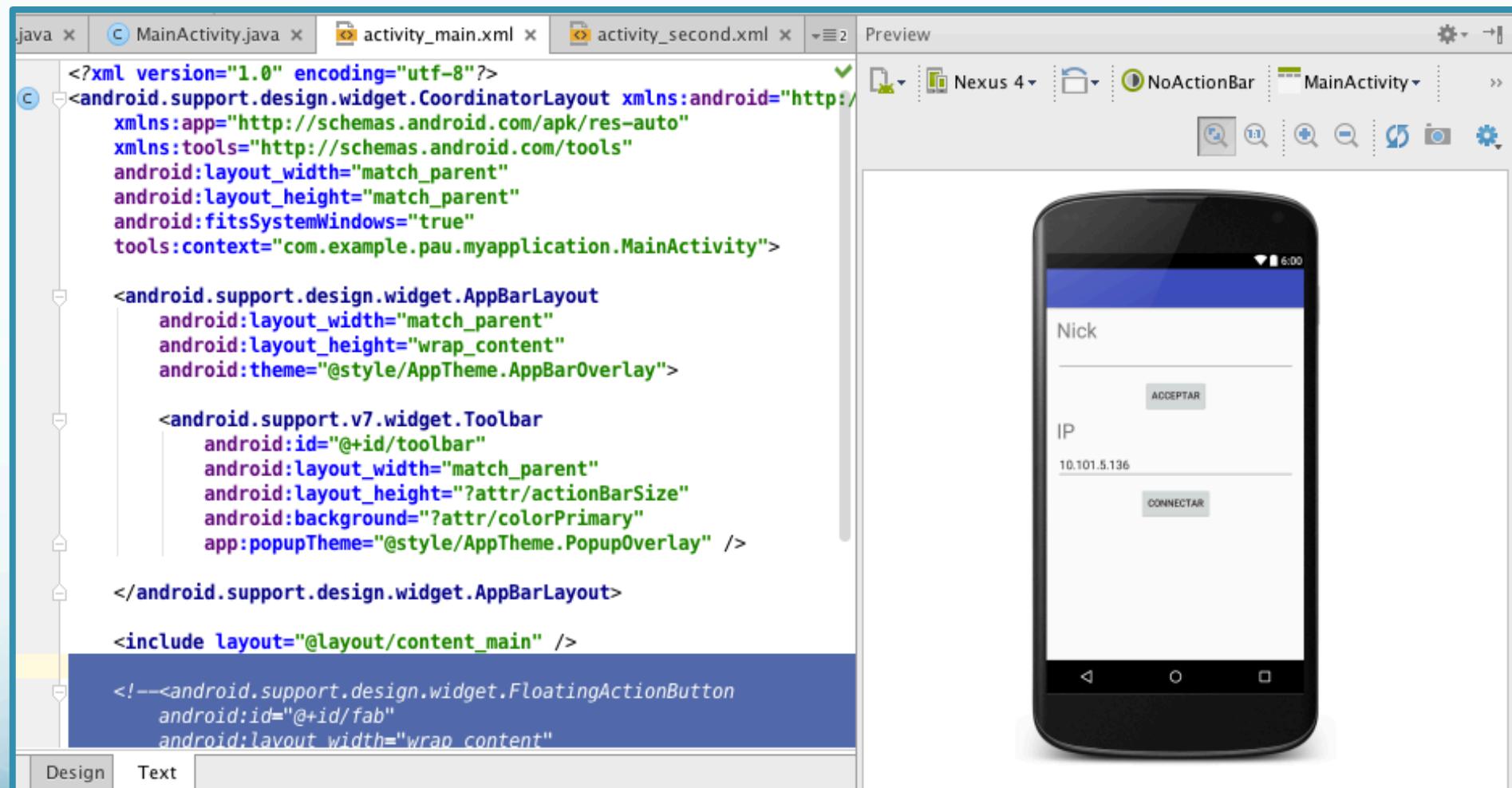


- Activity Manager.
- Content Provider.
- Notification Manager.
- View System.

# Applications



# Android Studio



The screenshot shows the Android Studio interface with the following details:

- Top Bar:** Shows tabs for `MainActivity.java`, `activity_main.xml`, and `activity_second.xml`.
- Left Panel:** A tree view of the XML file structure.
- XML Editor:** Displays the `activity_main.xml` code in XML format. The code includes a CoordinatorLayout with an AppBarLayout containing a Toolbar, and an include tag for the content\_main layout. A floatingActionButton tag is also present but commented out.
- Preview Pane:** Shows a smartphone screen with a blue header bar. The main content area has two text input fields labeled "Nick" and "IP" with the value "10.101.5.136". Below the IP field is a "CONNECTAR" button. Above the IP field is an "ACCEPTAR" button.
- Bottom Navigation:** Buttons for "Design" and "Text".

# Applications Development Frameworks

- Entorns de treball.
- Suposen una gran facilitat però també limitacions.
- Un *framework* no està lligat a un llenguatge encara que és típic: “Ruby on rails”.

## **Ruby on Rails: El desarrollo web que no molesta**

**Ruby on Rails** es un entorno de desarrollo web de código abierto que está optimizado para la satisfacción de los programadores y para la productividad sostenible. Te permite escribir un buen código evitando que te repitas y favoreciendo la convención antes que la configuración.

**¿Qué es Rails?** Rails es un armazón para construir aplicaciones web que acceden a bases de datos.

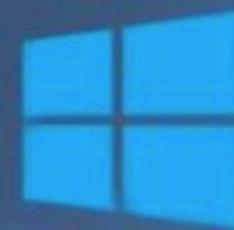
**¿Un arma-qué?** Un conjunto de librerías, automatismos y convenciones destinados a resolver los problemas más comunes a la hora de desarrollar una aplicación web, para que el programador pueda concentrarse en los aspectos únicos y diferenciales de su proyecto en lugar de los problemas recurrentes.

**¿Para qué se utiliza?** Para ayudar a construir aplicaciones modernas de internet como: [Twitter](#), [Scribd](#), [Hulu](#), [Xing](#), [Soundcloud](#), [Basecamp](#), [Github](#)...



Corona SDK: Make a game in 8 minutes

# Xamarin



create apps in C# for  
iOS, Android and Mac

# Xat Android

- AndroidManifest.xml
- Classes “.java”
- res.layouts

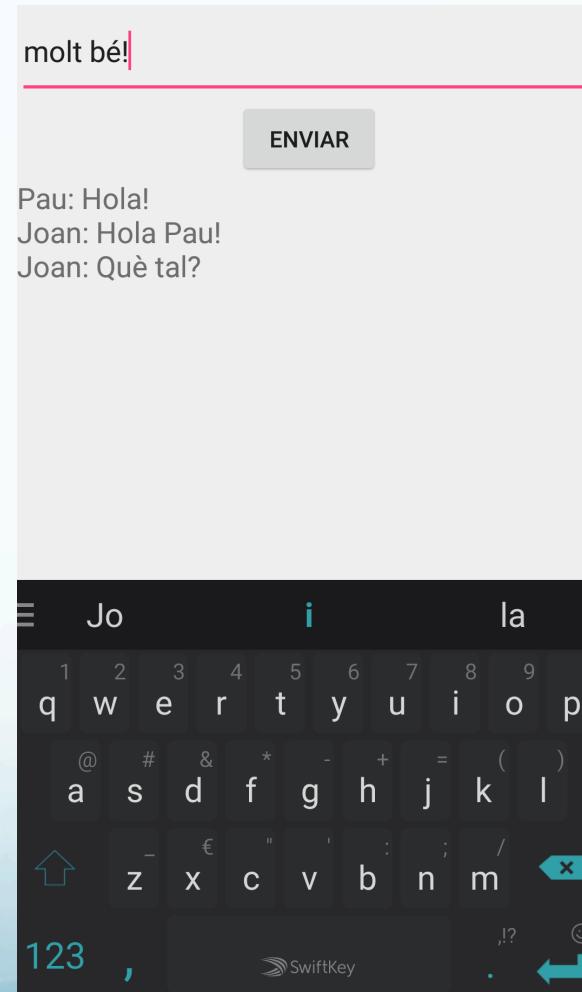
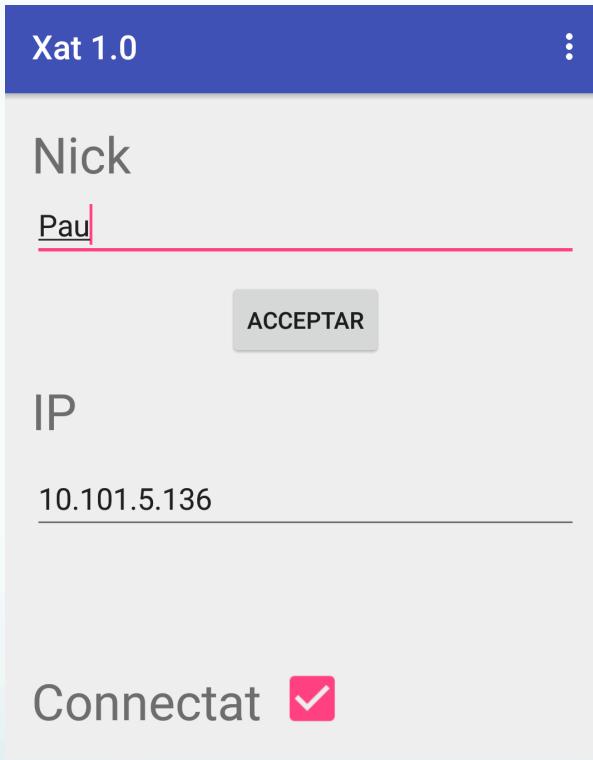
# AndroidManifest.xml

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
```

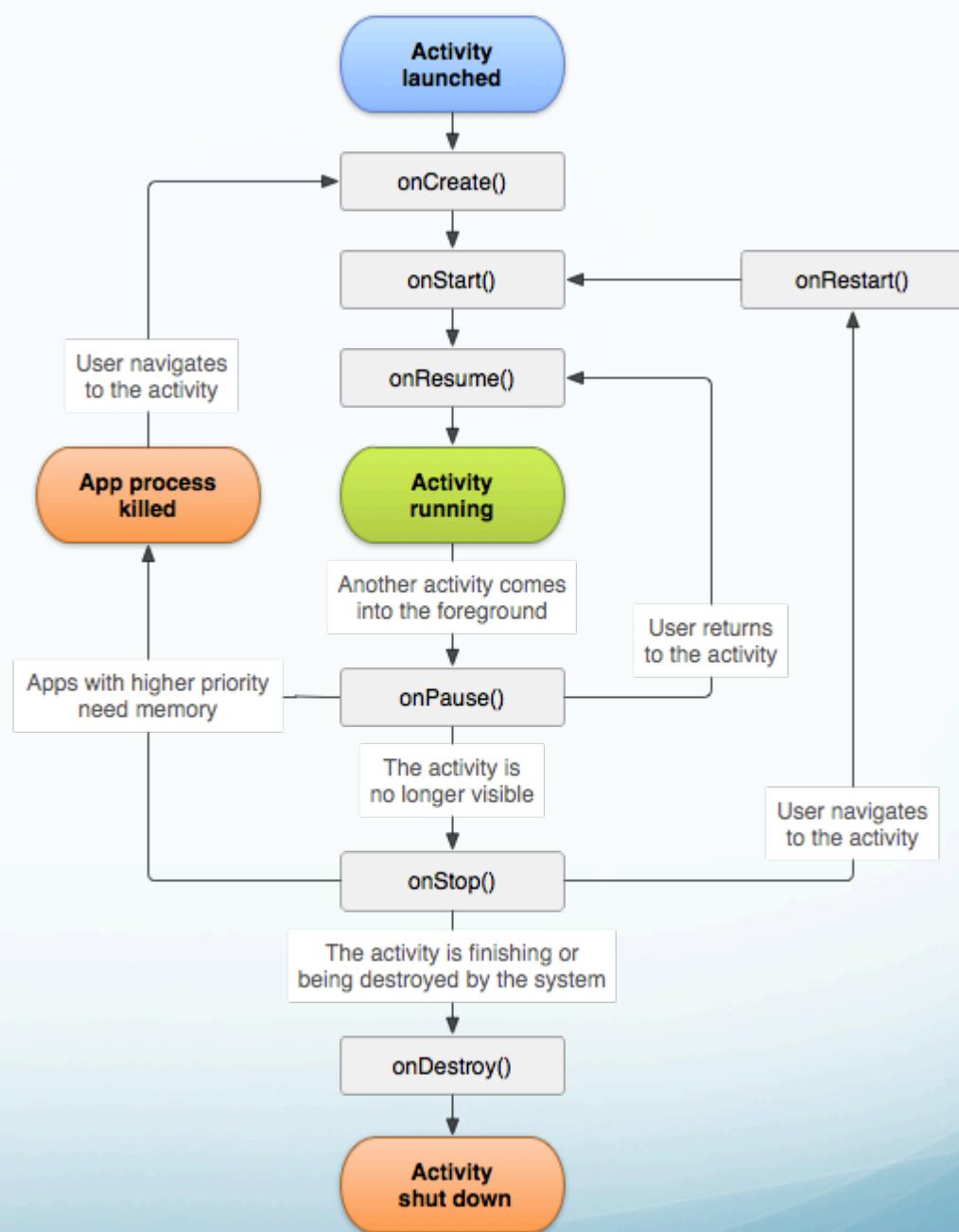
```
<activity
    android:name=".MainActivity"
    android:label="Xat 1.0"
    android:theme="@style/AppTheme.NoActionBar">
```

# Activity

- Cada pantalla d'una aplicació és una activitat.



# Activity lifecycle



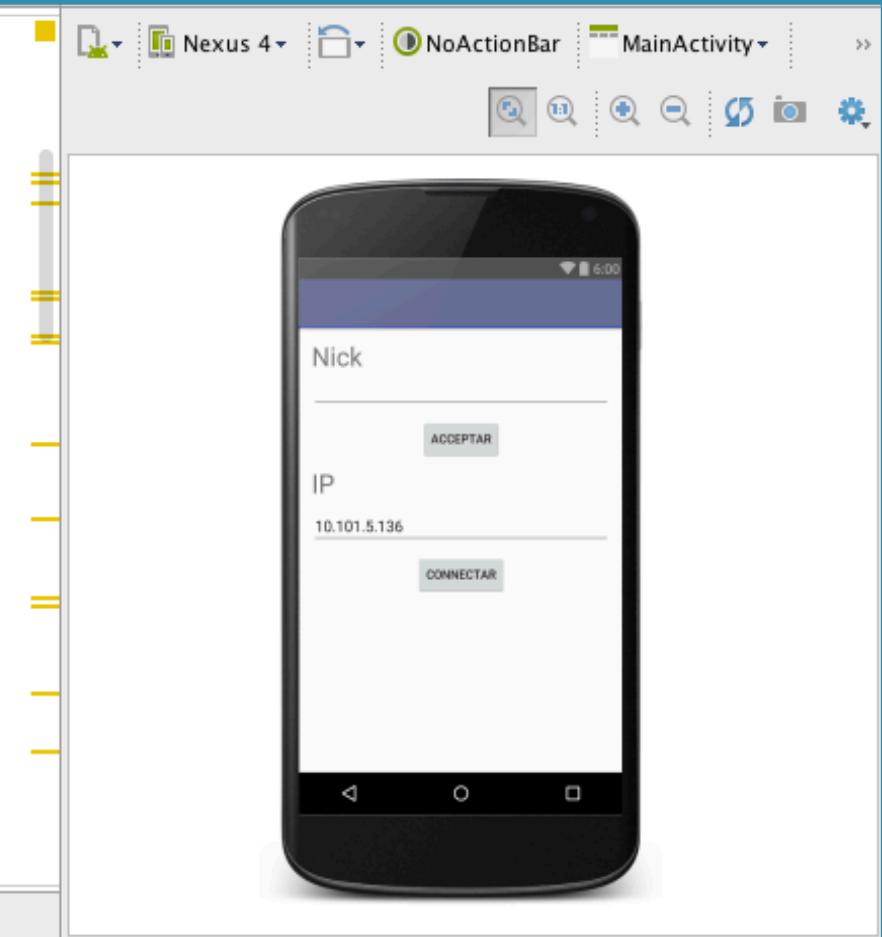
# Activity

- Té dues parts: la part lògica (\*.java, “extends Activity”) i la part gràfica (\*.xml).

```
public class MainActivity extends AppCompatActivity {  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {
```

# res.layout.content\_main.xml

```
<TextView  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="Nick"  
    android:textSize="30dp"  
    android:id="@+id/textView" />  
  
<EditText  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:id="@+id/nick" <--  
    android:layout_below="@+id/textView"  
    android:layout_alignParentLeft="true"  
    android:layout_alignParentStart="true"  
    android:layout_alignParentRight="true"  
    android:layout_alignParentEnd="true" />  
  
<TextView  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="IP"  
    android:textSize="30dp"  
    android:layout_below="@+id/button"  
    android:id="@+id/ipp"  
    android:layout_marginTop="10dp"  
    />  
<EditText  
    android:text="10.101.5.136"
```



# MainActivity.java

```
public void sendNick(View view) {
    if(connectat) {
        EditText et = (EditText) findViewById(R.id.nick);
        CharSequence cs = et.getText();

runOnUiThread(new Runnable() {
    @Override
    public void run() {
        connectat = true;
        Button b = (Button) findViewById(R.id.connect);
        b.setVisibility(Button.INVISIBLE);
```

# MainActivity.java

- import android.view.\*;

```
@Override  
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main);  
    Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);  
    setSupportActionBar(toolbar);  
  
    <Button  
        android:onClick="botoConnectar"  
  
public void botoConnectar(View view) {  
    EditText et = (EditText) findViewById(R.id.ipField);  
    CharSequence cs = et.getText();  
    if(checkIp(cs.toString())){  
        TextView tw = (TextView) findViewById(R.id.connectant);  
        tw.setVisibility        ProgressBar pb = (ProgressBar) findViewById(R.id.pbLoading);  
        pb.setVisibility(ProgressBar.VISIBLE);  
        iniciarClient();  
    }  
}
```

# Segona activitat

```
//Iniciar segona activitat
Intent intent = new Intent(MainActivity.this, SecondActivity.class);
startActivity(intent);
//acabar primera activitat
finish();
```

String nick = “Joan”;

intent.putExtra(“nick”, nick);

startActivity(intent);

A la segona activitat:

Intent intent = getIntent();

String nick = intent.getStringExtra(“nick”);

# MySocket.java

```
public void connect (int port, String ip) {  
    try {  
        InetAddress addr = InetAddress.getByName(ip);  
        SocketAddress socketAddress = new InetSocketAddress(addr, port);  
        socket = new Socket();  
        socket.bind(null);  
        socket.connect(socketAddress, 1000);  
        initStreams();  
    } catch (IOException ex) {  
        Log.e("MySocket", "IOException #####");  
        connect(12345,ip);  
    }  
}
```

- Root. KingoRoot.
- SuperSU.
- Custom ROMs.

