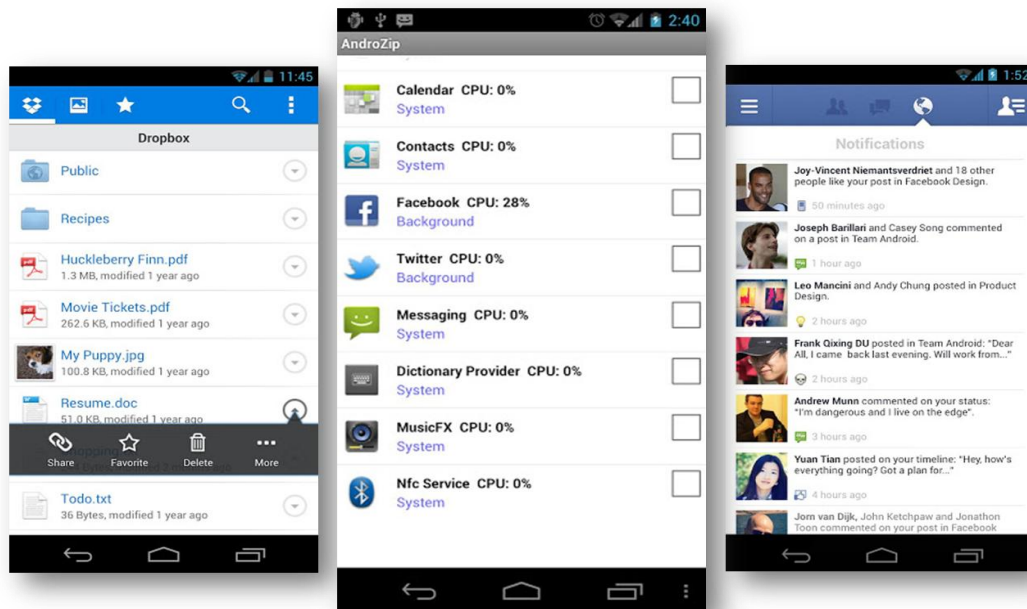


## ListView en Android

Son las llamadas listas que mostraran al usuario de forma ordenada y organizada una serie de datos:



## Tipos de ListView

En Android cuando estamos desarrollando podemos encontrarnos 2 tipos:

- Los que vienen ya diseñados y aporta android para poder usarlos por defecto
- ListView personalizados creados por nosotros

## Creación de un ListView no Personalizados

Lo primero que tenemos que hacer es añadirlo o crearlo en nuestra vista. Para ello añadimos el componente y le ponemos un id por defecto.

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.android.listviewejemplo.MainActivity"
    android:weightSum="1">
    //Nuestro ListView
    <ListView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/listaPersonas"
        android:layout_alignParentLeft="true"
        android:layout_alignParentTop="true"
        android:layout_weight="1.45" />
</LinearLayout>
```

Una vez añadido nos desplazamos hacia nuestra actividad (MainActivity.class). En ella tendremos que realizar los siguientes pasos:

- **Creamos las variables globales que necesitaremos**  
`String[] personas;`  
`ListView lista;`
- **Inicializamos el ListView**  
`lista = (ListView) findViewById(R.id.listaPersonas);`
- **Insertamos datos en nuestro Array:**  
`personas = new String[]`  
`{ "Pepe", "Miguel", "María", "Ana", "Paco", "Japón", "Antonio" };`
- **Creamos un Adaptador:**  
`ArrayAdapter<String> adaptador = new ArrayAdapter<String>(this,`  
`android.R.layout.simple_list_item_1, personas);`

Los parámetros a pasarle son:

- **Contexto:** si estamos trabajando en un Activity el contexto es (**this** o **NombreClaseActivity.this**). Si estamos trabajando con fragmento tendríamos que llamar al contexto con **getActivity()**
- **Nombre del tipo de layout para mostrar la lista:** En Android existen layout predefinidos que pueden ser usados llamándolos con (android.R.layout.**nombrelayout**)

Ejemplo: android.R.layout.**simple\_list\_item\_1**

- **Lista a mostrar**
- **Añadimos el adaptador a nuestro ListView**  
`lista.setAdapter(adaptador);`
- **Para que los ítems de la lista tenga Eventos implementamos el método **setOnItemClickListener()**. Nos pedirá que implementemos dicho método**

```
lista.setOnItemClickListener(this);
```

```
@Override  
public void onItemClick(AdapterView<?> parent, View view, int  
position, long id) {  
    //Operaciones a realizar  
}
```

## Código Final

```
public class MainActivity extends AppCompatActivity implements
AdapterView.OnItemClickListener {

    String[] personas;
    ListView lista;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        //Inicializamos nuestro listView
        lista = (ListView) findViewById(R.id.listaPersonas);
        //Creamos una lista de prueba para mostrarla
        personas = new String[]
{"Pepe", "Miguel", "María", "Ana", "Paco", "Japón", "Antonio"};
        //Creamos un adaptador
        ArrayAdapter<String> adaptador = new
ArrayAdapter<String>(this, android.R.layout.simple_list_item_1,
personas);
        //Añadimos el adaptador a nuestro listView para que se muestre
        lista.setAdapter(adaptador);

        //Para tener eventos implementamos el metodo
        setOnClickListener
        lista.setOnItemClickListener(this);
    }

    @Override
    public void onItemClick(AdapterView<?> parent, View view, int
position, long id) {
        Toast.makeText(MainActivity.this, personas[position],
Toast.LENGTH_SHORT).show();
    }
}
```

## Creación de ListView Personalizados