



Instituto Politécnico Nacional

Escuela Superior de Cómputo.



## **ACTIVIDADES Y EJERCICIOS**

### **ASIGNATURA:**

DESARROLLO DE APLICACIONES MOVILES NATIVAS

### **ALUMNO:**

PÉREZ MONDRAGÓN EDUARDO

### **Grupo:**

7CM1

### **Docente:**

JOSE ASUNCION ENRIQUEZ ZARATE

Enlace a Github:

<https://github.com/odraude67/Desarrollo-de-aplicaciones-moviles-nativas>

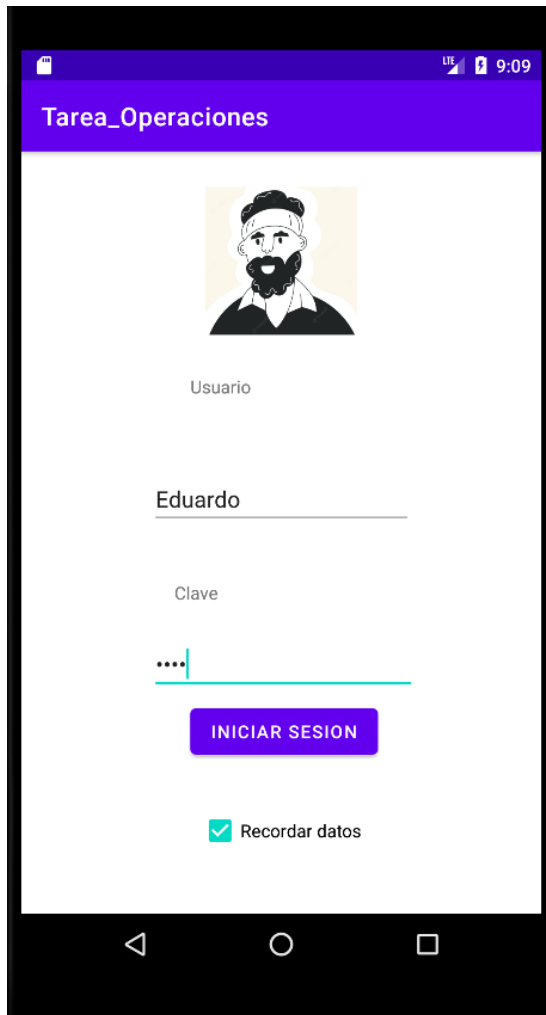
23/Junio/2023.

## índice

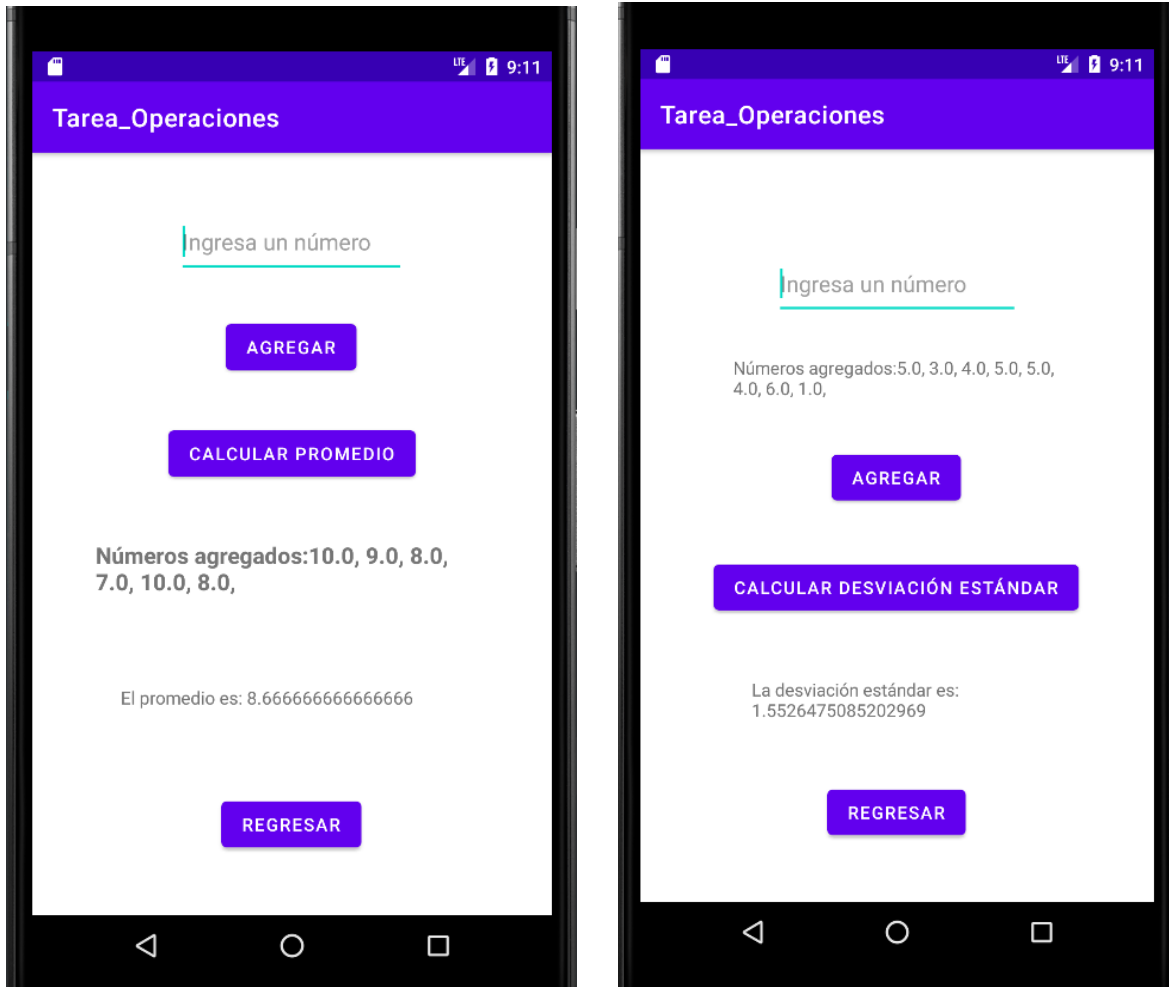
1. Operaciones .....	2
2. Login .....	5
3. Lista de frutas .....	8
4. Lista de Sensores.....	9
5. Lista de animes .....	10
6. Lista de frases .....	12
7. Ejemplo de BD - Frases .....	13
8. Retrofit - Pokedex.....	16

## 1. Operaciones

En la siguiente aplicación se muestra el cambio básico de vistas mediante el uso de un botón, en donde al cambiar de vista se visualiza una serie de botones para poder realizar las operaciones correspondientes de promedio, desviación estándar y factorial.



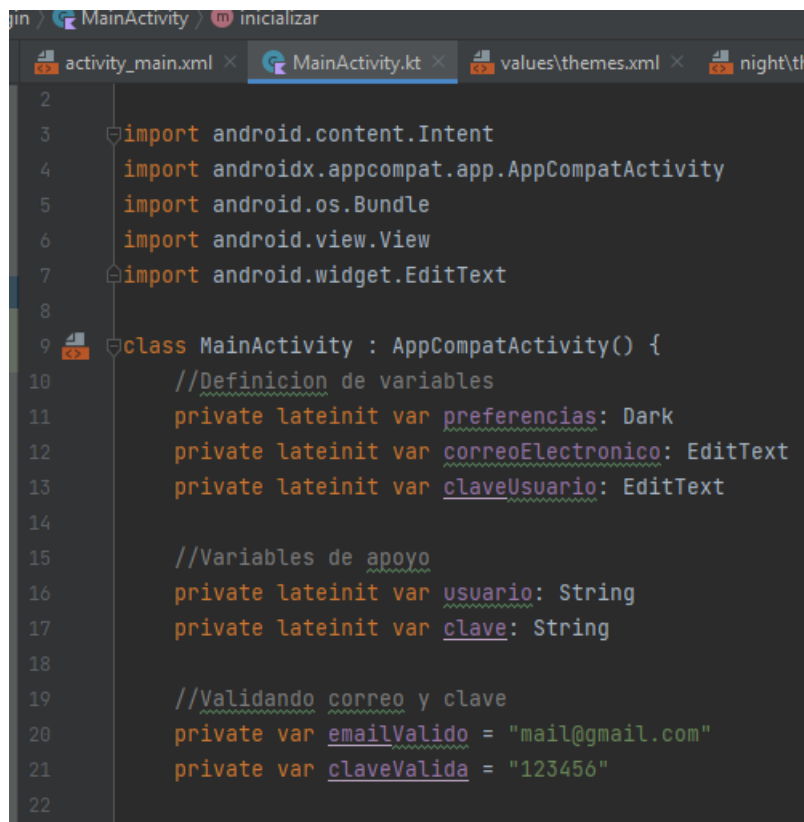
Posteriormente, se pueden visualizar las operaciones al momento de ejecutarse, esto mediante ir agregando números cualesquiera y posteriormente realizar el cálculo mediante el botón correspondiente.



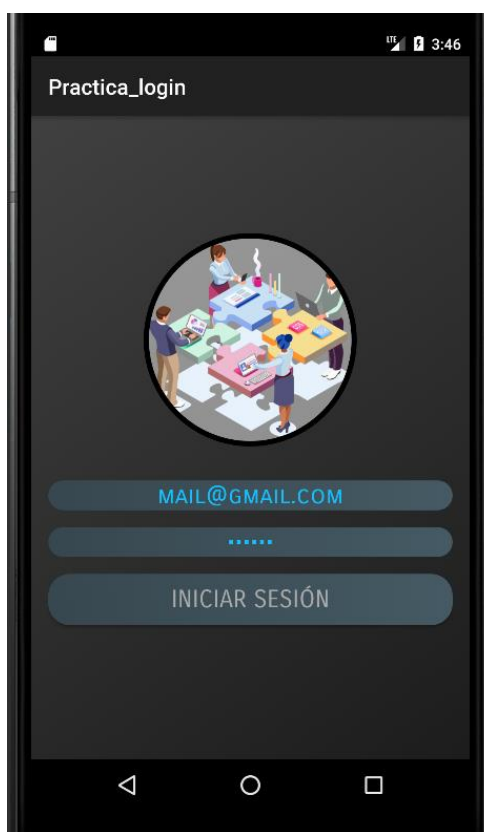
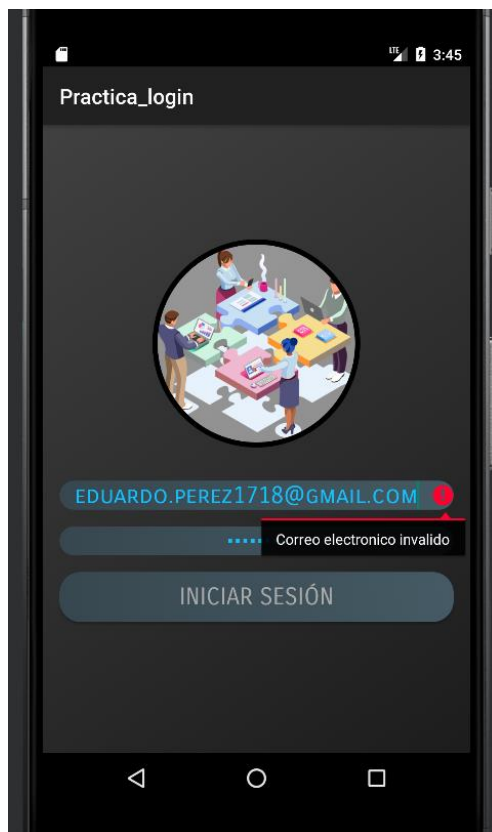
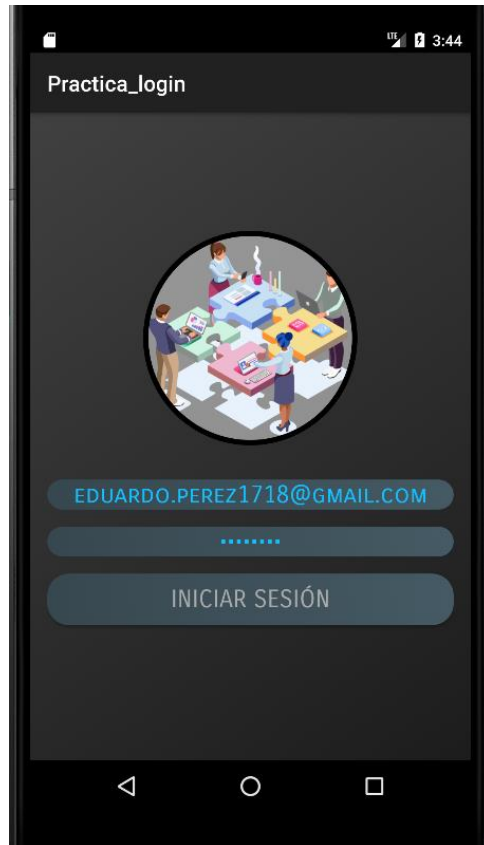
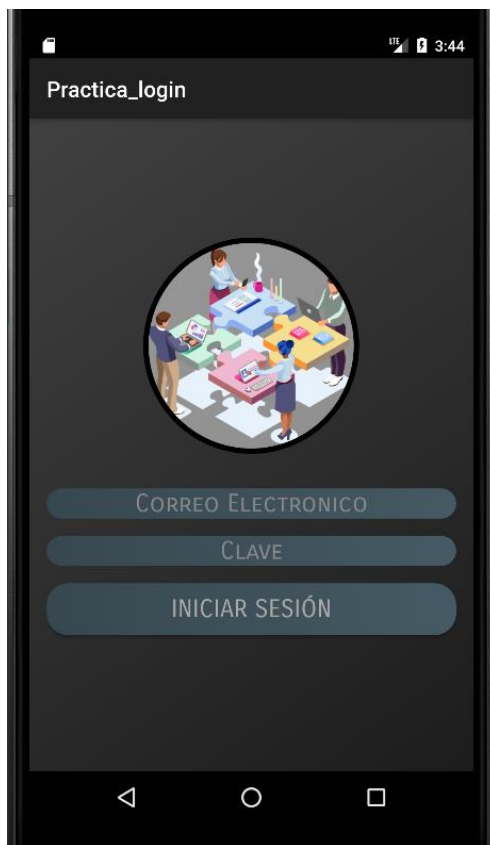


## 2. Login

En la siguiente aplicación se muestra inicio de sesión de un usuario. Cabe mencionar que solamente se podrá iniciar sesión con las credenciales que se han registrado previamente (en este caso las que se muestran a continuación que forman parte del código de MainActivity.kt). De otra forma no será posible el ingreso.



```
1  2
3  import android.content.Intent
4  import androidx.appcompat.app.AppCompatActivity
5  import android.os.Bundle
6  import android.view.View
7  import android.widget.EditText
8
9  class MainActivity : AppCompatActivity() {
10     //Definición de variables
11     private lateinit var preferencias: Dark
12     private lateinit var correoElectronico: EditText
13     private lateinit var claveUsuario: EditText
14
15     //Variables de apoyo
16     private lateinit var usuario: String
17     private lateinit var clave: String
18
19     //Validando correo y clave
20     private var emailValido = "mail@gmail.com"
21     private var claveValida = "123456"
22 }
```







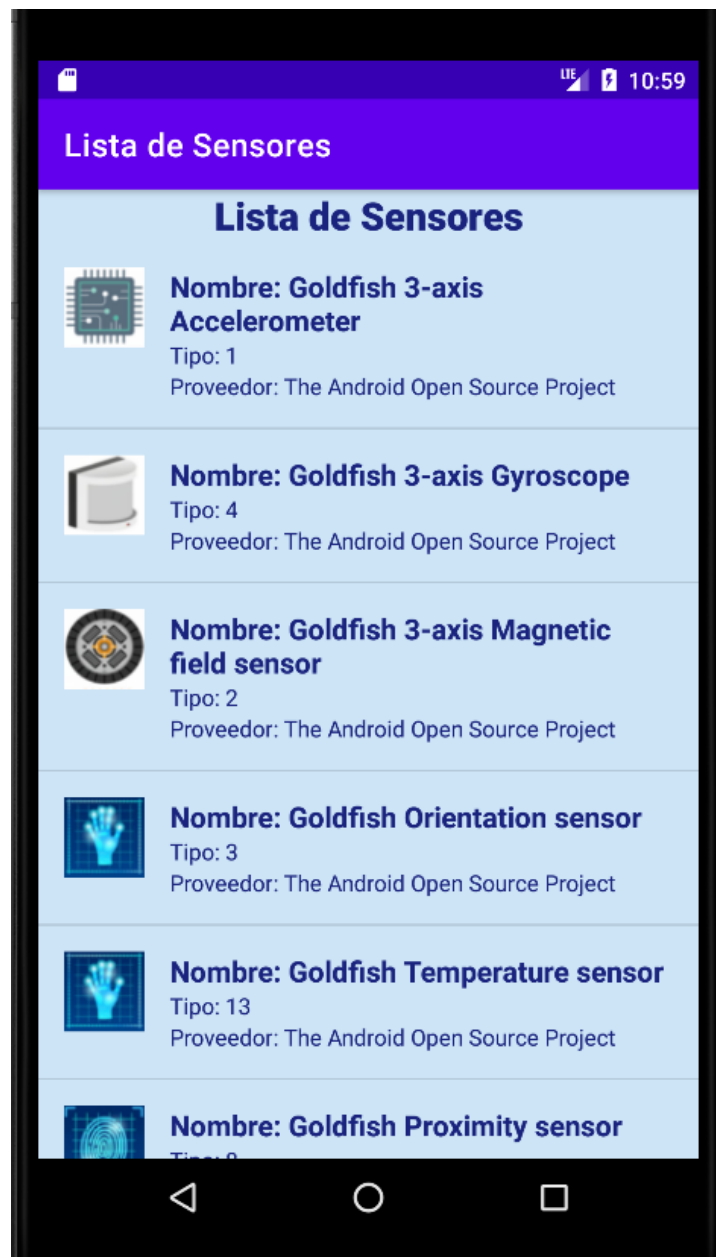
### 3. Lista de frutas

En la siguiente aplicación se muestra el listado de frutas, donde a su vez se puede visualizar su precio y descripción mediante un BaseAdapter, que vincula los datos de las frutas con las vistas correspondientes en un ListView.



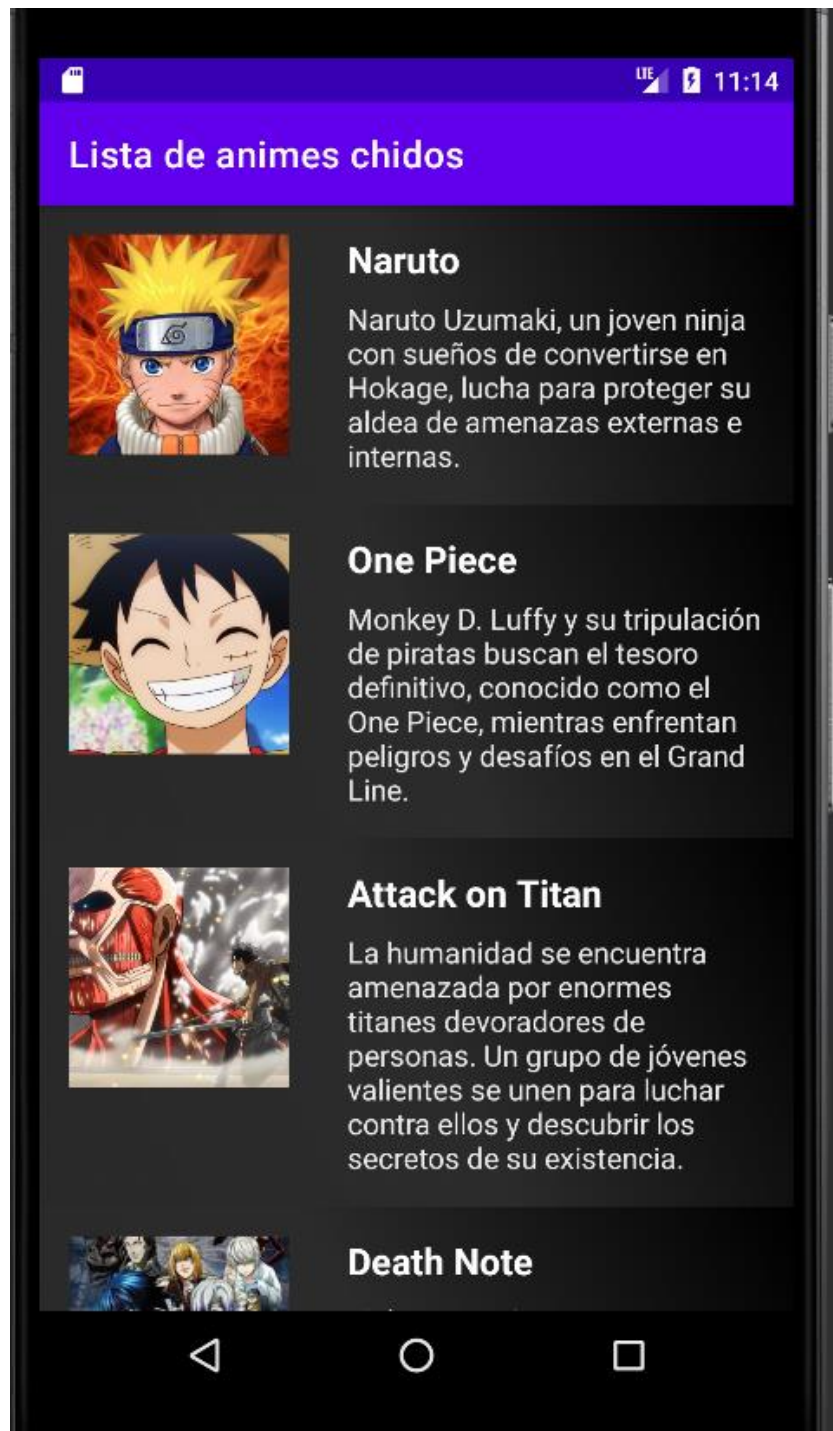
## 4. Lista de Sensores

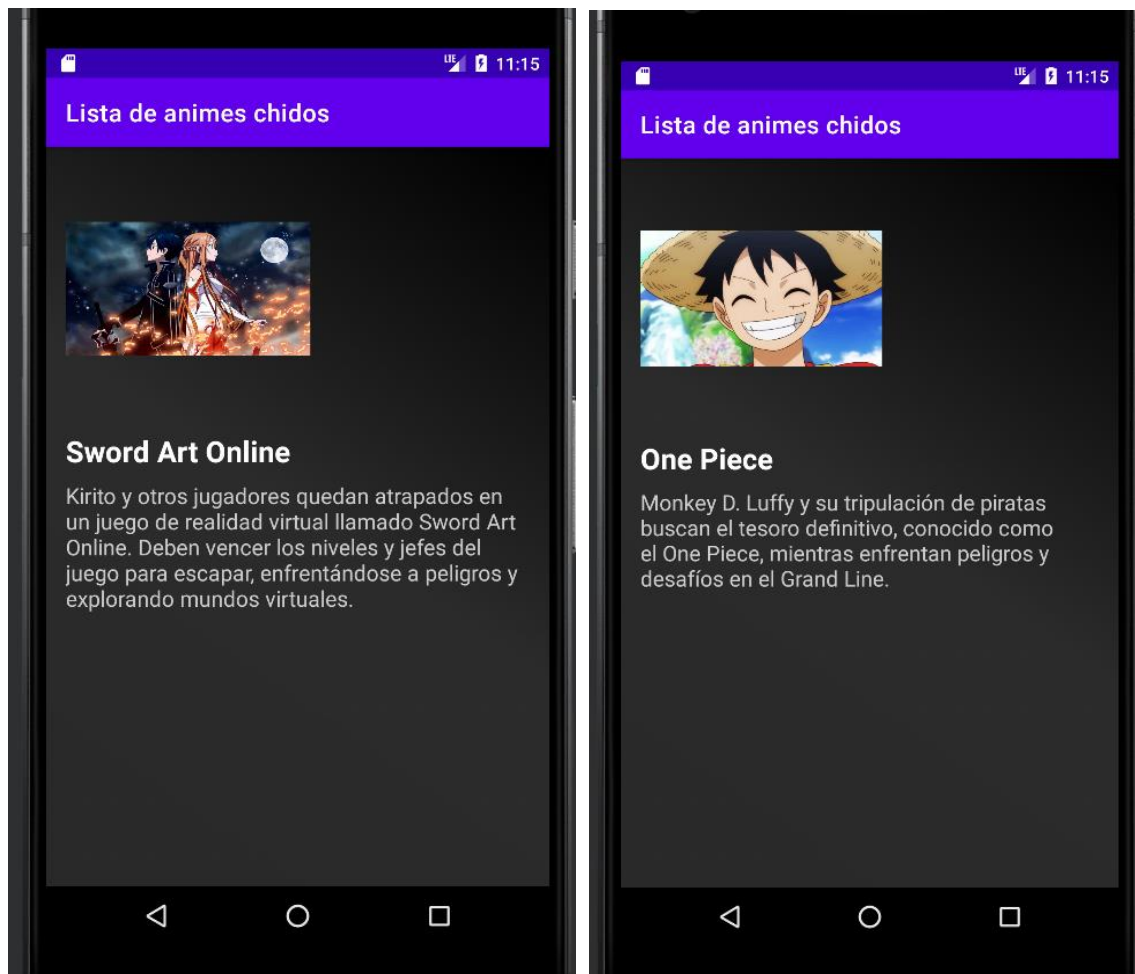
En la siguiente aplicación se muestran los sensores disponibles en el teléfono. Utilizando el `SensorManager`, se obtienen los sensores del dispositivo, como el acelerómetro, giroscopio, sensor de luz, sensor de proximidad, entre otros. Estos sensores brindan información sobre el entorno y el movimiento del dispositivo, lo que permite desarrollar aplicaciones interactivas y basadas en el contexto del usuario. El `BaseAdapter` se utiliza para vincular los datos de los sensores con las vistas correspondientes en un ListView.



## 5. Lista de animés

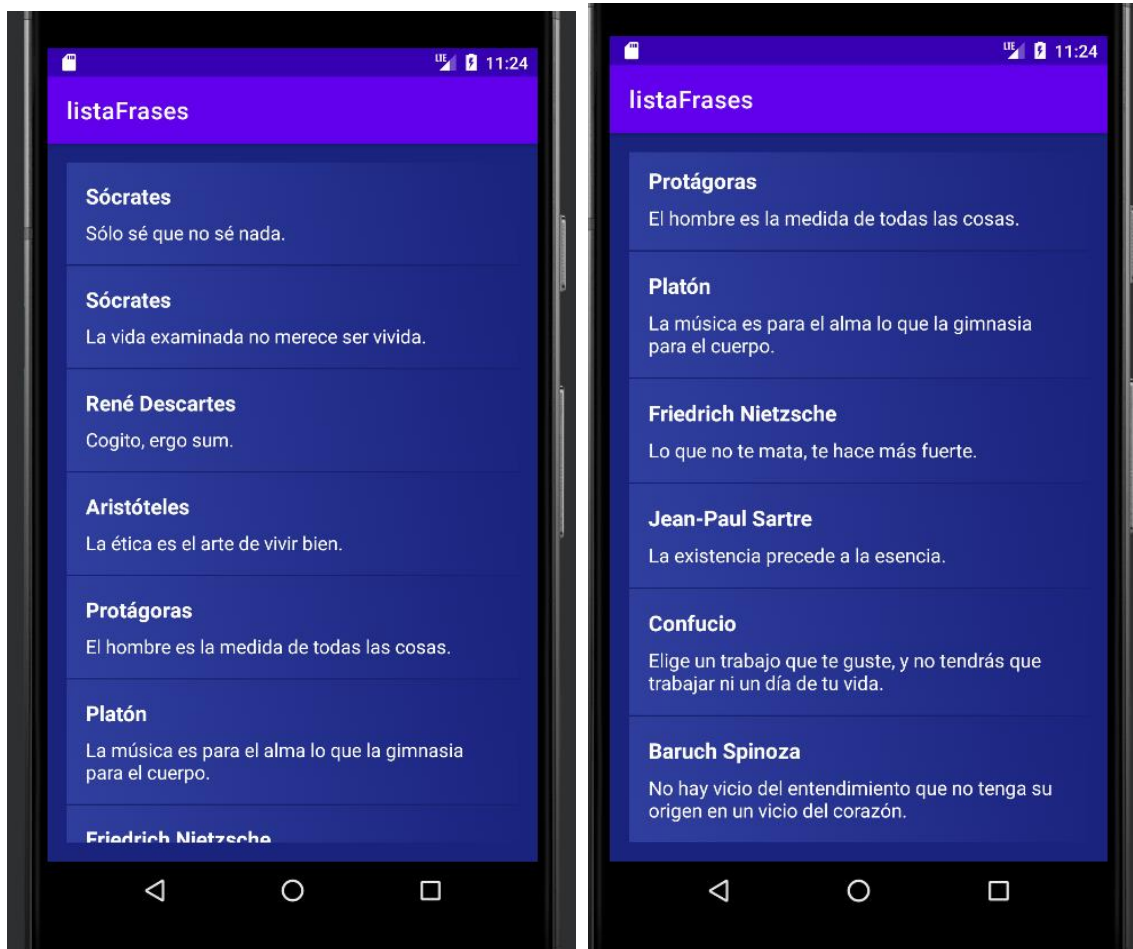
En la siguiente aplicación se muestran un listado de animés. Cada elemento de la lista representa un anime específico, mostrando su título, imagen y una breve descripción. Cuando el usuario hace clic en un elemento de la lista, se le redirige a una nueva vista que muestra información más detallada sobre ese anime en particular.





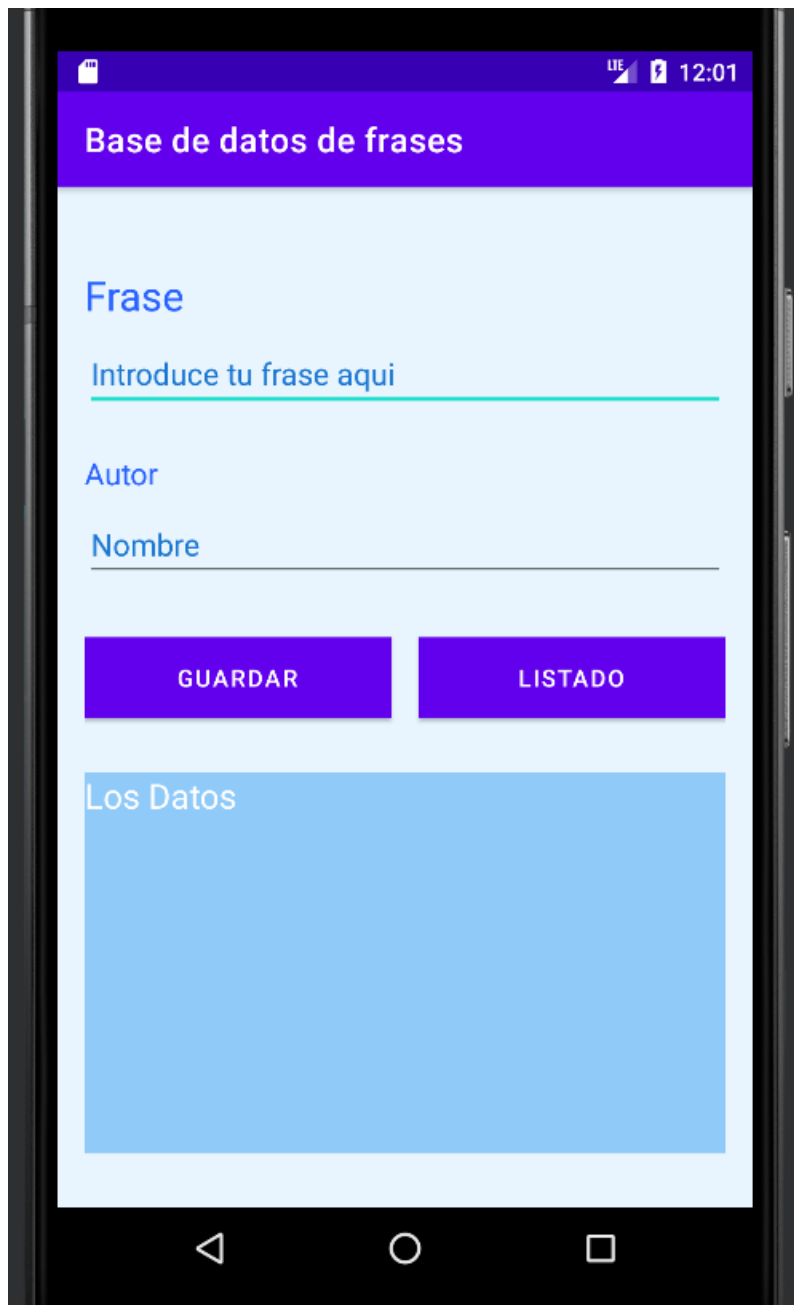
## 6. Lista de frases

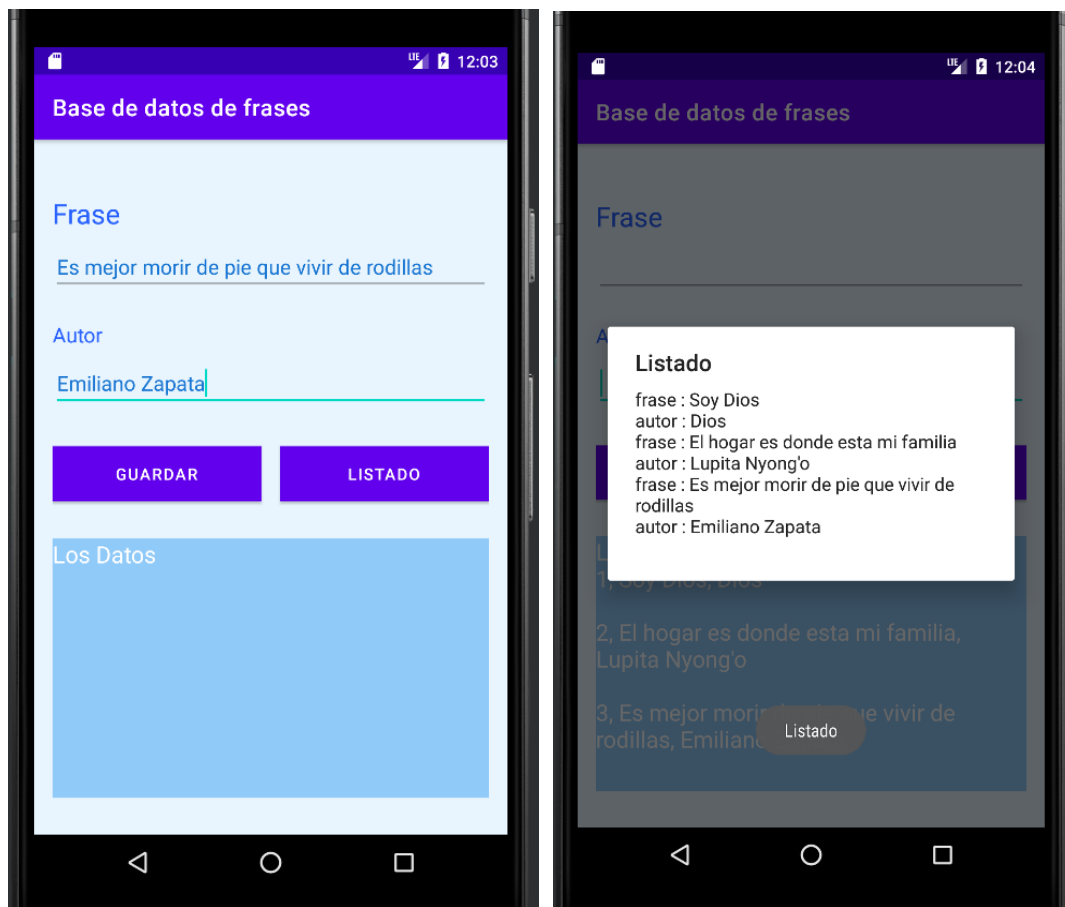
En la siguiente aplicación se muestra una colección de citas inspiradoras, dichos famosos o fragmentos de sabiduría. Cada elemento en la lista representa una frase única, compuesta por un autor y su frase asociada. La información se organiza y muestra en un ListView, donde cada elemento corresponde a una instancia de una clase personalizada, es decir, un objeto que encapsula los datos del autor y su frase. Al utilizar esta estructura de objetos, se facilita la manipulación y gestión de las frases en la aplicación.



## 7. Ejemplo de BD – Frases

En la siguiente aplicación se almacena y gestionan frases junto con el nombre de su autor, utilizando una base de datos SQLite y la clase SQLiteOpenHelper. Además de la funcionalidad de poder agregar y guardar, se puede visualizar a su vez las frases almacenadas mediante el botón de Listado, el cual desplegará una notificación con todos los registros almacenados y a su vez en el ListView de la parte inferior se podrán visualizar.









## 8. Retrofit – Pokedex

En la siguiente aplicación se creó una Pokedex, la cual utiliza la biblioteca Retrofit para consumir una API de Pokémon. Con esta aplicación, los usuarios pueden explorar y obtener información detallada sobre diferentes Pokémon. Al conectarse a la API de Pokémon, la aplicación recopila datos como el nombre, imagen, tipo, habilidades y una descripción de cada Pokémon.

