

PERSISTENCIA – MODELO VISTA PRESENTADOR

0. ANALISIS DE LA CLASE MASCOTA CON PERSISTENCIA

```
1. package com.gt.silviarossana.mascotasfavoritas.pojo;
2.
3. import java.io.Serializable;
4.
5. /**
6.  * Created by Silvia Rossana on 22/01/2017.
7.  */
8.
9. public class Mascota implements Serializable {
10.
11.     private int id;
12.     private String nombre;
13.     private int foto;
14.     private int numLike;
15.
16.     public Mascota(String nombre, int foto, int numLike) {
17.         this.nombre = nombre;
18.         this.foto = foto;
19.         this.numLike = numLike;
20.     }
21.
22.     public Mascota() {
23.
24.     }
25.
26.     public String getNombre() {
```

```
27.     return nombre;
28. }
29.
30. public void setNombre(String nombre) {
31.     this.nombre = nombre;
32. }
33.
34. public int getFoto() {
35.     return foto;
36. }
37.
38. public void setFoto(int foto) {
39.     this.foto = foto;
40. }
41.
42. public int getNumLike() {
43.     return numLike;
44. }
45.
46. public void setNumLike(int numLike) {
47.     this.numLike = numLike;
48. }
49.
50. public int getId() {
51.     return id;
52. }
53.
54. public void setId(int id) {
55.     this.id = id;
56. }
57. }
58.
```

0. ANALISIS DE LA CLASE BASEDATOS – AQUÍ SOLO EL MÉTODO DE CREACIÓN DE TABLAS

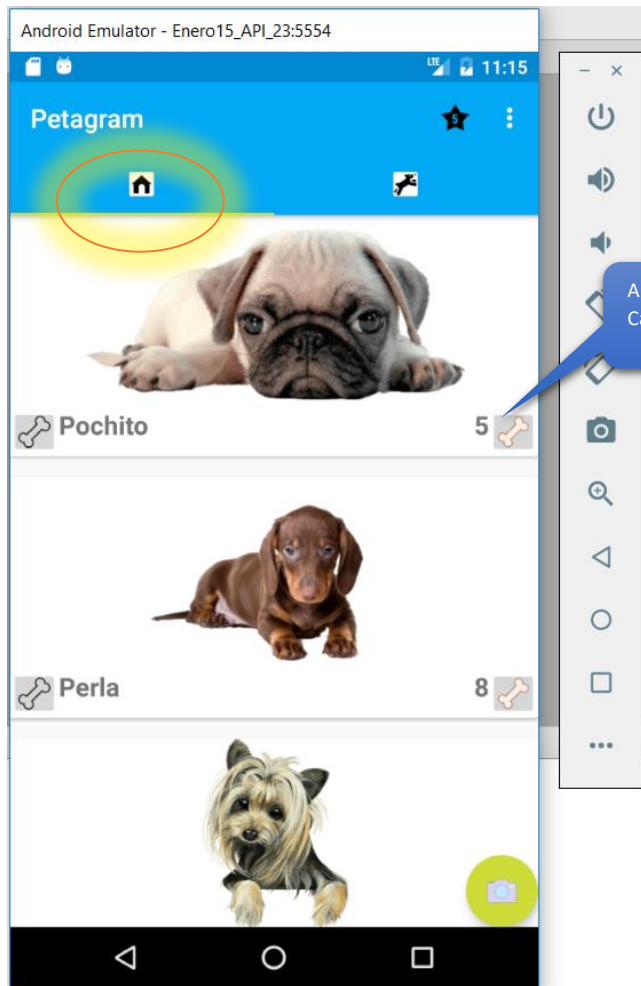
```
1. package com.gt.silviarossana.mascotasfavoritas.db;
2.
3. import android.content.ContentValues;
4. import android.content.Context;
5. import android.database.Cursor;
6. import android.database.sqlite.SQLiteDatabase;
7. import android.database.sqlite.SQLiteOpenHelper;
8. import android.widget.Toast;
9.
10. import com.gt.silviarossana.mascotasfavoritas.pojo.Mascota;
11.
12. import java.util.ArrayList;
13.
14. /**
15.  * Created by Silvia Rossana on 5/02/2017.
16.  */
17.
18. public class BaseDatos extends SQLiteOpenHelper {
19.
20.     private Context context;
21.
22.     public BaseDatos(Context context) {
23.         //super(context, name, factory, version);
24.         super(context, ConstantesBaseDatos.DATABASE_NAME, null, ConstantesBaseDatos.DATABASE_VERSION);
25.         this.context = context;
26.
27.     }
28.
29.     /*public BaseDatos(Context context, String name, SQLiteDatabase.CursorFactory factory, int version, DatabaseErrorHandler errorHandler) {
30.         super(context, name, factory, version, errorHandler);
31.     }*/
32.
33.     @Override
34.     public void onCreate(SQLiteDatabase sqLiteDatabase) {
```

```
35.
36. String queryCrearTablaContacto = "CREATE TABLE " + ConstantesBaseDatos.TABLE_MASCOTA
37.     + "(" + ConstantesBaseDatos.TABLE_MASCOTA_ID + " INTEGER PRIMARY KEY AUTOINCREMENT," +
38.     ConstantesBaseDatos.TABLE_MASCOTA_NOMBRE + " TEXT," +
39.     ConstantesBaseDatos.TABLE_MASCOTA_FOTO + " INTEGER" +
40.     ")";
41.
42. String queryCrearTablaLikesContacto = "CREATE TABLE " + ConstantesBaseDatos.TABLE_LIKE_MASCOTA
43.     + "(" + ConstantesBaseDatos.TABLE_LIKE_MASCOTA_ID + " INTEGER PRIMARY KEY AUTOINCREMENT," +
44.     ConstantesBaseDatos.TABLE_LIKE_MASCOTA_ID_MASCOTA + " INTEGER," +
45.     ConstantesBaseDatos.TABLE_LIKE_MASCOTA_NUMERO_LIKES + " INTEGER," +
46.     " FOREIGN KEY (" + ConstantesBaseDatos.TABLE_LIKE_MASCOTA_ID + ") " +
47.     " REFERENCES " + ConstantesBaseDatos.TABLE_MASCOTA + "(" + ConstantesBaseDatos.TABLE_MASCOTA_ID + ")" +
48.     ")";
49.
50. sqliteDatabase.execSQL(queryCrearTablaContacto);
51. sqliteDatabase.execSQL(queryCrearTablaLikesContacto);
52. }
53.
```

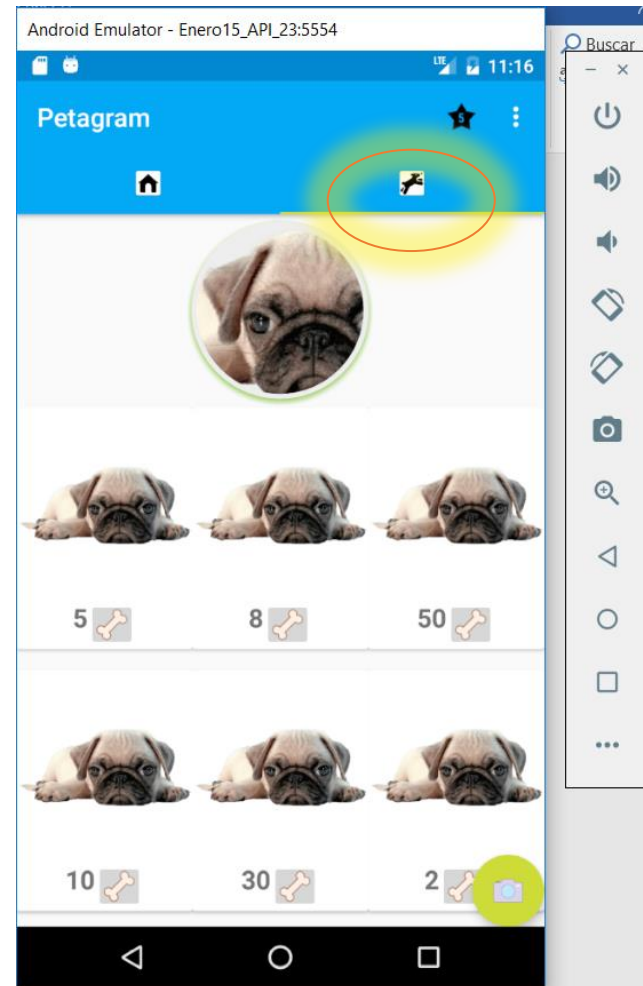
...

IMÁGENES APLICACIÓN CON PERSISTENCIA

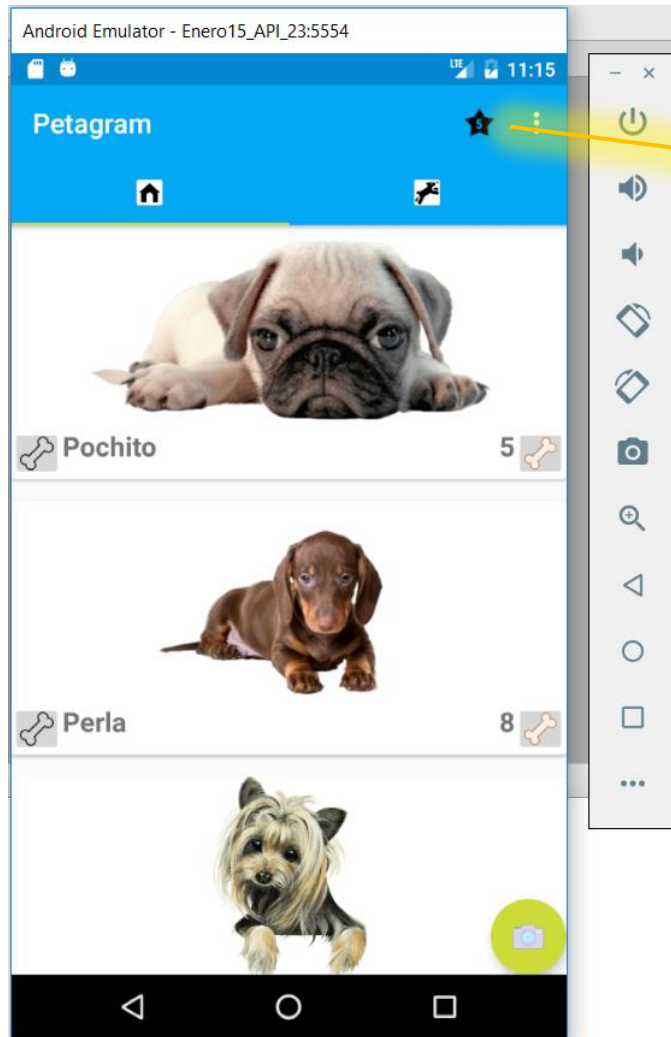
PÁGINA PRINCIPAL FRAGMENT HOME



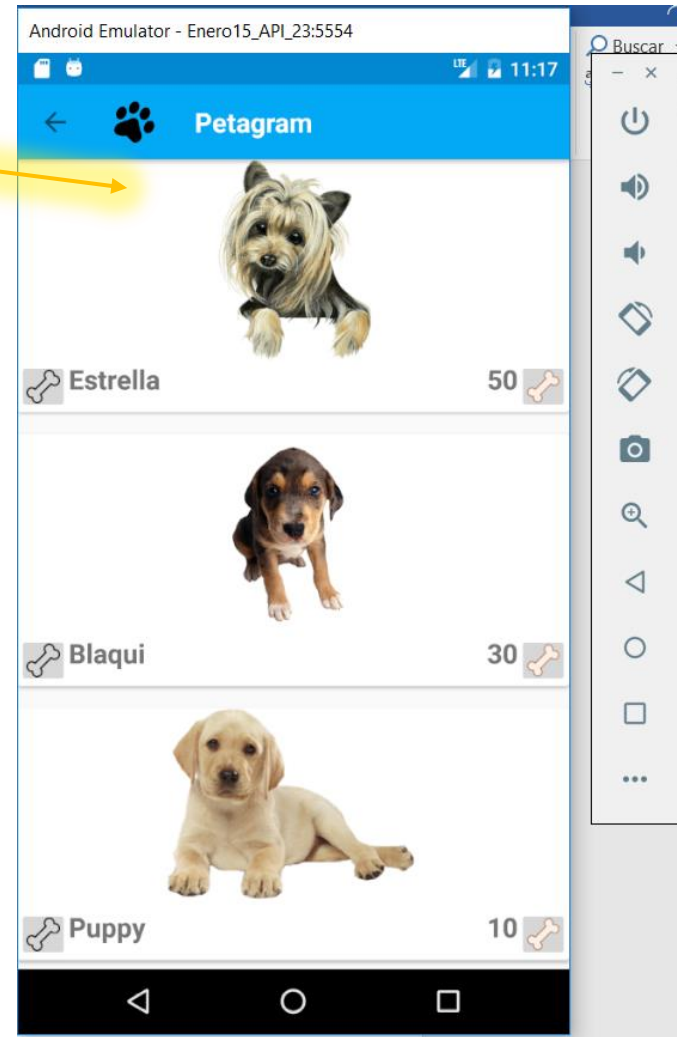
PÁGINA PRINCIPAL – FRAGMENT PERFIL



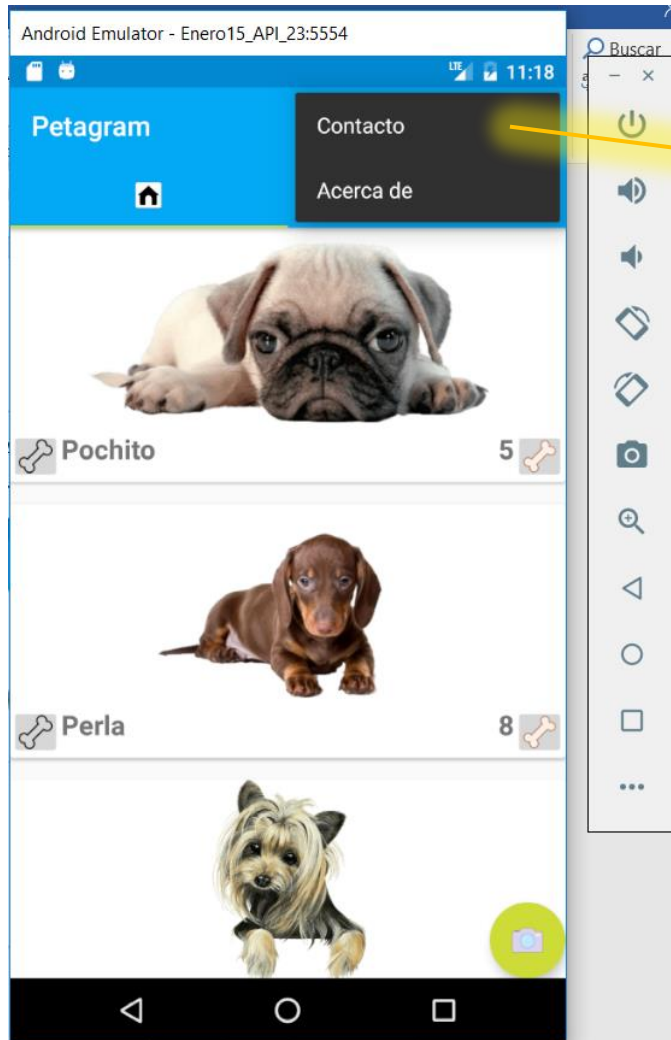
PÁGINA PRINCIPAL (CLICK EN 5 FAVORITOS)



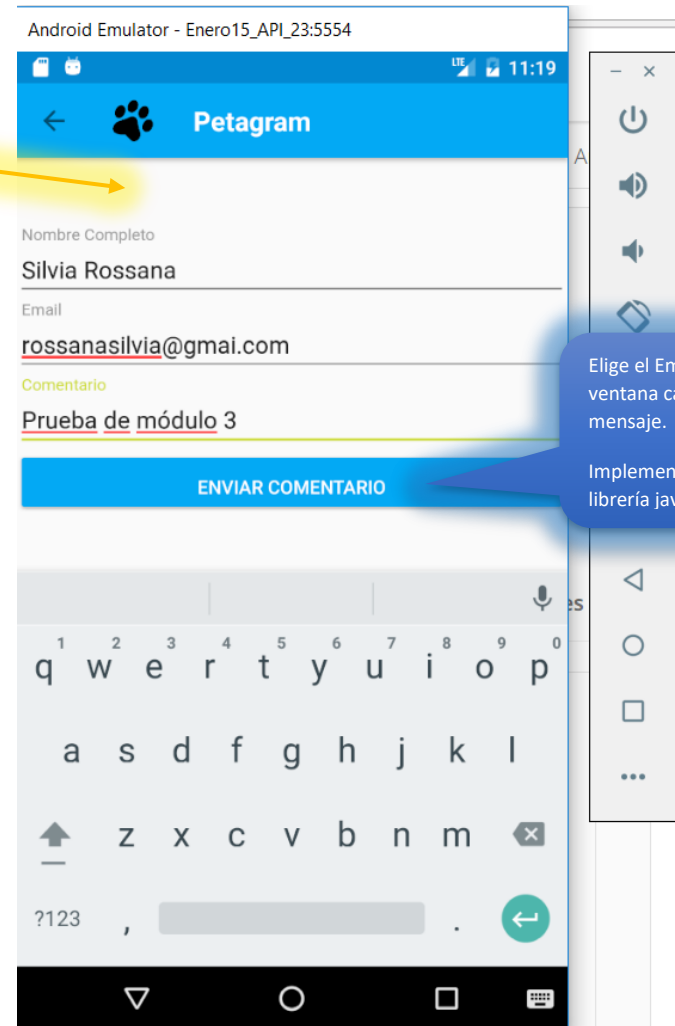
PÁGINA 5 FAVORITOS



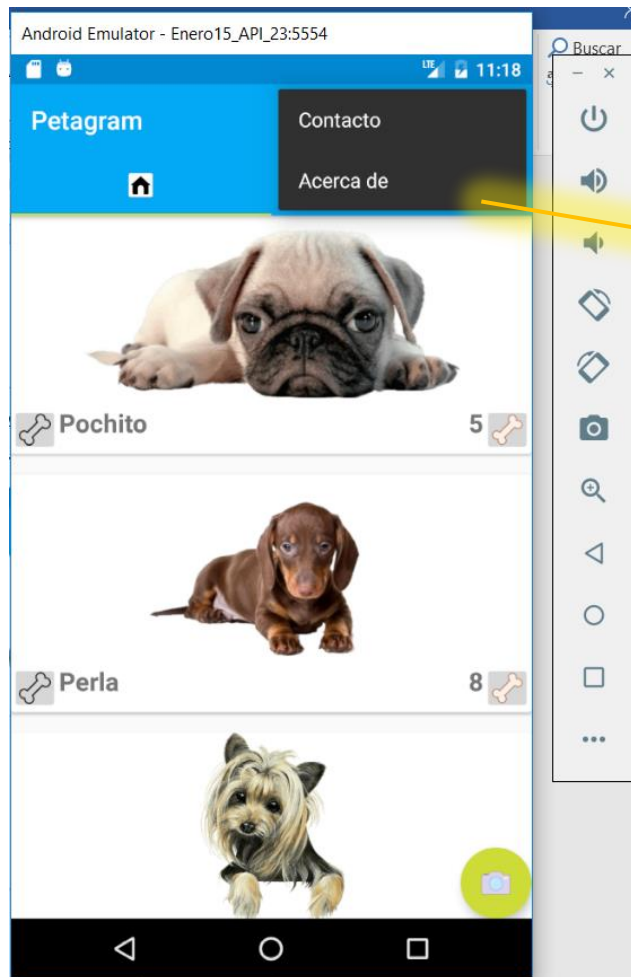
PÁGINA PRINCIPAL (CLIC EN CONTACTO)



CONTACTO PARA ENVIAR MAIL



PÁGINA PRINCIPAL (CLIC EN ACERCA DE)



PÁGINA ACERCA DE

