

ÍNDICE

[MainActivity](#)

[Entrar](#)

[Access](#)

[AccessAdapter](#)

[DetalleAccessAdapter](#)

[ListaAccesosActivity](#)

FUNCIONAMIENTO

AccessAdapter (Adaptador personalizado)

- Es el vínculo entre los datos y la interfaz visual de la lista.
- Extiende de `BaseAdapter`, que se usa con un `ListView`.

Funcionamiento paso a paso: (convierte cada objeto `Access` de la lista en una “fila” visual dentro del `ListView`).

1. El `AccessAdapter` recibe:
 - o El `Context` (actividad que lo usa).
 - o Una `List<Access>` con los accesos obtenidos de la base de datos.
2. Android llama a `getView()` tantas veces como elementos tenga la lista.
3. En cada llamada a `getView()`:
 - o Si hay una vista reutilizable (`convertView`), la usa; si no, la infla desde `item_access.xml`.
 - o Rellena los `TextView` e `ImageView` con los datos del objeto `Access` correspondiente:
 - Nombre de usuario (`username`).
 - Fecha (`createdAt`, formateada).
 - Imagen verde o roja según `valid`.
4. Devuelve la vista completamente configurada para que se muestre en el `ListView`.

ListaAccesosActivity

- Es la pantalla que muestra todos los intentos de acceso registrados.
- Pasos:
 1. Se abre la base de datos (`access-db`) con GreenDAO.
 2. Se crea una `DaoSession` y se obtiene el `AccessDao`.
 3. Se hace una consulta (`queryBuilder()`) para obtener todos los registros ordenados por fecha descendente.
 4. Se guarda el resultado en una lista `List<Access> lista`.

Se crea un `AccessAdapter` con esa lista y se asigna al `ListView`:

```
AccessAdapter adapter = new AccessAdapter(this, lista);
listView.setAdapter(adapter);
```

5. Se define un evento `setOnItemClickListener`:
 - Cuando el usuario toca una fila, se obtiene el `Access` de esa posición.
 - Se crea un `Intent` y se envía el `id` del acceso a `DetalleAccesoActivity`.
 - Se abre esa nueva actividad con los datos del acceso seleccionado.

Relación importante:

Base de datos → lista de objetos Access → AccessAdapter → ListView (pantalla)

DetalleAccesoActivity

- Recibe el `access_id` de la actividad anterior.
- Abre la base de datos y carga el registro concreto con: `Access acceso = accessDao.load(id);`

MainActivity

```
package es.unavarra.tlm.dscr_25_06;

import android.content.Intent; //Para navegar entre pantallas (Activitys)
import android.os.Bundle;
import android.widget.Button;
import android.widget.ImageView;

import androidx.appcompat.app.AppCompatActivity;

import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;

import org.greenrobot.greendao.database.Database;
import es.unavarra.tlm.dscr_25_06.DaoMaster;
import es.unavarra.tlm.dscr_25_06.DaoSession;
import es.unavarra.tlm.dscr_25_06.AccessDao;
import es.unavarra.tlm.dscr_25_06.AccessDao.Properties;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState); //esto es obligatorio en el código
        //Lo primero que hago es vincular esta activity con su Layout
        setContentView(R.layout.activity_main);
        ImageView imgStatus = findViewById(R.id.img_status);

        //Ahora hago referencias a los botones del layout
        Button btnEntrar = findViewById(R.id.btn_Eintrar);
        Button btnRegistro = findViewById(R.id.btn_Registro);
        Button btnAccesos = findViewById(R.id.btn_Accesos);

        //Ahora hago los listeners de los botones
        //Al pulsar ENTRAR abrimos la activity entrar
        btnEntrar.setOnClickListener(v -> { // v seria el objeto boton pulsado
            Intent i = new Intent(MainActivity.this, entrar.class);
            startActivity(i);
        });

        //Al pulsar REGISTRO abrimos la activity registro
        btnRegistro.setOnClickListener(v -> {
            Intent i = new Intent(MainActivity.this, registro.class);
            startActivity(i);
        });

        btnAccesos.setOnClickListener(v -> {
```

```

        Intent i = new Intent(MainActivity.this,
ListaAccesosActivity.class);
        startActivity(i);
    });
}

@Override
protected void onResume() { // Cada vez que volvamos a la pantalla principal
del semáforo, comprobamos la base de datos
    super.onResume();
    ImageView imgStatus = findViewById(R.id.img_status);

    //Abrimos (o creamos si no existe) la base de datos local
    DaoMaster.DevOpenHelper helper = new DaoMaster.DevOpenHelper(this,
"access-db");
    Database db = helper.getWritableDatabase();

    //Creamos una sesión (DaoSession) para trabajar con los datos
    DaoSession session = new DaoMaster(db).newSession();

    //Obtenemos el DAO de la tabla Access (donde guardas los intentos)
    AccessDao accessDao = session.getAccessDao();

    //Contamos cuántos intentos fueron correctos (valid = true)
    long aciertos = accessDao.queryBuilder()
        .where(Properties.Valid.eq(true))
        .count();

    // Contamos cuántos fueron incorrectos (valid = false)
    long fallos = accessDao.queryBuilder()
        .where(Properties.Valid.eq(false))
        .count();

    // Lógica para decidir qué color mostrar
    if (aciertos == 0 && fallos == 0) {
        imgStatus.setImageResource(R.drawable.white_icon); // sin intentos
todavía
    } else if (aciertos > fallos) {
        imgStatus.setImageResource(R.drawable.green_icon); // más aciertos
    } else if (fallos > aciertos) {
        imgStatus.setImageResource(R.drawable.red_icon); // más fallos
    } else {
        imgStatus.setImageResource(R.drawable.white_icon); // empate
    }

    // Cerramos la base de datos
    db.close();
}
}

```

[Entrar](#)

```
package es.unavarra.tlm.dscr_25_06;

import android.content.SharedPreferences; // Almacenamiento simple clave-valor
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View; // Para setVisibility(...)
import android.widget.Button;
import android.widget.EditText;
import android.widget.LinearLayout;
import android.widget.TextView;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;

import org.greenrobot.greendao.database.Database;

import java.util.Date;

public class entrar extends AppCompatActivity {

    private LinearLayout usuarioNoIdentificado, usuarioIdentificado;
    private TextView txtBienvenida;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        //vinculamos el layout con activity_entrar.xml
        setContentView(R.layout.activity_entrar);

        //Hago referencia a las vistas del Layout
        usuarioIdentificado = findViewById(R.id.usuarioIdentificado);
        usuarioNoIdentificado = findViewById(R.id.usuarioNoIdentificado);
        txtBienvenida = findViewById(R.id.Hola_usuario);

        //Configuro SharedPreferences
        SharedPreferences settings = getSharedPreferences("Config", 0);
        String usuario_id = settings.getString("usuario_id", "");
        //Toast.makeText(this, usuario_id, Toast.LENGTH_SHORT).show();
        boolean estaLogueado = !usuario_id.isEmpty(); //Para saber si está
logead
        if (estaLogueado) {

            usuarioNoIdentificado.setVisibility(View.GONE); //Hacemos que no se
muestre el Layout de usuario no identificado
            usuarioIdentificado.setVisibility(View.VISIBLE); //Hacemos que se
muestre el Layout de usuario identificado

            //mensaje de bienvenida -> cambiar Txt_bienvenida_id para q donde
ponga [usuario] coja lo de usuario_id
            String mensaje =
getString(R.string.hola_usuario).replace("[usuario]", usuario_id);
            txtBienvenida.setText(mensaje);
            //gestion del boton SALIR
            Button btnSalir = findViewById(R.id.Btn_salir);
```

```
        btnSalir.setOnClickListener(v -> cerrarSesion(usuario_id));
//Llamamos a la funcion que cierra la sesion del usuario
    }else{//Si el usuario no está identificado

        usuarioNoIdentificado.setVisibility(View.VISIBLE);
        usuarioIdentificado.setVisibility(View.GONE);
        loguin(); //Llamo a la funcion que gestiona el loguin

    }

}

private void loguin(){
    //Hago referencia a la vista del boton
    Button btnLogin = findViewById(R.id.Btn_login);
    btnLogin.setOnClickListener(v -> {
        //hago referencia a la vista de los editablenet
        EditText etUsuario = findViewById(R.id.Hint_usuarioLogin);
        EditText etContraseña = findViewById(R.id.Hint_contraseñaLogin);

        // Obtenemos el texto de los campos y quitamos espacios al
        inicio/fin (trim)
        String usuario = etUsuario.getText().toString().trim();
        String contraseña = etContraseña.getText().toString().trim();

        //Empezamos con la DB
        Boolean esValida = false;
        //Abrimos o creamos la base de datos local "access-db"
        DaoMaster.DevOpenHelper helper = new
        DaoMaster.DevOpenHelper(entrar.this, "access-db");
        Database db = helper.getWritableDatabase();

        //Creamos una sesión de trabajo
        DaoSession session = new DaoMaster(db).newSession();

        //Obtenemos el objeto AccessDao (DAO = Data Access Object)
        AccessDao accessDao = session.getAccessDao();

        //Validacion de que no haya dejado algun campo vacio
        if(usuario.isEmpty() && contraseña.isEmpty()){

Toast.makeText(this,getString(R.string.err_usuarioycontraseña),Toast.LENGTH_SHORT).show();

        }else if(usuario.isEmpty()){

Toast.makeText(this,getString(R.string.err_usuario),Toast.LENGTH_SHORT).show();

        }else if(contraseña.isEmpty()){

Toast.makeText(this,getString(R.string.err_contraseña),Toast.LENGTH_SHORT).show();
    }
}
```

```
        }else if(!contraseña.equals("dscr")){//compruebo que la contraseña
sea dscr

        esValida = false;
        Toast.makeText(this,getString(R.string.err_contraseñaIncorrecta),
Toast.LENGTH_SHORT).show();

        //Creamos un nuevo registro (objeto Access)
        Access intento = new Access();
        intento.setUsername(usuario);
        intento.setValid(esValida);
        intento.setCreatedAt(new Date());
        //Guardamos el registro en la base de datos
        accessDao.insert(intento);
        //Cerramos la base de datos (buena práctica)
        db.close();

    }

    else {

        esValida = true;
        SharedPreferences settings = getSharedPreferences("Config", 0);
        SharedPreferences.Editor editor = settings.edit();
        editor.putString("usuario_id", usuario);
        editor.apply();
        String texto =
getString(R.string.bienvenido_usuario).replace("[usuario]",
settings.getString("usuario_id",""));
        Toast.makeText(this, texto, Toast.LENGTH_SHORT).show();

        //Creamos un nuevo registro (objeto Access)
        Access intento = new Access();
        intento.setUsername(usuario);
        intento.setValid(esValida);
        intento.setCreatedAt(new Date());
        //Guardamos el registro en la base de datos
        accessDao.insert(intento);
        //Cerramos la base de datos (buena práctica)
        db.close();

        finish();
    }
    db.close();
}) ;

}

private void cerrarSesion(String usuario_id) {
```

```

        //Borro todos los valores guardados
        SharedPreferences settings = getSharedPreferences("Config", 0);
        SharedPreferences.Editor editor = settings.edit();
        editor.remove("usuario");
        editor.remove("usuario_id");
        editor.apply();
        //Volver al main_activity
        finish();
    }
}

```

Registro

```

package es.unavarra.tlm.dscr_25_06;

import android.content.SharedPreferences;
import android.os.Bundle;                                     // Estado de la Activity
import android.text.TextUtils;                                // Utilidad para comprobar strings vacíos
import android.widget.Button;                                // Botón
import android.widget.EditText;                             // Campo de texto
import android.widget.Toast;                               // Mensajes breves en pantalla
import androidx.appcompat.app.AppCompatActivity;

public class registro extends AppCompatActivity {

    //Referencias a los campos de texto
    private EditText etUsuario, etContraseña;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        //Vinculamos el layout con el activity_registro.xml
        setContentView(R.layout.activity_registro);

        //Ahora hago referencia a las vistas (boton y textos) del Layout
        etUsuario = findViewById(R.id.Hint_usuario);
        etContraseña = findViewById(R.id.Hint_contraseña);
        Button btnCrearCuenta = findViewById(R.id.Btn_crearCuenta);

        //la acción que ocurre al pulsar el botón, crearemos una función más abajo
        //para ver que es lo que hace
        btnCrearCuenta.setOnClickListener(v -> onCreateAccount());
    }

    private void onCreateAccount(){//Lo que va a pasar cuando pulsemos el botón
        // Obtenemos el texto de los campos y quitamos espacios al inicio/fín
        (trim)
        String usuario = etUsuario.getText().toString().trim();
        String contraseña = etContraseña.getText().toString().trim();

        //Validación de que no haya dejado algún campo vacío
        if(usuario.isEmpty() && contraseña.isEmpty()){

```

```

Toast.makeText(this,getString(R.string.err_usuarioycontraseña),Toast.LENGTH_SHORT).show();
    }else if(usuario.isEmpty()){

Toast.makeText(this,getString(R.string.err_usuario),Toast.LENGTH_SHORT).show();
    }else if(contraseña.isEmpty()){

Toast.makeText(this,getString(R.string.err_contraseña),Toast.LENGTH_SHORT).show();
};

    }else {
        //Para reemplazar el nombre que haya puesto el usuario
        String texto = getString(R.string.cuenta_creada).replace("[usuario]", usuario);
        Toast.makeText(this, texto, Toast.LENGTH_SHORT).show();
        finish(); //para salir de aqui y volver al login
    }
}
}

```

Access

```

package es.unavarra.tlm.dscre_25_06;

import org.greenrobot.greendao.annotation.Entity;
import org.greenrobot.greendao.annotation.Id;
import org.greenrobot.greendao.annotation.NotNull;
import org.greenrobot.greendao.annotation.Property;
import org.greenrobot.greendao.annotation.Generated;
import java.util.Date; //importante ponerlo el ultimo, si no da error

@Entity(nameInDb = "access")//Nombre de la tabla en la DB
public class Access {
    @Property(nameInDb = "id")//Nombre de la columna (buenas practicas poner nombres)
    @Id(autoincrement = true)//Id autoincremental
    private Long id;

    @Property(nameInDb = "username")
    @NotNull //Para que no pueda ser nullo
    private String username;

    @Property(nameInDb = "valid") //para ver si el intento fue valido o no
    (true/false)
    @NotNull
    private Boolean valid;

    @Property(nameInDb = "created_at")
    @NotNull
    private Date createdAt;

    @Generated(hash = 1235667477)
    public Access(Long id, @NotNull String username, @NotNull Boolean valid,
                 @NotNull Date createdAt) {

```

```

        this.id = id;
        this.username = username;
        this.valid = valid;
        this.createdAt = createdAt;
    }

    @Generated(hash = 1253708747)
    public Access() {
    }

    public Long getId() {
        return this.id;
    }

    public void setId(Long id) {
        this.id = id;
    }

    public String getUsername() {
        return this.username;
    }

    public void setUsername(String username) {
        this.username = username;
    }

    public Boolean getValid() {
        return this.valid;
    }

    public void setValid(Boolean valid) {
        this.valid = valid;
    }

    public Date getCreatedAt() {
        return this.createdAt;
    }

    public void setCreatedAt(Date createdAt) {
        this.createdAt = createdAt;
    }
}

```

AccessAdapter

```

package es.unavarra.tlm.dscre_25_06;

import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.BaseAdapter;
import android.widget.ImageView;
import android.widget.TextView;

```

```
import java.text.SimpleDateFormat;
import java.util.List;
import java.util.Locale;

import es.unavarra.tlm.dscr_25_06.R;
import es.unavarra.tlm.dscr_25_06.Access;

public class AccessAdapter extends BaseAdapter {

    private Context context;
    private List<Access> accesos;
    private LayoutInflater inflater;
    private SimpleDateFormat formatter = new SimpleDateFormat("yyyy-MM-dd
HH:mm:ss", Locale.getDefault());

    public AccessAdapter(Context context, List<Access> accesos) {
        this.context = context;
        this.accesos = accesos;
        this.inflater = LayoutInflater.from(context);
    }

    @Override
    public int getCount() {
        return accesos.size();
    }

    @Override
    public Object getItem(int position) {
        return accesos.get(position);
    }

    @Override
    public long getItemId(int position) {
        return accesos.get(position).getId();
    }

    @Override
    public View getView(int position, View convertView, ViewGroup parent) {
        // Reutiliza la vista si es posible
        if (convertView == null) {
            convertView = inflater.inflate(R.layout.item_access, parent, false);
        }

        // Referencias a los elementos del layout
        TextView txtUser = convertView.findViewById(R.id.txtUser);
        TextView txtDate = convertView.findViewById(R.id.txtDate);
        ImageView imgValid = convertView.findViewById(R.id.imgValid);

        // Obtenemos el objeto Access actual
        Access acceso = accesos.get(position);

        // Rellenamos los datos
        txtUser.setText(acceso.getUsername());
    }
}
```

```

        txtDate.setText(formatter.format(acceso.getCreatedAt()));

        // Cambiamos la imagen según si fue válido o no
        if (acceso.getValid()) {
            imgValid.setImageResource(R.drawable.green_icon);
        } else {
            imgValid.setImageResource(R.drawable.red_icon);
        }

        return convertView;
    }
}

```

DetalleAccessAdapter

```

package es.unavarra.tlm.dscr_25_06;

import android.os.Bundle;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;

import org.greenrobot.greendao.database.Database;
import java.text.SimpleDateFormat;
import java.util.Locale;

import es.unavarra.tlm.dscr_25_06.DaoMaster;
import es.unavarra.tlm.dscr_25_06.DaoSession;
import es.unavarra.tlm.dscr_25_06.Access;
import es.unavarra.tlm.dscr_25_06.AccessDao;

public class DetalleAccesoActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_detalle_acceso);

        //Referencias a los elementos del layout
        TextView txtUser = findViewById(R.id.txtUserDetail); //Asigno a variables
        los elementos de texto definidos en el XML del layout,
        TextView txtDate = findViewById(R.id.txtDateDetail); // para poder
        modificarlos desde el código.
        TextView txtValid = findViewById(R.id.txtValidDetail);

        // Recupero el ID pasado desde la lista, si no existe será -1
        long id = getIntent().getLongExtra("access_id", -1);

        // Accedo a la base de datos local
        DaoMaster.DevOpenHelper helper = new DaoMaster.DevOpenHelper(this,
        "access-db");
        Database db = helper.getReadableDb();
        DaoSession session = new DaoMaster(db).newSession();
        AccessDao accessDao = session.getAccessDao();
    }
}

```

```

// Cargo el registro correspondiente al id pasado desde la lista
Access acceso = accesoDao.load(id);

// Relleno los campos de la interfaz
SimpleDateFormat formatter = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss",
Locale.getDefault());
txtUser.setText("Usuario: " + acceso.getUsername());
txtDate.setText("Fecha: " + formatter.format(acceso.getCreatedAt()));
String valido = acceso.isValid() ? "Sí" : "No";
txtValid.setText("Es válido: " + valido);

// Cierro la base de datos
db.close();
}
}

```

ListaAccesosActivity

```

package es.unavarra.tlm.dscr_25_06;

import android.content.Intent;
import android.os.Bundle;
import android.widget.ListView;
import androidx.appcompat.app.AppCompatActivity;

import org.greenrobot.greendao.database.Database;
import java.util.List;
import es.unavarra.tlm.dscr_25_06.Access;
import es.unavarra.tlm.dscr_25_06.DaoMaster;
import es.unavarra.tlm.dscr_25_06.DaoSession;
import es.unavarra.tlm.dscr_25_06.AccessDao;
import es.unavarra.tlm.dscr_25_06.AccessAdapter;

public class ListaAccesosActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_lista_accesos);

        ListView listView = findViewById(R.id.list_accesos);

        //Abrimos la base de datos
        DaoMaster.DevOpenHelper helper = new DaoMaster.DevOpenHelper(this,
"access-db");
        Database db = helper.getReadableDb();
        DaoSession session = new DaoMaster(db).newSession();
        AccessDao accesoDao = session.getAccessDao();

        //Obtenemos todos los registros ordenados por fecha descendente
        List<Access> lista = accesoDao.queryBuilder()
            .orderDesc(AccessDao.Properties.CreatedAt)
            .list();

        //Creamos el adapter personalizado y lo asignamos al ListView

```

```
AccessAdapter adapter = new AccessAdapter(this, lista);
listView.setAdapter(adapter);

//Evento al pulsar una fila
listView.setOnItemClickListener((parent, view, position, id) -> {
    Access acceso = lista.get(position);
    // Pasamos el id del acceso al nuevo Activity
    Intent intent = new Intent(ListaAccesosActivity.this,
DetalleAccesoActivity.class);
    intent.putExtra("access_id", acceso.getId());
    startActivity(intent);
}) ;

db.close();
}
}
```