



L

Instituto Politécnico Nacional  
Escuela Superior de Cómputo



## Desarrollo de Aplicaciones Móviles Nativas

### “Login”

Alumno:

Malagón Baeza Alan Adrian

Profesor:

M. en C. José Asunción Enríquez Zárate

Grupo: 7CM1

## Introducción

Se muestra un ejemplo de una aplicación de inicio de sesión en Android utilizando el patrón de diseño de arquitectura Modelo-Vista-Controlador (MVC). La aplicación consta de dos actividades principales: MainActivity, donde los usuarios pueden ingresar su correo electrónico y contraseña para iniciar sesión, y BienvenidoActivity, que muestra un mensaje de bienvenida junto con algunas opciones adicionales una vez que el usuario ha iniciado sesión correctamente.

El MainActivity tiene un formulario de inicio de sesión con campos para ingresar el correo electrónico y la contraseña. Se realiza una validación básica para verificar si los campos están vacíos o si los valores ingresados no coinciden con los valores predefinidos de correo electrónico y contraseña válidos. Si la validación es exitosa, se almacena la información del inicio de sesión en las preferencias compartidas utilizando la clase DarkPreferenceManager.

La clase DarkPreferenceManager se utiliza para administrar las preferencias compartidas de la aplicación. Proporciona métodos para guardar y recuperar información, como el estado de inicio de sesión y el nombre de usuario, utilizando SharedPreferences.

Una vez que el usuario ha iniciado sesión correctamente, se muestra la actividad BienvenidoActivity. Aquí se muestra un mensaje de bienvenida que incluye el nombre de usuario del usuario. Además, se proporcionan opciones para cerrar sesión, enviar un correo electrónico o realizar una llamada telefónica.

## Desarrollo

```
class Bienvenido : AppCompatActivity() {
    // Definicion de las variables
    private lateinit var preferencias: DarkPreferenceManager

    // Variables de apoyo
    private lateinit var usuario: String
    private lateinit var lblDatos: TextView

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_bienvenido)
        inicializar()
        verificarLogin()
    }

    private fun inicializar() {
        preferencias = DarkPreferenceManager(this)
        usuario = preferencias.getNombreDeUsuario().toString()
        lblDatos = findViewById(R.id.lblBienvenido)
        setLblDatos()
    }

    private fun verificarLogin() {
        if (preferencias.isLogin() == false) {
            val mainIntent = Intent(this, MainActivity::class.java)
            startActivity(mainIntent)
            finish()
        }
    }

    private fun setLblDatos() {
        lblDatos.text = "Bienvenido : $usuario"
    }

    fun cerrarSesion(view: View) {
        preferencias.limpiarPreferencias()
        val mainIntent = Intent(this, MainActivity::class.java)
        startActivity(mainIntent)
        finish()
    }

    fun correo(view: View) {
        val enviarCorreo = Intent().apply {
            action = Intent.ACTION_SEND
            data = Uri.parse("mailto:mail@mail.com")
        }
        startActivity(Intent.createChooser(enviarCorreo, "Edge"))
    }

    fun telefono(view: View) {
        val llamar = Intent().apply {
            action = Intent.ACTION_DIAL
            data = Uri.parse("tel:00000000")
        }
    }
}
```

```

        startActivity(llamar)
    }
}

```

```

class DarkPreferenceManager(contexto: Context?) {

    // Definir el modo del Shared Preferences
    val MODO_PRIVADO = 0

    // Nombre del Archivo de Preferencias
    private val PREFERENCE_NAME = "DarkSharedPrefences"
    private val IS_LOGIN = "is_login"

    val preferencias: SharedPreferences? =
contexto?.getSharedPreferences(PREFERENCE_NAME, MODO_PRIVADO)
    val editor: SharedPreferences.Editor? = preferencias?.edit()

    fun setLogin(isLogin: Boolean) {
        editor?.putBoolean(IS_LOGIN, isLogin)
        editor?.commit()
    }

    fun isLogin() : Boolean? {
        return preferencias?.getBoolean(IS_LOGIN, false)
    }

    fun setNombreDeUsuario(nombreDeUsuario: String?) {
        editor?.putString("nombreDeUsuario", nombreDeUsuario)
        editor?.commit()
    }

    fun getNombreDeUsuario() : String?{
        return preferencias?.getString("nombreDeUsuario", "")
    }

    fun limpiarPreferencias() {
        editor?.clear()
        editor?.commit()
    }
}

```

```

class MainActivity : AppCompatActivity() {
    // Definicion de las variables
    private lateinit var preferencias: DarkPreferenceManager
    private lateinit var correoElectronico: EditText
    private lateinit var claveUsuario: EditText

    // Variables de apoyo
    private lateinit var usuario: String
    private lateinit var clave: String

```

```

// Validando correo y clave
private var emailValido = "mail@gmail.com"
private var claveValida = "123456"

override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    setContentView(R.layout.activity_main)
    inicializar()
    verificarLogin()
}

private fun verificarLogin() {
    if(preferencias.isLogin()){
        val bienvenidoIntent = Intent(this, Bienvenido::class.java)
        startActivity(bienvenidoIntent)
        finish()
    }
}

private fun inicializar() {
    preferencias = DarkPreferenceManager(this)
    correoElectronico = findViewById(R.id.txtEmail)
    claveUsuario = findViewById(R.id.txtClave)
}

fun iniciarSesion(view: View){
    usuario = correoElectronico.text.toString().trim()
    clave = claveUsuario.text.toString().trim()
    if(usuario.isEmpty() || usuario == ""){
        correoElectronico.error = "Correo electrónico requerido "
        correoElectronico.requestFocus()
    } else if (clave.isEmpty() || clave == ""){
        claveUsuario.error = "Clave requerida "
        claveUsuario.requestFocus()
    } else if (usuario != emailValido){
        correoElectronico.error = "Correo electrónico invalido "
        correoElectronico.requestFocus()
    } else if (clave != claveValida) {
        claveUsuario.error = "Clave invalida "
        claveUsuario.requestFocus()
    } else {
        preferencias.setLogin(true)
        preferencias.setNombreDeUsuario(usuario)

        val bienvenidoIntent = Intent(this, Bienvenido::class.java)
        startActivity(bienvenidoIntent)
        finish()
    }
}
}

```

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"

```

```

xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".Bienvenido">

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_marginStart="16dp"
    android:layout_marginTop="16dp"
    android:layout_marginEnd="16dp"
    android:layout_marginBottom="16dp"
    android:background="@drawable/background"
    android:orientation="vertical"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.0"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="1.0">

    <TextView
        android:id="@+id/lblBienvenido"
        style="@style/estiloTextView"
        android:layout_height="126dp"
        android:layout_marginTop="40dp"
        android:hint="Bienvenido" />

    <Button
        android:id="@+id/btnAceptar"
        style="@style/estiloBoton"
        android:layout_marginTop="16dp"
        android:hint="Cerrar Sesión"
        android:onClick="cerrarSesion"
        />

    <Button
        android:id="@+id/btnTelefono"
        style="@style/estiloBoton"
        android:layout_marginTop="16dp"
        android:hint="LLamar"
        android:onClick="telefono"
        />

    <Button
        android:id="@+id/btnCorreo"
        style="@style/estiloBoton"
        android:layout_marginTop="16dp"
        android:hint="Enviar Correo"
        android:onClick="correo"
        />

    <de.hdodenhof.circleimageview.CircleImageView
        android:id="@+id/imgLogo"
        android:layout_width="match_parent"
        android:layout_height="191dp"
        android:layout_marginTop="100dp"
        android:src="@mipmap/prime"
        app:civ_border_color="#FF000000"

```

```

        app:civ_border_width="5dp"
        app:civ_circle_background_color="#79ffffff" />
    </LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>

```

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".MainActivity">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginStart="16dp"
        android:layout_marginTop="16dp"
        android:layout_marginEnd="16dp"
        android:layout_marginBottom="16dp"
        android:background="@drawable/background"
        android:orientation="vertical"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="1.0">

        <EditText
            android:id="@+id/txtEmail"
            style="@style/estiloEditText"
            android:layout_width="match_parent"
            android:layout_marginTop="40dp"
            android:autofillHints=""
            android:hint="@string/email"
            android:inputType="textPersonName" />

        <EditText
            android:id="@+id/txtClave"
            style="@style/estiloEditText"
            android:layout_marginTop="16dp"
            android:hint="@string/clave"
            android:inputType="textPassword"
            android:autofillHints="" />

        <Button
            android:id="@+id/btnAceptar"
            android:layout_marginTop="16dp"
            style="@style/estiloBoton"
            android:hint="@string/login"
            android:onClick="iniciarSesion"/>

    </LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
<de.hdodenhof.circleimageview.CircleImageView

```

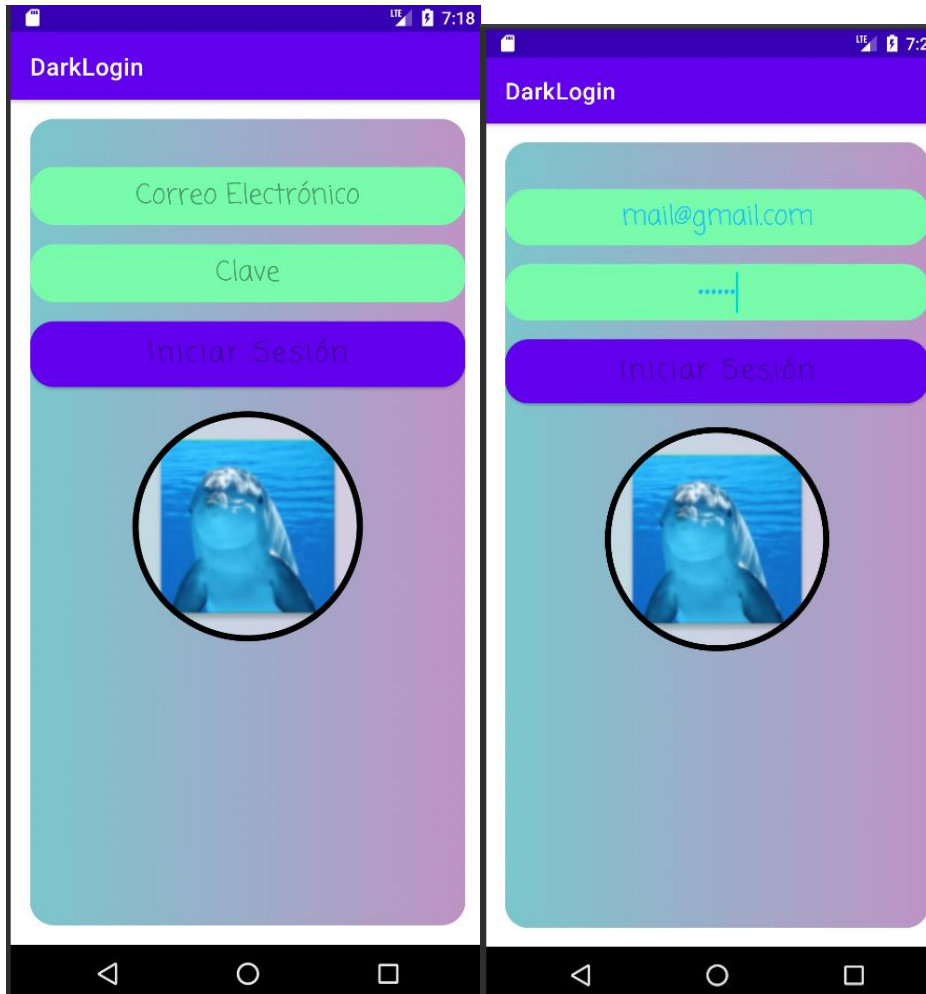
```
        android:id="@+id/imgLogo"
        android:layout_width="match_parent"
        android:layout_height="191dp"
        android:layout_marginTop="20dp"
        android:src="@mipmap/prime"
        app:civ_border_color="#FF000000"
        app:civ_border_width="5dp"
        app:civ_circle_background_color="#79ffffff" />

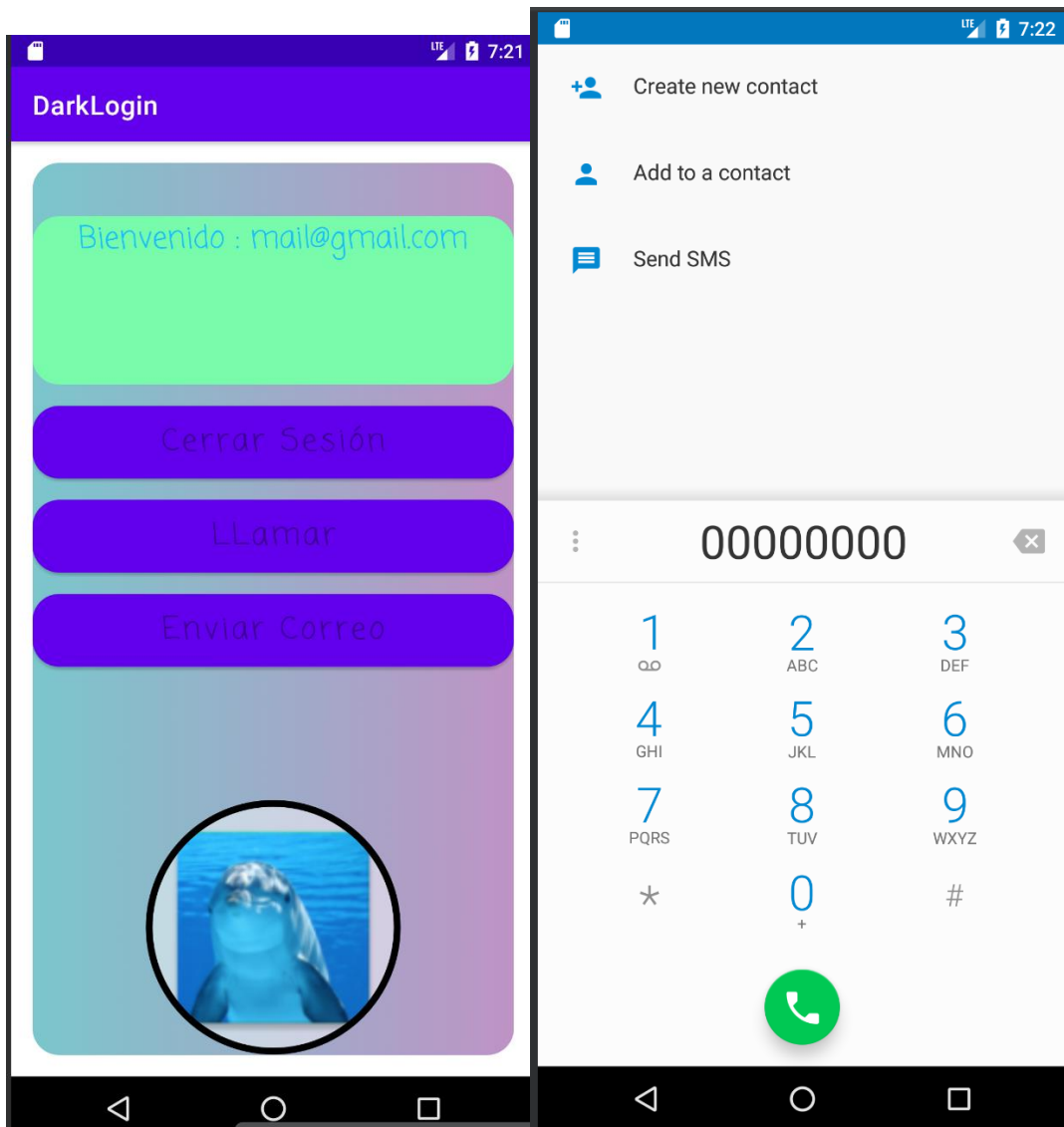
    </LinearLayout>

</androidx.constraintlayout.widget.ConstraintLayout>
```



## Resultados





## Conclusiones

El código presentado es un ejemplo básico de una aplicación de inicio de sesión en Android utilizando el patrón MVC y SharedPreferences para el almacenamiento de datos. Proporciona una estructura simple pero funcional para permitir a los usuarios iniciar sesión, almacenar y recuperar información de inicio de sesión, y realizar algunas acciones adicionales una vez que han iniciado sesión correctamente.

Este ejemplo puede servir como punto de partida para desarrollar aplicaciones más complejas que requieran un inicio de sesión seguro y la gestión de preferencias compartidas. Al comprender cómo se estructura el código y cómo se utilizan las diferentes clases y componentes de Android, los desarrolladores pueden personalizar y ampliar la funcionalidad según sus necesidades específicas.