

CLINICA VETERINARIA



MANUAL TECNICO

YESENIA NICOLE NOVOA VELASQUEZ NV210134

DIEGO BENJAMIN GUERRERO ZELAYA GZ210369

REQUERIMIENTOS DEL PROYECTO

Android Studio es un entorno de desarrollo integrado (IDE) oficial para el desarrollo de aplicaciones para Android



Git es un sistema de control de versiones distribuido (DVCS) que permite a los desarrolladores de software colaborar en proyectos de forma eficiente



Firebase es una plataforma de desarrollo de aplicaciones móviles y web que ofrece una gama de servicios que pueden ayudar a los desarrolladores a crear aplicaciones de alta calidad de forma rápida y sencilla.



Kotlin es un lenguaje de programación multiplataforma, estáticamente tipado, de alto nivel y propósito general con inferencia de tipos.



ESTRUCTURA DEL PROYECTO

Android Manifest

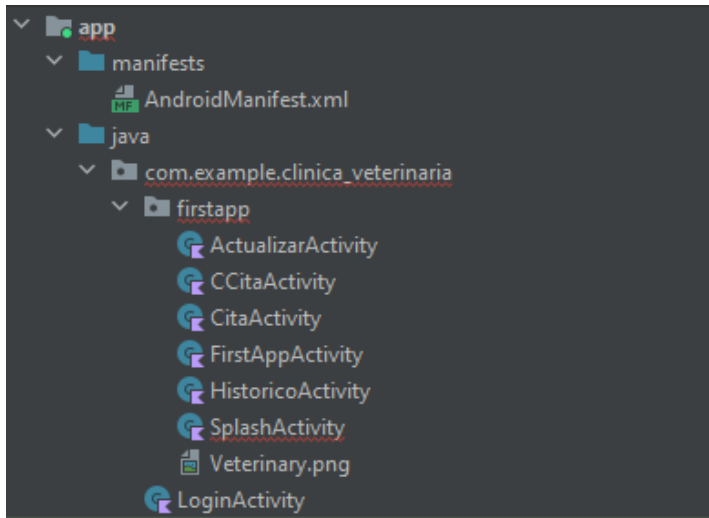
El archivo de manifiesto se utiliza para describir el nombre de la aplicación, la versión de la aplicación, los componentes de la aplicación y los permisos necesarios para que la aplicación funcione.

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="Clinica_Veterinaria"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Clinica_Veterinaria"
        tools:targetApi="31">
        <activity
            android:name=".firstapp.HistoricoActivity"
            android:exported="false" />
    </activity>
</manifest>
```

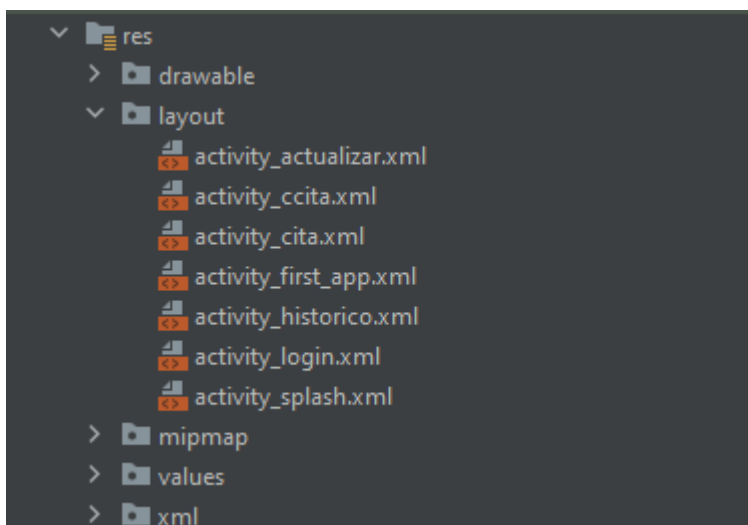
Activities

Aquí están alojadas las actividades que se utilizan para mostrar contenido, interactuar con los usuarios y proporcionar una experiencia de usuario coherente.

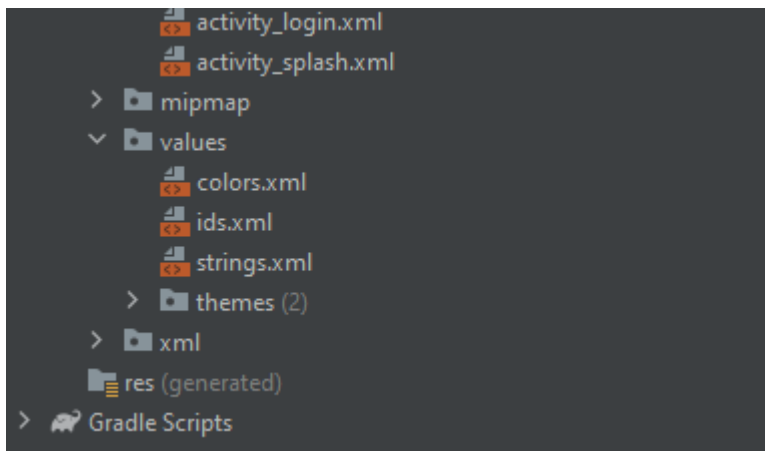


XML

Dentro de la carpeta res tenemos varias carpetas, la carpeta layout se utiliza para diseñar las pantallas de la aplicación.



En la misma carpeta res tenemos la carpeta VALUES, aquí se guardan ID, colores bases, entre otros.



STRINGS.XML

Aquí se encuentran todos los ID registrados

```
<resources>
  <string name="app_name">Clinica_Veterinaria</string>
  <string name="Login">Iniciar sesion\n</string>
  <string name="hint_email">Ingrese su correo electronico\n</string>
  <string name="hint_pass">Ingrese su contraseña</string>
  <string name="modificar_articulo">Modificar articulo</string>
  <string name="foto_mascota">Foto de la mascota</string>
  <string name="modificar_nuevo_articulo">ACEPTAR\n</string>
  <string name="citas">CITAS</string>
  <string name="pacientes">PACIENTES</string>
  <string name="dui">DUI</string>
  <string name="nmascota">Nombre de la mascota</string>
  <string name="campo">Cambie el campo deseado</string>
  <string name="actualizar_cita">Actualizar Cita</string>
  <string name="historico_cita">Historico Cita</string>
  <string name="historico_pacientes">Historico Pacientes</string>
  <string name="I_dui">Ingrese el DUI del dueño para poder acceder a los datos del paciente</string>
  <string name="medico">Medico encargado</string>
  <string name="verificar">Verificar</string>
  <string name="aceptar">Aceptar</string>
</resources>
```

SplashActivity

En esta Activity está el código de la pantalla tipo Splash

```
package com.example.clinica_veterinaria.firstapp

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import com.example.clinica_veterinaria.R

class SplashActivity : AppCompatActivity() {

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_splash)

        //Retraso de 2 segundos
        Handler().postDelayed({
            // Inicia la actividad principal
            startActivity(Intent(this, CitaActivity::class.java))
            finish()
        }, 2000)
    }
}
```

```

package com.example.veterinaria_kotlin

import ...

class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        val preferences = PreferenceHelper.defaultPrefs(context: this)
        if(preferences["session", false])
            goToMenu()

        val tvGoRegister = findViewById<TextView>(R.id.tv_go_to_register)
        tvGoRegister.setOnClickListener{ it: View!
            goToRegister()
        }

        val btnGoMenu = findViewById<Button>(R.id.btn_go_to_menu)
        btnGoMenu.setOnClickListener{ it: View!
            goToMenu()
        }
    }

    private fun goToRegister(){
        val i = Intent(packageContext: this, RegisterActivity::class.java)
        startActivity(i)
    }

    private fun goToMenu(){
        val i = Intent(packageContext: this, MenuActivity::class.java)
        createSessionPreference()
        startActivity(i)
        finish()
    }

    private fun createSessionPreference(){
        val preferences = PreferenceHelper.defaultPrefs(context: this)
        preferences["session"] = true
    }
}

```



```
package com.example.veterinaria_kotlin

import ...

class RegisterActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_register)
        val tvGoLogin = findViewById<TextView>(R.id.tv_go_to_login)
        tvGoLogin.setOnClickListener { it: View!
            goToLogin()
        }
    }

    private fun goToLogin(){
        val i = Intent( packageContext: this, MainActivity::class.java)
        startActivity(i)
    }
}
```

```

package com.example.veterinaria_kotlin

import ...

class MenuActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_menu)

        val btnLogout = findViewById<Button>(R.id.btn_Logout)
        btnLogout.setOnClickListener { it: View!
            clearSessionPreferences()
        }

        val btnReservarCita = findViewById<Button>(R.id.btn_reservar_cita)
        btnReservarCita.setOnClickListener { it: View!

        }

    }

    private fun goToCreateAppointment(){
        val i = Intent( packageContext: this, CreateAppointmentActivity::class.java)
        startActivity(i)
    }

    private fun goToLogin(){
        val i = Intent( packageContext: this, MainActivity::class.java)
        startActivity(i)
        finish()
    }

    private fun clearSessionPreferences(){
        val preferences = PreferenceHelper.defaultPrefs( context: this)
        preferences["session"] = false
    }
}

```