Web Services (servicios web)

Son servicios que a través de Api’s puede realizar cualquier operación (CRUD) desde un celular, Tablet u otro dispositivo en un servidor Web



Requerimientos

* Tener Instalado Xampp, Visual Studio Code y Android Studio
* Tener conexión de Internet

Pasos

1. Crear una base de datos llamada Empresa en MySql con una tabla llamada Usuario con los siguientes campos:



Crear al menos tres registros

insert into Usuario (usr,nombre,correo,clave)

values('FreddyMC','Freddy Moscoso','fmc@gmail.com','1234'),

('LuisaFer','Luisa Fernanda','lfmg@gmail.com','1234'),

('DianaP','Diana Patricia','dpga@gmail.com','1234')

1. Crear la carpeta WebServices en C:\Xampp\htdocs

Crear las Api’s correspondientes

* Ingreso.php

<?php

if (isset($\_REQUEST['correo']) && isset($\_REQUEST['clave']))

{

$correo=$\_REQUEST['correo'];

$clave=$\_REQUEST['clave'];

$cnx = mysqli\_connect("localhost","root","","Empresa");

$res=$cnx->query("select \* from Usuario where correo = '$correo' and clave = '$clave'");

$json = array();

foreach ($res as $row)

{

$json['datos'][]=$row;

}

//pasar los datos del array a JSON con información o vacío

echo json\_encode($json);

}

else

{

echo "El correo y la clave son obligatorios";

}

?>

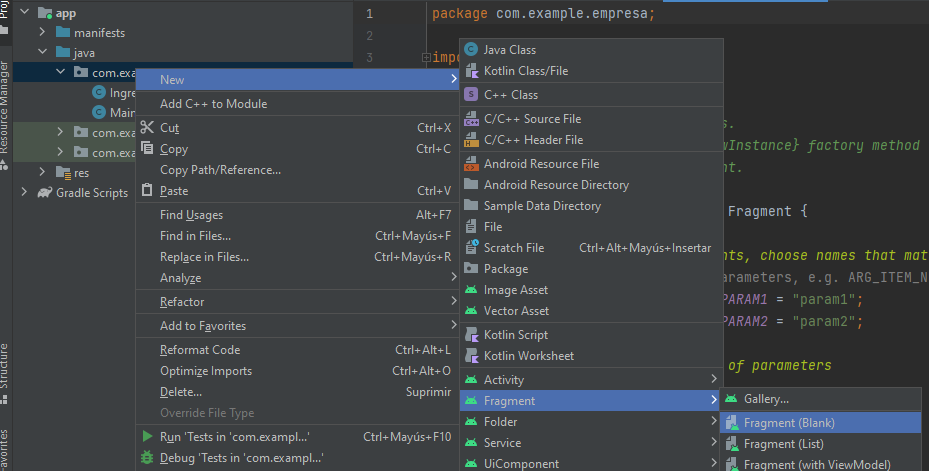
1. Crear una aplicación en Android Studio llamada Empresa con un formulario con un solo layout (con nombre y orientación horizontal), queda así:

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout 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:id="@+id/Escenario"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"  
 tools:context=".MainActivity">  
</LinearLayout>

* Crear una vista (fragment) dentro de la actividad



* Los pasos para crearla son los siguientes



* Dar el nombre de IngresoFragment
* Cambiar en el diseño del fragment el tipo de Layout por LinearLayout y su orientación debe ser vertical
* Agregar el siguiente diseño en fragment\_ingreso.xml

<TextView  
 android:layout\_marginTop="200dp"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Inicio de Sesion"  
 android:gravity="center\_horizontal" />  
  
<TextView  
 android:layout\_marginTop="10dp"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Correo Electronico"  
 android:gravity="center\_horizontal" />  
  
<EditText  
 android:id="@+id/etcorreo"  
 android:layout\_width="300dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center\_horizontal"  
 android:hint="p.e. micorreo@gmail.com"  
 android:inputType="textEmailAddress"  
 android:gravity="center\_horizontal" />  
  
<TextView  
 android:layout\_marginTop="10dp"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Contraseña"  
 android:gravity="center\_horizontal" />  
  
<EditText  
 android:id="@+id/etclave"  
 android:layout\_width="300dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center\_horizontal"  
 android:hint="Contraseña"  
 android:inputType="textPassword"  
 android:gravity="center\_horizontal" />  
  
<Button  
 android:layout\_marginTop="20dp"  
 android:id="@+id/btingresar"  
 android:layout\_height="wrap\_content"  
 android:layout\_width="200dp"  
 android:text="Ingresar"  
 android:layout\_gravity="center\_horizontal" />  
<TextView  
 android:layout\_marginTop="10dp"  
 android:id="@+id/tvregistrar"  
 android:layout\_width="200dp"  
 android:layout\_height="wrap\_content"  
 android:text="Haz clic AQUI para Registrarse!"  
 android:textColor="#3F51B5"  
 android:layout\_gravity="center\_horizontal"  
 android:gravity="center\_horizontal" />

* Eliminar el código en IngresoFragment.Java para que solo quede el ultimo método así:

package com.example.empresa;  
  
import android.os.Bundle;  
import androidx.fragment.app.Fragment;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.view.ViewGroup;  
  
public class IngresoFragment extends Fragment {  
   
 @Override  
 public View onCreateView(LayoutInflater inflater, ViewGroup container,  
 Bundle savedInstanceState) {  
 // Inflate the layout for this fragment  
 return inflater.inflate(R.layout.*fragment\_ingreso*, container, false);  
 }  
}

* Cambiar el layout del MainActivity por la vista, se ingresa al método OnCreate del MainActivity.Java y se adiciona el siguiente código:

//Ocultar la barra de menu por defecto  
getSupportActionBar().hide();  
//Libreria para el manejo de vistas (fragment)  
FragmentManager fm = getSupportFragmentManager();  
fm.beginTransaction().replace(R.id.*Escenario*,new IngresoFragment()).commit();

* Ejecutar desde el emulador...
* Especificar en GradleScripts (build.gradle(Module.app)) la siguiente dependencia que permita hacer conexiones http: (permite hacer conexiones remotas)

implementation 'com.android.volley:volley:1.2.1'

* Hacer clic en Sync now....(parte superior derecha)
* En el archivo AndroidManifest.xml (está en la carpeta manifest, encima de la carpeta Java), especificar la línea que permite tomar datos de internet:

<uses-permission android:name="android.permission.INTERNET"/>

Nota: debe colocarse antes de <Aplication

* En la vista fragment (IngresoFragment.java) agregar al encabezado de la clase el siguiente código:

implements Response.Listener<JSONObject>,Response.ErrorListener

debe quedar:

public class SesionFragment extends Fragment implements Response.Listener<JSONObject>,Response.ErrorListener {  
  
 EditText jetcorreo,jetclave;  
 Button jbtingresar;  
 TextView jtvregistrar;  
 RequestQueue rq;  
 JsonRequest jrq;  
  
  
 @Override  
 public View onCreateView(LayoutInflater inflater, ViewGroup container,  
 Bundle savedInstanceState) {  
 // Inflate the layout for this fragment  
 //return inflater.inflate(R.layout.fragment\_sesion, container, false);  
 View vista = inflater.inflate(R.layout.*fragment\_sesion*,container,false);  
 rq = Volley.*newRequestQueue*(getContext());//requerimiento Volley  
 jetcorreo=vista.findViewById(R.id.*etcorreo*);  
 jetclave=vista.findViewById(R.id.*etclave*);  
 jbtingresar=vista.findViewById(R.id.*btingresar*);  
 jtvregistrar=vista.findViewById(R.id.*tvregistrar*);  
 jbtingresar.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Iniciar\_sesion();  
 }  
 });  
 return vista;  
 }  
  
 private void Iniciar\_sesion(){  
 String correo,clave;  
 correo=jetcorreo.getText().toString();  
 clave=jetclave.getText().toString();  
 if (correo.isEmpty() || clave.isEmpty()){  
 Toast.*makeText*(getContext(), "Correo y clave son requeridos", Toast.*LENGTH\_SHORT*).show();  
 jetcorreo.requestFocus();  
 }  
 else{  
 String url = "http://172.16.60.36:8080/WebServices/Sesion.php?correo="+correo+"&clave="+clave+"";  
 jrq = new JsonObjectRequest(Request.Method.*GET*,url,null,this,this);  
 rq.add(jrq);  
 }  
 }  
  
 @Override  
 public void onErrorResponse(VolleyError error) {  
 Toast.*makeText*(getContext(), "Error en inicio de sesion", Toast.*LENGTH\_SHORT*).show();  
 }  
  
 @Override  
 public void onResponse(JSONObject response) {  
 Toast.*makeText*(getContext(), "Sesion iniciada", Toast.*LENGTH\_SHORT*).show();  
 }  
}

Nota: La clase a implementar es una clase abstracta y por lo tanto necesita que se implementen los métodos abstractos

(click derecho sobre el encabezado de la clase, generate, implements methods y Aceptar)

....y se genera 2 métodos...onResponse y onErrorResponse ....

* Definir los objetos que se requieren para la conexión

EditText jetcorreo,jetclave;

Button jbtingresar;

TextView jtvregistrar;

RequestQueue rq;

JsonRequest jrq;

* Asociar objetos Xml con objetos Java

Se debe poner en comentario el return por defecto y cambiarlo por mi propia vista

// return inflater.inflate(R.layout.fragment\_ingreso, container, false);  
  
 //Asociar objetos  
 View vista = inflater.inflate(R.layout.*fragment\_ingreso*,container,false);  
 jetcorreo = vista.findViewById(R.id.*etcorreo*);  
 jetclave = vista.findViewById(R.id.*etclave*);  
 jbtingresar = vista.findViewById(R.id.*btingresar*);  
 jtvregistrar=vista.findViewById(R.id.*tvregistrar*);  
  
 return vista;

Instrucciones nuevas

rq = Volley.newRequestQueue(getContext());//requerimiento Volley

String url = "http://192.168.1.62/serviciosandroidphp/sesion.php?usr="+usr.getText().toString()+"&clave="+clave.getText().toString();

jrq = new JsonObjectRequest(Request.Method.GET,url,null,this,this);

Adicionar una actividad llamada UsuarioActivity

Instrucciones UsuarioActivity

Código Xml

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout 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=".UsuarioActivity"  
 android:orientation="horizontal">  
  
 <LinearLayout  
 android:layout\_width="409dp"  
 android:layout\_height="729dp"  
 android:orientation="vertical"  
 tools:layout\_editor\_absoluteX="1dp"  
 tools:layout\_editor\_absoluteY="1dp">  
  
 <TextView  
 android:layout\_width="410dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="60dp"  
 android:gravity="center"  
 android:text="Usuarios"  
 android:textSize="40dp" />  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="70dp"  
 android:layout\_marginTop="40dp">  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="20dp"  
 android:layout\_marginTop="20dp"  
 android:text="Usuario"  
 android:textSize="30dp" />  
  
 <EditText  
 android:id="@+id/etusuario"  
 android:layout\_width="240dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="20dp"  
 android:layout\_marginTop="20dp"  
 android:textSize="30dp" />  
  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="70dp">  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="20dp"  
 android:layout\_marginTop="20dp"  
 android:text="Nombre"  
 android:textSize="30dp" />  
  
 <EditText  
 android:id="@+id/etnombre"  
 android:layout\_width="240dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="20dp"  
 android:layout\_marginTop="20dp"  
 android:textSize="30dp" />  
  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="70dp">  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="20dp"  
 android:layout\_marginTop="20dp"  
 android:text="Correo"  
 android:textSize="30dp" />  
  
 <EditText  
 android:id="@+id/etcorreo"  
 android:layout\_width="260dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="20dp"

android:inputType="textEmailAddress"  
 android:layout\_marginTop="20dp"  
 android:textSize="30dp" />  
  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="70dp">  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="20dp"  
 android:layout\_marginTop="20dp"  
 android:text="Clave"  
 android:textSize="30dp" />  
  
 <EditText  
 android:id="@+id/etclave"  
 android:layout\_width="270dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="20dp"

android:inputType="textPassword"  
 android:layout\_marginTop="20dp"  
 android:textSize="30dp" />  
  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="200dp"  
 android:layout\_height="70dp"  
 android:layout\_gravity="center">  
  
 <CheckBox  
 android:id="@+id/cbactivo"  
 android:layout\_width="200dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 android:enabled="false"  
 android:layout\_weight="1"  
 android:gravity="center"  
 android:text=" Activo"  
 android:textSize="30dp" />  
  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="50dp"  
 android:layout\_gravity="center"  
 android:layout\_marginTop="60dp"  
 android:gravity="center">  
  
  
 <Button  
 android:id="@+id/btnadicionar"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Adicionar" />  
  
 <Button  
 android:id="@+id/btnconsultar"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="20dp"  
 android:text="Consultar" />  
  
 <Button  
 android:id="@+id/btnmodificar"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="20dp"  
 android:text="Modificar" />  
  
  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="wrap\_content"  
 android:layout\_height="50dp"  
 android:layout\_gravity="center"  
 android:gravity="center">  
  
  
 <Button  
 android:id="@+id/btnanular"  
 android:layout\_width="100dp"  
 android:layout\_height="wrap\_content"  
 android:text="Anular" />  
  
 <Button  
 android:id="@+id/btnlimpiar"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="20dp"  
 android:text="Limpiar" />  
  
 <Button  
 android:id="@+id/btnregresar"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="20dp"  
 android:text="Regresar" />  
  
 </LinearLayout>  
 </LinearLayout>  
</LinearLayout>

Código Java

EditText jetusuario,jetnombre,jetcorreo,jetclave;  
CheckBox jcbactivo;  
RequestQueue rq;  
JsonRequest jrq;  
String usr,nombre,correo,clave;

Instrucciones dentro del método OnCreate

//ocultar la barra, asociar objetos Java con objetos Xml  
// inicializar la cola de consultar  
getSupportActionBar().hide();  
jetusuario=findViewById(R.id.*etusuario*);  
jetnombre=findViewById(R.id.*etnombre*);  
jetcorreo=findViewById(R.id.*etcorreo*);  
jetclave=findViewById(R.id.*etclave*);  
jcbactivo=findViewById(R.id.*cbactivo*);  
rq = Volley.*newRequestQueue*(this);

Método para consultar usuarios

Primero la clase debe heredar de Android Studio la librería correspondiente

implements Response.Listener<JSONObject>,Response.ErrorListener

Nota: recuerde, va en el encabezado de la clase y al ser una clase abstracta se le deben implementar los métodos OnResponse y OnErrorResponse

Las instrucciones para capturar la información enviada por Json son las siguientes

public void Consultar(View view){  
 usr=jetusuario.getText().toString();  
 if (usr.isEmpty()){  
 Toast.*makeText*(this, "Usuario es requerido para la busqueda", Toast.*LENGTH\_SHORT*).show();  
 jetusuario.requestFocus();  
 }  
 else{  
 String url = "http://172.16.60.36:8080/WebServices/consulta.php?usr="+usr;  
 jrq = new JsonObjectRequest(Request.Method.*GET*,url,null,this,this);  
 rq.add(jrq);  
 }  
}

@Override  
public void onErrorResponse(VolleyError error) {  
 Toast.*makeText*(this, "Usuario no registrado", Toast.*LENGTH\_SHORT*).show();  
}  
  
@Override  
public void onResponse(JSONObject response) {  
 Toast.*makeText*(this, "Usuario registrado", Toast.*LENGTH\_SHORT*).show();  
 JSONArray jsonArray = response.optJSONArray("datos");  
 JSONObject jsonObject = null;  
 try {  
 jsonObject = jsonArray.getJSONObject(0);//posicion 0 del arreglo....  
 jetnombre.setText(jsonObject.optString("nombre"));  
 jetcorreo.setText(jsonObject.optString("correo"));  
 jetclave.setText(jsonObject.optString("clave"));  
 if (jsonObject.optString("activo").equals("si"))  
 jcbactivo.setChecked(true);  
 else  
 jcbactivo.setChecked(false);  
 } catch (JSONException e) {  
 e.printStackTrace();  
 }  
}

Métodos para Limpiar campos y regresar

public void Limpiar(View view){  
 Limpiar\_campos();  
}  
  
public void Regresar(View view){  
 Intent intmain=new Intent(this,MainActivity.class);  
 startActivity(intmain);  
}  
  
private void Limpiar\_campos(){  
 jetnombre.setText("");  
 jetcorreo.setText("");  
 jetclave.setText("");  
 jetusuario.setText("");  
 jcbactivo.setChecked(false);  
 jetusuario.requestFocus();  
}

Método para guardar

public void Guardar(View view){  
 usr=jetusuario.getText().toString();  
 nombre=jetnombre.getText().toString();  
 correo=jetcorreo.getText().toString();  
 clave=jetclave.getText().toString();  
 if (usr.isEmpty() || nombre.isEmpty() || correo.isEmpty() || clave.isEmpty()){  
 Toast.*makeText*(this, "Todos los datos son requeridos", Toast.*LENGTH\_SHORT*).show();  
 jetusuario.requestFocus();  
 }  
 else{  
 String url = "http://172.16.60.36:8080/WebServices/registrocorreo.php";  
 StringRequest postRequest = new StringRequest(Request.Method.*POST*, url,  
 new Response.Listener<String>()  
 {  
 @Override  
 public void onResponse(String response) {  
 Limpiar\_campos();  
 Toast.*makeText*(getApplicationContext(), "Registro de usuario realizado!", Toast.*LENGTH\_LONG*).show();  
 }  
 },  
 new Response.ErrorListener()  
 {  
 @Override  
 public void onErrorResponse(VolleyError error) {  
 Toast.*makeText*(getApplicationContext(), "Registro de usuario incorrecto!", Toast.*LENGTH\_LONG*).show();  
 }  
 }  
 ) {  
 @Override  
 protected Map<String, String> getParams()  
 {  
 Map<String, String> params = new HashMap<String, String>();  
 params.put("usr",jetusuario.getText().toString().trim());  
 params.put("nombre", jetnombre.getText().toString().trim());  
 params.put("correo",jetcorreo.getText().toString().trim());  
 params.put("clave",jetclave.getText().toString().trim());  
 return params;  
 }  
 };  
 RequestQueue requestQueue = Volley.*newRequestQueue*(this);  
 requestQueue.add(postRequest);  
  
 }  
}

Método Eliminar

public void Eliminar(View view){  
 usr=jetusuario.getText().toString();  
 if (usr.isEmpty()){  
 Toast.*makeText*(this, "El usuario es requerido", Toast.*LENGTH\_SHORT*).show();  
 jetusuario.requestFocus();  
 }  
 else{  
 url = "http://172.16.60.36:8080/WebServices/elimina.php";  
 StringRequest postRequest = new StringRequest(Request.Method.*POST*, url,  
 new Response.Listener<String>()  
 {  
 @Override  
 public void onResponse(String response) {  
 Limpiar\_campos();  
 Toast.*makeText*(getApplicationContext(), "Registro eliminado!", Toast.*LENGTH\_LONG*).show();  
 }  
 },  
 new Response.ErrorListener()  
 {  
 @Override  
 public void onErrorResponse(VolleyError error) {  
 Toast.*makeText*(getApplicationContext(), "Registro no eliminado!", Toast.*LENGTH\_LONG*).show();  
 }  
 }  
 ) {  
 @Override  
 protected Map<String, String> getParams()  
 {  
 Map<String, String> params = new HashMap<String, String>();  
 params.put("usr",jetusuario.getText().toString().trim());  
 return params;  
 }  
 };  
 RequestQueue requestQueue = Volley.*newRequestQueue*(this);  
 requestQueue.add(postRequest);  
 }  
}

Método Anular

public void Anular(View view){  
 usr=jetusuario.getText().toString();  
 if (usr.isEmpty()){  
 Toast.*makeText*(this, "El usuario es requerido", Toast.*LENGTH\_SHORT*).show();  
 jetusuario.requestFocus();  
 }  
 else{  
 url = "http://172.16.60.36:8080/WebServices/anula.php";  
 StringRequest postRequest = new StringRequest(Request.Method.*POST*, url,  
 new Response.Listener<String>()  
 {  
 @Override  
 public void onResponse(String response) {  
 Limpiar\_campos();  
 Toast.*makeText*(getApplicationContext(), "Registro anulado!", Toast.*LENGTH\_LONG*).show();  
 }  
 },  
 new Response.ErrorListener()  
 {  
 @Override  
 public void onErrorResponse(VolleyError error) {  
 Toast.*makeText*(getApplicationContext(), "Registro no anulado!", Toast.*LENGTH\_LONG*).show();  
 }  
 }  
 ) {  
 @Override  
 protected Map<String, String> getParams()  
 {  
 Map<String, String> params = new HashMap<String, String>();  
 params.put("usr",jetusuario.getText().toString().trim());  
 return params;  
 }  
 };  
 RequestQueue requestQueue = Volley.*newRequestQueue*(this);  
 requestQueue.add(postRequest);  
 }  
}