

VOLLEY

Uso de Volley para recibir solicitudes Json, String o archivos de imagen de un servidor., facilitando solicitudes y reduciendo el código.

Volley trabaja con caché php, se guarda copia en el caché.

StringRequest->Clase de librería Volley

Request.Method.POST-> Método

Conexión.URL_WEB_SERVICES ->Link donde se encuentra la Web Services

onResponse->Respuesta

onErrorResponse->Error

1. Crear tabla usuarios

Email como BINARY en atributos para que haga la diferencia entre mayúsculas y minúsculas.

Nombre de la tabla: usuarios Agregar 1 columna(s) Continuar

Nombre	Tipo	Longitud/Valores	Predeterminado	Cotejamiento	Atributos	Nulo	Índice	A.I	Comentarios
id_usuario	INT		Ninguno			<input type="checkbox"/>	PRIMARY	<input checked="" type="checkbox"/>	
nombre	VARCHAR	60	Ninguno			<input type="checkbox"/>	---	<input type="checkbox"/>	
email	VARCHAR	60	Ninguno		BINARY	<input type="checkbox"/>	---	<input type="checkbox"/>	
password	VARCHAR	100	Ninguno			<input type="checkbox"/>	---	<input type="checkbox"/>	
telefono	VARCHAR	12	Ninguno			<input type="checkbox"/>	---	<input type="checkbox"/>	

Comentarios de la tabla: Cotejamiento: Motor de almacenamiento: InnoDB

definición de la PARTICIÓN: Dividido por: (Expresión o lista de colur) Particiones:

Previsualizar SQL Guardar

2. Insertar un registro con función MD5 en password.

Servidor: localhost:3306 » Base de datos: moveinto_ejercicioandroid » Tabla: usuarios

Examinar Estructura SQL Buscar Insertar Exportar Importar Operaciones Disparadores

Columna	Tipo	Función	Nulo	Valor
id_usuario	int(11)			
nombre	varchar(60)			Raquel Ochoa Ornelas
email	varchar(60)			raqueoo@msn.com
password	varchar(100)	MD5		12345
telefono	varchar(12)			3411022306

Continuar

3. Consultar registro

Servidor: localhost:3306 » Base de datos: moveinto_ejercicioandroid » Tabla: usuarios

Examinar Estructura SQL Buscar Insertar Exportar Importar Operaciones Disparadores

✓ Mostrando filas 0 - 0 (total de 1, La consulta tardó 0.0002 segundos.)

SELECT * FROM `usuarios`

☐ Perfilando [Editar en línea]

☐ Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla

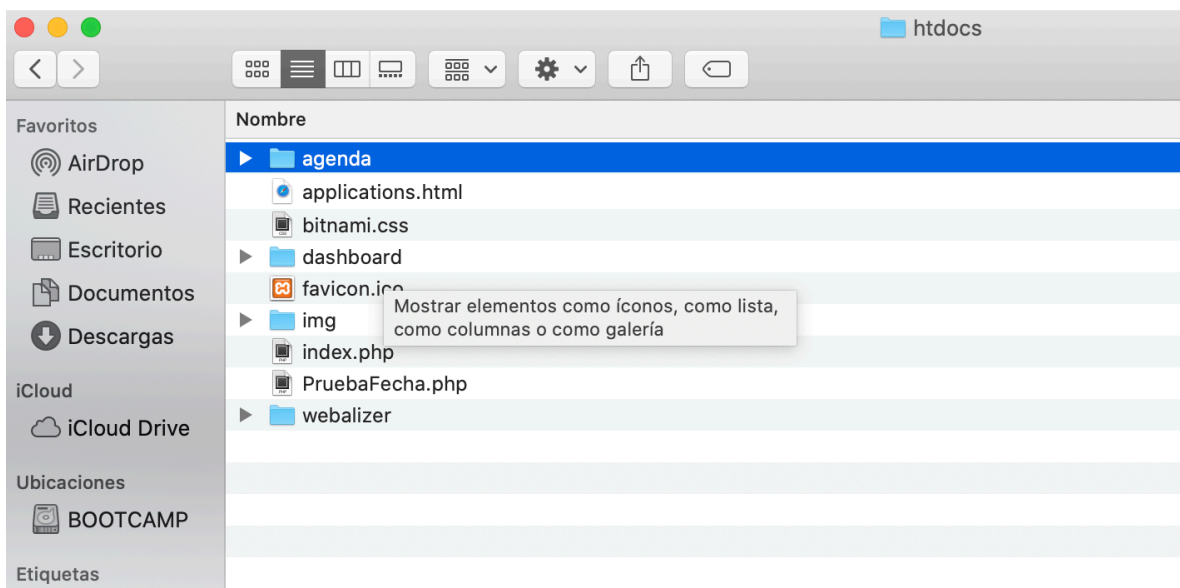
+ Opciones

	id_usuario	nombre	email	password	telefono
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	1	Raquel Ochoa Ornelas	raqueoo@msn.com	827ccb0eea8a706c4c34a16891f84e7b	3411022306

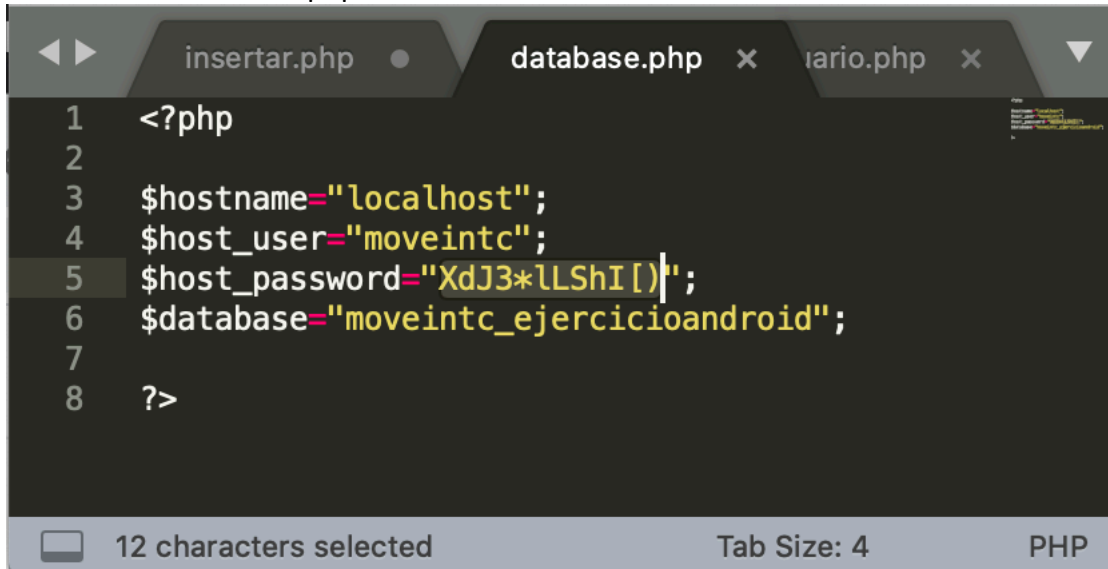
☐ Seleccionar todo Para los elementos que están marcados: ☐ Editar ☐ Copiar ☐ Borrar ☐ Exportar

☐ Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla

4. Crear carpeta en htdocs con el nombre agenda.



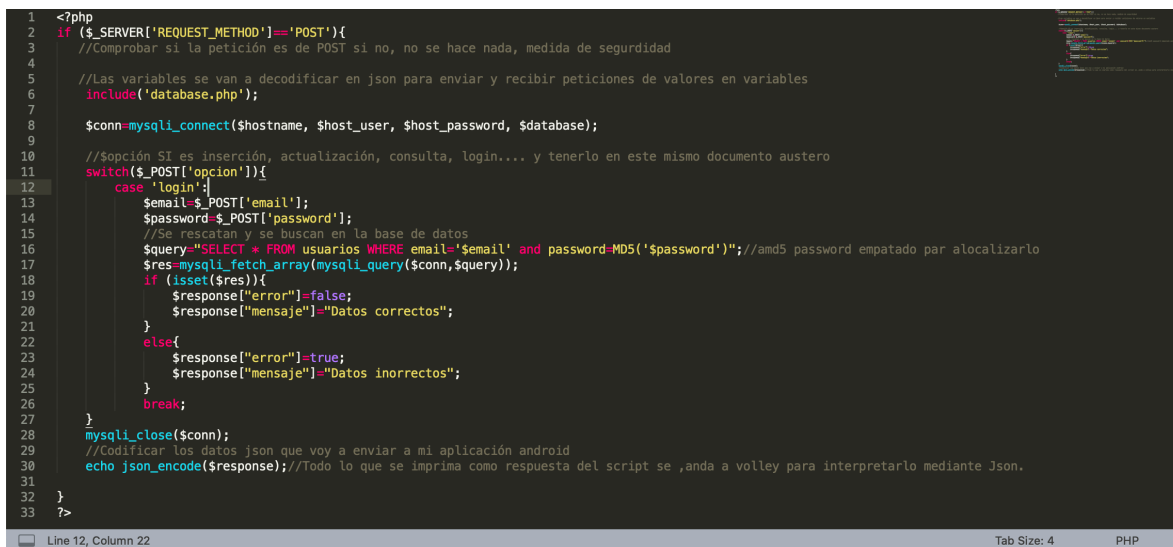
5. Crear database.php con la conexión a la base de datos



```
1 <?php
2
3 $hostname="localhost";
4 $host_user="moveintc";
5 $host_password="XdJ3*LLShI[]";
6 $database="moveintc_ejercicioandroid";
7
8 ?>
```

12 characters selected Tab Size: 4 PHP

6. Crear el archivo usuario.php para el diseño del webservice, el cual conecta la base de datos si se hizo una petición POST y realiza la búsqueda del usuario.

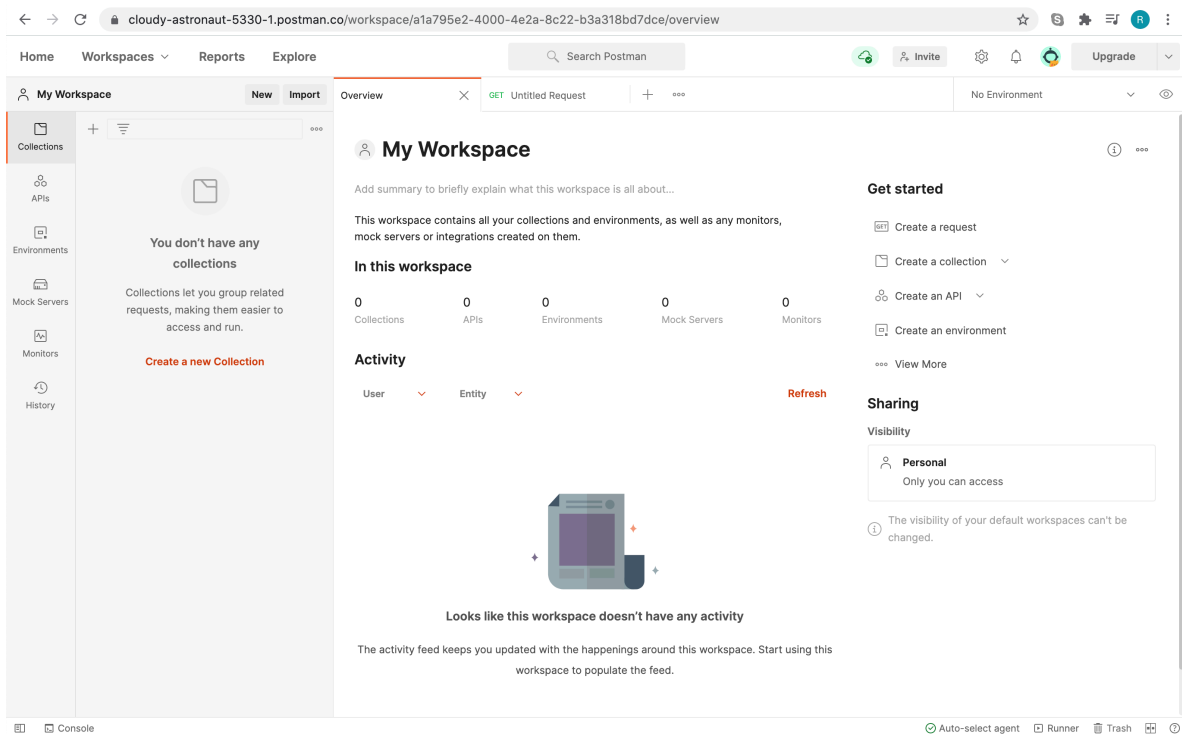
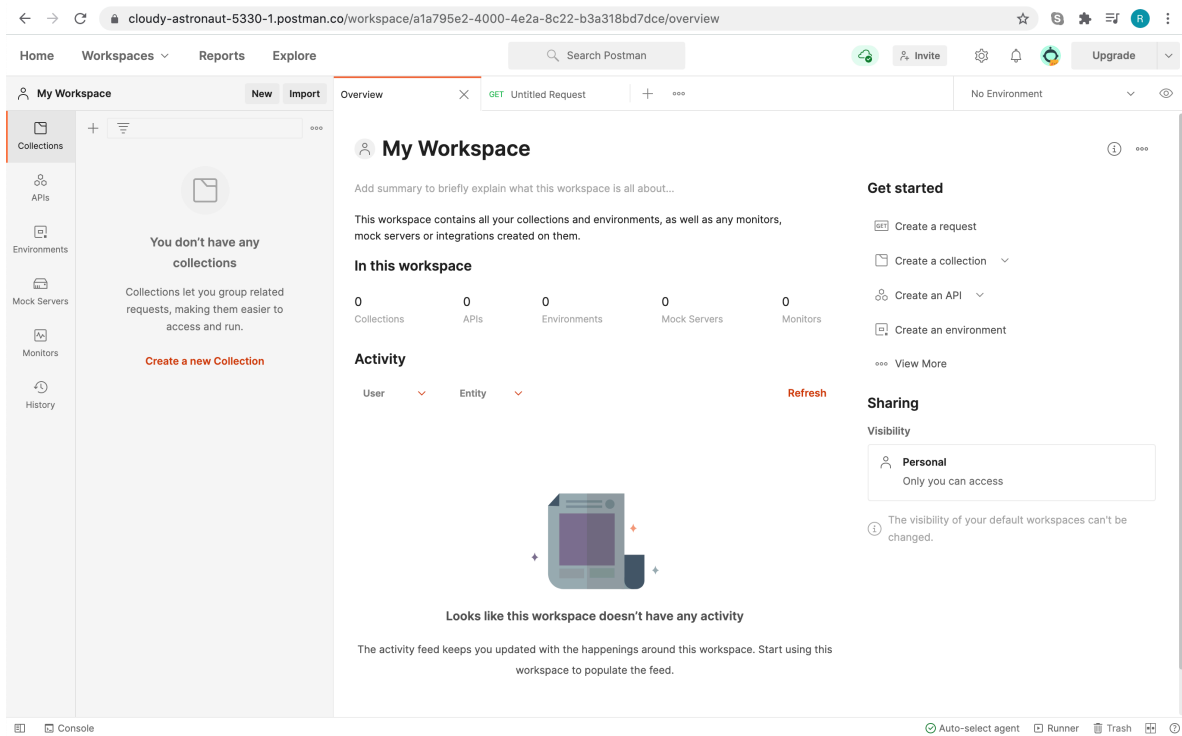


```
1 <?php
2 if ($_SERVER['REQUEST_METHOD']=='POST'){
3     //Comprobar si la petición es de POST si no, no se hace nada, medida de seguridad
4
5     //Las variables se van a decodificar en json para enviar y recibir peticiones de valores en variables
6     include('database.php');
7
8     $conn=mysqli_connect($hostname, $host_user, $host_password, $database);
9
10    //Opción SI es inserción, actualización, consulta, login... y tenerlo en este mismo documento austero
11    switch($_POST['opcion']){
12        case 'login':
13            $email=$_POST['email'];
14            $password=$_POST['password'];
15            //Se rescatan y se buscan en la base de datos
16            $query="SELECT * FROM usuarios WHERE email='".$email.'" and password=MD5('".$password."')"; //amd5 password empatado par alocalizarlo
17            $res=mysqli_fetch_array(mysqli_query($conn,$query));
18            if (isset($res)){
19                $response["error"]=false;
20                $response["mensaje"]="Datos correctos";
21            }
22            else{
23                $response["error"]=true;
24                $response["mensaje"]="Datos incorrectos";
25            }
26            break;
27        }
28    mysqli_close($conn);
29    //Codificar los datos json que voy a enviar a mi aplicación android
30    echo json_encode($response); //Todo lo que se imprima como respuesta del script se ,anda a volley para interpretarlo mediante Json.
31 }
32 ?>
```

Line 12, Column 22 Tab Size: 4 PHP

7. Registrarse en Postman on the web desde <http://postman.com>

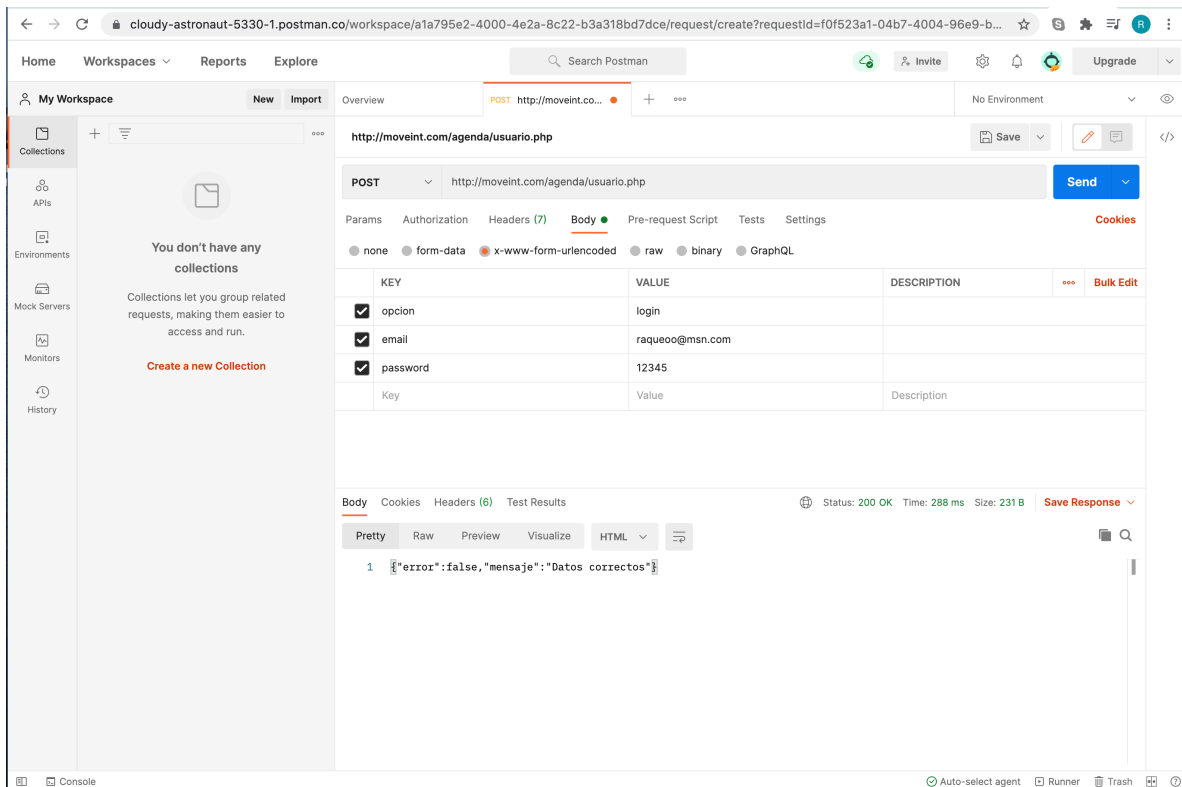
8. Ingresar a Postman ->My Workspace y seleccionar Create a request



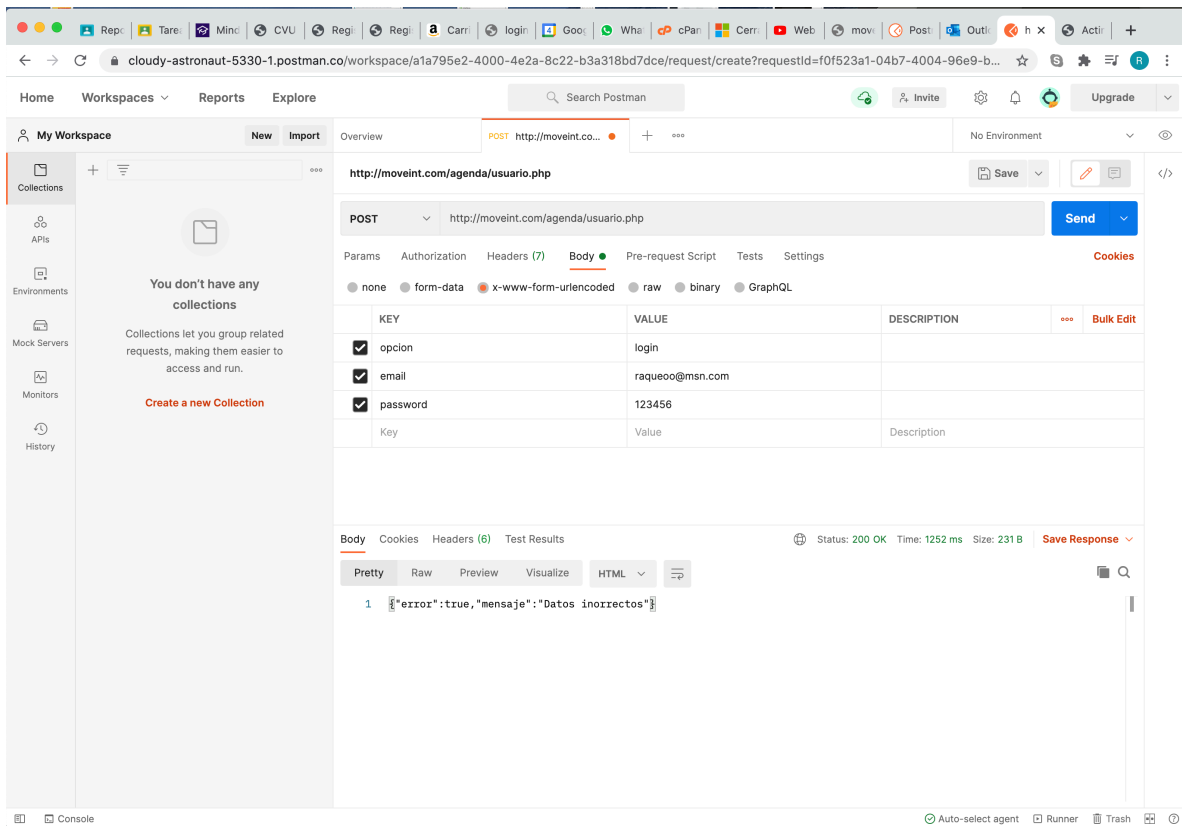
9. Seleccionar POST y escribir la URL del sitio para ejecutar el Webservice. Seleccionar Body y x-www-form-urlencoded. Dar de alta los parámetros que espera el documento php

Elaborada por: Dra. Raquel Ochoa Ornelas

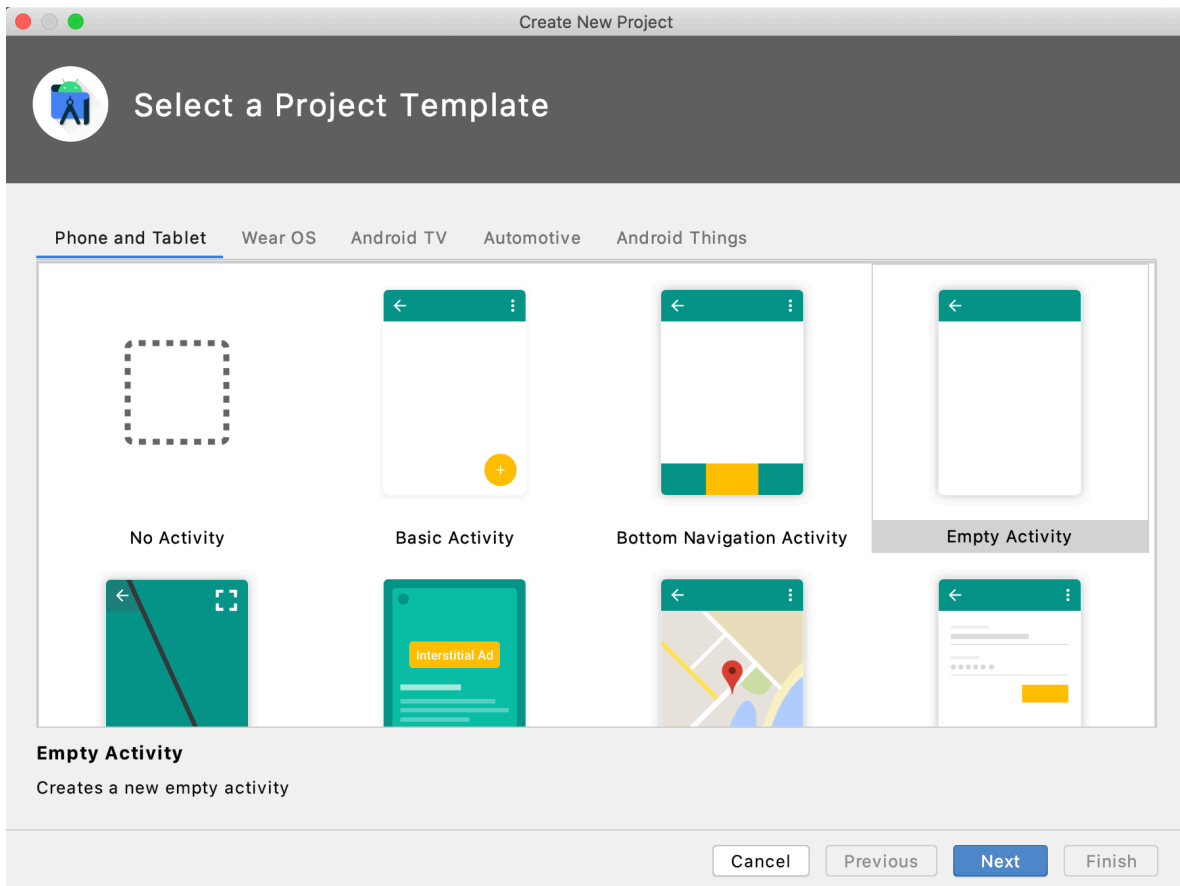
con POST: opcion, email y password. Después de establecer los parámetros, se debe dar clic en el botón Send. Probar con datos correctos que existen en la base de datos.



Posteriormente, probar con datos incorrectos.

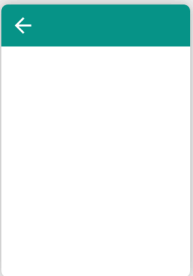


10. A continuación diseñar la app en Android, creando una Empty Activity:



Create New Project

Configure Your Project



Empty Activity

Creates a new empty activity


Name
AgendaAppWS


Package name
com.example.agendaappws


Save location
nelas/Documents/Semestre Enero-Junio2021/AgendaAppWS2

Language
Java

Minimum SDK
API 30: Android 11.0 (R)

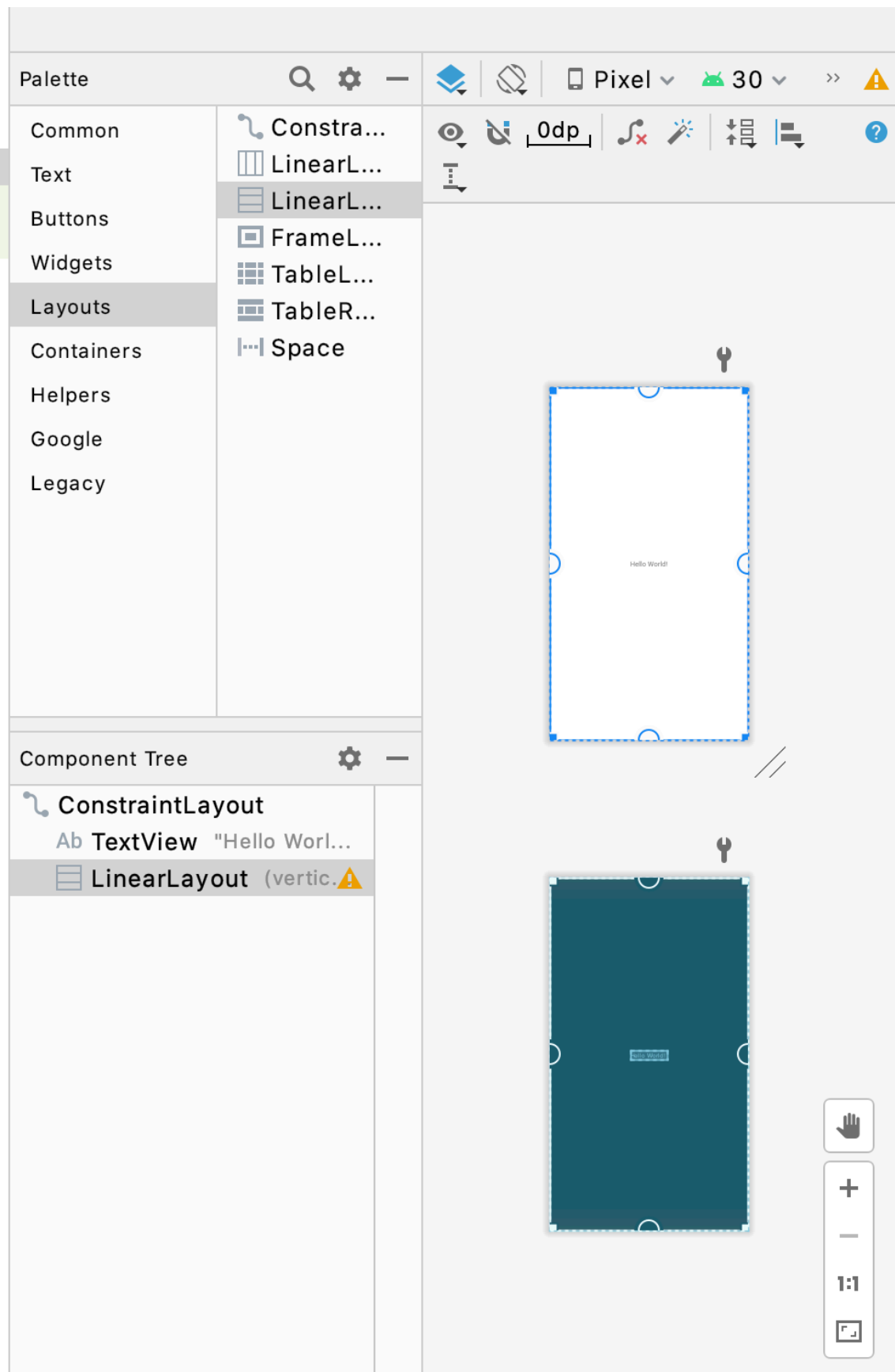
 Your app will run on < 1% of devices.
[Help me choose](#)

☐ Use legacy android.support libraries 

 project location should not contain whitespace, as this can cause problems with the NDK tools.

Cancel Previous Next Finish

Trazar un LinearLayout Vertical



Completar el siguiente código en activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    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:orientation="vertical"
        tools:layout_editor_absoluteX="51dp"
        tools:layout_editor_absoluteY="24dp">

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Agenda de contactos"
            android:gravity="center"
            android:textSize="25sp"
            android:textStyle="bold"
            android:layout_marginBottom="10dp"
            />

        <EditText
            android:id="@+id/etEmail"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Email"
            android:gravity="center"
            android:inputType="textEmailAddress"
            android:layout_marginBottom="7dp"
            />

        <EditText
            android:id="@+id/etPassword"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Password"
            android:gravity="center"
            android:inputType="textPassword"
            android:layout_marginBottom="7dp"
            />

        <Button
            android:id="@+id/btnlogin"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Acceder"/>

    </LinearLayout>

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

Abrir línea de comandos

Ejecutar IPConfig

Verificar Dirección IPv4

IP para acceder al Webservice

Verificar la comunicación, llendo al emulador, ir flecha atrás y poner la dirección en el navegador Chrome.

Añadir la librería de Volley en Gradle (Module).

```
dependencies {  
    implementation 'androidx.appcompat:appcompat:1.2.0'  
    implementation 'com.google.android.material:material:1.2.1'  
    implementation 'androidx.constraintlayout:constraintlayout:2.0.4'  
    testImplementation 'junit:junit:4.+'  
    androidTestImplementation 'androidx.test.ext:junit:1.1.2'  
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.3.0'  
    implementation 'com.google.android.material:material:1.0.0'  
    implementation 'com.android.volley:volley:1.1.1'  
}
```

Dar clic a Sync now (sincronizr ahora)

Escribir el siguiente código en MainActivity.java

```
package com.example.agendaappws;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.app.AlertDialog;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.Toast;  
  
import com.android.volley.Request;  
import com.android.volley.RequestQueue;  
import com.android.volley.Response;  
import com.android.volley.VolleyError;  
import com.android.volley.toolbox.StringRequest;  
import com.android.volley.toolbox.Volley;  
  
import org.json.JSONException;  
import org.json.JSONObject;  
  
import java.util.HashMap;  
import java.util.Map;
```

Elaborada por: Dra. Raquel Ochoa Ornelas

```

public class MainActivity extends AppCompatActivity {

    //Objetos de la vista
    EditText etEmail, etPassword;
    Button btnLogin;
    String e,p;

    //Animación de carga del webservice para ver que hace la aplicación

    ProgressDialog progressDialog;

    //Cuando mandamos petición, se debe enviar string con muchos elementos para
    ejecutar volley

    RequestQueue requestQueue;

    String httpURI="https://moveint.com/agenda/usuario.php";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        //Enlazamos controladores
        etEmail=(EditText) findViewById(R.id.etEmail);
        etPassword=(EditText) findViewById(R.id.etPassword);
        btnLogin=(Button) findViewById(R.id.btnlogin);
        //Inicializar o enlazar a requestQueue y el progressDialog

        requestQueue= Volley.newRequestQueue(MainActivity.this);
        //Indicar dónde se ejecutará progressdialog
        progressDialog=new ProgressDialog(MainActivity.this);
        //Falta el oyente del botón login armar la cadena que se ejecuta en
        webservice y ls acciones
        //que va realizar cuando tenga éxito o no tenga éxito.

        //Oyente para el botón
        btnLogin.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Login();
            }
        });
    }
    private void Login(){
        e= etEmail.getText().toString();
        p= etPassword.getText().toString();
        //No podemos mandar valores nulos al servidor
        if (e.isEmpty() || p.isEmpty()){
            Toast.makeText(getApplicationContext(), "Debes introducir los datos
en los dos campos", Toast.LENGTH_LONG).show();
        }
        else {
            //Mostramos el progressDialog
            progressDialog.setMessage("Procesando...");
            progressDialog.show();

            //Creación de la cadena a ejecutar en el web service mediante Volley
            //Enviar por petición POST
            //Implementar los métodos del Request con Option+Enter

```

```

        //Cambiar el Response por serverResponse en onResponse
        //Agregar new Response.ErrorListener para manejo de errores y un
        Toast ; y mapeo a web service
        StringRequest stringRequest=new StringRequest(Request.Method.POST,
        httpURI, new Response.Listener<String>() {
            @Override
            public void onResponse(String serverResponse) {
                //Quitar el progressDialog porque ya recibimos la respuesta y
                ocultamos el progressDialog
                progressDialog.dismiss();
                //Manejo de errores con respecto a la respuesta que
                obtengamos del webservice que es un objeto de tipo Json
                try {
                    //Crear objeto Json el cual va a obtener los valores que
                    tenga de parte del webservice en ejecución
                    JSONObject obj = new JSONObject(serverResponse);
                    //Interpretar los valores que retorne por medio de obj
                    Boolean error=obj.getBoolean("error");
                    //Obtener el mensaje
                    String mensaje=obj.getString("mensaje");
                    //Interpretar los dos valores
                    if (error)//Datos incorrectos
                        Toast.makeText(getApplicationContext(),
mensaje,Toast.LENGTH_LONG).show();
                    else{
                        Toast.makeText(getApplicationContext(), "Acceso
correcto",Toast.LENGTH_LONG).show();
                        //Presentar otra activity....
                    }
                } catch (JSONException e) {
                    e.printStackTrace();
                }
            }
        }, new Response.ErrorListener() {
            @Override
            public void onErrorResponse(VolleyError error) {
                //Aunque marque error detener progressDialog
                progressDialog.dismiss();
                //Mostrar el error de Volley exacto a través de la librería
                Toast.makeText(getApplicationContext(),
error.toString(),Toast.LENGTH_LONG).show();
            }
        }){
        protected Map<String,String> getParams(){
            Map<String, String> parametros=new HashMap<>();
            //Parámetros que se esperan en el webservice
            parametros.put("email",e);
            parametros.put("password",p);
            parametros.put("opcion","login");
            return parametros;
        }

    };

    //requestQueue ejecute la cadena
    requestQueue.add(stringRequest);
}

```

```
}  
}
```

Colocar los permisos de Internet en manifest

```
<?xml version="1.0" encoding="utf-8"?>  
<manifest xmlns:android="http://schemas.android.com/apk/res/android"  
    package="com.example.agendaappws" >  
    <uses-permission android:name="android.permission.INTERNET"/>  
    <application  
        android:allowBackup="true"  
        android:icon="@mipmap/ic_launcher"  
        android:label="@string/app_name"  
        android:roundIcon="@mipmap/ic_launcher_round"  
        android:supportRtl="true"  
        android:theme="@style/Theme.AgendaAppWS" >  
        <activity android:name=".MainActivity" >  
            <intent-filter>  
                <action android:name="android.intent.action.MAIN" />  
  
                <category android:name="android.intent.category.LAUNCHER" />  
            </intent-filter>  
        </activity>  
    </application>  
</manifest>
```

Corrida:

