



Programador de Aplicaciones para Moviles

APPCONVERT

GRUPO 11

Integrantes:

Godoy, Gustavo
Rodriguez, Milagros
Toso, Fabricio

ÍNDICE



Contenido

- Página 3: Introducción
- Página 4: Objetivos
- Página 5: Login
- Página 6-7-9: Conexión con base de datos externa
- Página 10: Icono de aplicación
- Página 11 y 13: Menú OverFlow
- Página 12: Activity inicio
- Página 14-15-16: Activity conversiones

INTRODUCCION

La conversión de unidades de medida es una tarea común en numerosos campos, desde la ingeniería y la física hasta la cocina y la vida cotidiana. Ya sea que estemos calculando distancias, pesos, temperaturas o volúmenes, la capacidad de convertir rápidamente entre diferentes sistemas de unidades es esencial para realizar cálculos precisos y comprender mejor el mundo que nos rodea.

En este documento, presentaremos una aplicación convertidora de unidades de medida que brinda a los usuarios una solución práctica y eficiente para convertir fácilmente entre diversas unidades. Esta aplicación ha sido desarrollada con el objetivo de simplificar el proceso de conversión, proporcionando una interfaz intuitiva y herramientas versátiles para satisfacer las necesidades de usuarios de todos los niveles de experiencia.

A lo largo de esta documentación, exploraremos las características y funcionalidades clave de nuestra aplicación convertidora de unidades. Discutiremos cómo utilizarla para convertir distintos tipos de unidades, cómo personalizar las preferencias y ajustes según las necesidades individuales, así como también proporcionaremos ejemplos concretos y guías paso a paso para aprovechar al máximo esta herramienta.

Ya sea que esté realizando cálculos científicos complejos o simplemente necesite convertir unidades básicas en su vida diaria, esta aplicación le permitirá ahorrar tiempo y evitar errores en el proceso de conversión de medidas. Esperamos que esta documentación sea una guía útil para aprovechar todas las capacidades de nuestra aplicación convertidora de unidades y facilitar sus tareas de conversión de manera eficiente y precisa.

OBJETIVO

El objetivo de nuestra aplicación convertidora de unidades de medida es proporcionar a los usuarios una herramienta fácil de usar y confiable que les permita convertir rápida y precisamente entre diferentes sistemas de unidades. Nuestro objetivo principal es simplificar el proceso de conversión de unidades y eliminar la necesidad de realizar cálculos manuales propensos a errores.

Algunos objetivos específicos de nuestra aplicación incluyen:

1- Facilitar la conversión de unidades: Busca simplificar el proceso de conversión de unidades, brindando a los usuarios una interfaz intuitiva y fácil de entender. El objetivo es permitir a los usuarios realizar conversiones rápidas y precisas sin tener que lidiar con fórmulas complejas o referencias externas.

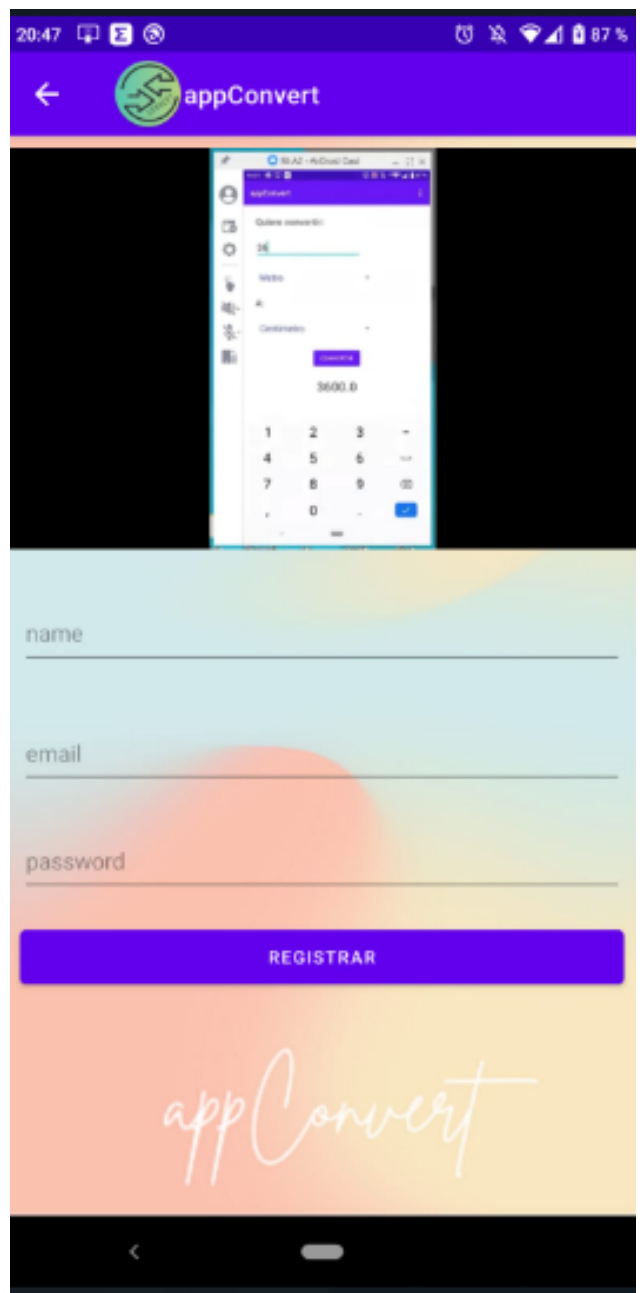
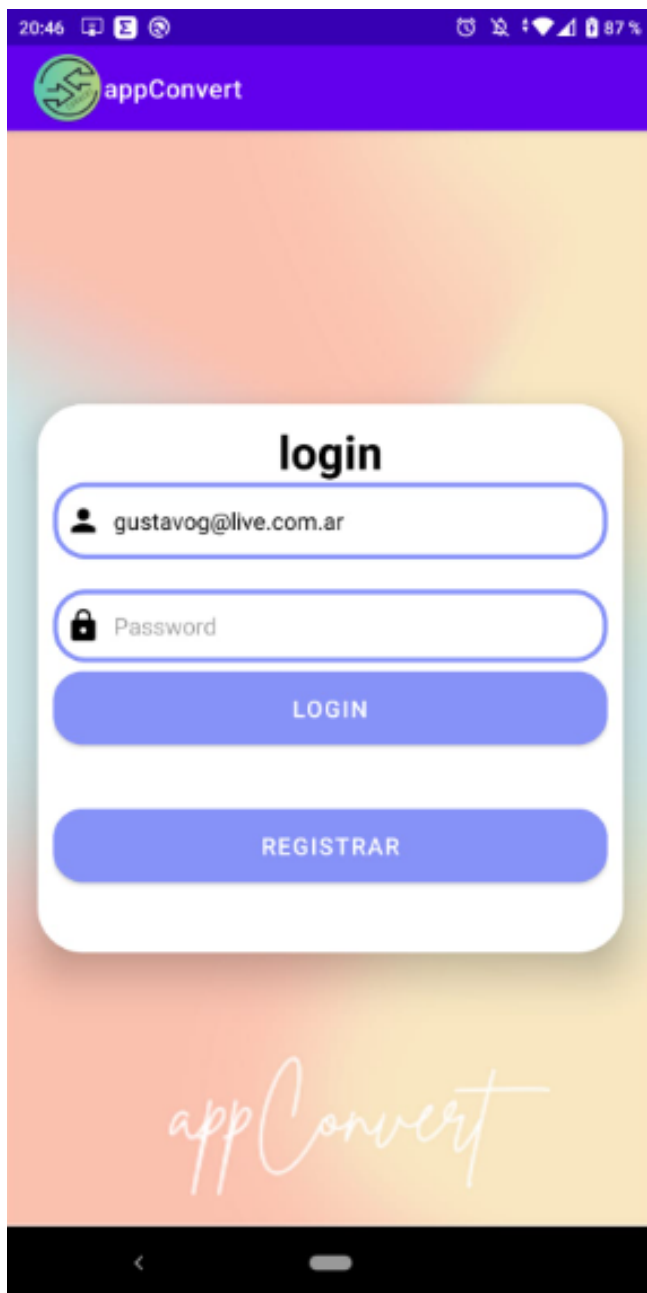
2- Cubrir una amplia gama de unidades: Nos esforzamos por incluir una amplia variedad de unidades de medida en nuestra aplicación. Desde unidades básicas como metros y kilogramos, hasta unidades más especializadas, nuestro objetivo es abarcar tantas unidades comunes como sea posible.

3- Proporcionar resultados precisos y confiables: La precisión es fundamental en la conversión de unidades de medida. Nos comprometemos a garantizar que nuestra aplicación proporcione resultados precisos y confiables en todas las conversiones realizadas.

En resumen, el objetivo de nuestra aplicación convertidora de unidades de medida es ofrecer a los usuarios una herramienta confiable, fácil de usar y personalizable que simplifique el proceso de conversión de unidades, permitiéndoles realizar conversiones precisas en diferentes campos y sistemas de unidades.

LOGIN

La aplicación parte desde un login en donde da la posibilidad de iniciar sesion si es que ya tenemos cuenta o, presionar el boton "registrar" ir a la segunda activity para cargar los datos necesarios para el registro



Coneccion con base de datos externa

La aplicación esta realizada con base de datos MySql y dos archivos en Php.

Para poder validar y registrar un registro con una base de datos en un Hosting alojado en la web, primero debemos dar los siguientes permisos:

- En el archivo AndroidManifest antes de las llaves de <application> se debe agregar el siguiente código:

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    package="com.planidear.appconvert">
```

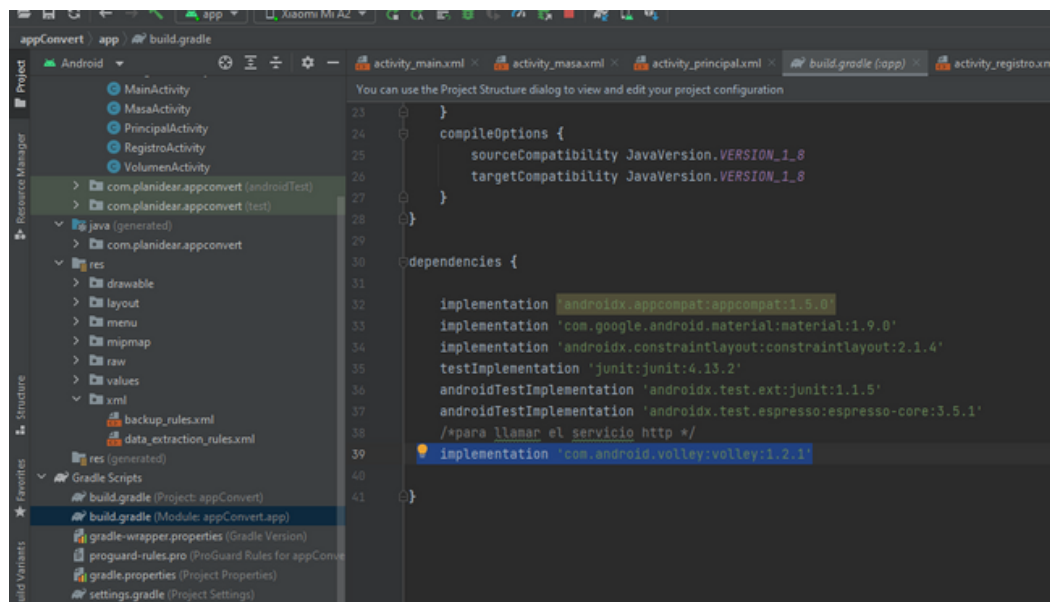
```
<uses-permission android:name="android.permission.INTERNET" />
```

```
<application
```

de esta forma daremos autorización a android para conectarse con Internet.

Luego dentro de la carpeta build.gradle, dentro de las llaves dependencia agregar el siguiente código:

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



```

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

    email = findViewById(R.id.username);
    password = findViewById(R.id.password);
    loginButton = findViewById(R.id.loginButton);
    loginButton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) { validarUsuario(url1); }
    });
}

```

El metodo anterior llama a validarUsuario, éste permite brindar una serie de mensajes informativos al usuario como "Ingresaste" o "Error al querer ingresar" Además guarda el mail ingresado.

```

private void validarUsuario(String URL){
    StringRequest stringRequest = new StringRequest(Request.Method.POST, URL, new Response.Listener<String> () {
        3 usages
    @Override
    public void onResponse(String response){
        if(response.equals("ingresaste")){
            Toast.makeText(context: MainActivity.this, text: "ingresaste", Toast.LENGTH_LONG).show();
            Intent intent = new Intent(getApplicationContext(), PrincipalActivity.class);
            startActivity(intent);
            EditText editText = (EditText) findViewById(R.id.username);
            String message = editText.getText().toString();
            intent.putExtra(EXTRA_MESSAGE, message);
            startActivity(intent);

            //Metodo para guardar el email
            SharedPreferences preferencias = getSharedPreferences( name: "datos", Context.MODE_PRIVATE);
            //Editor es para guardar un archivo
            SharedPreferences.Editor obj_editor = preferencias.edit();
            obj_editor.putString( s: "mail", email.getText().toString());
            //confirmar lo que declaramos arriba se va a guardar
            obj_editor.commit();
            finish();
        }else{
            Toast.makeText(context: MainActivity.this, text: "Error al querer ingresar", Toast.LENGTH_LONG).show();
            /* Intent intent = new Intent(getApplicationContext(), PrincipalActivity.class);
            startActivity(intent);*/
        }
    }
}

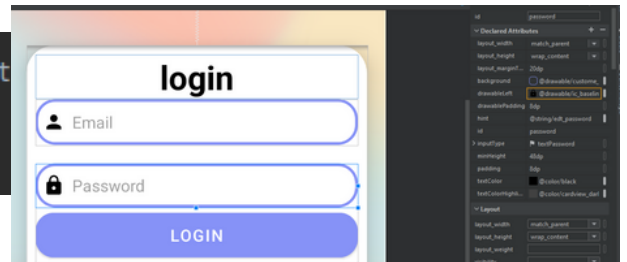
```

Activity Login

Dentro del MainActivity.java

Declaramos los EditText y boton:

```
public class MainActivity extends AppCompatActivity {
    private EditText email;
    private EditText password;
    Button loginButton;
```



Además, se declara el url del código Php

```
//conexion con base de datos:
String url1 = "https://planidear.com.ar/ConexionAndroid/validar_usuario.php";
```

El codigo de la pagina web:

```
<?php
include 'conexion.php';
$susu_usuario=$_POST['email'];
$susu_password=$_POST['password'];
//susu_usuario="email";
//susu_password="1234";

$sentencia=$conexion->prepare("SELECT * FROM usersandroid WHERE email=? AND password=?");
$sentencia->bind_param('ss',$susu_usuario,$susu_password);
$sentencia->execute();

$resultado = $sentencia->get_result();
if ($fila = $resultado->fetch_assoc()) {
    //echo json_encode($fila,JSON_UNESCAPED_UNICODE);
    echo "ingresaste";
}else{
    echo "ERROR";
}
$sentencia->close();
$conexion->close();
```

En el mismo archivo MainActivity, escribimos el siguiente código:

```
private void validarUsuario(String URL){
    StringRequest stringRequest = new StringRequest(Request.Method.POST, URL, new Response.Listener<String> () {
        @Override
        public void onResponse(String response){
            if(response.equals("ingresaste")){
                Toast.makeText(context, MainActivity.this, text "ingresaste", Toast.LENGTH_LONG).show();
                Intent intent = new Intent(getApplicationContext(), PrincipalActivity.class);
                startActivity(intent);
                EditText editText = (EditText) findViewById(R.id.username);
                String message = editText.getText().toString();
                intent.putExtra(EXTRA_MESSAGE, message);
                startActivity(intent);

                //Metodo para guardar el email
                SharedPreferences preferencias = getSharedPreferences( name: "datos", Context.MODE_PRIVATE);
                //Editor es para guardar un archivo
                SharedPreferences.Editor obj_editor = preferencias.edit();
                obj_editor.putString( "email", email.getText().toString());
                //confirmar lo que declaramos arriba se va a guardar
                obj_editor.commit();
                finish();
            }else{
                Toast.makeText(context, MainActivity.this, text "Error al querer ingresar", Toast.LENGTH_LONG).show();
            }
        }
    }, new Response.ErrorListener(){
        @Override
        public void onErrorResponse(VolleyError error) {
            Toast.makeText(context, MainActivity.this, error.toString(), Toast.LENGTH_LONG).show();
        }
    });
```

De esta forma, luego de hacer el request, envía los datos a través del Post a la URL. Luego, a través del Response recibe la respuesta y la compara dentro del bucle if la palabra true del condicional archivo Php, en caso de ser false, avisa con el Toast que el usuario es incorrecto.

Activity Login

Como se dijo anteriormente, los datos del editext usuario como password se envía través de la función `getParams()`:

```

    }{
        @Override
        protected Map<String, String> getParams() throws AuthFailureError {
            Map<String, String> params = new HashMap<>();
            params.put( "email", email.getText().toString().trim());
            params.put( "password", password.getText().toString().trim());
            return params;
        }
    };
    RequestQueue requestQueue = Volley.newRequestQueue( context: MainActivity.this);
    requestQueue.add(stringRequest);
}

```

Activity Registro

En caso de no tener un usuario, con el botón Registrar, nos llevara al Activity Registro. Este tiene los campos de usuarios, email y password. Además, tiene un video explicativo en su cabecera.

Para ver el video:

```

//Inicio video
VideoView videoView = findViewById(R.id.videoView);
String videop = "android.resource://" + getPackageName() + "/" + R.raw.videol;
Uri uri = Uri.parse(videop);
videoView.setVideoURI(uri);
MediaController mediaController = new MediaController( context: this);
videoView.setMediaController(mediaController);
mediaController.setAnchorView(videoView);
//Final video

```

Enviar datos al url:

```

private void registroUsuario(String URL) {
    StringRequest stringRequest = new StringRequest(Request.Method.POST, URL, new Response.Listener<String> () {
        @Override
        public void onResponse(String response){
            if(response.equals("Registrado")){
                Toast.makeText( context: RegistroActivity.this, text: "Registrado", Toast.LENGTH_LONG).show();
                Intent intent = new Intent(getApplicationContext(), MainActivity.class);
                startActivity(intent);
                EditText editText = (EditText) findViewById(R.id.username);
                String message = editText.getText().toString();
                intent.putExtra(EXTRA_MESSAGE, message);
                startActivity(intent);
            }else{
                Toast.makeText( context: RegistroActivity.this, text: "Error al querer ingresar", Toast.LENGTH_LONG).show();
                // Intent intent = new Intent(getApplicationContext(), PrincipalActivity.class);
                startActivity(intent);
            }
        }
    });
}

```

Codigo de el url (Php):

```

<?php
include 'conexion.php';
$susu_name=$_POST['name'];
$susu_email=$_POST['email'];
$susu_password=$_POST['password'];

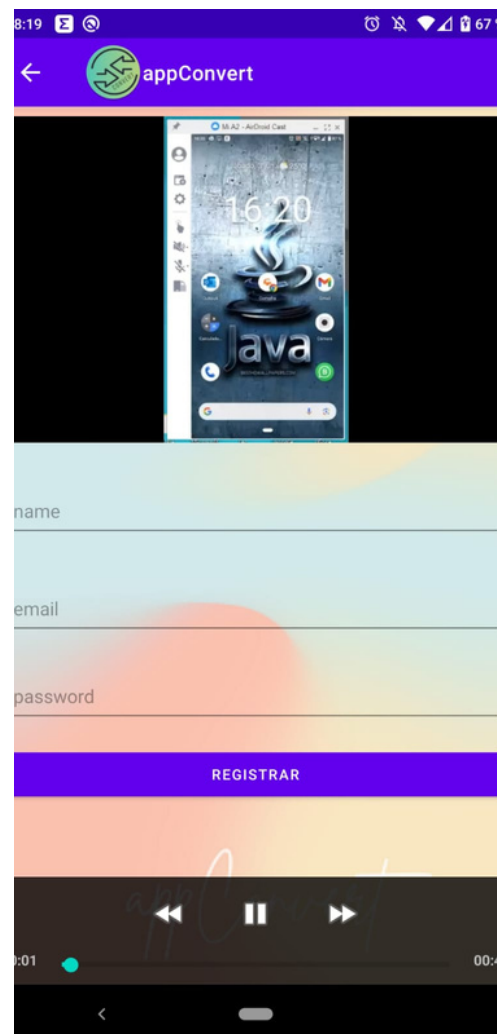
$sentencia=$conexion->prepare("INSERT INTO 'usersandroid' ('id', 'name', 'email', 'password', 'phone', 'rol')
VALUES (NULL, '$susu_name', '$susu_email', '$susu_password', '', 'user');");

$sentencia->bind_param('ss',$susu_name,$susu_email,$susu_password);
$sentencia->execute();

$resultado = $sentencia->get_result();
if ($resultado > 0) {
    //echo json_encode($fila,JSON_UNESCAPED_UNICODE);
    $titulo="Usted se ha registrado correctamente";
    $mensaje="Name: ".$susu_name." - Email: ".$susu_email." - password: ".$susu_password." - Puede disfrutar de nuestra aplicacion ";
    $para=$susu_email;

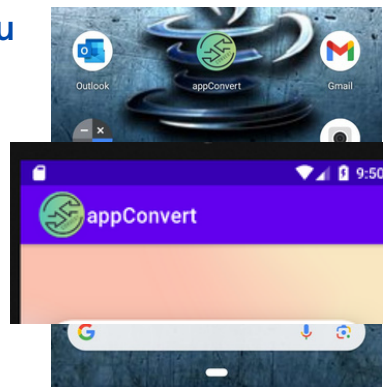
    $cabeceras = 'From: gustavogustavog@polandear.com.ar';
    $enviado = mail($para, $titulo, $mensaje,$cabeceras);
    echo "Registrado";
}else{
    //
}

```



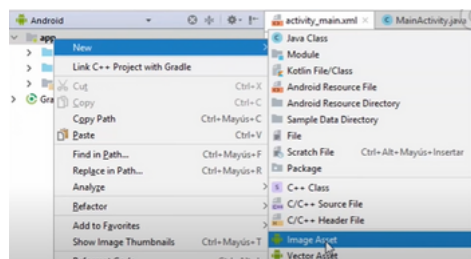
El código de php tiene la particularidad, de luego cargar el correo, la misma url le enviara un email confirmando el registro.

Icono de la aplicación y menú



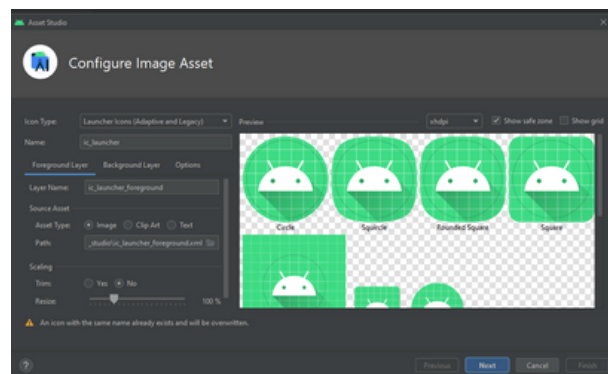
Android studio por defecto nos da el logo de Android, para cambiar el icono y agregarlo en el menú, se debe agregar una imagen preferentemente .png y todo el nombre de la imagen este completamente en minúscula.

El primer paso debemos ir la carpeta app, botón derecho, New y la opción Image Asset:

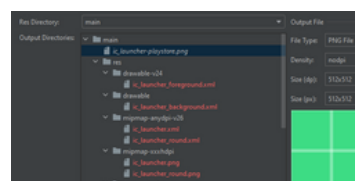


Aparecerá la siguiente pantalla:
Trabajaremos con la primera opción de

- Icono Type: Launcher Icons (Legacy only)
- En la opción de Name: dejar el nombre que tiene: ic_launcher
- Dentro la carpeta Paht: buscar y seleccionar la imagen.
- Las advertencias que aparezcan, se debe aceptar y seguir.



Una vez que se cambió el icono de la aplicación. Para poder cambiar el icono del menú:

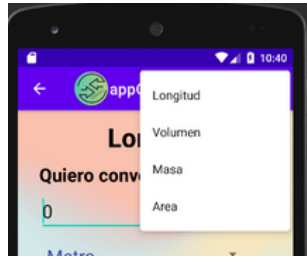


Dentro de MainActivity.java despues del onCreate agregar el siguiente codigo:

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

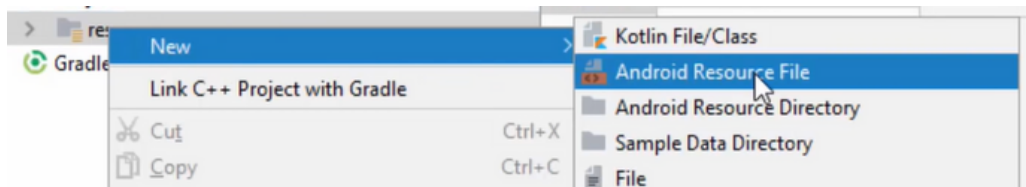
    //para que tenga el logo en el action bar
    getSupportActionBar().setDisplayHomeAsUpEnabled(true);
    getSupportActionBar().setIcon(R.mipmap.ic_launcher_round);
}
```

Menu OverFlow

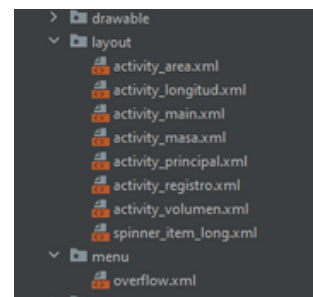
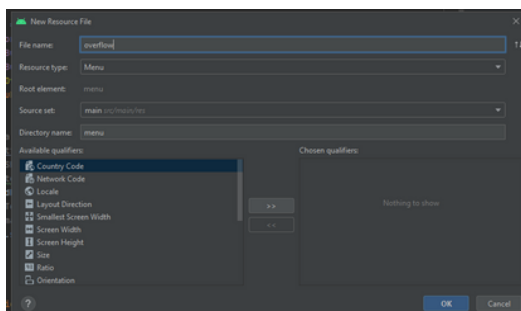


Primero debemos crear un activity nuevo:

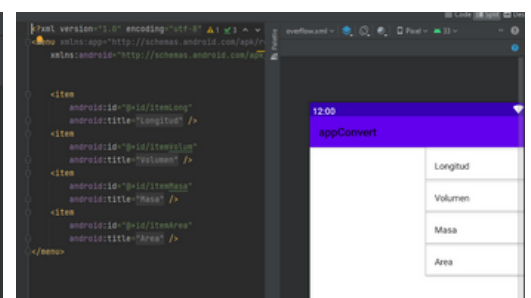
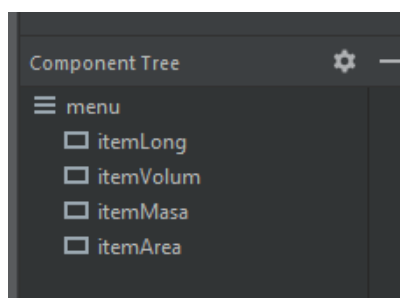
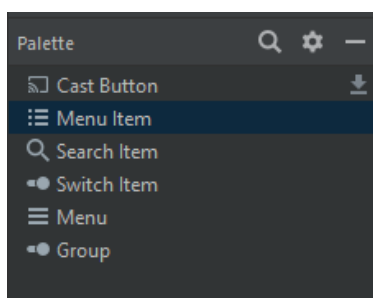
- Clic derecho sobre la carpeta res, New --> Android Resource File



- En File Name: agregar el nombre que nosotros necesitemos, todo en minúscula.
- En segundo lugar, selecciona Menu en Resource type.



- Dentro del archivo .xml agregar los Menu Item que necesitamos que se vean:



- Para poder mostrar el menu y ver el efecto de los tres puntos, dentro de cada Activity fuera del bucle onCreate agregar el siguiente código:

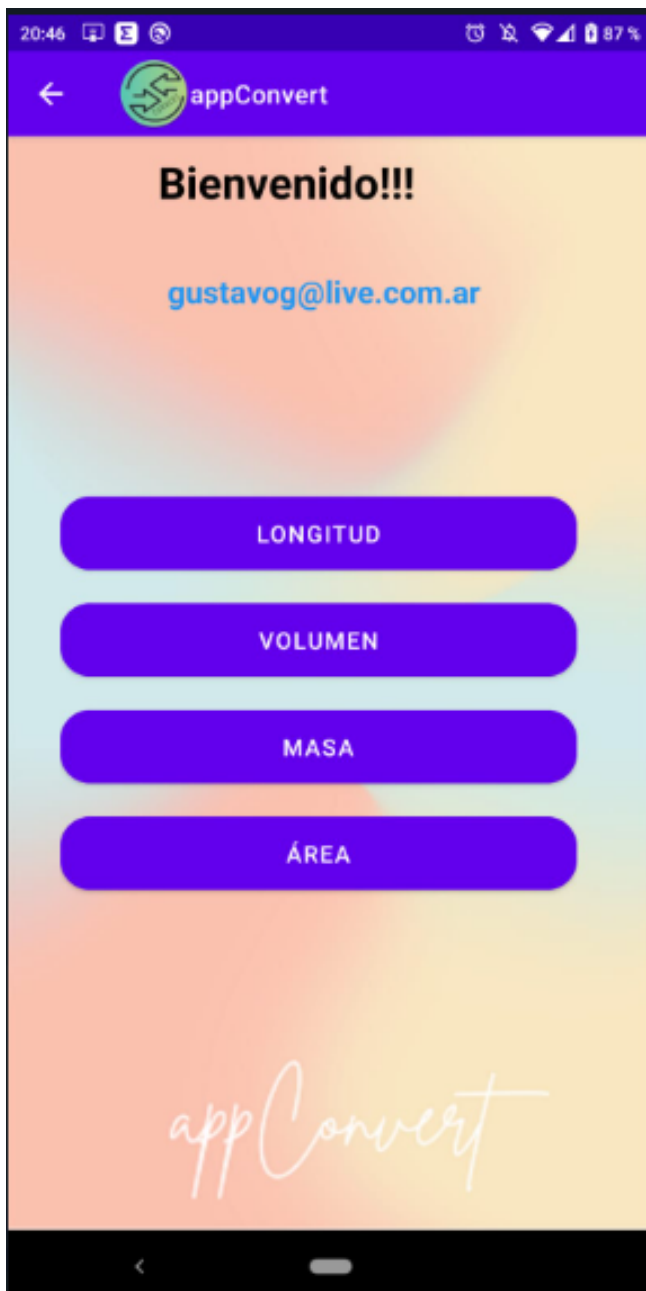
```
//para mostrar y ocultar el menu
public boolean onCreateOptionsMenu(Menu menu){
    getMenuInflater().inflate(R.menu.overflow, menu);
    return true;
}
```

```
//Metodo para asignar las funciones acorrespondiente a las opciones
public boolean onOptionsItemSelected(MenuItem item){
    int id = item.getItemId();//para recuperar el item que se esta seleccionando
    if(id == R.id.itemLong){
        Toast.makeText( context, "Longitud", Toast.LENGTH_LONG).show();
        //para abrir el activity
        Intent intent = new Intent(getApplicationContext(), LongitudActivity.class);
        startActivity(intent);
    }else if(id == R.id.itemMasa){
```

ELECCIÓN DE UNIDAD A CONVERTIR

Una vez logueado, el siguiente activity toma el correo ingresado y da un mensaje de bienvenida, de esta manera se vinculan entre ambas.

Además se puede seleccionar la unidad de medida a convertir



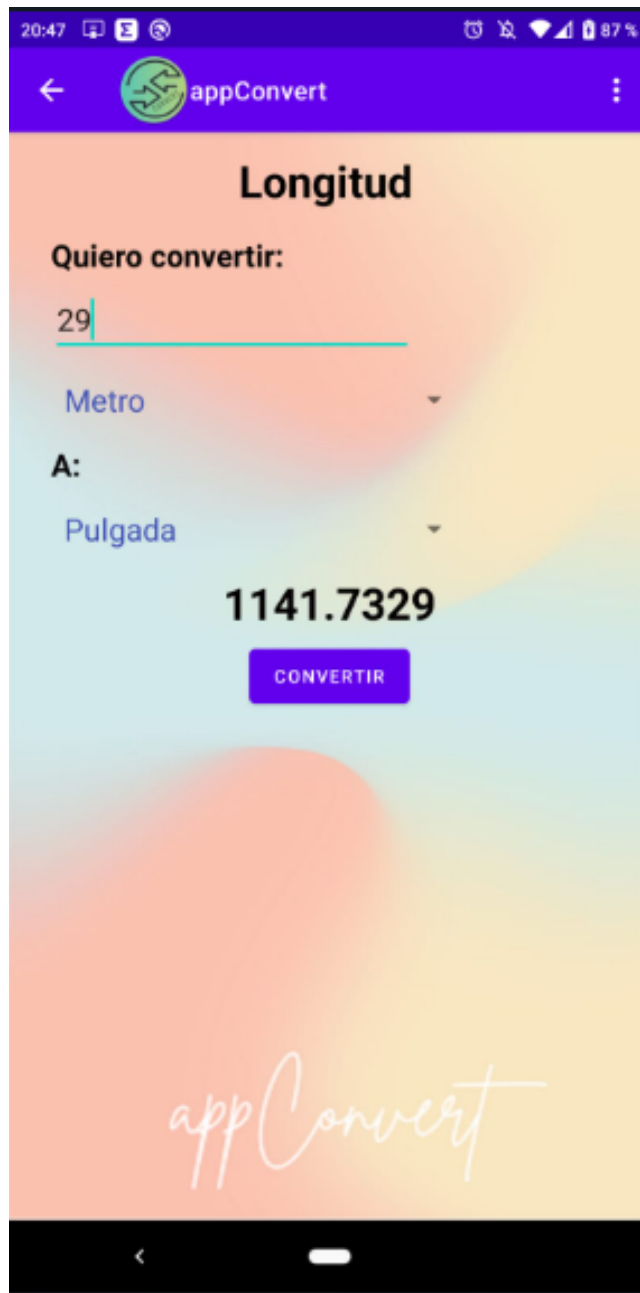
El menú esta creado con el siguiente metodo

```
public boolean onOptionsItemSelected(MenuItem item){
    int id = item.getItemId();//para recuperar el item que se esta seleccionando
    if(id == R.id.itemLong){
        Toast.makeText( context: this, text: "Longitud", Toast.LENGTH_LONG).show();
        //para abrir el activity
        Intent intent = new Intent(getApplicationContext(), LongitudActivity.class);
        startActivity(intent);
    }else if(id == R.id.itemMasa){
        Toast.makeText( context: this, text: "Masa", Toast.LENGTH_LONG).show();
    }else if(id == R.id.itemVolum){
        Toast.makeText( context: this, text: "Volumen", Toast.LENGTH_LONG).show();
    }else if(id == R.id.itemArea){
        Toast.makeText( context: this, text: "Masa", Toast.LENGTH_LONG).show();
    }

    return super.onOptionsItemSelected(item);
}
```

CONVERSION DE MEDIDAS

Una vez elegida la unidad de medida, mediante el método Convertir se plantea la logica que permite convertir las distintas unidades de medida pidiendo al usuario que primero ingrese las medidas expresadas en la unidad correspondiente. En el siguiente ejemplo elegimos "Longitud"



En principio se piden los valores a convertir:

```
public void Convertir(View view) {
    //para recuperra el valor
    String valor_string = etValor.getText().toString();

    //cambiar los valores a entero
    Double valor_int = Double.parseDouble(valor_string);

    String seleccion1 = spinner1.getSelectedItem().toString();
    String seleccion2 = spinner2.getSelectedItem().toString();
    //seleccion1.equals compara cadena de texto

    if (valor_int ==0 ) {
        Toast.makeText(LongitudActivity.this, "Debes ingresar un numero", Toast.LENGTH_LONG).show();
    }
}
```

Y segun la unidad de medida elegida desde la que se quiere partir, metros en el caso del ejemplo, a la que queremos convertir (Pulgadas) en el siguiente fragmento de codigo se muestra como valida que se cumplan las condiciones para realizar el calculo correspondiente y devolver el resultado

```
Double ConvMetro = valor_int * 1000;
String resultado = String.valueOf(ConvMetro);
tvResul.setText(resultado);
} else if (seleccion1.equals("Metro") & seleccion2.equals("Micrometro")) {
    Double ConvMetro = valor_int * 1000000;
    String resultado = String.valueOf(ConvMetro);
    tvResul.setText(resultado);
} else if (seleccion1.equals("Metro") & seleccion2.equals("Metro")) {
    Double ConvMetro = valor_int * 1;
    String resultado = String.valueOf(ConvMetro);
    tvResul.setText(resultado);
} else if (seleccion1.equals("Metro") & seleccion2.equals("Pulgada")) {
    Double ConvMetro = valor_int * 39.3701;
    String resultado = String.valueOf(ConvMetro);
    tvResul.setText(resultado);
}
```

Si existiese algun error devuelve un mensaje

```
else {  
    Toast.makeText(context: LongitudActivity.this, text: "ERROR DE VALOR", Toast.LENGTH_LONG).show();  
}  
  
}
```