

Evidencia Día 3 Semana 11

Tarea en clases: Crear interfaz gráfica y método para capturar una foto.

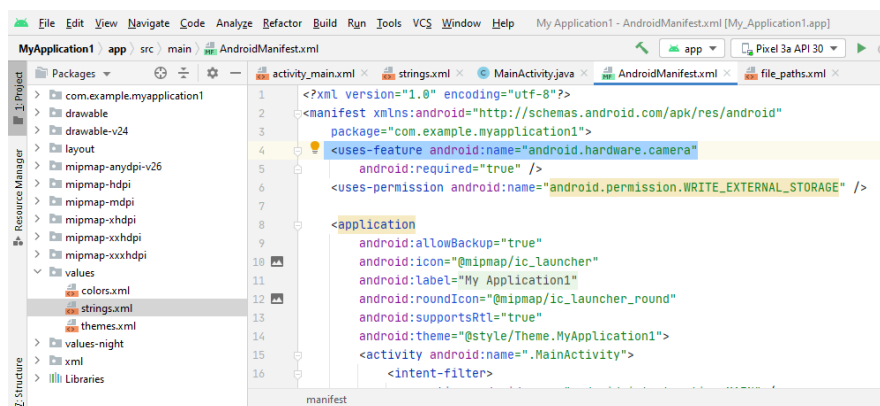
Para la ejecución de este ejercicio, se revisó el siguiente video y posterior documento en referencia.

-Curso Android desde cero #46 | Cámara de los dispositivos Android - Tomar fotografías

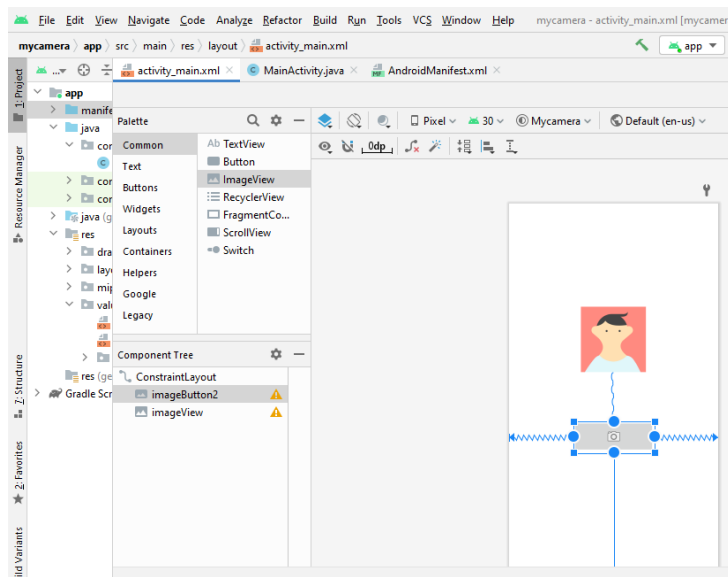
<https://www.youtube.com/watch?v=qYeVXGNG-b4>.

-<https://developer.android.com/training/camera/photobasics>

Para anunciar que la aplicación depende de tener una cámara, se debe colocar una etiqueta `<uses-feature>` en el archivo de manifiesto:



En la parte Gráfica se arrastran 2 componentes un imageView y un ImageButton el cual tendrá asociado el método capturar o tomar foto y se realizan los constrains a los componentes.



En el MainActivity digitamos el siguiente código que nos va permitir capturar una foto y crear el archivo de la foto (Bitmap).

```
package com.example.mycamera;
```

```

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.content.Intent;
import android.graphics.Bitmap;
import android.os.Environment;
import android.provider.MediaStore;
import android.view.View;
import android.widget.ImageButton;
import android.widget.ImageView;
import android.widget.Toast;

import java.io.File;
import java.io.FileOutputStream;
import java.text.SimpleDateFormat;
import java.util.Date;

public class MainActivity extends AppCompatActivity {

    ImageButton btncamara;
    ImageView img;

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

        btncamara = findViewById(R.id.imageButton2);
        img = findViewById(R.id.imageView);

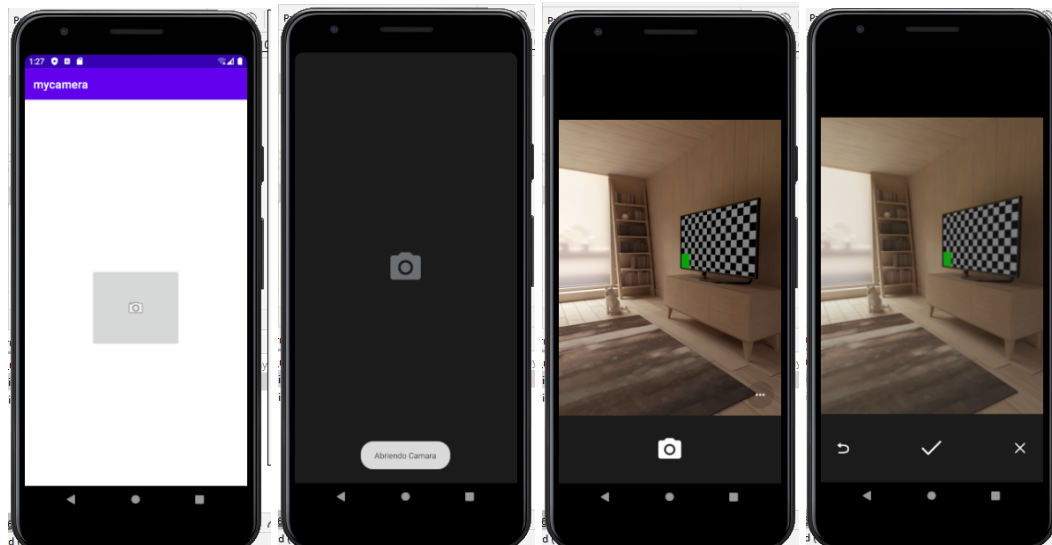
        btncamara.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick (View view) {
                abrirCamara();
            }
        });
    }

    private void abrirCamara() {
        Toast.makeText(this, "entro abrir camara", Toast.LENGTH_LONG).show();
        Intent intent = new Intent (MediaStore.ACTION_IMAGE_CAPTURE);
        //if (intent.resolveActivity(getPackageManager()) !=null){
            startActivityForResult(intent, 1);
        //}
    }

    protected void onActivityResult (int requestCode, int resultCode, Intent data) {
        super.onActivityResult(requestCode, resultCode, data);
        if (requestCode == 1 && resultCode == RESULT_OK) {
            Bundle extras = data.getExtras();
            Bitmap imgBitmap = (Bitmap) extras.get("data");
            img.setImageBitmap (imgBitmap);
            saveTempBitmap(imgBitmap);
        }
    }
}

```

Luego nos vamos a la Activity que estamos trabajando, pinchamos el ImageButton, atributos, OnClick y asociamos el método. Ejecutamos el emulador:



Reflexión: La función era bastante difícil, sobretodo al momento de buscar información al respecto ya que era muy complejo de entender el tema de los permisos, para poder ejecutar el código.