**Cours : Utiliser la caméra d’un smartphone dans une application Android**

**🎯 Objectif :**

* Ouvrir la **caméra native** du téléphone
* Capturer une image
* Afficher l’image capturée dans un ImageView
* Sauvegarder éventuellement l’image dans un fichier JPEG

**1. Configuration du projet Android**

 **Fichier AndroidManifest.xml :**

Ajoutez les permissions nécessaires :

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

<uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE" />

<uses-permission android:name="android.permission.READ\_EXTERNAL\_STORAGE" />

<application ...>

 <provider

 android:name="androidx.core.content.FileProvider"

 android:authorities="${applicationId}.fileprovider"

 android:exported="false"

 android:grantUriPermissions="true">

 <meta-data

 android:name="android.support.FILE\_PROVIDER\_PATHS"

 android:resource="@xml/file\_paths" />

 </provider>

</application>

 **Créez un fichier res/xml/file\_paths.xml :**

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

<paths xmlns:android="http://schemas.android.com/apk/res/android">

 <external-path name="external\_files" path="." />

</paths>

**2. Layout XML (res/layout/activity\_main.xml)**

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

 android:orientation="vertical" android:layout\_width="match\_parent"

 android:layout\_height="match\_parent" android:padding="16dp">

 <Button

 android:id="@+id/btnCapture"

 android:layout\_width="match\_parent"

 android:layout\_height="wrap\_content"

 android:text="Prendre une photo" />

 <ImageView

 android:id="@+id/imageView"

 android:layout\_width="match\_parent"

 android:layout\_height="400dp"

 android:scaleType="centerCrop"

 android:layout\_marginTop="16dp"/>

</LinearLayout>

**3. Code Java (MainActivity.java)**

public class MainActivity extends AppCompatActivity {

 // Code de demande d'autorisation de la caméra

 private static final int REQUEST\_IMAGE\_CAPTURE = 1;

 private static final int REQUEST\_CAMERA\_PERMISSION = 100;

 // Composants de l'interface

 private ImageView imageView; // Pour afficher la photo prise

 private Uri photoUri; // URI du fichier image temporaire

 private File photoFile; // Fichier temporaire où l'image sera stockée

 @Override

 protected void onCreate(Bundle savedInstanceState) {

 super.onCreate(savedInstanceState);

 setContentView(R.layout.activity\_main); // Charge le layout XML

 Button btnCapture = findViewById(R.id.btnCapture); // Récupère le bouton

 imageView = findViewById(R.id.imageView); // Récupère l'imageView

 // Déclenche l'action quand l'utilisateur clique sur "Prendre une photo"

 btnCapture.setOnClickListener(v -> {

 // Vérifie si la permission caméra est accordée

 if (ContextCompat.checkSelfPermission(this, Manifest.permission.CAMERA)

 != PackageManager.PERMISSION\_GRANTED) {

 // Si non, demande la permission

 ActivityCompat.requestPermissions(this,

 new String[]{Manifest.permission.CAMERA}, REQUEST\_CAMERA\_PERMISSION);

 } else {

 // Si oui, ouvrir la caméra

 openCamera();

 }

 });

 }

 // Ouvre l'application de caméra native du téléphone

 private void openCamera() {

 Intent intent = new Intent(MediaStore.ACTION\_IMAGE\_CAPTURE); // Demande de capture d'image

 if (intent.resolveActivity(getPackageManager()) != null) { // Vérifie qu'une appli peut gérer cette action

 try {

 // Crée un fichier temporaire pour stocker la photo

 photoFile = createImageFile();

 // Crée une URI compatible avec FileProvider pour accéder au fichier

 photoUri = FileProvider.getUriForFile(this,

 getPackageName() + ".fileprovider", photoFile);

 // Passe l'URI à la caméra pour y stocker l'image

 intent.putExtra(MediaStore.EXTRA\_OUTPUT, photoUri);

 // Lance l'application de caméra

 startActivityForResult(intent, REQUEST\_IMAGE\_CAPTURE);

 } catch (IOException e) {

 e.printStackTrace(); // En cas d'erreur de création du fichier

 }

 }

 }

 // Crée un fichier image temporaire dans le dossier Pictures de l'application

 private File createImageFile() throws IOException {

 String fileName = "photo\_" + System.currentTimeMillis(); // Nom unique basé sur le temps

 File storageDir = getExternalFilesDir(Environment.DIRECTORY\_PICTURES); // Répertoire de stockage

 return File.createTempFile(fileName, ".jpg", storageDir); // Crée le fichier temporaire

 }

 // Gère le retour de l'activité (ici, la caméra)

 @Override

 protected void onActivityResult(int requestCode, int resultCode, Intent data) {

 super.onActivityResult(requestCode, resultCode, data);

 // Vérifie que la photo a bien été capturée

 if (requestCode == REQUEST\_IMAGE\_CAPTURE && resultCode == RESULT\_OK) {

 // Affiche la photo dans l'ImageView

 imageView.setImageURI(photoUri);

 }

 }

 // Gère le résultat de la demande de permission

 @Override

 public void onRequestPermissionsResult(int requestCode,

 @NonNull String[] permissions, @NonNull int[] grantResults) {

 // Si la permission est accordée, on ouvre la caméra

 if (requestCode == REQUEST\_CAMERA\_PERMISSION &&

 grantResults.length > 0 &&

 grantResults[0] == PackageManager.PERMISSION\_GRANTED) {

 openCamera();

 } else {

 // Sinon, on affiche un message d'erreur

 Toast.makeText(this, "Permission caméra refusée", Toast.LENGTH\_SHORT).show();

 }

 }

}

**🔁 Résumé du flux :**

1. Clic sur bouton → vérifie la permission
2. Si OK → ouvre la caméra
3. L’image est enregistrée dans un fichier temporaire
4. Le fichier est affiché dans un ImageView
5. Le tout fonctionne grâce à FileProvider pour éviter les erreurs de sécurité