

Desarrollo de Aplicaciones en Android

Profesora: Lina E. Arias Hdz.

Unidad 5: *Persistencia de Datos y Conectividad*

Proyecto: *App Instagram*

***#21130598 Edmundo Viramontes Gutierrez
#20130762 Reyes Lazalde Joahan Gilberto***

Fecha de entrega: 22/11/2024

Tabla de Contenidos

1. INTRODUCCIÓN	3
2. COMPETENCIAS POR DESARROLLAR	3
3. EXPLICACIÓN DE CÓDIGO	3
4. CONCLUSIONES Y RECOMENDACIONES	7
5. REFERENCIAS	7

1. INTRODUCCIÓN

Este proyecto trata acerca de la utilización de Android Studio, con el objetivo a cumplir, en nuestro caso siendo este el poder seleccionar una imagen y poder compartirla en Instagram.

2. COMPETENCIAS POR DESARROLLAR

Como se mencionó en la introducción nuestro tema asignado fue el poder hacer publicación de imágenes en instagram, esto en nuestro proyecto se lleva a cabo mediante la selección de una imagen su vista previa dentro de la app y el proceso de publicación dentro de la aplicación de instagram.

Porque se realiza el proceso de publicación dentro de la aplicación de instagram? Debido a que para cargar imágenes directamente a Instagram sin interacción del usuario, necesitaría integrar la API Graph de Instagram, pero esto requiere autenticación de la aplicación y tiene requisitos específicos de Instagram.

Esta solución depende de que Instagram esté instalado en el dispositivo del usuario. Si Instagram no está instalado, la aplicación permitirá al usuario elegir otra aplicación que admita compartir imágenes.

3. EXPLICACIÓN DE CÓDIGO

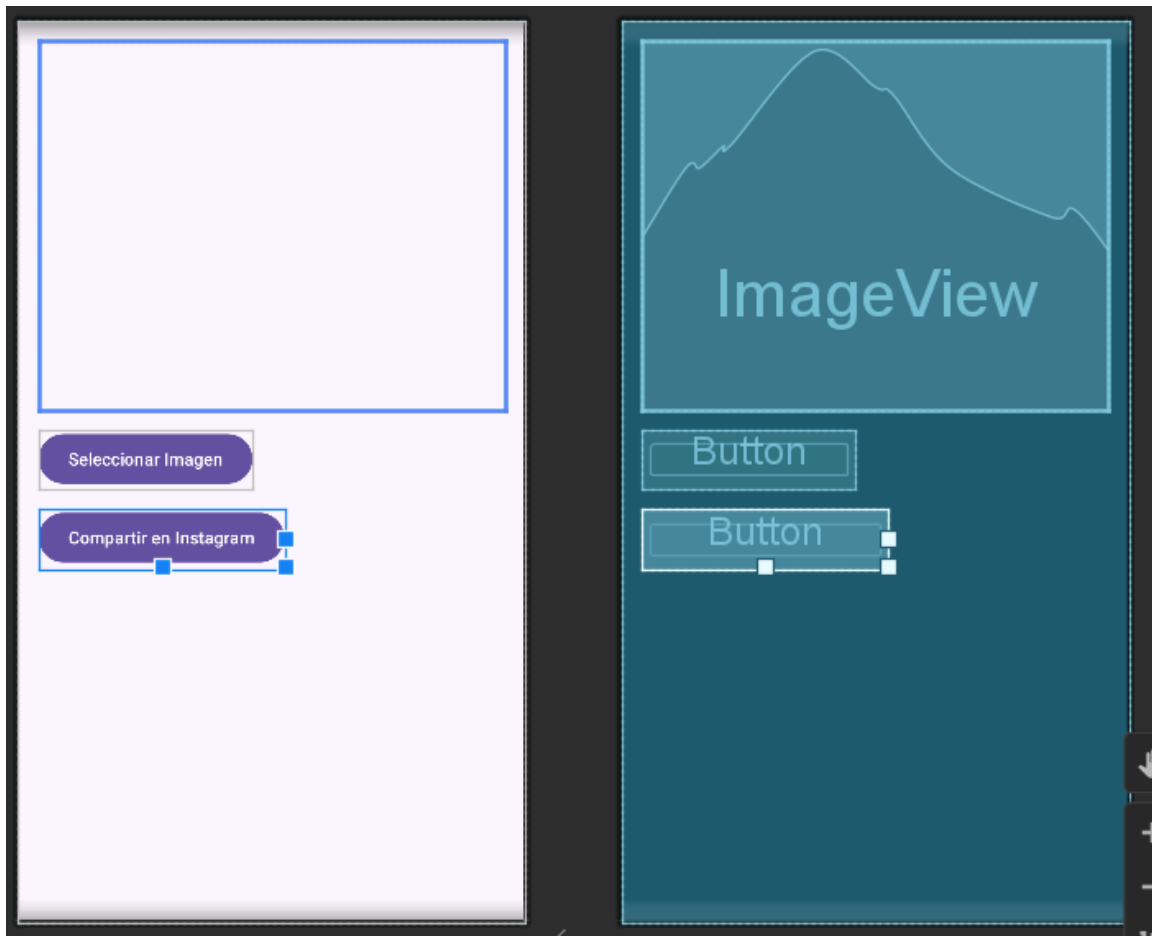
Este apartado se utilizará para explicar el funcionamiento específico de la aplicación y aquellos code - snippets esenciales para la utilización.

Dentro de AndroidManifest.xml agregamos:

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

Esto ya que así informas al sistema que tu app necesita permiso para leer el almacenamiento externo (como la galería).

GUI:



Nuestra interfaz grafica es una sencilla pero la cual cumple con los requisitos necesarios para cumplir los objetivos del proyecto, estos siendo selección de imagen, visualización previa de la imagen seleccionada y la capacidad de compartir dicha imagen.

Dentro del ActivityMain.java se encuentran todos los metodos que nos permiten el funcionamiento de la app, estos siendo:

Variables:

PICK_IMAGE_REQUEST: Una constante para identificar la solicitud de selección de imagen. Se usa para distinguir qué solicitud está regresando en onActivityResult.

selectedImageUri: Una variable que contiene la URI (Identificador Uniforme de Recursos) de la imagen seleccionada.

imageView: Referencia al ImageView en el layout para mostrar la imagen seleccionada.

```
private static final int PICK_IMAGE_REQUEST = 1;
private Uri selectedImageUri;
private ImageView imageView;
```

Dentro del método onCreate() tenemos la inicialización de nuestros elementos y la creación de métodos OnClickListener los cuales mandan a llamar métodos según la funcionalidad de cada botón.

```
imageView = findViewById(R.id.imageView);
Button selectImageBtn = findViewById(R.id.selectImageBtn);
Button shareToInstagramBtn = findViewById(R.id.shareToInstagramBtn);

// BBoton para seleccionar imagen

selectImageBtn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        openImageSelector();
    }
});

// Boton para compartir imagen a Instagram

shareToInstagramBtn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        shareImageToInstagram();
    }
});
}
```

El siguiente método es el openImageSelector(), este como su nombre nos puede decir, abre la galería para seleccionar imagen

```
private void openImageSelector() {
    Intent intent = new Intent(Intent.ACTION_PICK,
    MediaStore.Images.Media.EXTERNAL_CONTENT_URI);
    startActivityForResult(intent, PICK_IMAGE_REQUEST);
}
```

Se crea un Intent con Intent.ACTION_PICK para abrir la galería.

MediaStore.Images.Media.EXTERNAL_CONTENT_URI especifica que estás buscando imágenes en el almacenamiento externo.

startActivityForResult() se llama para iniciar la actividad de la galería y esperar el resultado (la imagen seleccionada).

Después se tiene dentro del método `@Override onActivityResult()`, este es el que se encarga de procesar la imagen seleccionada, tenemos un `if()` el cual nos extrae la URI de la imagen, URI es la posición donde se encuentra la imagen y actualiza el `imageView` para la visualización previa de la imagen seleccionada.

```
@Override
protected void onActivityResult(int requestCode, int resultCode,
@Nullable Intent data) {
    super.onActivityResult(requestCode, resultCode, data);

    if (requestCode == PICK_IMAGE_REQUEST && resultCode == RESULT_OK &&
data != null) {
        selectedImageUri = data.getData();
        imageView.setImageURI(selectedImageUri);
    }
}
```

`requestCode`: Identifica a qué solicitud pertenece este resultado. Lo comparas con `PICK_IMAGE_REQUEST` para asegurarte de que es la respuesta de la galería.

`resultCode`: Verifica si la operación fue exitosa (`RESULT_OK`), si tiene éxito, se obtiene la URI de la imagen con `data.getData()`.

`imageView.setImageURI(selectedImageUri)` actualiza el `ImageView` para mostrar la imagen seleccionada.

Por último tenemos el método que hace la magia, esté siendo `shareImageToInstagram()`, como podremos inferir de su nombre, es el que se encarga de compartir la imagen a Instagram. Tenemos dentro de él un `if()` el cual nos si se tiene una URI almacenada se da luz verde al proceso, esté siendo:

```
//Comparte la imagen seleccionada a Instagram
private void shareImageToInstagram() {
    if (selectedImageUri != null) {
        Intent shareIntent = new Intent();
        shareIntent.setAction(Intent.ACTION_SEND);
        shareIntent.setType("image/*");
        shareIntent.putExtra(Intent.EXTRA_STREAM, selectedImageUri);

        //Se establece Instagram como el app destino
        shareIntent.setPackage("com.instagram.android");

        //Se checa si Instagram esta instalado mediante getPackageManager
        if (shareIntent.resolveActivity(getPackageManager()) != null) {
            startActivity(shareIntent);
        } else {
            //Si Instagram no está instalado, se avisa al usuario
            startActivity(Intent.createChooser(shareIntent, "Share
Image"));
        }
    }
}
```

```
}
```

Explicación:

- Se crea un nuevo Intent con ACTION_SEND para indicar que se desea compartir contenido.
- setType("image/*") especifica que el tipo de contenido es una imagen.
- putExtra(Intent.EXTRA_STREAM, selectedImageUri) adjunta la URI de la imagen seleccionada al Intent.
- setPackage("com.instagram.android") asegura que Instagram sea la aplicación de destino. Si Instagram no está instalado, resolveActivity() devolverá null, y se mostrará un cuadro de diálogo para que el usuario elija otra app compatible, de esta manera, el usuario aún puede compartir la imagen a través de otra aplicación compatible.

Y eso sería todo para la explicación de nuestro código, es una aplicación sencilla pero cumplidora.

4. CONCLUSIONES Y RECOMENDACIONES

Esta aplicación nos sirvió bastante para entender bien el manejo de los Intents, que tipos hay y para que se utilizan, además de reforzar lo visto en clase en tema de URI's.

También que la documentación para la utilización de la API de meta Graph API es muy poco entendible lo cual lleva a errores y poca comprensión.

5. REFERENCIAS

Maheshwari, A. (2021, Mayo 13). *Android app — share multiple files to Instagram*.

Medium. Retrieved Noviembre 18, 2024, from

<https://abskmhswri-er.medium.com/android-app-share-multiple-files-to-instagram-be8ed12a652a>

Mani, A. (2013, Abril 30). *android - post image in Instagram*. Stack Overflow. Retrieved

Noviembre 21, 2024, from

<https://stackoverflow.com/questions/16296770/post-image-in-instagram>

Meta Platforms, Inc. (n.d.). *Instagram APIs | Facebook for Developers*. Meta for Developers. Retrieved Noviembre 18, 2024, from <https://developers.facebook.com/products/instagram/apis/>

michaeltys. (2018, Octubre 10). *Sharing an image to instagram stories or create a post*. GitHub Gist. Retrieved Noviembre 18, 2024, from <https://gist.github.com/michaeltys/a8613e5aea9db8e4684bf85568e40160>