









Clic para adjuntar un archivo multimedia




7:14

Tiendas App


**LN ATLIXCO MORALEDA LNPK091PUE**  
**Izucar de Matamoros**  
**LN-Izucar de Matamoros**


**SUPERVISOR: HERNANDEZ LIMANTUR DAVID**

 Responder

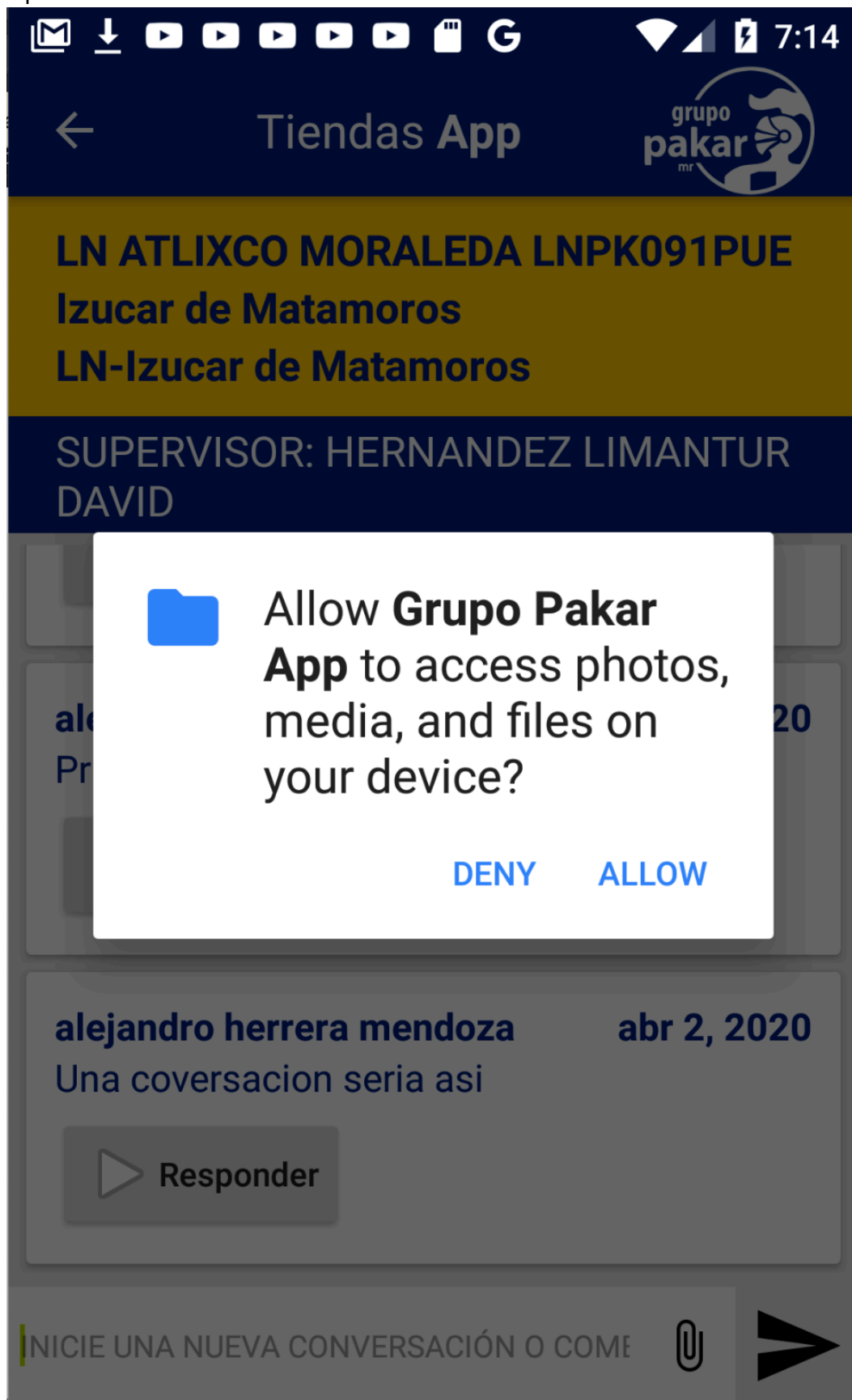
**alejandros herrera mendoza** **abr 2, 2020**  
Prueba comentario  
 Responder

**alejandros herrera mendoza** **abr 2, 2020**  
Una conversacion seria asi  
 Responder

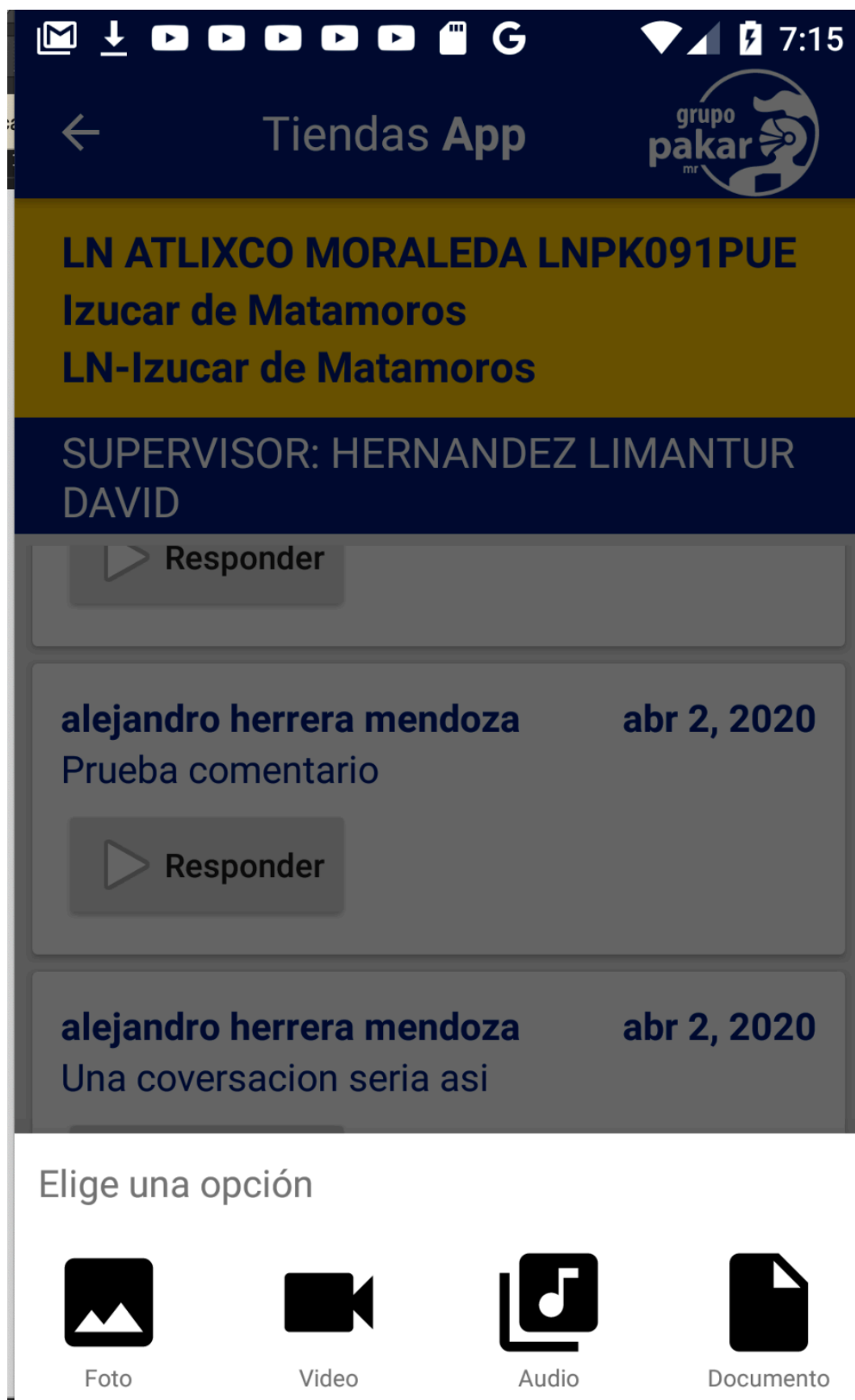
INICIE UNA NUEVA CONVERSACIÓN O COME 



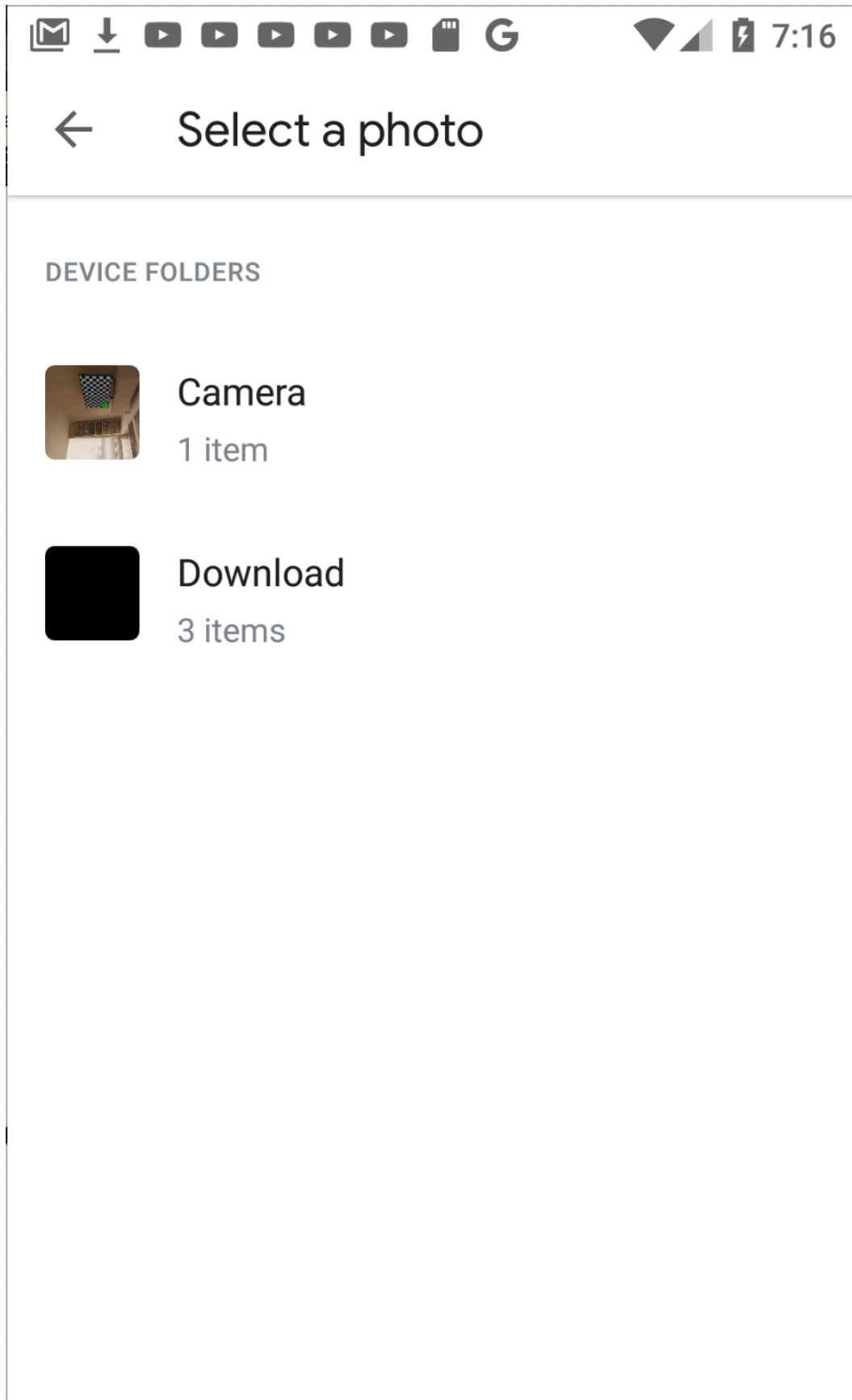
Se pide acceso para acceder a sus datos



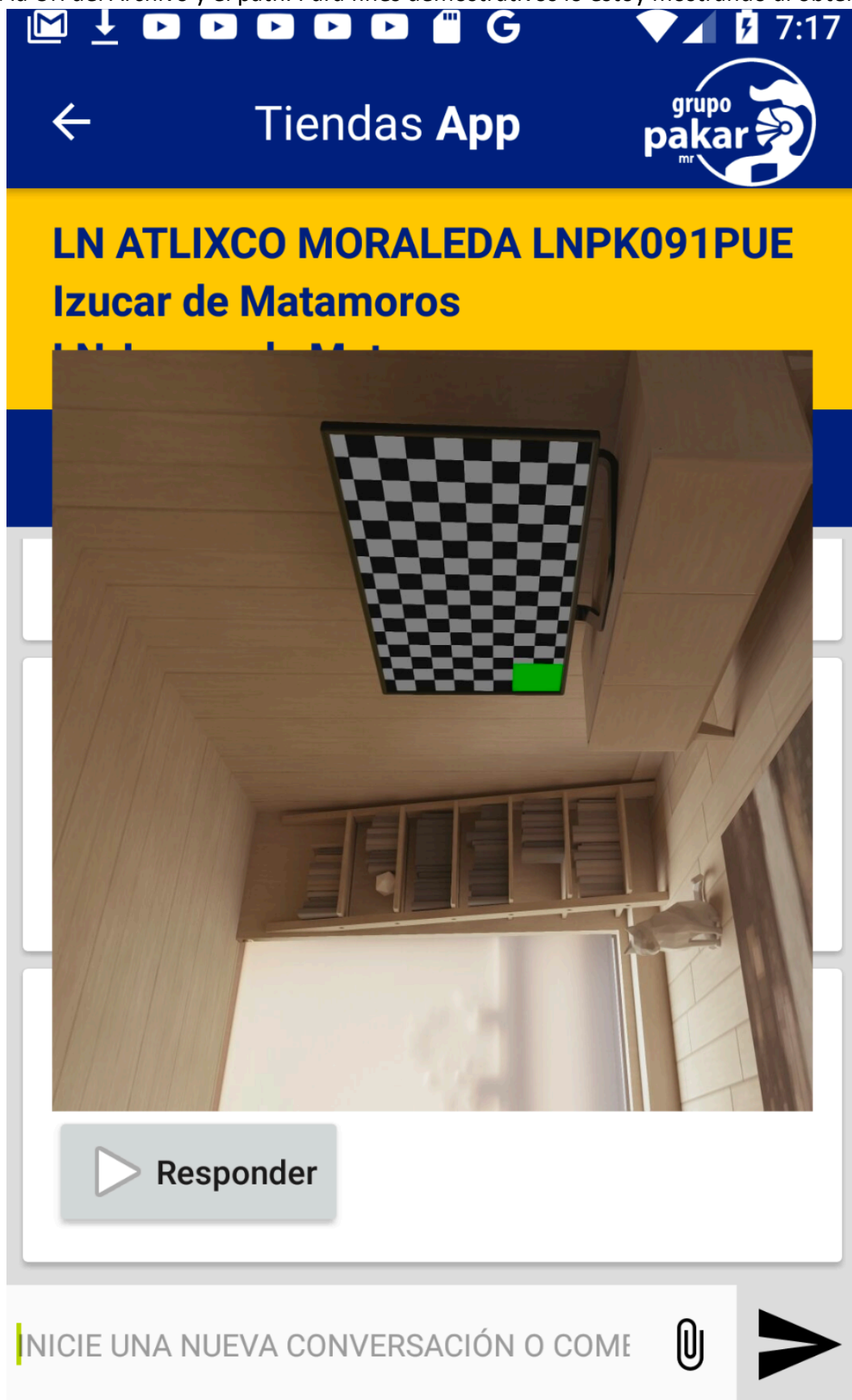
Se selecciona una opción del menú



Se elige opción Foto



Ya se retoma la Uri del Archivo y el path. Para fines demostrativos lo estoy mostrando al obtener la uri.



Se creo la clase Archivos.java para acceder al tipo de multimedia.

```
Archivos.java x Comentarios.java x activity_comentarios.xml x Constante.java x elemento_comentario_archivo.xml x
4 import android.content.Intent;
5 import android.database.Cursor;
6 import android.net.Uri;
7 import android.provider.MediaStore;
8
9
10 import static com.grupopakar.grupopakarapp.util.Constante.ARCHIVO_CODE;
11 import static com.grupopakar.grupopakarapp.util.Constante.AUDIO_CODE;
12 import static com.grupopakar.grupopakarapp.util.Constante.IMAGE_CODE;
13 import static com.grupopakar.grupopakarapp.util.Constante.VIDEO_CODE;
14
15 public class Archivos{
16
17 @   public void openGallery(Activity activity) {
18     Intent gallery = new Intent(Intent.ACTION_PICK, MediaStore.Images.Media.INTERNAL_CONTENT_URI);
19     activity.startActivityForResult(gallery, IMAGE_CODE);
20 }
21
22 @   public void openMusic(Activity activity) {
23     Intent gallery = new Intent(Intent.ACTION_PICK, MediaStore.Audio.Media.EXTERNAL_CONTENT_URI);
24     activity.startActivityForResult(gallery, AUDIO_CODE);
25 }
26
27 @   public void openVideo(Activity activity) {
28     Intent gallery = new Intent(Intent.ACTION_PICK, MediaStore.Video.Media.INTERNAL_CONTENT_URI);
29     activity.startActivityForResult(gallery, VIDEO_CODE);
30 }
31
32 @   public void openDocument(Activity activity) {
33     Intent intent = new Intent(Intent.ACTION_OPEN_DOCUMENT);
34     intent.setType("*/*");
35     intent.addCategory(Intent.CATEGORY_OPENABLE);
36     activity.startActivityForResult(intent, ARCHIVO_CODE);
37 }
38 }
```

También se agregó un método en esta clase para retornar el path del archivo(imagen, video y audio)

```
38
39 @   public String rutaArchivo(Uri uri, String[] media, Activity activity) {
40     String fotoPath = null;
41     Cursor cursor = activity.getContentResolver().query(uri,
42         media, selection: null, selectionArgs: null, sortOrder: null);
43     if (cursor != null) {
44         cursor.moveToFirst();
45         int archivoIndex = cursor.getColumnIndex(media[0]);
46         fotoPath = cursor.getString(archivoIndex);
47         cursor.close();
48     }
49     return fotoPath;
50 }
```

Tipo de acción para acceder al tipo de multimedia.

```
@Override
public void onClick(View v) {
    switch (v.getId()) {
        case R.id.imageButton:
            permisos();
            break;
        case R.id.imageButtonSend:
            enviarComentario();
            break;
        case R.id.actionFoto:
            archivos.openGallery( activity: this);
            mBottomSheetDialog.dismiss();
            break;
        case R.id.actionDocumento:
            archivos.openDocument( activity: this);
            mBottomSheetDialog.dismiss();
            break;
        case R.id.actionAudio:
            archivos.openMusic( activity: this);
            mBottomSheetDialog.dismiss();
            break;
        case R.id.actionVideo:
            archivos.openVideo( activity: this);
            mBottomSheetDialog.dismiss();
            break;
    }
}
```

Método @Override para capturar el resultado del Intent.

```
@SuppressWarnings("MissingSuperCall")
@Override
protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {
    if (data != null) {
        Uri uri = data.getData();
        assert uri != null;
        switch (requestCode) {
            case IMAGE_CODE:
            case AUDIO_CODE:
            case VIDEO_CODE:
                String ruta = archivos.rutaArchivo(uri, filePathColumn, activity: this);
                System.out.println("Ruta **: " + ruta);
                archivoImage.setVisibility(VISIBLE);
                archivoImage.setImageURI(uri);
                break;
            case ARCHIVO_CODE:
                Uri content_describer = data.getData();
                getArchivo(content_describer);
                break;
        }
    }
}
```

Método para construir el archivo, ya que no se puede obtener su path como de los demás (imagen, audio, video)

```
private void getArchivo(Uri content_describer) {
    BufferedReader reader = null;
    try {
        InputStream in = getContentResolver().openInputStream(content_describer);
        reader = new BufferedReader(new InputStreamReader(in));
        String line;
        StringBuilder builder = new StringBuilder();
        while ((line = reader.readLine()) != null) {
            builder.append(line);
        }
        System.out.println("getArchivo " + builder.toString());
    } catch (IOException e) {
        e.printStackTrace();
    } finally {
        if (reader != null) {
            try {
                reader.close();
            } catch (IOException e) {
                e.printStackTrace();
            }
        }
    }
}
```