Bloque 3. Exercici Resolt Intent Implícit

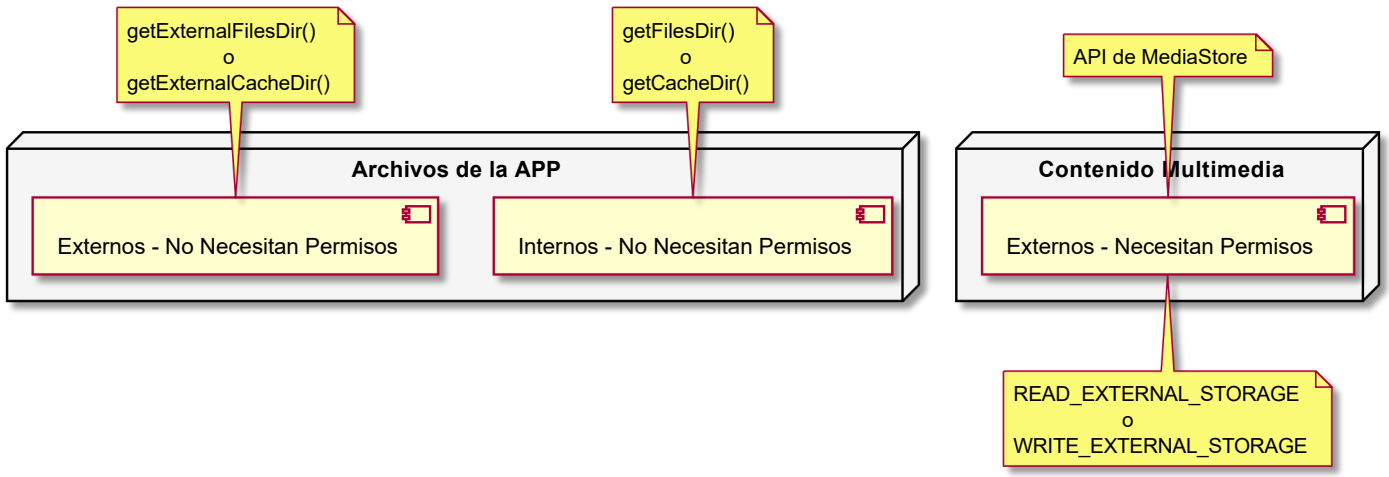
Descarregar aquests apunts

Emmagatzematge d'informació en Android

Android utilitza un sistema d'arxius que és similar als sistemes d'arxius basats en discos d'altres plataformes. El sistema proporciona diverses opcions perquè guardes les dades de la teua app:

- **Emmagatzematge específic de l'app**: Emmagatzema arxius dissenyats només per a la teua app, ja siga en directoris dedicats dins d'un volum de **emmagatzematge intern** o en directoris dedicats diferents dins del **emmagatzematge extern**. Usa els directoris de l'emmagatzematge intern per a guardar informació sensible a la qual altres apps no haurien d'accedir.
- **Emmagatzematge compartit**: Emmagatzema arxius que la teua app pretenga compartir amb altres apps, inclosos arxius multimèdia, documents i altres.
- **Preferències**: Emmagatzema dades primitives i privades en parells clau-valor.
- **Bases de dades**: Emmagatzema dades estructurades en una base de dades privada mitjançant la biblioteca de persistències Room.

> ⚠ **Avis**: Per a donar-los als usuaris més control sobre els seus arxius i delimitar el desordre, les aplicacions orientades a Android 10 (nivell de API 29) i versions posteriors reben accés amb abast determinat a l'emmagatzematge extern, o **emmagatzematge específic**, de manera predeterminada. Aquestes apps només tenen accés al directori específic de l'aplicació en l'emmagatzematge extern, així com a tipus específics de contingut multimèdia que va crear l'app.



> 📌 **Nota:** Els emmagatzematges externs, siguen arxius pertanyents a l'App o contingut multimèdia seran accessibles per altres aplicacions. Els arxius de l'App s'eliminaran quan s'esborre aquesta, tant els interns com els externs.
>
> Els volums que es poden extraure, com les **targetes SD**, apareixen en el sistema d'arxius com a part de l'emmagatzematge extern. Android representa aquests dispositius mitjançant una ruta d'accés, com **/sdcard**. Per a accedir a aquest emmagatzematge, a més de demanar permís s'haurà de declarar un content provider en el manifest.

Exercici

Realitzar una aplicació que continga tres botons. El primer botó té obrir el navegador amb la següent cerca *"iesdoctorbalmis"*. El segon botó ens permetrà obrir una aplicació externa que hàgeu instal·lat prèviament. I el tercer botó llançarà una cançó mp3 que es trobe en la targeta **sd** del dispositiu.

PAS I. Preparar la vista de l'aplicació

Per a això haurem de modificar l'arxiu **activity_main.xml** amb el necessari per a crear els tres botons que estiguen més o menys centrats. Una possible solució podria ser la següent:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:orientation="vertical"
    android:gravity="center"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
    <Button
        android:id="@+id/navegador"
        android:layout_width="200dp"
        android:layout_height="wrap_content"
        android:text="@string/abrir_navegador"/>
    <Button
        android:id="@+id/abrir"
        android:layout_width="200dp"
        android:layout_height="wrap_content"
        android:text="@string/abrir_aplicaci_n"/>
    <Button
        android:id="@+id/escuchar"
        android:layout_width="200dp"
        android:layout_height="wrap_content"
        android:text="@string/escuchar_canci_n"/>
</LinearLayout>
```

PAS II. Codificar l'execució del primer botó

El mètode següent és el que correspon amb la codificació del primer botó. Haurem d'afegir-ho dins del mètode `onCreate` de la `MainActivity.kt`

```
override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    setContentView(R.layout.activity_main)
    findViewById<Button>(R.id.navegador).setOnClickListener (View.OnClickListener {
        val intent = Intent(Intent.ACTION_WEB_SEARCH).apply {
            putExtra(SearchManager.QUERY, "iesdoctorbalmis")
        }
        startActivity(intent)
    })
}
```

> 📌 **Nota:** la **línia 3** el que fa és carregar el layout que volem associar amb l'activitat principal i assignar-ho al context de l'aplicació, **Línia 4** el mètode findViewById cerca la vista en el layout del context i la seguidora (crear l'objecte del tipus corresponent associat a aqueixa vista, en aquest cas Button). **Línia 7 i 8** creen el Intent de cerca en la web i inicia l'activitat que ho resolga, sempre que hi haja alguna activitat instal·lada en el dispositiu que el puga resoldre.

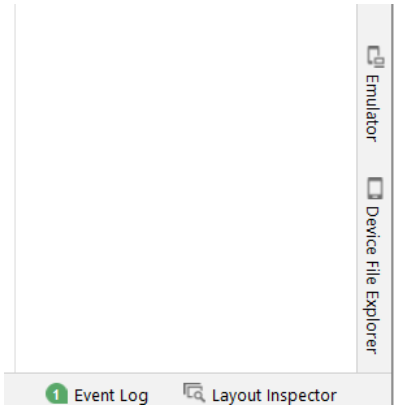
PAS III. Obrir aplicació externa

El codi següent llança el intent que permet obrir una app que està instal·lada en el dispositiu. S'haurà de tindre coneixement del domini per a indicar-lo en el nom del component **Línia3**.

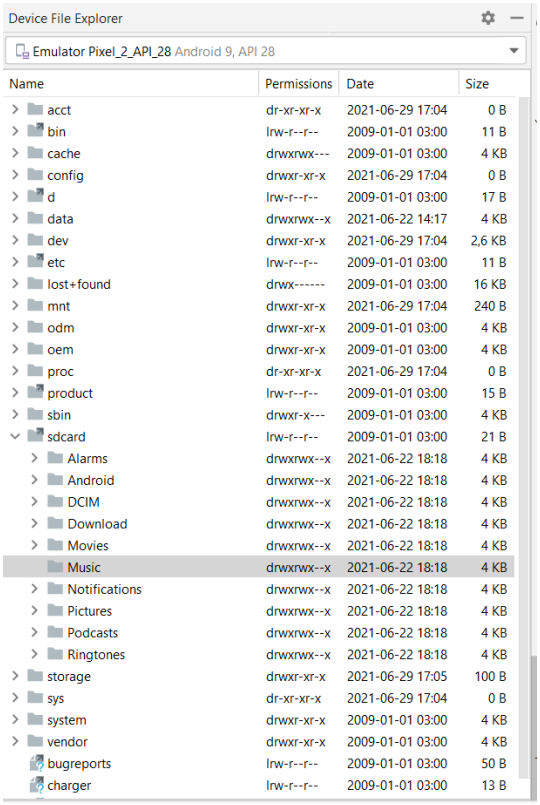
```
findViewById<Button>(R.id.abrir).setOnClickListener {
    val intent= Intent();
    intent.setComponent(ComponentName("com.ejemplos.myapplication",
            "com.ejemplos.myapplication.MainActivity"))
    if (intent.resolveActivity(packageManager) != null) startActivity(intent)
}
```

PAS IV. Accés a la targeta SD

Després de crear el projecte hem d'accedir al Device File Explorer, hi ha una pestanya en la part inferior dreta del IDE.



Això ens obrirà un explorador que dona accés a les carpetes dels dispositius/emuladors que estiguen arrancats en aqueix moment. Accedirem a la carpeta Music de sdcard i amb el menú contextual podrem pujar o descarregar elements al/del dispositiu. Procedeix a afegir un arxiu .mp3 (es passen dos com a recursos de l'exercici).



PAS V. Lògica Botó Escoltar cançó

Tenint en compte la teoria sobre l'emmagatzematge que usen els dispositius Android, que ens trobem al principi del tema. En aquest cas s'accedirà al contingut multimèdia, per la qual cosa s'haurà de sol·licitar permís d'accés a la targeta externa, a més de declarar un content provider. Vegem el codi i passem a les explicacions...

Demanar permisos

Aquest codi és l'encarregat de mostrar els diàlegs que permeten a l'usuari atorgar el permís sol·licitat. Es comenta en el tema, per la qual cosa no es dirà res més sobre aquest tema.

```
private val RESPUESTA_PERMISOS = 111

@RequiresApi(Build.VERSION_CODES.Q)
fun solicitarPermisosYEscucharCancion() {
    if (checkSelfPermission(READ_EXTERNAL_STORAGE) ==
    PackageManager.PERMISSION_DENIED) {
            requestPermissions(arrayOf(READ_EXTERNAL_STORAGE),
             RESPUESTA_PERMISOS)
    } else escucharCancion()
}
override fun onRequestPermissionsResult(
        requestCode: Int,               //codigo de identificación resultado
        permissions: Array<out String>, //array con los nombres
                                        //de los permisos
        grantResults: IntArray          //array de 0 y -1 (permitido,
                                        //no permitido) en orden
        ) {
        super.onRequestPermissionsResult(requestCode, permissions,
        grantResults)
        when (requestCode) {
            RESPUESTA_PERMISOS -> {
                if (grantResults[0] == PackageManager.PERMISSION_GRANTED) {
                    Toast.makeText(
                        applicationContext,
                        permissions[0] + " Permiso concedido",
                        Toast.LENGTH_SHORT
                    ).show()
                    escucharCancion()
                }
            }
        }
}
```

Declarar el content provider

Per a la versions de SDK d'Android majors a la 24, no es pot accedir directament a la memòria externa que no corresponga a l'aplicació, per la qual cosa serà necessari realitzar els següents passos:

1. Crear una carpeta xml en els recursos, **`res\xml`** i dins un fitxer, per exemple **`provider_paths`**. On afegirem el següent codi per a referenciar al directori arrel.

   ```
   <?xml version="1.0" encoding="utf-8"?>
   <paths>
       <external-path name="external_files" path="."/>
   </paths>
   ```

2. Afegir en el Manifest un provider que faça referència al recurs creat. Dins de l'etiqueta aplicació i tenint en compte no oblidar els permisos de **`READ_EXTERNAL_STORAGE`**.

```
<provider
    android:name="androidx.core.content.FileProvider"
    android:authorities="${applicationId}.provider"
    android:exported="false"
    android:grantUriPermissions="true">
    <meta-data
        android:name="android.support.FILE_PROVIDER_PATHS"
        android:resource="@xml/provider_paths" />
</provider>
```

Codificar el intent

```
1  class MainActivity : AppCompatActivity() {
2
3      @RequiresApi(Build.VERSION_CODES.Q)
4      override fun onCreate(savedInstanceState: Bundle?) {
5        ...
6
7         findViewById<Button>(R.id.escuchar).setOnClickListener
8         {
9              solicitarPermisosYEscucharCancion()
10        }
11     }
12     private fun escucharCancion() {
13         //Ruta de la carpeta que usa android para guardar recursos
14         //de la aplicación, consta del dominio y es privada a esta
15         var directorio=getExternalFilesDir(null)?.absolutePath
16         //Cogemos el inicio  de la ruta, que corresponde con
17         //los archivos multimedia 'storage/emulated/0/'
18         directorio=directorio?.substring(0,directorio?.indexOf("Android") ?: 0)
19         val data = File(directorio + "music/minions.mp3")
20         val uri = FileProvider.getUriForFile(
21             this@MainActivity,
22             BuildConfig.APPLICATION_ID + ".provider",
23             data
24         )
25         val intent = Intent(Intent.ACTION_VIEW)
26                     .setDataAndType(uri, "audio/.mp3")
27                     .addFlags(Intent.FLAG_GRANT_READ_URI_PERMISSION);
28         startActivity(intent)
29     }
30     ...
31 }
```

> 📌 **Nota:** En prémer el botó d'escoltar cançó es dirà el mètode `solicitarPermisosYEscucharCancion()` , **Línia 9** que al seu torn cridarà al mètode escucharCancion() quan s'hagen acceptat els permisos. En aquest últim mètode **Línia 12 a**

24, es crea un objecte per a crear la uri amb la direcció d'accés usant el provider declarat abans **Línia 15**. No s'ha d'oblidar afegir el Flags amb els permisos de lectura de Uri, **Línia 27**.