

UF2 Elements multimèdia. Voice recording

En aquesta pràctica volem gravar un missatge d'àudio i mostrar-lo al camp de text.

Interfície:

1. Crea una interfície que tingui un botó (ImageButton) i un camp de text (TextView)
2. Fes que el botó tingui una imatge d'un micro tancat

ex:



3. Ara volem fer que si cliquem el botó comenci a gravar.
4. Fes que la imatge del botó es modifiqui per un micro obert si comences a gravar.

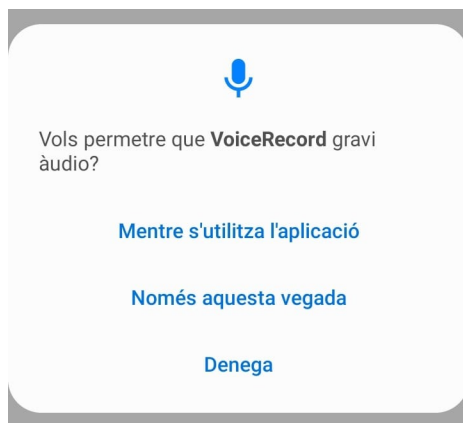


Permisos:

1. A l'Android Manifest afegirem els següents permisos

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

2. L'aplicació requereix tenir permisos de gravació pel que hem de demanar a l'usuari que faci la validació



A l'inici del codi, al mètode `onCreate` farem aquesta petició. Per a fer-ho primer comprovarem que l'aplicació tingui els permisos ben definits a l'Android Manifest. Si això es compleix farem una petició de permisos mitjançant el mètode `requestPermissions`.

```
if (ContextCompat.checkSelfPermission(this, Manifest.permission.RECORD_AUDIO) !=
PackageManager.PERMISSION_GRANTED) {
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {
        ActivityCompat.requestPermissions(this, new
            String[] {Manifest.permission.RECORD_AUDIO}, RecordAudioRequestCode);
    }
}
```

Fixeu-vos que estem fent una petició de permisos i per tant necessitem analitzar la resposta, aquesta estarà definida en un altre mètode `onRequestPermissionsResult`.

Imagineu que tenim més d'un permís per aplicació, però només tenim un mètode predefinit per analitzar la resposta, ja que aquest és un `@Override` del propi `android`. És per això que a cada petició de permisos hem d'afegir-hi un request code. En aquest cas de l'exemple, he definit la variable `RecordAudioRequestCode` per analitzar aquesta crida.

```
@Override
public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions,
@NonNull int[] grantResults) {
    super.onRequestPermissionsResult(requestCode, permissions, grantResults);
    if (requestCode == RecordAudioRequestCode && grantResults.length > 0) {
        if (grantResults[0] == PackageManager.PERMISSION_GRANTED)
            Toast.makeText(this, "Permission Granted", Toast.LENGTH_SHORT).show();
    }
}
```

Gravem i transofrmem:

1. En fer click al botó cridarem a un mètode que cridarà a l'acció de gravar

```
Intent speechRecognizerIntent = new Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEECH);  
speechRecognizerIntent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL, RecognizerIntent.LA  
NGUAGE_MODEL_FREE_FORM);  
speechRecognizerIntent.putExtra(RecognizerIntent.EXTRA_LANGUAGE, Locale.getDefault());  
speechRecognizerIntent.putExtra(RecognizerIntent.EXTRA_PROMPT, "Hola, digues quelcom!");
```

2. Creem un intent que cridarà a `ACTION_RECOGNIZE_SPEECH` li passem diferents paràmetres per customitzar el dialog, també l'idioma amb el que analitzarà la veu.
3. Una vegada tinguís l'intent preparat, envia'l mitjançant la petició `startActivityForResult` passant-li l'intent i un codi identificador per la resposta (`requestCode`).
4. Quan hagi acabat de parlar hauràs d'analitzar el retorn de l'intent.

Per a fer-ho crea un mètode que es digui

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

En aquest, el primer que hauràs de comprovar és si el `requestCode` és el de la crida. Si ho és, hauràs d'analitzar si el `resultCode` és `RESULT_OK`, és a dir, si ha estat satisfactori i si `data` (tot el que retorna l'intent) és diferent a null.

Si aquestes condicions es compleixen pots analitzar `data`.

```
ArrayList<String> result=data.getStringArrayListExtra( RecognizerIntent.EXTRA_RESULTS );
```

La variable `result` emmagatzemarà tota la informació de l'acció de gravar. A `result.get(0)` hi trobaràs el text del teu missatge d'àudio.

5. Fes que quan acabi de gravar es mostri el text en el camp de text i que el botó es torni a posar en vermell