



TUTORIAL 1

USO DE SENSOR DE VOZ

PEDRO GÓNGORA SORIA

Descripción

- En este tutorial voy a explicar el uso de control de voz en ANDROID STUDIO
- Gracias a esto podemos ingresar datos mediante la voz y colocarlos en los respectivos TextView o EditText o, simplemente, usarlos como evento.

COMO FUNCIONA

- La voz es registrada en una vector que podemos pasar a tipo String, y ahí detectar comprobando qué palabra se ha introducido.
- Además, hay algunas instrucciones que nos permiten separar el contenido por espacios, y así poder agregar varios elementos a la vez.

USANDOLO

- Lo primero, es crear una variable global dentro de la activity con esta estructura :

```
private static final int VOICE_RECOGNITION_REQUEST_CODE = 1;
```
- Y además, en el manifest darle permisos a la aplicación para acceder a internet.

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

USANDOLO

Cuando hemos establecido estas variables, la idea es llamar a esta función en el evento que deseemos. En este caso, voy a usarlo al pulsar un botón y asignarle un valor para que sepa qué tiene que hacer con la información que obtenga:

```
nombre.setOnClickListener(new
View.OnClickListener() {
    @Override
    public void onClick(View v) {
        //Lanzamos el reconocimiento de voz
        opcion = 0;
        startVoiceRecognitionActivity();
    }
});
```

USANDOLO

El paso siguiente, es implementar la función dentro de la activity para que sepa qué hacer en caso de que se llame, como en la transparencia anterior, a la función por un evento.

```
private void startVoiceRecognitionActivity() {  
    Intent intent = new Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEECH); // Definición del intent para realizar en análisis del mensaje  
  
    intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL, // Indicamos el modelo de lenguaje para el intent  
        RecognizerIntent.LANGUAGE_MODEL_FREE_FORM);  
    intent.putExtra(RecognizerIntent.EXTRA_PROMPT, ""); // Definimos el mensaje que aparecerá  
  
    startActivityForResult(intent, VOICE_RECOGNITION_REQUEST_CODE); // Lanzamos la actividad esperando resultados  
}
```

USANDOLO

Ahora voy a establecer dos maneras de usarlo:

La primera, es si quieres que todo el contenido de tu mensaje se use sin tener en cuenta que son palabras separadas.

Por ejemplo, si quiero meter el nombre de un alimento en su casilla, simplemente obtendría el nombre sin mas, pero si quiero llamar a alguien, necesitaría reconocer la palabra, llamar primero y luego la palabra que indique a quién llamar.

USANDOLO

- Entonces en este primer caso :

```
protected void onActivityResult(int requestCode, int resultCode, Intent data) {  
  
    if (requestCode == VOICE_RECOGNITION_REQUEST_CODE && resultCode == RESULT_OK) {  
  
        ArrayList<String> matches =  
data.getStringArrayListExtra(RecognizerIntent.EXTRA_RESULTS);  
  
        final CharSequence[] items = new CharSequence[matches.size()+1];  
  
        for(int i = 0; i < matches.size(); i++){  
  
            items[i] = matches.get(i);  
  
        }  
  
        if (opcion == 0)  
            nombre.setText(items[0].toString());  
  
    }  
  
    super.onActivityResult(requestCode, resultCode, data);  
  
}
```


USANDOLO

En este caso, tendremos en ítem[0] el String de lo que hemos dicho por micro, y podremos usarlo entero como se ve en el ejemplo de

```
if (opcion == 0)
    nombre.setText(items[0].toString());
```

USANDOLO

El otro caso, es separar lo que hemos dicho por espacios , guardarlo en un vector y usarlos según corresponda; por ejemplo usando Llamar a Pepe:

Llamar sería palabra[0], a palabra[1] y pepe palabra[2]

```
String [ ] palabras = new String[6];  
if (requestCode == VOICE_RECOGNITION_REQUEST_CODE && resultCode == RESULT_OK) {  
  
    ArrayList<String> matches = data.getStringArrayListExtra(RecognizerIntent.EXTRA_RESULTS);  
  
    palabras = matches.get(0).toString().split(" "); //CODIGO PARA SEPARAR LAS PALABRAS POR  
    ESPACIOS
```

USANDOLO

De este modo podríamos usar dichas partes según convenga :

En mi caso seria así:

```
nombre.setText (palabras[0].toString());  
gramos.setText (palabras[1].toString());  
Kcal.setText (palabras[2].toString());  
Prot.setText (palabras[3].toString());  
Hidr.setText (palabras[4].toString());  
Grasas.setText (palabras[5].toString());
```