

BRUJULA

En este tutorial vamos a mostrar como realizar una app de brújula en Android.

En primer lugar necesitamos modificar nuestra clase principal dado que vamos a utilizar un `SensorEventListener`:

```
Public class MainActivity extends Activity implements SensorEventListener{
```

Dado que necesitaremos utilizar el método `onSensorChanged` del sensor que queremos usar, en este caso es el `TYPE_ORIENTATION`

Pero antes debemos de crear una imagen, que enlazaremos con nuestra imagen alojada en los recursos, en nuestro caso utilizamos una rosa de los vientos. La imagen debe de almacenarse como `.png` y debe de tener el fondo transparente. Esta imagen la cargaremos en el `xml` y la enlazaremos en el código de nuestra aplicación.

También crearemos un dato de tipo `float` que utilizaremos para recalcular la posición de la imagen cuando esta gire apuntando al norte y también crearemos nuestra instancia del `SensorManager`.

```
Private ImageView image;
```

```
Private float currentDegree =0f;
```

```
Private SensorManager miSensorManager;
```

En el método `onCreate` enlazamos la imagen y lanzamos el sensor, cuyo funcionamiento implementaremos en la función `onSensor Changed`.

```
Protected void onCreate....{
```

```
Image = (ImageView)findViewById(R.id.imageViewCompass);
```

```
miSensorManager = (SensorManager) getSystemService(SENSOR_SERVICE);
```

```
...}
```

En el método `onSensorChanged` recibimos la orientación del sensor y giramos nuestra imagen para que el norte apunte al norte y además veamos donde está apuntando el dispositivo en ese momento, actualizando la posición periódicamente. El proceso es sencillo y podemos encontrarlo en la dirección de GitHub o en el siguiente enlace:

<https://www.codeofaninja.com/2013/08/android-compass-code-example.html>

En nuestra aplicación utilizamos una clase auxiliar `low filter` para recalcular la posición del norte, pero hemos comprobado que no es necesario, se ruega obviar la clase. En el

código viene comentado como se recalcula, basta con desbloquear dicho comentario y comentar la sentencia que usa el low filter.