

Sensores y eventos - Ejercicios

Índice

1 Pantalla táctil.....	2
2 Gestos.....	2
3 Manipulación de gráficos 3D.....	2
4 Acelerómetro.....	3
5 Geolocalización.....	3
6 Reconocimiento del habla.....	4

1. Pantalla táctil

Vamos a implementar una nueva aplicación `Eventos`, en la que mostraremos una caja en la pantalla (un rectángulo de 20x20) y la moveremos utilizando la pantalla táctil. Se pide:

- a) Empezar haciendo que se mueva la caja al punto en el que el usuario pone el dedo y comprobar que funciona correctamente (sólo hace falta reconocer el evento `DOWN`).
- b) Implementar ahora también el evento de movimiento (`MOVE`), para hacer que la caja se desplace conforme movemos el dedo.
- c) Sólo queremos que la caja se mueva si cuando pusimos el dedo en la pantalla lo hicimos sobre la caja. En el evento `DOWN` ya no moveremos la caja, sino que simplemente comprobaremos si hemos pulsado encima de ella.

Ayuda

Esto último se puede conseguir de forma sencilla devolviendo `true` o `false` cuando se produzca el evento `DOWN`, según si queremos seguir recibiendo eventos para ese gesto o no. Si se pulsa fuera de la caja podemos devolver `false` para así no recibir ningún evento de movimiento correspondiente a ese gesto.

2. Gestos

Continuaremos trabajando con el proyecto anterior, en este caso para reconocer gestos. Se pide:

- a) Modificar el ejercicio anterior para utilizar un detector de gestos para desplazar la caja. Utilizaremos el evento `onDown` para determinar si el gesto ha comenzado sobre la caja, y `onScroll` para desplazarla.
- b) Reconocer el evento `tap` realizado sobre la caja. Cuando esto ocurra se deberá cambiar el color de la caja.
- c) Reconocer el gesto `fling` ejercido sobre la caja. Cuando esto ocurra mostraremos un vector (línea) saliendo de la posición en la que terminó el gesto indicando la velocidad y dirección con la que se lanzó.
- d) De forma optativa, se puede hacer que al realizar el gesto `fling` sobre la caja ésta se lance con cierta inercia. Para hacer esto necesitaremos un hilo o temporizador que vaya actualizando la posición de la caja según su velocidad, e irá disminuyendo su velocidad debido al rozamiento.

3. Manipulación de gráficos 3D

Vamos a recuperar la aplicación `Graficos` de la sesión anterior, en la que veíamos una figura rotando, y vamos a hacer que en lugar de rotar automáticamente la podamos rotar nosotros conforme arrastremos el dedo por la pantalla, utilizando para ello un detector de gestos. Se pide:

- a) Hacer que el objeto rote al desplazar el dedo por la pantalla. Utilizaremos el evento `onScroll`.
- b) Hacer que el objeto rote con una determinada inercia al realizar un evento de tipo `onFling`.
- c) Hacer que el objeto pare cuando hagamos un *double tap*.

Ayuda

Si utilizamos `GLSurfaceView` ya no podremos sobrescribir el método `onTouchEvent`, ya que no hemos creado una subclase, sino que simplemente hemos definido un `Renderer`. Para poder recibir los eventos podemos definir un objeto `OnTouchListener` y registrarlo en el `GLSurfaceView`. Puede ser buena idea hacer que la misma clase del `Renderer` actúe también como *listener*, y así el mismo evento de *touch* podrá modificar directamente el ángulo de rotación actual.

4. Acelerómetro

Implementar una aplicación `Acelerometro` que muestre en una tabla los valores de aceleración para las coordenadas X, Y, Z.

Nota

Sólo podremos hacer este ejercicio si contamos con un dispositivo real, ya que el emulador no soporta este tipo de sensor.

5. Geolocalización

Implementar una nueva aplicación `Geolocalizacion` que nos localice geográficamente utilizando GPS, y nos muestre tanto nuestras coordenadas como nuestra dirección en forma de texto.

Para poder probar esto en el emulador deberemos indicarle manualmente al emulador las coordenadas en las que queremos que se localice. Esto lo podemos hacer de dos formas: mediante línea de comando o mediante la aplicación `DDMS`. Vamos a verlas a continuación.

Para comunicar las coordenadas al emulador mediante línea de comando deberemos conectarnos a él mediante `telnet`. Por ejemplo, si nuestro emulador está funcionando en

el puerto 5554, haremos un telnet a localhost y a dicho puerto:

```
telnet localhost 5554
```

Una vez dentro de la línea de comando del emulador, invocaremos el comando `geo` para suministrarle las coordenadas. Por ejemplo, las siguientes coordenadas corresponden a la Universidad de Alicante:

```
geo fix -0.51515 38.3852333
```

Si no queremos tener que ir a línea de comando, podemos utilizar la aplicación DDMS a la que se puede acceder de forma independiente o desde dentro de Eclipse. Dado que estamos ejecutando el emulador desde Eclipse, deberemos lanzar DDMS también dentro de este entorno. Para ello deberemos mostrar la vista *Emulator Control*. En ella veremos unos cuadros de texto y un botón con los que enviar las coordenadas al emulador, siempre que esté en funcionamiento.

Advertencia

Debido a un *bug* del SDK de Android, el DDMS no envía correctamente las coordenadas al emulador si nuestro *locale* no está configurado con idioma inglés. Para solucionar esto de forma sencilla, podemos editar el fichero `eclipse.ini` y añadir dentro de él la opción `-Duser.language=en`. Si no hacemos esto, el emulador recibirá siempre las coordenadas 0, 0.

Para utilizar el *geocoder*, deberemos utilizar un emulador que incorpore las APIs de Google (para así poder acceder a la API de mapas). Además, dado que necesitará conectarse a Internet para obtener las direcciones, deberemos solicitar el permiso `INTERNET`.

Advertencia

En el emulador de la plataforma Android 2.2 no funciona correctamente el *geocoder*. Funcionará correctamente si utilizamos, por ejemplo, un emulador con Google APIs de nivel 3 (versión 1.5 de la plataforma).

6. Reconocimiento del habla

Implementar una nueva aplicación `Habla` con un campo de texto y un botón. Al pulsar sobre el botón se lanzará el módulo de reconocimiento del habla, y una vez finalizado mostraremos lo que se haya reconocido en el campo de texto.

Nota

Sólo podremos hacer este ejercicio si contamos con un dispositivo real, ya que el emulador no soporta el reconocimiento del habla.

