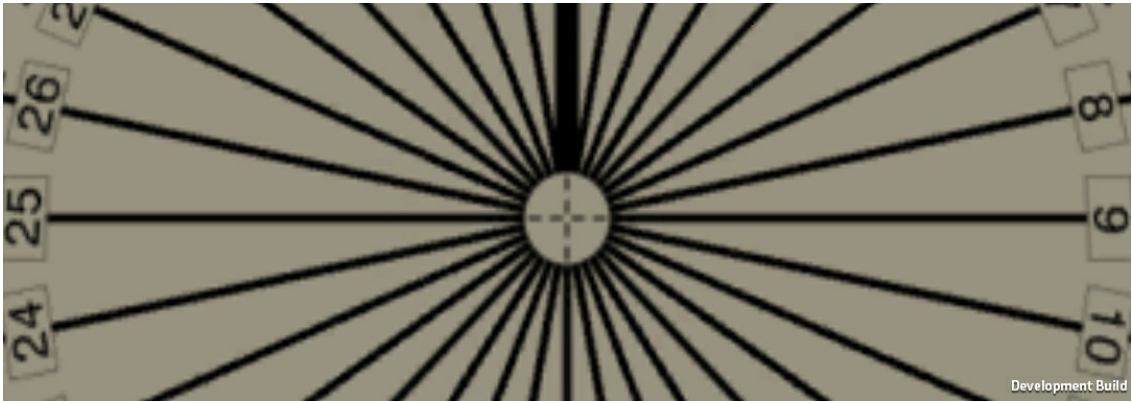


Nombre: Aram Pérez Dios
Correo: alu0101244488@ull.edu.es
Universidad: Universidad de La Laguna
Asignatura: Interfaces Inteligentes
Grupo: PE102

Introducción

En esta práctica se pretende implementar una interfaz multimodal en la cual se implementen tres componentes que se encuentran en todos los dispositivos móviles. Estos son la Brújula, los acelerómetros y el GPS. Todos estos elementos pueden ser accedidos desde unity mediante su API.





Script

Todas las funcionalidades de la aplicación Se encuentran en un mismo script **PhoneComponent.cs**. Esto es debido a la ejecución en serie del código. Ya que es necesario activar los componentes aantes de poder usarlos. Además de que realizar todas las acciones desde un mismo script facilita en gran medida la generación de la aplicación.

La gran mayoría del código se encuentra en **Start()** ya que es aquí donde deberían inicializarse todas las herramientas. Así pues lo primero que nos encontramos en el script es la activación del componente de la brújula:

```
Input.compass.enabled = true;
```

Una vez hecho esto el siguiente paso es el de activar las funcionalidades de localización (en el caso de que todavía no estuviesen activas):

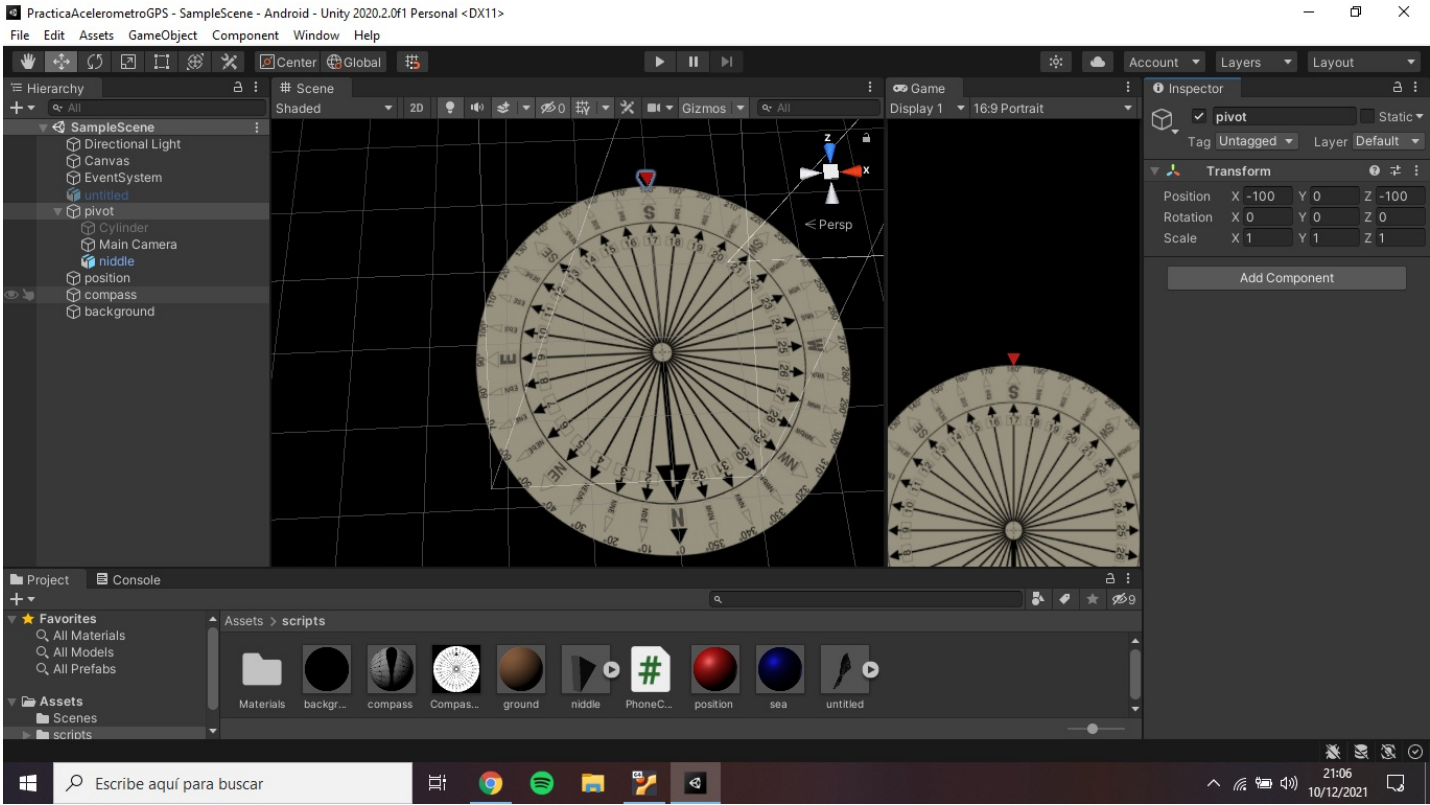
```
Input.location.Start();
```

Después de esto podemos introducir código para comprobar si estas funcionalidades se han activa correctamente o no y actuar en consecuencia. En este caso simplemente desactivamos estas funcionalidades en caso de que los elementos no comiencen a funcionar.

Hay que tener en cuenta de que **Input.LocationService** no puede activar (o desactivar) las herramientas de localización del dispositivo, sino que comprueba si estas están activas y si se puede hacer uso de ellas por lo que es necesario activarlas antes de comenzar el uso dela aplicación (todo esto se detalla en el apartado de configuración del dispositivo).

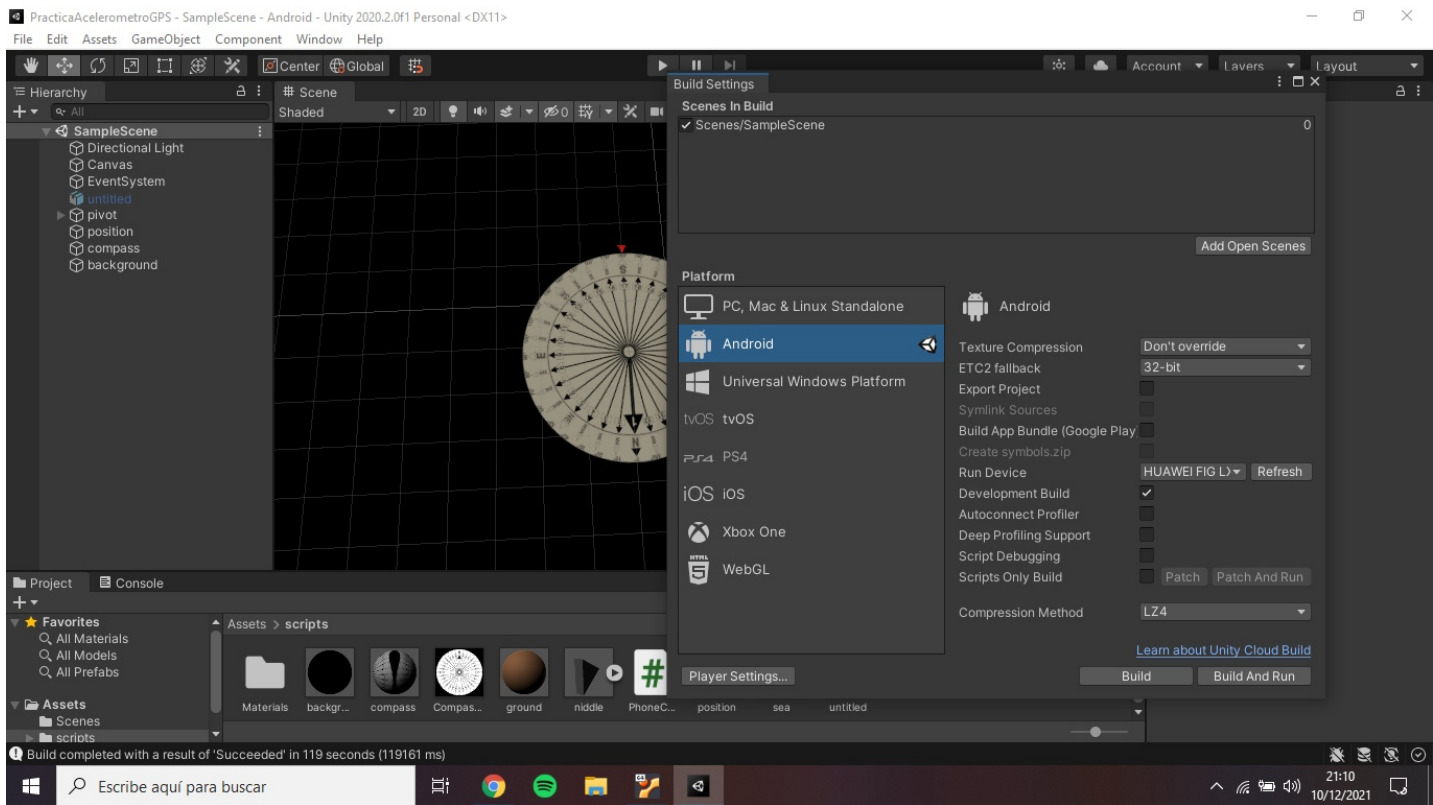
Después de esto nos encontramos con la función **Update()** la cual en este caso solamente se encarga de obtener el valor de **trueHeading** de la brújula y luego lo empleamos para rotar un objeto.

El último elemento es **OnGUI()** el cual nos permite en este caso imprimir en un elemento de GUI que se autogenera todas las lecturas de los componentes (brújula, acelerómetros y GPS). Mostrando primero la lectura de obteniendo el valor de la brújula de la misma forma que hacíamos en **Update()**. Además aquí se muestra de forma actualizada la posición via GPS y la acción de los acelerómetros.



Build (generar aplicación)

El siguiente paso es generar la APK para poder ejecutar la aplicación en un dispositivo móvil. Lo primero es instalar los módulos de unity para android en este caso. Después de instalarlos pasamos a la sección **Build Settings** donde cambiamos de plataforma a android. Y si tenemos conectado el móvil y hemos activado el modo desarrollo del dispositivo podremos realizar la build y ejecutarla directamente en el móvil.



Configuración del dispositivo

Un paso previo al de generar la build es activar el modo de desarrollador de android para poder ejecutar la aplicación directamente en el dispositivo. Aunque también hay que decir que este paso es opcional y existen otras maneras de generar la versión.

El otro paso a realizar si es obligatorio y aque sin esten la aplicación no podrá acceder a las funciones del gps. Para ello lo primero es dar permisos a la aplicación para poder hacer uso de estos. Luego se requiere la activación de las funciones de ubicación. Una vez hecho esto, podemos acceder a la aplicación sin problemas.

Muestra de la ejecución del juego

En el siguiente gif se puede observar una ejecución del entorno empleando todas las acciones que se pueden llevar a cabo:

