**MORSE TO LIGHT**

https://web.microsoftstream.com/video/ed02b92c-0b34-4994-bbe8-3c0c17adf99e

Muy buenas, en esta ocasión vamos a aprender a realizar una app con Xamarin para dispositivos Android. Básicamente la función de nuestra app será traducir un texto a código morse y transmitirlo usando para ello el flash de la cámara de nuestro smartphone.

Para ello vamos a apoyarnos en la API multiplataforma Xamarin Essentials, la cual permite hacer uso de los sensores del dispositivo (giroscopio, brújula, etc..) de una manera sencilla.

Comenzamos:

Abrimos un nuevo proyecto en Visual Studio. En este caso nuestro proyecto será del tipo Xamarin.forms como se indica en la imagen.

https://flic.kr/p/27Lt5yR

Además vamos a instalar el paquete Nugget Xamarin Essentials, que nos permitirá hacer uso del flash de la cámara.

https://www.flickr.com/photos/139260067@N04/42510042201/in/dateposted-public/

Para ello es importante que marquemos la casilla Include prerelease, ya que al ser una versión pre-lanzamiento si no la marcamos no nos aparecerá.

Ahora necesitamos dar permisos a nuestra APP para que pueda hacer uso del flash, para ello debemos editar el archivo AssemblyInfo.cs en Android e incluir las siguientes líneas de código.

[assembly: UsesPermission(Android.Manifest.Permission.Flashlight)]  
[assembly: UsesPermission(Android.Manifest.Permission.Camera)]

https://www.flickr.com/photos/139260067@N04/27639512667/in/dateposted-public/

Nos crearemos una Carperta Views y dentro de ella una vista del tipo ContentPage:

https://www.flickr.com/photos/139260067@N04/42510042251/in/dateposted-public/

En este tipo de elementos podremos incluir la parte visual (archivo XAML) , donde crearemos cuatro botones y un "entry" donde escribiremos el texto a traducir , además podremos implementar  la parte lógica de nuestra APP en el archivo CS de nuestro ContentPage.

En el archivo XAML escribiremos el siguiente código:

https://www.flickr.com/photos/139260067@N04/42510042521/in/dateposted-public/

En el archivo CS de nuestro ContentPage vamos a escribir las funciones que nos van a permitir encender y apagar el flash y además implementaremos el traductor a Código Morse.

https://www.flickr.com/photos/139260067@N04/27639512397/in/dateposted-public/

Como veis haciendo uso del Nugget Xamarin.Essentials podemos encender y apagar el flash con una simple línea de código:

https://www.flickr.com/photos/139260067@N04/27639512537/in/dateposted-public/

Ahora vamos a crear la lógica que nos permitirá encender y apagar el flash en base a un texto traducido a código Morse, el cuál traduce un texto a una serie de rayas, puntos y espacios;  nosotros convendremos en que las rayas sean un destello de luz largo, los puntos destello corto y los espacios entre palabras ausencia de luz de dos segundos.

Para ello vamos a necesitar un diccionario que contenga las correspondencias de los caracteres alfabéticos y numéricos con el sistema raya/punto del morse.

https://www.flickr.com/photos/139260067@N04/42510042421/in/dateposted-public/

Y este diccionario lo usaremos en la función traductora del texto :

https://www.flickr.com/photos/139260067@N04/28637060188/in/dateposted-public/

Ahora nuestro texto ya está representado en código Morse como una cadena de rayas y puntos  y le diremos al flash del teléfono la duración del destello cada vez que se encuentre con uno de estos caracteres:

https://www.flickr.com/photos/139260067@N04/27639512577/in/dateposted-public/

Espero que os haya gustado, esta APP es totalmente funcional para teléfonos Android con una version 4.4 o superiores; os dejaré un enlace GitHub por si queréis probar la APP en vuestros teléfonos.

Un saludo .

**Autor** :   Jesús Giganto Diez  
**Curso**:    Microsoft MCSA Web Applications + Microsoft MCSD App Builder + Xamarin  
**Centro**: Tajamar  
**Año académico**: 2017-2018

www.linkedin.com/in/jesusgiganto

**Github:**

https://github.com/jgiganto/MorseToLight