

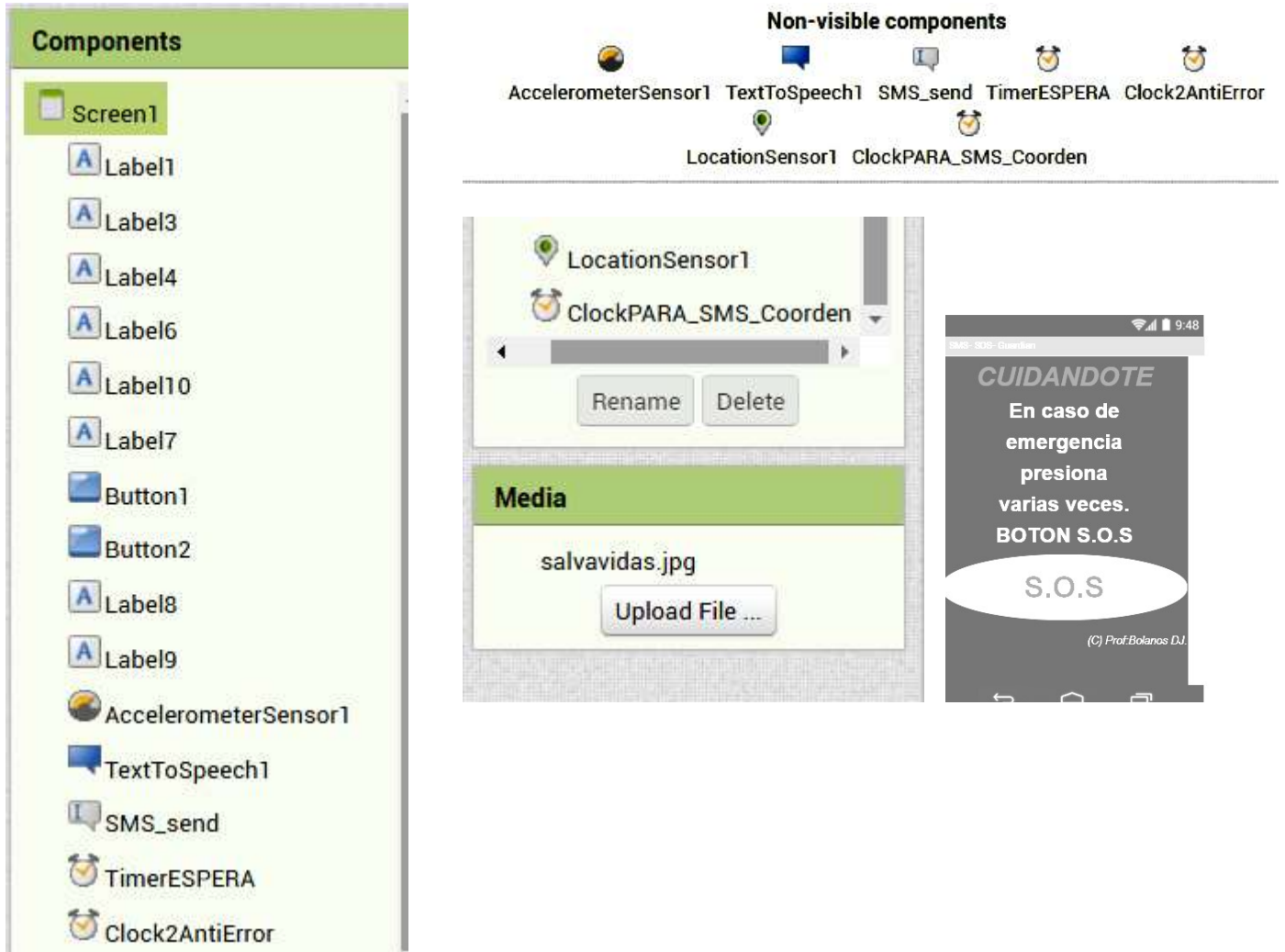
## INTRODUCCION A LA PROGRAMACION EN ANDROID

### Utilizando MIT App Inventor 2

El objeto SMS send ahora se llama TEXTING pero se trabaja de igual manera.

### Ejercicio 4

En este ejercicio se propone realizar un botón de pánico. Cuando el usuario presione (en apariencia) varias veces un botón en la pantalla, se envía en SMS el pedido de ayuda y las coordenadas de la posición actual.



Los componentes NO VISIBLES necesarios son:

- TextToSpeech1
- SMS\_send
- Clock1 (renombrado como TimerESPERA).

- Clock2 (renombrado Clock2\_Antierror),
- ClockPARA\_SMS\_Coorden
- LocationSensor1.

Además se agrega un AcelorometroSensor pero no se usa en esta versión, este permitirá en versiones posteriores, que el llamado se produzca al agitar

### Aclaraciones:

Para el SMS\_Send, **En PhoneNumber el teléfono destino al cual se enviara el mensaje.**

(Debería colocar aquí su número de teléfono, ya que de lo contrario necesitaría otro teléfono para probar esta APP). La mayoría de las compañías telefónicas ofrecen mensajes gratis a sus usuarios.

Los componentes clock funcionan de igual manera que los Timer en Builder C++.

Los tiempos de lo Clock se controlan desde las propiedades del componente en la ventana de diseño.

**A continuación de muestra el diagrama de bloques sugerido para esta APP.**

```

when Button1 .Click
do
  set Button1 . Enabled to false
  set Clock2AntiError . TimerEnabled to true
  set TimerESPERA . TimerEnabled to true
  set LocationSensor1 . Enabled to true

```

```

when TimerESPERA .Timer
do
  set Button2 . Enabled to true
  set Button2 . Visible to true
  set Button1 . Visible to false
  set TimerESPERA . TimerEnabled to false

```

```

when Button2 .Click
do
  set Button1 . Visible to true
  call TextToSpeech1 .Speak
  message " Sistema de emergencia activado, la ayuda viene e..."
  set Screen1 . BackgroundColor to #000000
  set SMS_send . Message to
    join " S.O.S Guardian te necesita. Ubicacion: "
    join " Lat: "
    join LocationSensor1 . Latitude
    join " Long: "
    LocationSensor1 . Longitude
  call SMS_send .SendMessage
  set ClockPARA_SMS_Coorden . TimerEnabled to true
  set Button2 . Visible to false

```

```

when Clock2AntiError .Timer
do
  set Button1 . Visible to true
  set Button2 . Visible to false
  set Clock2AntiError . TimerEnabled to false
  set Button1 . Enabled to true
  set Button2 . Enabled to false

```

```

when ClockPARA_SMS_Coorden .Timer
do
  set SMS_send . Message to
    join " S.O.S Guardian te necesita. Ubicacion: "
    join " Lat: "
    join LocationSensor1 . Latitude
    join " Long: "
    join LocationSensor1 . Longitude
    join " http://maps.google.com/maps?f=q&q="
    join LocationSensor1 . Latitude
    join " "
    join LocationSensor1 . Longitude
    join "&z=16 "
  call SMS_send .SendMessage
  set Screen1 . BackgroundColor to make color
  make a list 0
  set ClockPARA_SMS_Coorden . TimerEnabled to false

```

**Funcionamiento:** Button1 y Button2 están superpuestos uno sobre el otro, pero son visibles alternativamente. Cuando el usuario presiona el Button1 se inician los 2 Clock. Cuando acaba el Clock **Tiempo de Espera**, podemos ver el Button2 y presionarlo, este es el que envía un primer SMS al teléfono destino, junto con las coordenadas del **GPS** del teléfono, que podrían ser Lat=0 Long=0 si no estaba el GPS en funcionamiento, pero esta acción pone a trabajar al **GPS**, para que luego de acabado el **ClockPARA\_SMS\_Coord**, vuelva a enviar de nuevo las coordenadas, posiblemente correctas en esta ocasión. Evidentemente se puede mejorar esta APP.

Se pide pensar y agregar mejoramientos a esta APP, como un visualizador de la información que se envía por SMS. No olvide colocar en la pantalla principal el Copyright con el nombre del programador (alumno). Debería mejorar la interfaz grafica. Sea creativo.