Programación Bluetooth con AppInventor

Cristina Díaz García Noviembre 2018



Índice

Ín	ndice general	
1.	Pantalla Inicial	2
2.	Pantalla del Cliente	2
3.	Pantalla del Servidor	5
4.	COMUNICACIÓN CLIENTE-SERVIDOR	8

1. Pantalla Inicial

En la pantalla inicial tenemos dos botones, con los que iniciar el servidor o conectarse a uno ya iniciado.



2. Pantalla del Cliente

En la pantalla del cliente tenemos dos botones en la cabecera, con los que conectarse y desconectarse, respectivamente. Al tocar conectar, nos saldrá una lista de los dispositivos con los que podemos conectarnos.

Debajo de estos botones, tenemos el estado de la conexión, que por defecto es desconectado.

En la parte de abajo de la pantalla hay una caja de texto en la que podremos escribir el mensaje que queramos escribir, y a su lado hay un botón con el que enviarlo.

Debajo del estado tenemos una pantalla en la que saldrá la conversación que se está teniendo, en la que sale $\it Yo$ para los mensajes enviados y $\it Servidor$ para los mensajes recibidos.



Desconectado



Los bloques quedarían así:

```
when ListPicker1 v . Elements v to BluetoothClient1 v . AddressesAndNames v

when Button3 v .Click

do tif BluetoothClient1 v . IsConnected v then call BluetoothClient1 v .SendText

text TextBox2 v . Text v to set TextBox1 v . Text v to set TextBox2 v . Text v

set TextBox2 v . Text v to v ... v ...
```

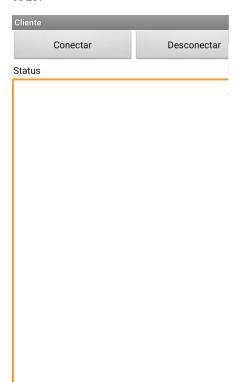
Esta pantalla tiene, a mi parecer, unos cuantos problemas de implementación que no he sabido arreglar, y son:

- Se puede editar la pantalla de texto de la conversación.
- Se pueden "enviar" mensajes aunque no haya conexión (no se envían por Bluetooth, pero sí al chat)

Lo intenté arreglar con estos bloques:

```
ListPicker1 ▼ .BeforePicking
                       set ListPicker1 ▼ . Elements ▼ to (
                                                         BluetoothClient1 v
                                                                           AddressesAndNames v
                                                                                 Cliente v
                                                                                set TextBox2 ▼ . Enabled ▼ to false ▼
  when Button3 ▼ .Cl
                                                                                set TextBox2 ▼ . Visible ▼ to true ▼
                  BluetoothClient1 v . IsConnected v = v false v
                                                                                set Button3 ▼ . Enabled ▼ to false ▼
             call BluetoothClient1 ▼ .SendText
                                                                                set Button3 ▼ . Enabled ▼ to true ▼
                                       text ( TextBox2 ▼ . Text ▼
                                                                                 set TextBox1 ▼ . Enabled ▼ to false ▼
                                         join TextBox1 v Text v
             set TextBox1 ▼ . Text ▼ to
                                                            " Yo:
                                                                                     hen Button2 ▼ .C
                                                                                        call BluetoothClient1 ▼ .Disconnect
                                                           TextBox2 ▼
                                                                         Text ▼
             set TextBox2 ▼ . Text ▼ to
                                                                                         set TextBox2 ▼ . Enabled ▼ to false ▼
                                                                                         set Button3 ▼ . Enabled ▼ to false ▼
Show Warnings
                                                  when ListPicker1 .AfterPicking
                                                               call BluetoothClient1 ▼ .Connect
                                                                                       address (ListPicker1 ▼
                                                           set Label1 ▼ . Text ▼ to
                                                                                       " Conectado "
                                                      set TextBox2 ▼ . Enabled ▼ to true ▼
when Clock1 ▼ .Time
                                                      set Button3 ▼ . Enabled ▼ to true ▼
          BluetoothClient1 ▼ . IsConnected ▼
                  call BluetoothClient1 .BytesAvailableToReceive
                set TextBox1 v . Text v to ( join TextBox1 v . Text v
                                                                   Servidor:
                                                                  call BluetoothClient1 ▼ .ReceiveText
                                                                                                     call BluetoothClient1 ▼ .BytesAvailab
```

Y me dio fallos de poder editarlos aun así, no poder enviar mensajes cuando en teoría sí se debería poder o que no aparezcan las cajas de texto de abajo del todo.



3. Pantalla del Servidor

En la pantalla del servidor tenemos un botón para desconectar el servidor (que nos lleva a la pantalla inicial), una caja de texto con la conversación, una caja de texto para el mensaje que queramos enviar y un botón para hacerlo.



```
Escribe aquí tu mensaje Enviar
```

Los bloques quedarían así:

```
when Sutton1 v. Click
do call BluetoothServer1 v. Disconnect
open another screen screenName v. Screen1 v.

when Clock1 v. Timer
do if BluetoothServer1 v. IsConnected v.
then if call BluetoothServer1 v. Servidor. Text v.

bytesAvailable To Receive v. Text v.

call BluetoothServer1 v. Servidor. Text v.

call BluetoothServer1 v.

call BluetoothServer1 v. Servidor. Text v.

c
```

Al igual que en el cliente, los fallos que veo son que hay bloques de texto que no deberían ser editables y lo son. Lo inteneté solventar con los siguientes bloques:

```
when Button1 \( \text{Click} \)
do call \( \text{BluetoothServer1 \( \text{Point} \) Disconnect open another screen screenName \( \text{Screen1 \( \text{Point} \) Screen1 \( \text{Point} \) AcceptConnection serviceName \( \text{Point} \) another screen screenName \( \text{Point} \) Screen1 \( \text{Point} \) ServiceName \( \text{Point} \) AcceptConnection serviceName \( \text{Point} \) mychal \( \text{Point} \) when \( \text{Clock1 \( \text{Point} \) . Timer \( \text{Coll BluetoothServer1 \( \text{Point} \) . Enabled \( \text{Point} \) to \( \text{Point} \) and \( \text{Point} \) if \( \text{Click Double of the normal serviceName \( \text{Point} \) if \( \text{Point} \) and \( \text{Point} \) if \( \text{Point} \) and \( \text{Point} \) if \( \text{Point} \) is \( \text{Point} \) i
```

Y me daba los mismos problemas que en el cliente:



4. COMUNICACIÓN CLIENTE-SERVIDOR

Esta es la vista desde el cliente:





Y esta es la vista desde el servidor:



Como se puede ver, otro de los fallos existentes es que los mensajes no saltan de línea al escribirse.