



TECNICATURA SUPERIOR EN
Telecomunicaciones

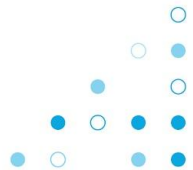
Arquitectura y conectividad

Trabajo Práctico N°6

Grupo N°1

INTEGRANTES

- Fernando Gimenez Coria
- Nicolás Nahuel Barrionuevo
- Macarena Aylen Carballo
 - Raul Antonio Jara
 - Diego Ezequiel Ares
- Juan Diego González Antoniazzi

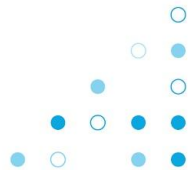


Proyecto 2

Implementar un proyecto con ESP32, mediante el cual puedan realizar una conexión con un Smartphone.

La consigna es encender/apagar 2 Led's y comando por voz de los mismos, además poder ver la variación de voltaje de un potenciómetro usando protocolo de comunicaciones Bluetooth.

La comunicación es entre el Smartphone o entorno virtual (PC) y el ESP32.



Codigo

```
1  #include <Arduino.h>
2  #include <BluetoothSerial.h>
3  #define L_PATIO 32
4  #define L_FRENTE 33
5
6  static unsigned long ultimoEnvio = 0;
7  const unsigned long intervalo_lectura = 5000;
8
9  BluetoothSerial controladorBT;
10
11
```

```
12 void setup() {
13     Serial.begin(115200);
14     controladorBT.begin("SKYNET");
15     Serial.println("El dispositivo se ha inicializado y esta listo para conectarse");
16     pinMode(L_PATIO, OUTPUT);
17     digitalWrite(L_PATIO, LOW);
18     pinMode(L_FRENTE, OUTPUT);
19     digitalWrite(L_FRENTE, LOW);
20 }
21
```

```
22 void loop() {
23     if (controladorBT.available() > 0){
24         String mensaje = controladorBT.readStringUntil('\n'); //lee el strig proveniente del telefono hasta el caracter de nueva línea
25         mensaje.trim(); //recorta espacios y saltos
26         Serial.println("Comando Recibido");
27         Serial.println(mensaje);
28
29         if (mensaje.equalsIgnoreCase("encender lámpara patio")){
30             digitalWrite(L_PATIO, HIGH);
31             Serial.println("Lámpara patio encendida");
32         }
33         else if (mensaje.equalsIgnoreCase("apagar lámpara patio")){
34             digitalWrite(L_PATIO, LOW);
35             Serial.println("Lámpara patio apagada");
36         }
37         if (mensaje.equalsIgnoreCase("encender lámpara frente")){
38             digitalWrite(L_FRENTE, HIGH);
39             Serial.println("Lámpara frente encendida");
40         }
41         else if (mensaje.equalsIgnoreCase("apagar lámpara frente")){
42             digitalWrite(L_FRENTE, LOW);
43             Serial.println("Lámpara frente apagada");
44         }
45         else if (mensaje.equalsIgnoreCase("informar intensidad de luz")){
46             int LDR = map(analogRead(34), 0, 4096, 0, 100);
47             Serial.print("La intensidad luminica es de: ");
48             Serial.print(LDR);
49             Serial.println("%");
50             controladorBT.println("La intensidad luminica es de: " + String(LDR)+ "%");
51         }
52         else {
53             Serial.println("Comando incorrecto");
54         }
55
56         if (millis() - ultimoEnvio > intervalo_lectura){
57             int LDR = map(analogRead(34), 0, 4095, 0, 100);
58             String luz = "La intensidad luminica es de: " + String(LDR) + "%\n";
59             controladorBT.println(luz);
60             Serial.println("Enviado a app: " + luz);
61             ultimoEnvio = millis();
62         }
63     }
64 }
```



Codigo de MIT App Inventor

←

controlPorVozBT

Integrados

Control

Lógica

Matemáticas

Texto

Listas

Diccionarios

Colores

Variables

Procedimientos

Screen1

HorizontalArrangement

VerticalArrangement

Cambiar nombre

Borrar

Medios

apagado.png

bt.png

comando-de-voz.png

encendido.png

Subir archivo...

cuando ListPicker1 . AntesDeSelección

ejecutar

poner ListPicker1 . Elementos como BluetoothClient1 . DireccionesYNombres

cuando ListPicker1 . DespuésDeSelección

ejecutar

si

llamar BluetoothClient1 . Conectar

dirección ListPicker1 . Selección

entonces

poner ListPicker1 . Texto como

unir Conectado a

ListPicker1 . Selección

poner BluetoothClient1 . ByteDelimitador como

else

poner ListPicker1 . Texto como No se pudo conectar

cuando PATIO_ON . Clic

ejecutar

si

BluetoothClient1 . Conectado

entonces

llamar BluetoothClient1 . EnviarTexto

texto encender lámpara patio'n

cuando PATIO_OFF . Clic

ejecutar

si

BluetoothClient1 . Conectado

entonces

llamar BluetoothClient1 . EnviarTexto

texto apagar lámpara patio'n

cuando FRETE_ON . Clic

ejecutar

si

BluetoothClient1 . Conectado

entonces

llamar BluetoothClient1 . EnviarTexto

texto encender lámpara frente'n

cuando FRETE_OFF . Clic

ejecutar

si

BluetoothClient1 . Conectado

entonces

llamar BluetoothClient1 . EnviarTexto

texto apagar lámpara frente'n

cuando VOICE_COM . Clic

ejecutar

llamar SpeechRecognizer1 . ObtenerTexto

cuando SpeechRecognizer1 . DespuésDeObtenerTexto

Resultado parcial

ejecutar

si

BluetoothClient1 . Conectado

entonces

llamar BluetoothClient1 . EnviarTexto

texto tomar Resultado

poner Lbl_Com . Texto como

Comando enviado'n

cuando Clock1 . Temporizador

ejecutar

si

BluetoothClient1 . Conectado

entonces

si

llamar BluetoothClient1 . BytesDisponiblesParaRecibir

>= 0

entonces

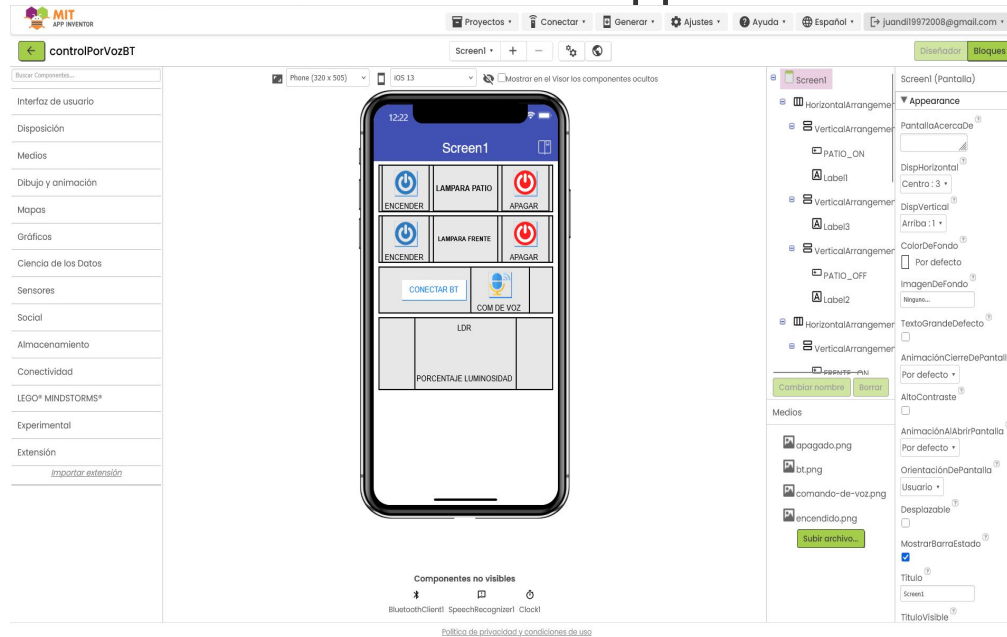
poner LDR_Lbl . Texto como

llamar BluetoothClient1 . RecibirTexto

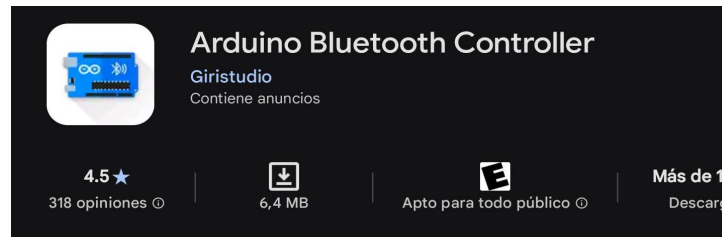
númeroDeBytes 1

ISPC / Tecnicatura Superior en Telecomunicaciones

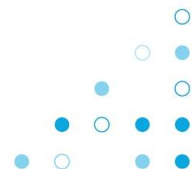
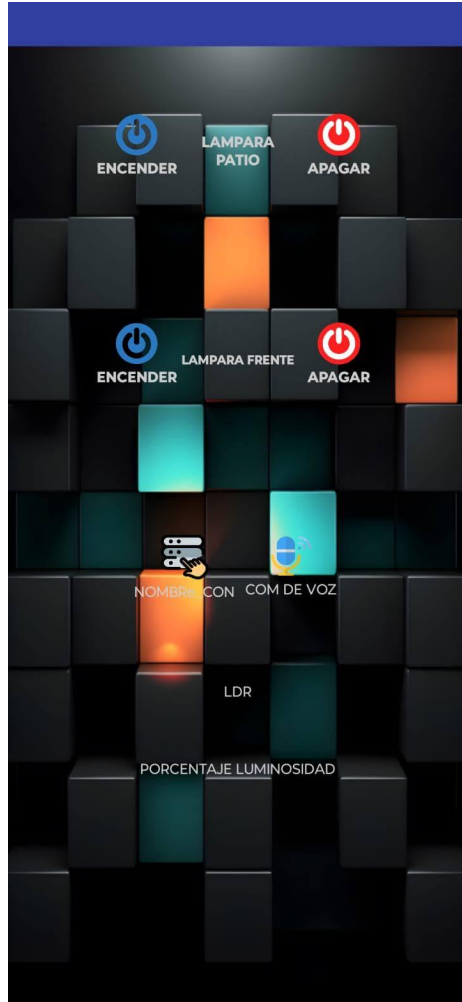
Interfaz desde MIT App Inventor



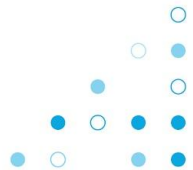
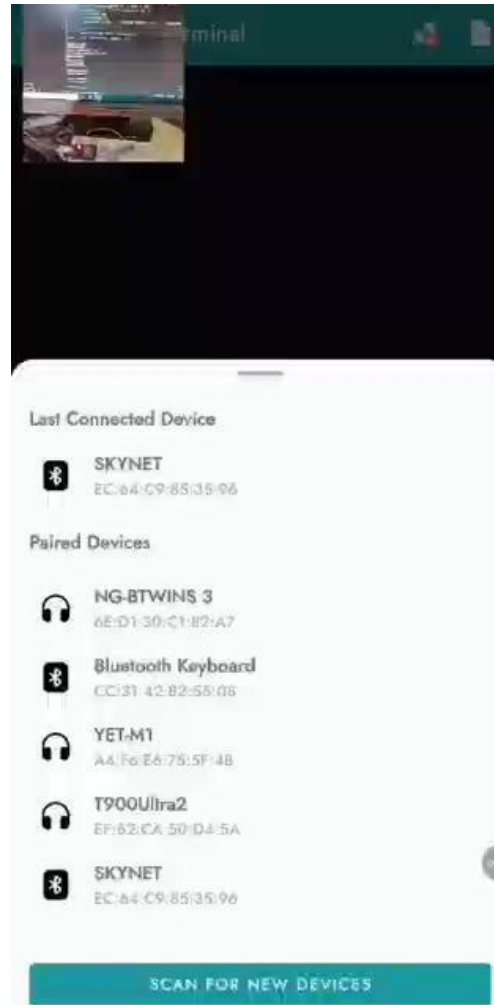
APP de Android



Diseño desde el celular



Funcionamiento comando por voz





¡Muchas gracias!