**MENSAJES LCD VIA BLUETOOTH.**

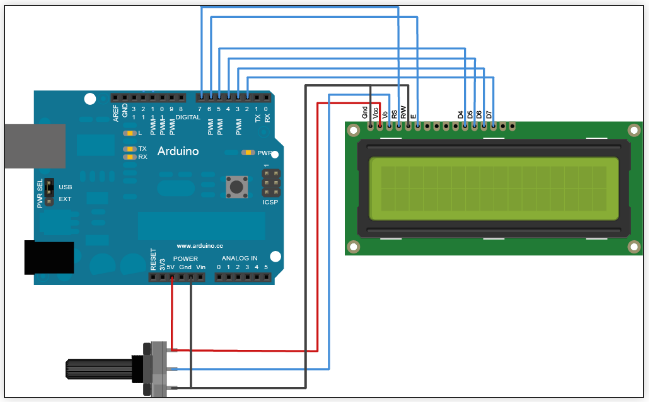
Dentro de lo que podemos hacer con Arduino como amantes de la electrónica, es imprescindible hoy en día que nuestros proyectos tengan una comunicación con los dispositivos móviles, ya sean tabletas, teléfonos móviles inteligentes y computadoras portátiles.

En este ensayo vamos a realizar una comunicación a través de un módulo Bluetooth HC-05 entre un Arduino Uno y un teléfono móvil inteligente y mostraremos los mensajes que sean enviados desde el teléfono en un LCD de 2X16 conectado al Arduino Uno.

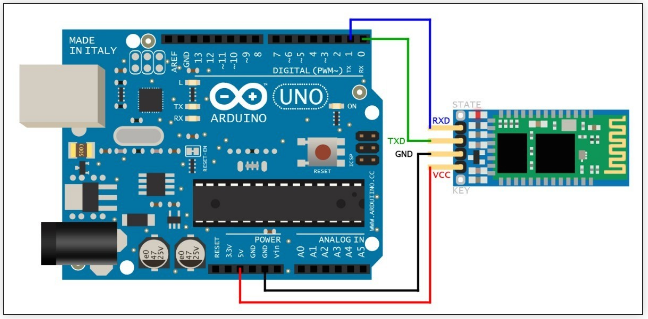
Para comenzar describiré a continuación lo que necesitamos para el desarrollo:

* Modulo Bluetooth HC-05,
* Arduino Uno,
* Telefono inteligente con sistema operativo Android,
* Un LCD 2X16,
* Potenciómetro de 5000 ohmios y
* Cables de conexión.

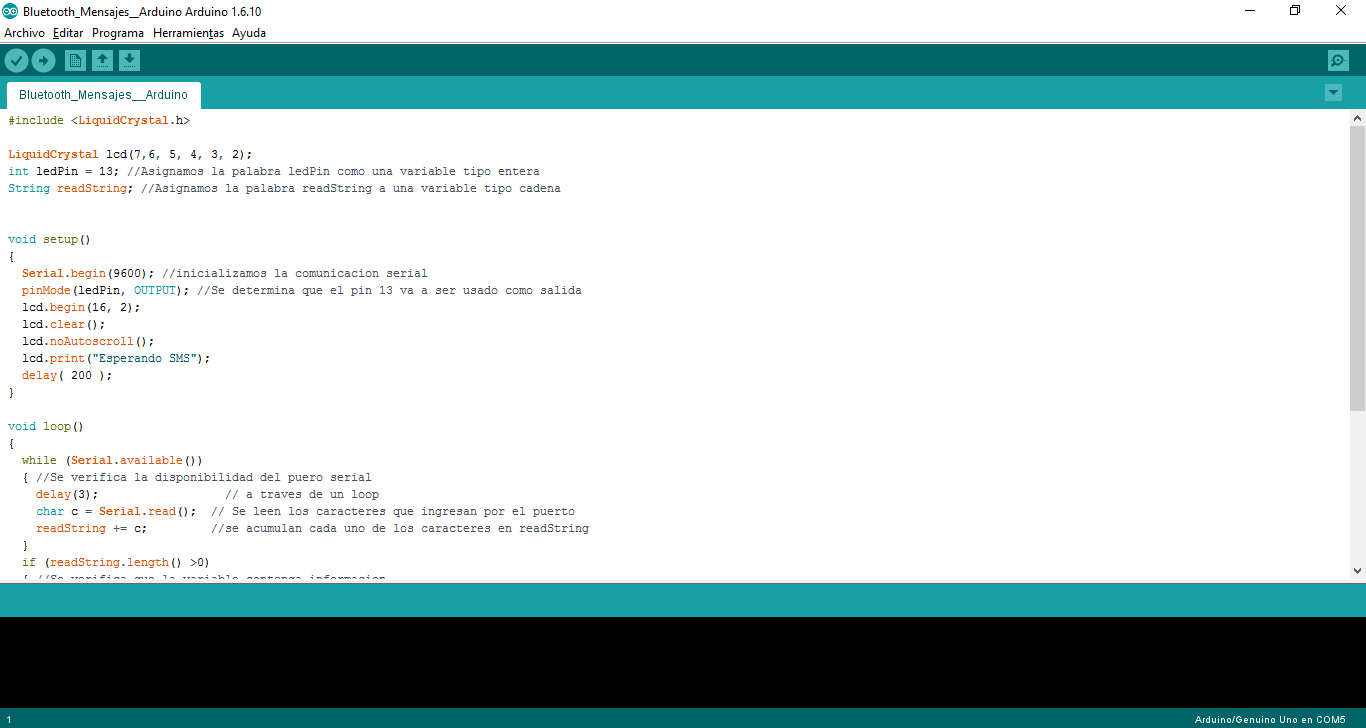
Para controlar el LCD 2X16, he realizado la conexión de los pines de la siguiente manera para el caso de un Arduino Uno usamos los pines 2 al 7 y VCC y GND de la tarjeta Arduino.



Para el caso del módulo bluetooth, este se conecta de la siguiente manera, con los pines 0 y 1 del Arduino conectados al TX y RX del módulo Bluetooth y el modulo alimentado por VCC (5V) y GND del Arduino.



El código que he implementado en el Arduino es muy sencillo, e incluye únicamente la configuración del LCD y la comunicación serial a través de la cual podemos controlar la comunicación con el modulo Bluetooth.



#include <LiquidCrystal.h>

LiquidCrystal lcd(7,6, 5, 4, 3, 2);

int ledPin = 13; //Asignamos la palabra ledPin como una variable tipo entera

String readString; //Asignamos la palabra readString a una variable tipo cadena

void setup()

{

Serial.begin(9600); //inicializamos la comunicacion serial

pinMode(ledPin, OUTPUT); //Se determina que el pin 13 va a ser usado como salida

lcd.begin(16, 2);

lcd.clear();

lcd.noAutoscroll();

lcd.print("Esperando SMS");

delay( 200 );

}

void loop()

{

while (Serial.available())

{ //Se verifica la disponibilidad del puero serial

delay(3); // a traves de un loop

char c = Serial.read(); // Se leen los caracteres que ingresan por el puerto

readString += c; //se acumulan cada uno de los caracteres en readString

}

if (readString.length() >0)

{ //Se verifica que la variable contenga informacion

lcd.clear();

Serial.println(readString); //se envia lo que tenga la variable

lcd.print(readString);

if (readString == "On") //se compara si la palabra que llega es "on"

{

digitalWrite(ledPin, HIGH); //Si es "on" se enciende el led del pin 13

}

if (readString == "Off") // Se compara si la palabra que llega es "off"

{

digitalWrite(ledPin, LOW); //si es "off" se apaga el led del pin 13

}

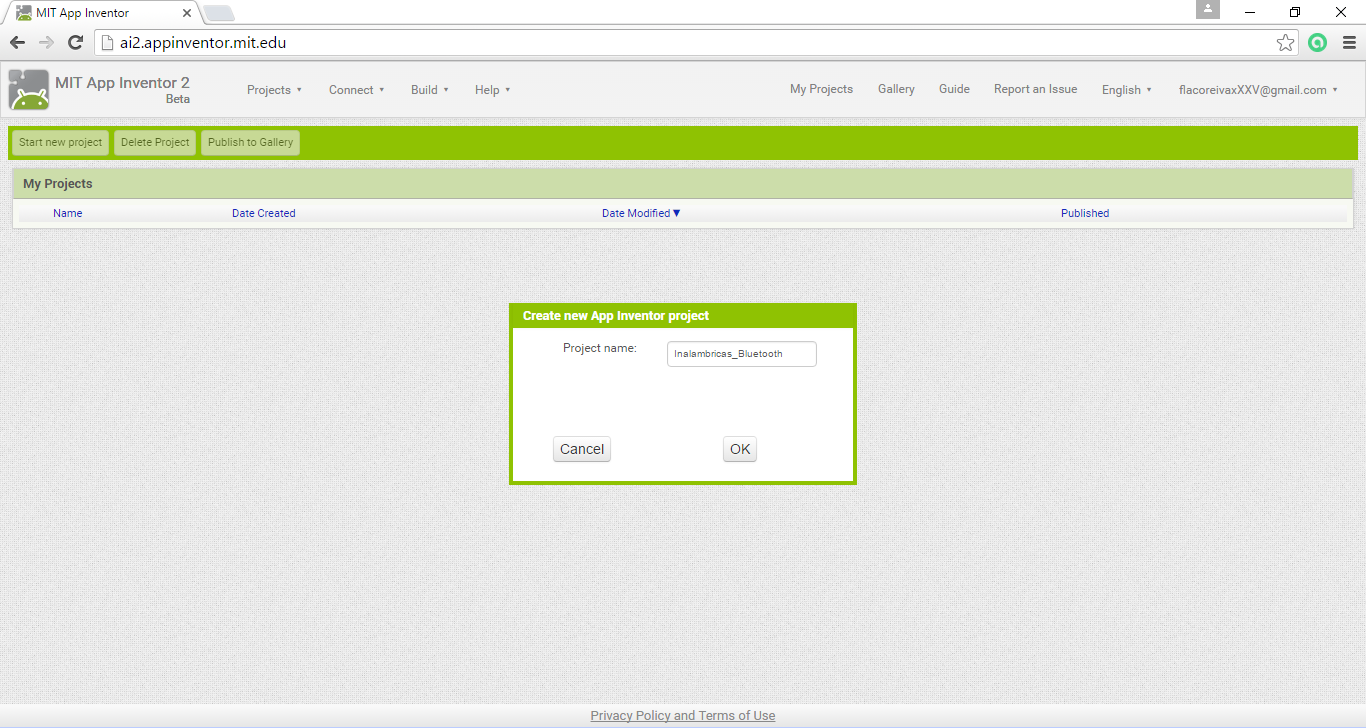
readString=""; // Se limpia la variable

}

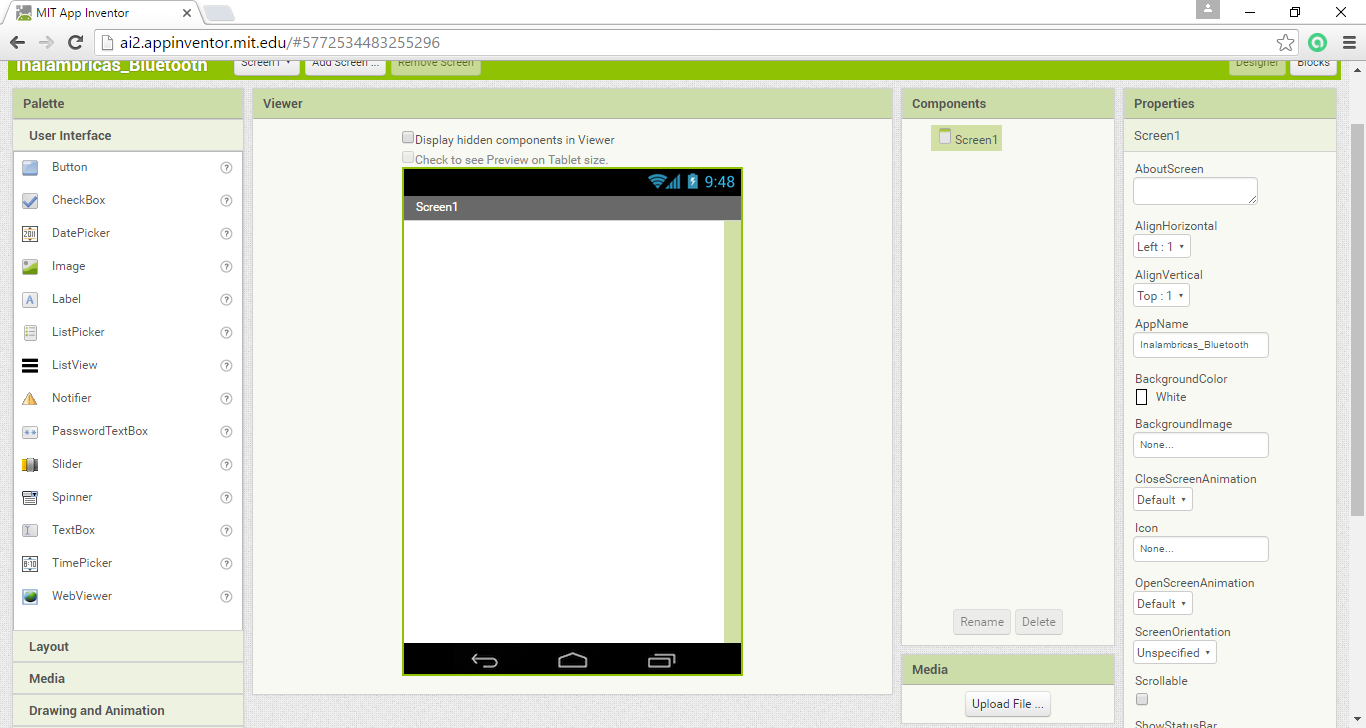
}

Ahora la implementación de la Aplicación Móvil la realizamos haciendo uso de un software online creado por Google Labs y en mi concepto es de las mejores formas de programar aplicaciones móviles que se han desarrollado, por su flexibilidad, sencillez y facilidad.

E App Inventor, en una aplicación web desarrollada por Google Labs en asociación con el MIT. Es una aplicación desarrollada para los que no están familiarizados con la programación tradicional y se basaron en investigaciones de informática educativa, de allí se deriva su simplicidad.



Primer paso es el agregar la interfaz gráfica.



Agregamos el código ara cada elemento:.

