# TUTORIAL

## CYTRON UNO WITH favoriot PLATFORM

*“Monitor your data easily by connecting your IoT board to our IoT platform”*

**FAVORIOT SDN. BHD.**

Suite 28, 6th Floor.
IOI Business Park, 47100
Puchong ,Selangor, Malaysia

Website : www.favoriot.com
Email: info@favoriot.com

# Contents

*"This tutorial will teach you on how to create an Arduino program that uses temperature sensor and push button while connecting to FAVORIOT platform. It is important for you to have your own account in the FAVORIOT platform before trying out this tutorial."*

# FAVORIOT Platform

FAVORIOT a middleware platform specifically designed for any Internet of Things (IoT) and Machine to Machine (M2M) solutions. The platform is developed to support the integration of data from various sensors, actuators and other sources of data. Collecting and storing data from IOT devices become much easier. Moreover, the platform also helps developers to build vertical applications. Develops does not need to worry about hosting and storing the data generated by their IoT devices. You may subscribe to our FAVORIOT platform through this link http://favoriot.com/pricing.

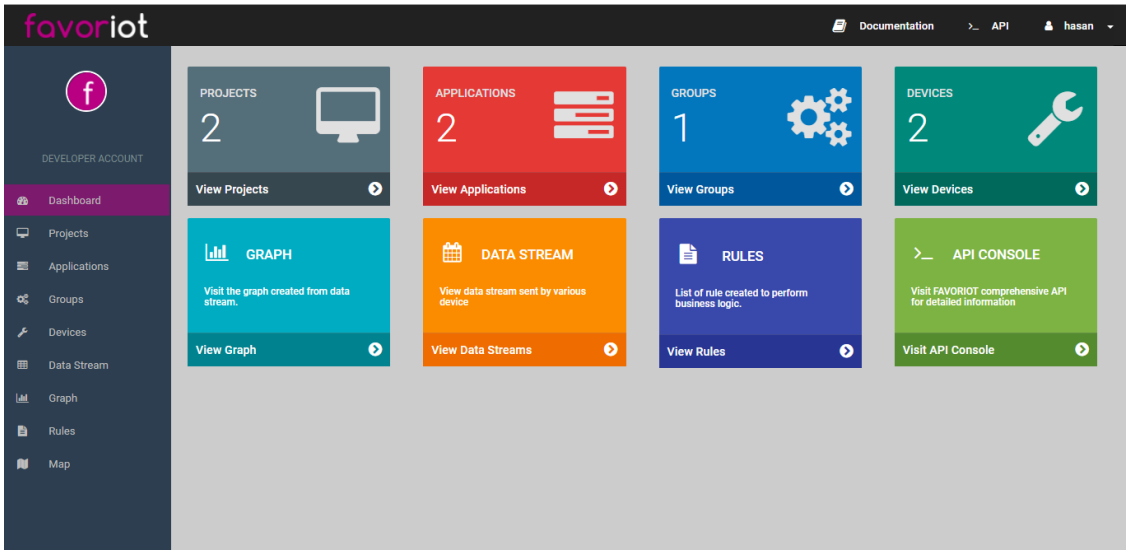



Figure 1 : FAVORIOT Platform Dashboard

# Cytron UNO

CT UNO is the Cytron version of Arduino Uno. In other words, it is Arduino Uno compatible with improvements. The CT UNO combines the simplicity of the UNO's Optiboot bootloader (which loads program faster), the stability of the FTDI and the R3 shield compatibility of the latest Arduino Uno R3.

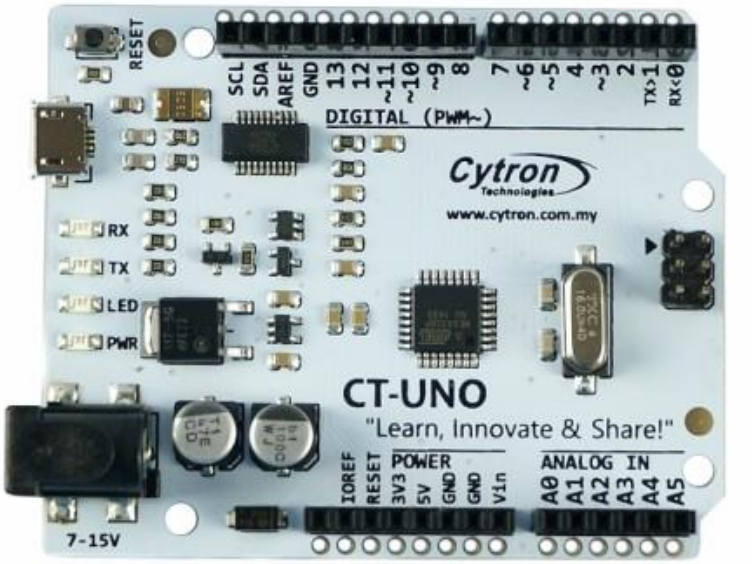
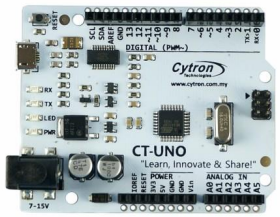
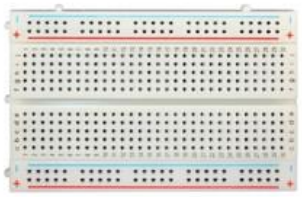
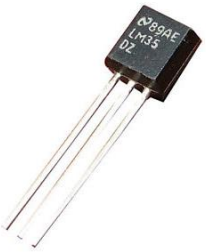



Figure 2 : Cytron UNO Board

## CT-UNO Features

- SMD ATmega328 microcontroller with Optiboot (UNO) Bootloader.
- USB Programming Facilitated by the Ubiquitous FTDI FT231X (more stable).
- Input voltage: DC715V.
- 1A (maximum) 5V voltage regulator.
- 500mA (maximum) 3.3V voltage regulator.
- 05V outputs with 3.3V compatible inputs.
- 14 Digital I/O Pins (6 PWM outputs).
- 6 Analog Inputs.
- ISP 6pin Header.
- 32k Flash Memory.
- 16MHz Clock Speed.
- R3 Shield Compatible.
- TX, RX, Power, pin 13 LEDs are moved to edge.
- Utilize USB MicroB socket.

# Components

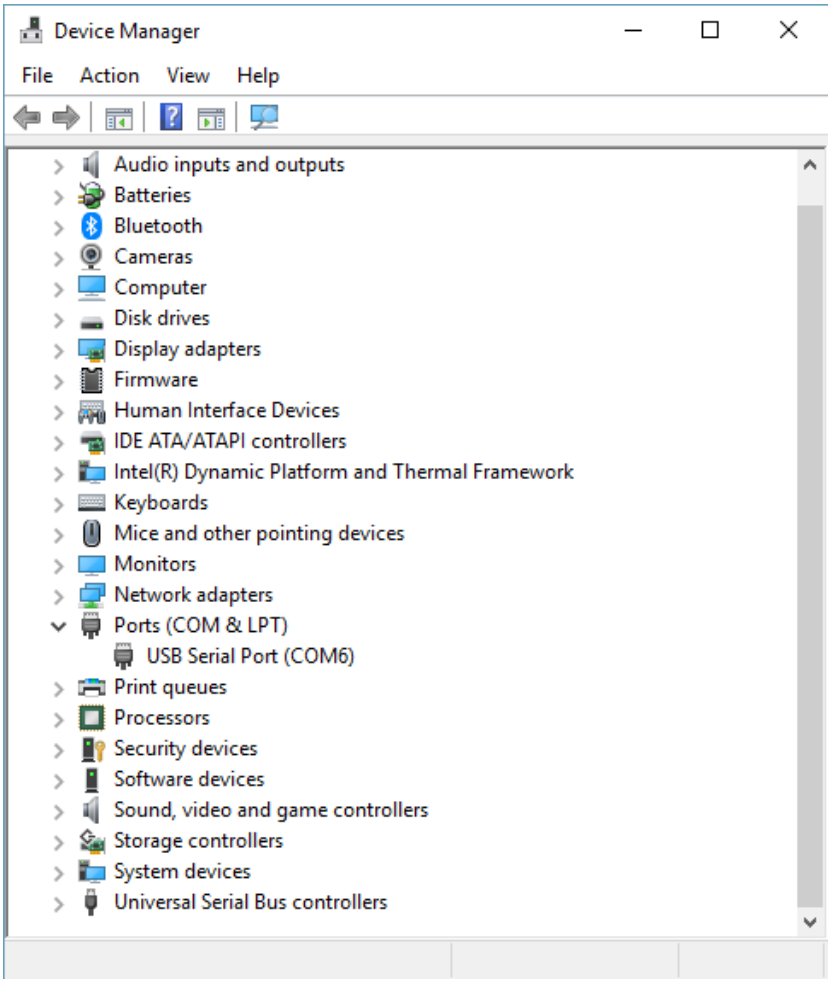
| COMPONENT | |
|---|---|
|  Cytron UNO board – (1) |  Breadboard – (1) |
|  Cytron WiFi Shield – (1) |  USB MicroB Type Cable – (1) |
|  Resistor 1.25W 5% (10kΩ) – (1) |  6x6x1 Push Button 2 Pins – (1) |
|  Male to Male Jumper – (7) |  Temperature Sensor LM35 – (1) |

# Arduino IDE

This tutorial requires to run Arduino IDE to write and upload the code into the CT-UNO board. This is an open-source software that can be used with any Arduino board. Download link is available at (https://www.arduino.cc/en/Main/Software).

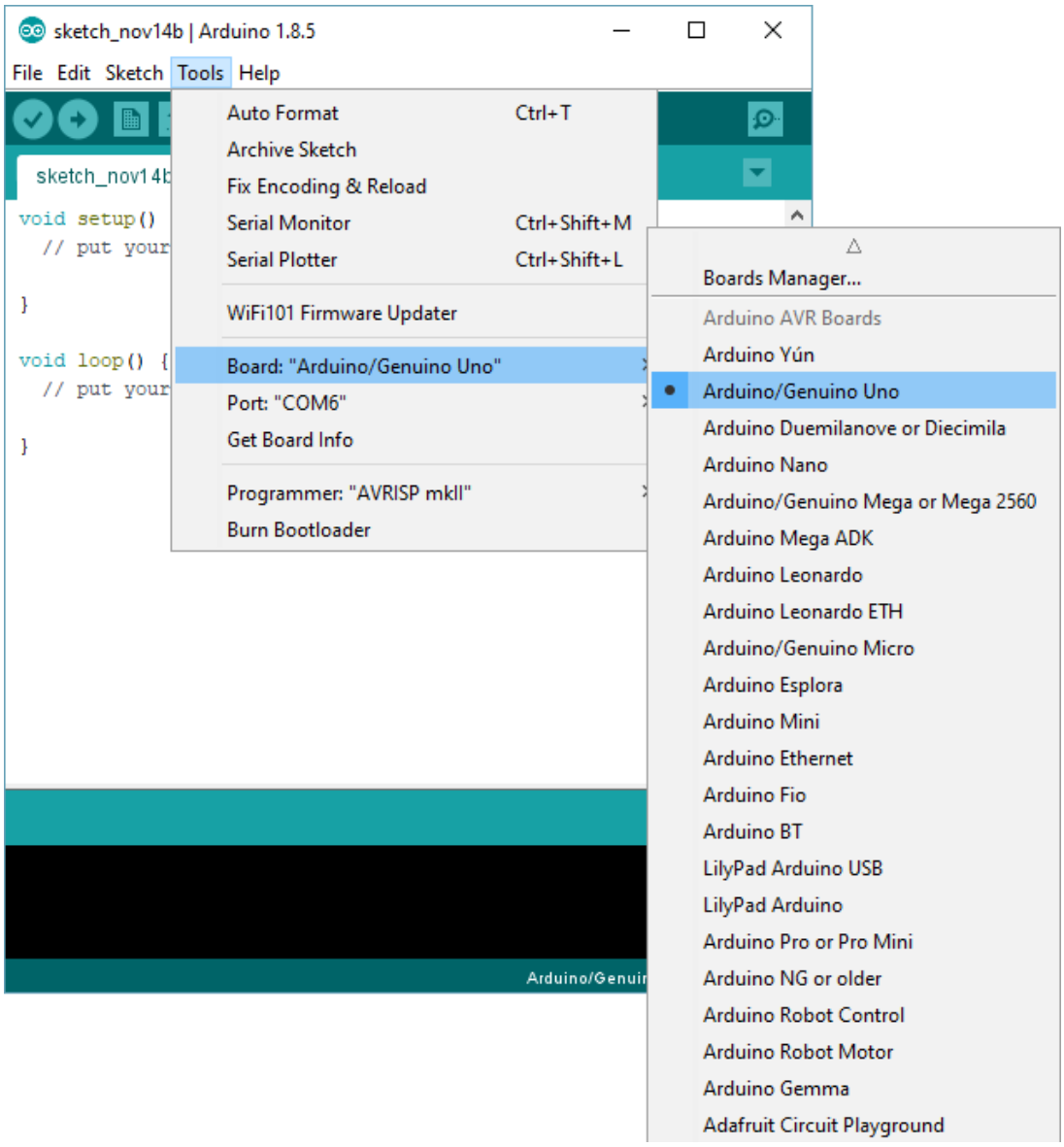
## Setup Arduino IDE with CT-UNO

1. Connect CT-UNO board to your computer.
2. Open Device Manager and check whether the board is recognized as USB Serial Port with COM port number, e.g. *USB Serial Port (COM6).*




** If you get an unrecognized device, please download and install the latest FTDI driver from (http://www.ftdichip.com/Drivers/VCP.htm) or you can try connect your board to a different USB ports in your computer.

3. Open the Arduino IDE software > Tools > Board: "***Arduino/genuine Uno***" and set Serial Port: "***Port that connected to your board***".




4. Now, you can start write the code. Once written, click on the Verify button and Upload code to your board. You will be able to see the output through the Serial Monitor.

# Setting Up the Components

The components used are listed in the previous section. Please set up the hardware as shown in the diagram below:

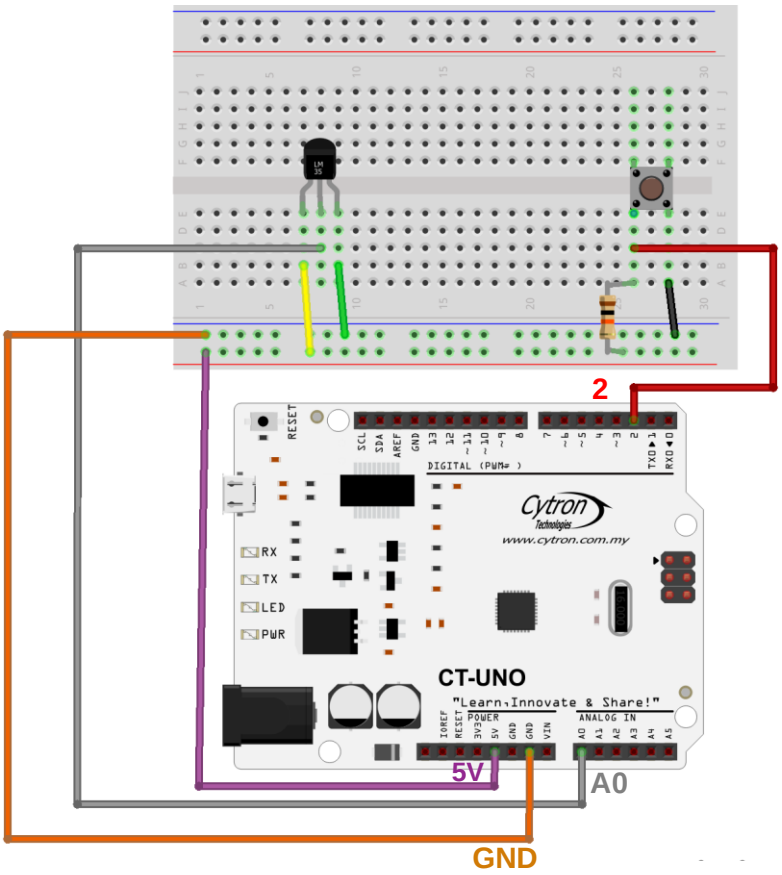



Figure 3 : Breadboard View

# Write the Code

## Temperature Sensor with Push Button

This code is mainly for testing out the program without connecting to any internet connection and favoriot platform. This program will read the current temperature and display its value once you pressed the button. You may skip this part and continue to the complete code.

**Arduino Code:**

```
#define sensorPin A0
#define pushButton 2
int sensorValue = 0;
int celcius = 0;

void setup()
{
  Serial.begin(9600);
  pinMode(pushButton, INPUT);
  pinMode(sensorPin, INPUT);
  Serial.println("Press to get temperature");

}

void loop()
{
  if(digitalRead(pushButton) == LOW)  // Execute when push button is pressed
  {
     // Read data from temperature sensor and convert ADC value to celcius
     celcius = analogRead(sensorPin) * 0.488;
     // Display current temperature
     Serial.println("Temperature : " + String(celcius) + " Celcius");
     delay(1000);
  }
}
```

## Wi-Fi Shield Demo

In order to connect to FAVORIOT platform, internet connection is a need. In this case, we are using Cytron Wi-Fi Shield to support wireless internet connectivity. Make sure you attached the Wi-Fi Shield to the Arduino board and install the Cytron Wi-Fi shield library. This demo is optional, you may skip this part and continue to the complete code.

**How to install:**

- Arduino IDE > Sketch > Include Library > Manage Libraries. Search for **Cytron ESPWiFi Shield** and install.

**Test out Wi-Fi connection:**

- Arduino IDE > File > Examples > Cytron ESPWiFi Shield > CytronWiFiDemo.
- In the void setup(), comment the ***wifi.config(ip);*** line by insert **//** symbols in front of the line and continue with compiling and uploading the code to test out the connection.

**Arduino Code (*void setup() with changes*):**

```
void setup() {

  // put your setup code here, to run once:
  Serial.begin(9600);
  while (!Serial) {
    ; // wait for serial port to connect. Needed for Leonardo only
  }
  if(!wifi.begin(2, 3))
  {
    Serial.println(F("Error talking to shield"));
    while(1);
  }
  Serial.println(wifi.firmwareVersion());
  Serial.print(F("Mode: "));Serial.println(wifi.getMode());// 1- station
  mode, 2- softap mode, 3- both
  Serial.println(F("Setup wifi config"));
  //wifi.config(ip);
  Serial.println(F("Start wifi connection"));
  if(!wifi.connectAP(ssid, pass))
  {
    Serial.println(F("Error connecting to WiFi"));
    while(1);
  }
  Serial.print(F("Connected to "));Serial.println(wifi.SSID());
  Serial.println(F("IP address: "));
  Serial.println(wifi.localIP());
  wifi.updateStatus();
  Serial.println(wifi.status()); //2- wifi connected with ip, 3- got
  connection with servers or clients, 4- disconnect with clients or servers,
  5- no wifi
  clientTest();
  espblink(100);
  server.begin();  }
```

## Connecting Arduino program with FAVORIOT platform

You have to combine the codes from Arduino program and Wi-Fi demo. Then, add some codes to enable your Arduino board to send the data retrieved from the sensors to the FAVORIOT platform. The complete code has several changes where some parts of original codes are excluded. There are instructions included in the code for your reference.

**Arduino Code** (*code needed to connect to the FAVORIOT platform*):

```
String apikey = "Your API key";     // replace with your api key from the
FAVORIOT platform account setting
char serverAdd[] = "apiv2.favoriot.com";
datatype yourData = " " ;

void loop() {
{
   dataStream();
}

void dataStream()
{
  // Json Data to send to Platform
  String json = "{\"device_developer_id\":\"insert your device_id from
FAVORIOT platform here \",\"data\":{\"Data\":\""+String(yourData)+"\"}}";

  if (client.connect(serverAdd, 80)) {
    // Make a HTTP request:
    client.println("POST /v2/streams HTTP/1.1");
    client.println("Host: apiv2.favoriot.com");
    client.print(String("apikey: "));
    client.println(apikey);
    client.println("Content-Type: application/json");
    client.println("cache-control: no-cache");
    client.print("Content-Length: ");
    int thisLength = json.length();
    client.println(thisLength);
    client.println("Connection: close");

    client.println();
    client.println(json);
    }
}
```

**Arduino Code** (*complete code*):

```
#include <CytronWiFiShield.h>
#include <CytronWiFiClient.h>
#include <CytronWiFiServer.h>
#include <SoftwareSerial.h>
#define sensorPin A0
#define pushButton 2

const char *ssid = "Your Wi-Fi SSID";  // replace the SSID
const char *pass = "Your Wi-Fi password";  // replace the password
IPAddress ip(192, 168, 1 ,242);
ESP8266Server server(80);
ESP8266Client client;

int sensorValue = 0;
int celcius = 0;
unsigned long start, finished, elapsed;
String apikey = "Your API key";    // replace with your api key from the
FAVORIOT platform account setting
char serverAdd[] = "apiv2.favoriot.com";

void setup() {

  // put your setup code here, to run once:
  Serial.begin(9600);
  pinMode(pushButton, INPUT);
  pinMode(sensorPin, INPUT);

  while (!Serial)
  {
    ; // wait for serial port to connect. Needed for Leonardo only
  }

  if(!wifi.begin(2, 3))
  {
    Serial.println(F("Error talking to shield"));
    while(1);
  }

  Serial.println(wifi.firmwareVersion());
  Serial.print(F("Mode: "));Serial.println(wifi.getMode());// 1- station
mode, 2- softap mode, 3- both
  Serial.println(F("Setup wifi config"));
  //wifi.config(ip); //only if you have a static ip
  Serial.println(F("Start wifi connection to"));
  Serial.print(F("Wifi SSID: "));Serial.println(ssid);

  if(!wifi.connectAP(ssid, pass))
  {
    Serial.println(F("Error connecting to WiFi"));
    while(1);
  }

  Serial.print(F("Connected to "));Serial.println(wifi.SSID());
  Serial.println(F("IP address: "));
  Serial.println(wifi.localIP());
  wifi.updateStatus();
  Serial.println(wifi.status()); //2- wifi connected with ip, 3- got
connection with servers or clients, 4- disconnect with clients or servers, 5-
no wifi
```

```
  clientTest();
  server.begin();
}

void loop() {

  if(digitalRead(pushButton) == LOW) { //disable this line to retrieved data
automatically
     //execute when the push button is pressed only
     celcius = analogRead(sensorPin) * 0.488;  //read the temperature and
convert to celcius
     start=millis();  //start counting time
     dataStream(celcius);  //send data to FAVORIOT platform
     finished=millis();  //stop counting time
     elapsed=(finished-start) / 1000;  //calculate time taken to send data
     Serial.print("        PROCESSING TIME : " + String(elapsed) + "
seconds\n");  //display time taken
     Serial.print("*************************************************");
  }
//delay(1000); //enable this line if you disable the 'if' condition
}

void dataStream(int celcius)
{
  // Json Data to send to Platform
  String json = "{\"device_developer_id\":\"insert your device_id from
FAVORIOT platform here\",\"data\":{\"Temperature\":\""+String(celcius)+
"\"}}";
  Serial.println("\n        TEMPERATURE : " + String(celcius)+ " Celcius");
// display temperature value

  if (client.connect(serverAdd, 80)) {
    // Make a HTTP request:
    Serial.println("        STATUS : Sending data...");  //Display sending
status
    client.println("POST /v2/streams HTTP/1.1");
    client.println("Host: apiv2.favoriot.com");
    client.print(String("apikey: "));
    client.println(apikey);
    client.println("Content-Type: application/json");
    client.println("cache-control: no-cache");
    client.print("Content-Length: ");
    int thisLength = json.length();
    client.println(thisLength);
    client.println("Connection: close");

    client.println();
    client.println(json);
    Serial.println("        STATUS : Data sent!");  //display sent status
    }
}

void clientTest()
{
  const char destServer[] = "www.google.com";
  ESP8266Client client;

  Serial.print(F("Starting connection to: "));Serial.println(destServer);
  Serial.print("\n*************************************************");
  if (!client.connect(destServer, 80))
  {
```

```
    Serial.println(F("Failed to connect to server."));
    client.stop();
    return;
  }
   while (client.available()>0)
  {
    Serial.write(client.read());
  }

  if (!client.connected()) {
    client.stop();
  }
}
```

## Download Link

You may download the codes at https://github.com/Favoriot/MyFirstArduinoProject.

# Data Stream

Log in to your FAVORIOT platform account at https://platform.favoriot.com/# and click on the Data Stream. Once the data has been sent, you will be able to see the records in the table as shown in the figure below.

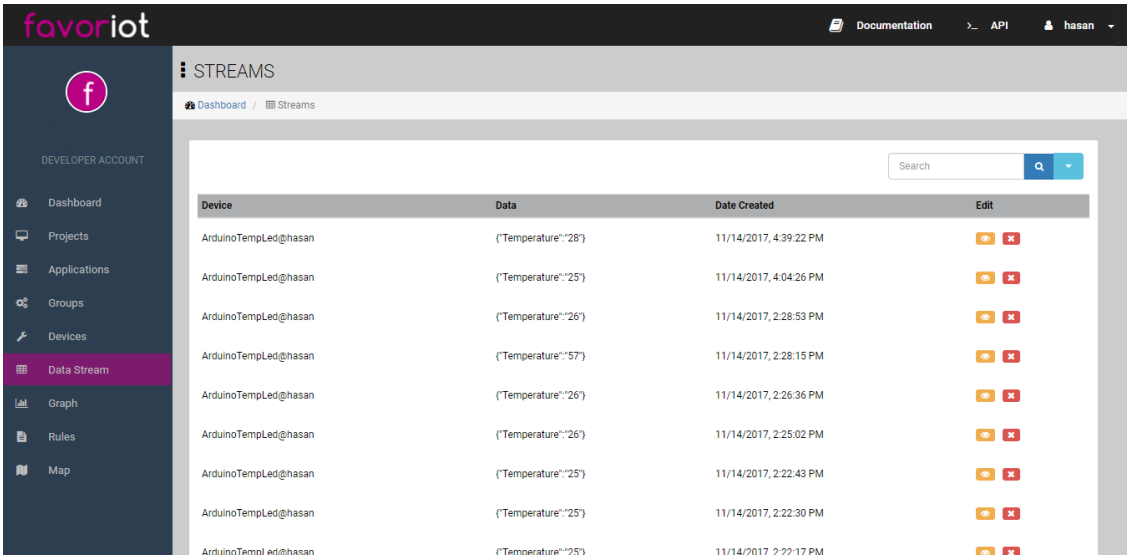



Figure 4 : Data Stream Page


Figure 5 : Data Details

# Company Information

**FAVORIOT SDN. BHD.**

Suite 28, 6th Floor.
IOI Business Park, 47100
Puchong ,Selangor, Malaysia

Website : www.favoriot.com
Email: info@favoriot.com

favoriot