Workshop: Build Your First IoT Project!

http://makerden.io

November 1, 2015

Cisco ‘MakerBro’ Zabala

About Makerden

We’re an Educational Makerspace where we combine the necessary tools, equipment, and instruction for hands-on learning through building things.

Members of our group vary both widely and wildly in terms of skill level and age range (10 to 68).

About Makerden

Events organized at our Educational Makerspace:

[DIY] These events include show-and-tells by our members followed by a work-on-your-project(s) and ask-for-help sessions.

[Workshop] An instructor leads a hands-on workshop for learning or polishing a practical skill useful for Makers (including schematic-drawing, PCB design, 3D-Printing, soldering).

[Class] An instructor leads a formal lecture featuring instructor-led activities that allow participants to learn, develop, or refine a technical skill.

About ACROBOTIC

Makerden's Educational Electronics partner

Small, bootstrapped Open-Source electronics startup dedicated to the design of hardware and software products for use in education, DIY, hobby, arts, science, and more!

- Online store: **https://acrobotic.com**
- Develops online tutorials: **http://learn.acrobotic.com**
- Supplies all the electronic components for this class!
- Helps Makerden members take their ideas from concept to implementation

Downloading this presentation

MakerdenIO/First_IoT_Proje ×

GitHub, Inc. [US] https://github.com/MakerdenIO/First_IoT_Project

MakerdenIO / First_IoT_Project

Unwatch 1 Star 0 Fork 0

Navigate to:
https://github.com/makerdenio/First_IoT_Project

Build an Internet-connected Weather Station using the ESP8266 Development Board! — Edit

13 commits 1 branch 0 releases 1 contributor

Branch: master First_IoT_Project / +

themakerbro updating — Latest commit 1a88405 2 minutes ago

iot_weather_station	updating widgets section	5 days ago
presentation	updating	9 minutes ago
LICENSE.txt	updating	2 minutes ago
README.md	updating	2 minutes ago

README.md

First_IoT_Project

Build an Internet-connected Weather Station using the ESP8266 Development Board!

Description

For more details, check out the tutorial page at:

Code
Issues 0
Pull requests 0
Wiki
Pulse
Graphs
Settings

HTTPS clone URL
https://github.com

You can clone with HTTPS, SSH, or Subversion.

Clone in Desktop
Download ZIP

Intro

Overview of IoT and ESP8266

Project description

What is *the* Internet?

The Internet is a global system of interconnected computer networks that communicate with one another to link **billions** of devices worldwide.

Facebook's Map of World 'Friendship'

What devices are found on *the* Internet?

"interconnected computer networks…"

What is the goal *the* IoT?

Extending the internet to include “everyday objects”.

System Overview

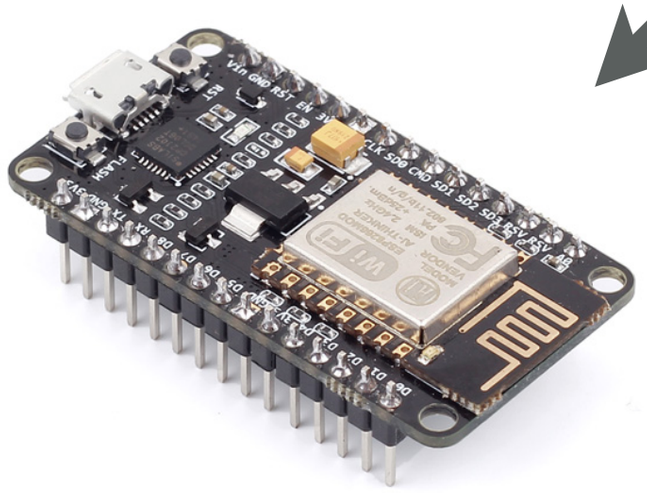
Cloud

Things

Users
&
endpoints

System Overview

Electronics



Backend

Frontend

Fundamental Concepts

Hardware prototyping

Electronic Fundamentals

How the solderless breadboard works:

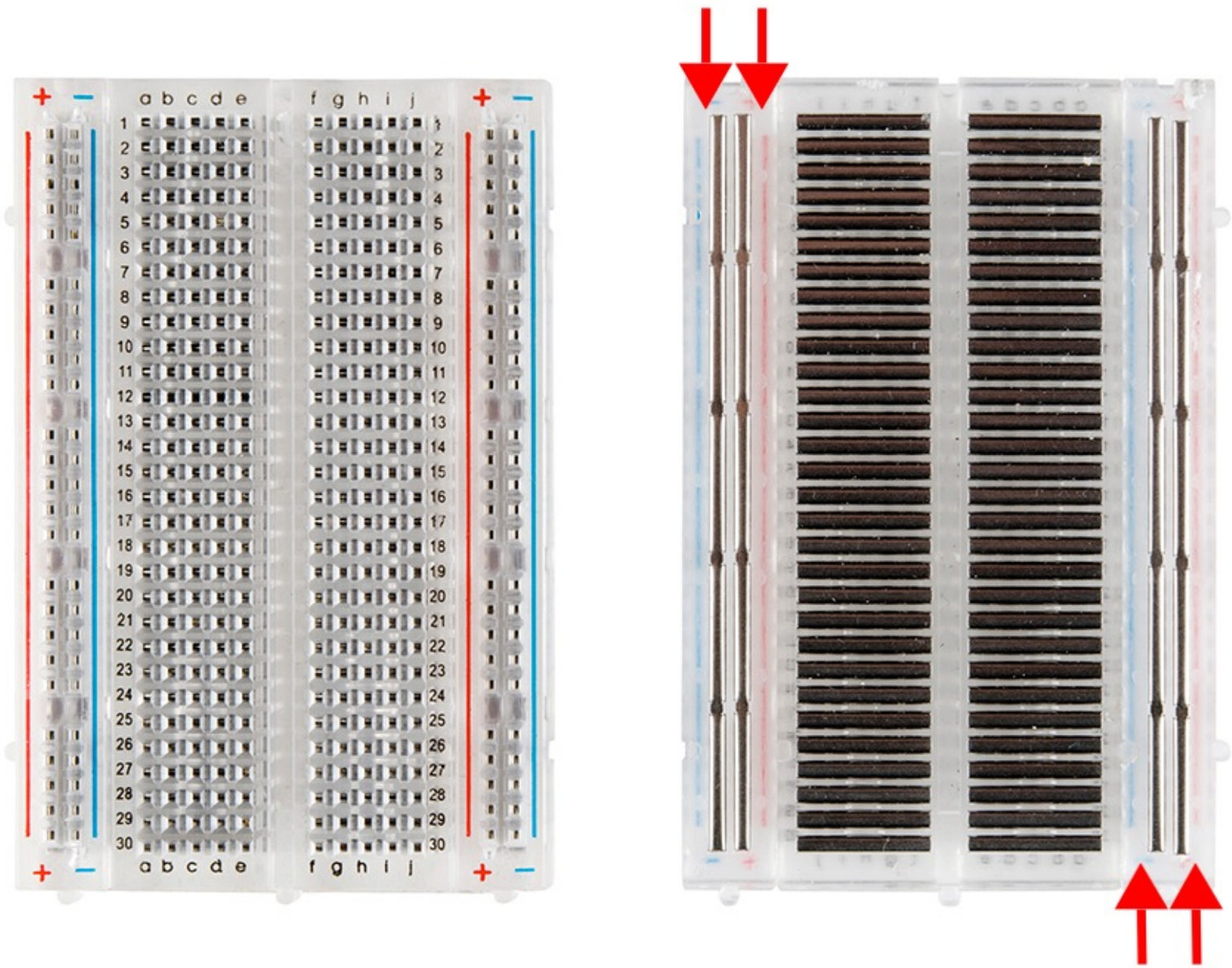


Photo credit: Sparkfun Electronics, Inc.

Electronic Fundamentals

Reading Resistor Codes:

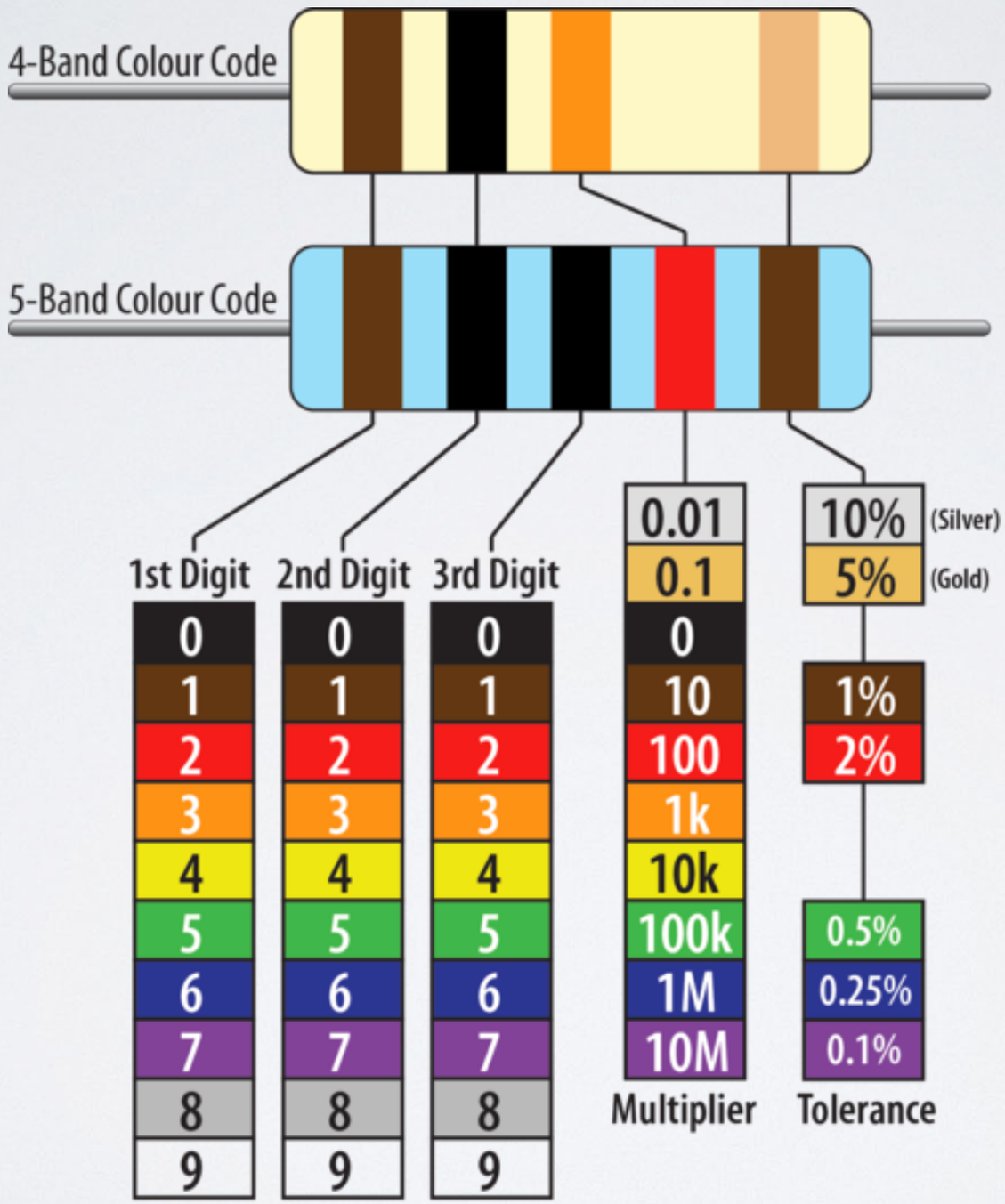
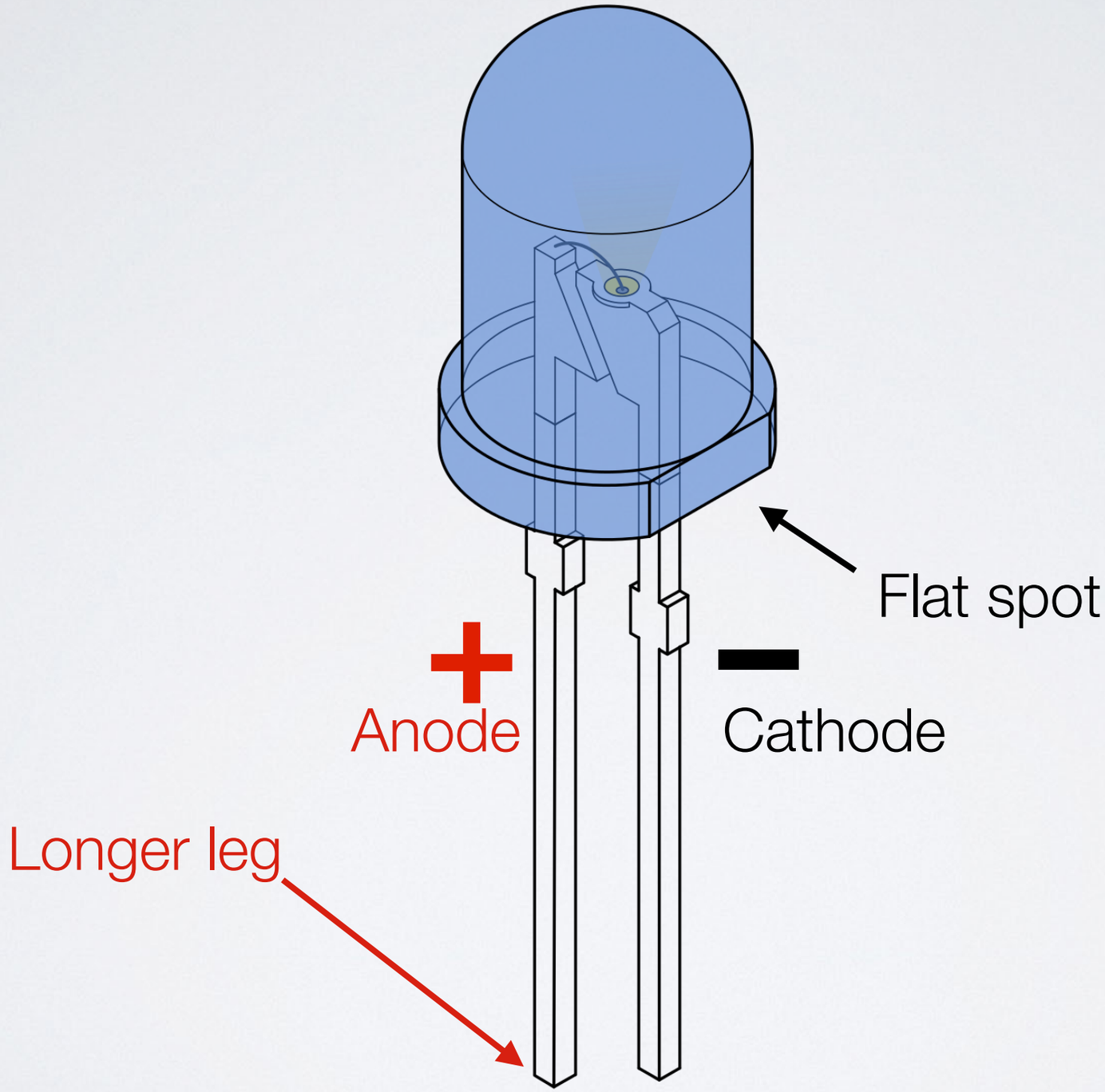


Photo credit: www.freetronics.com

Electronic Fundamentals

Reading LED polarity:



Electronic Fundamentals

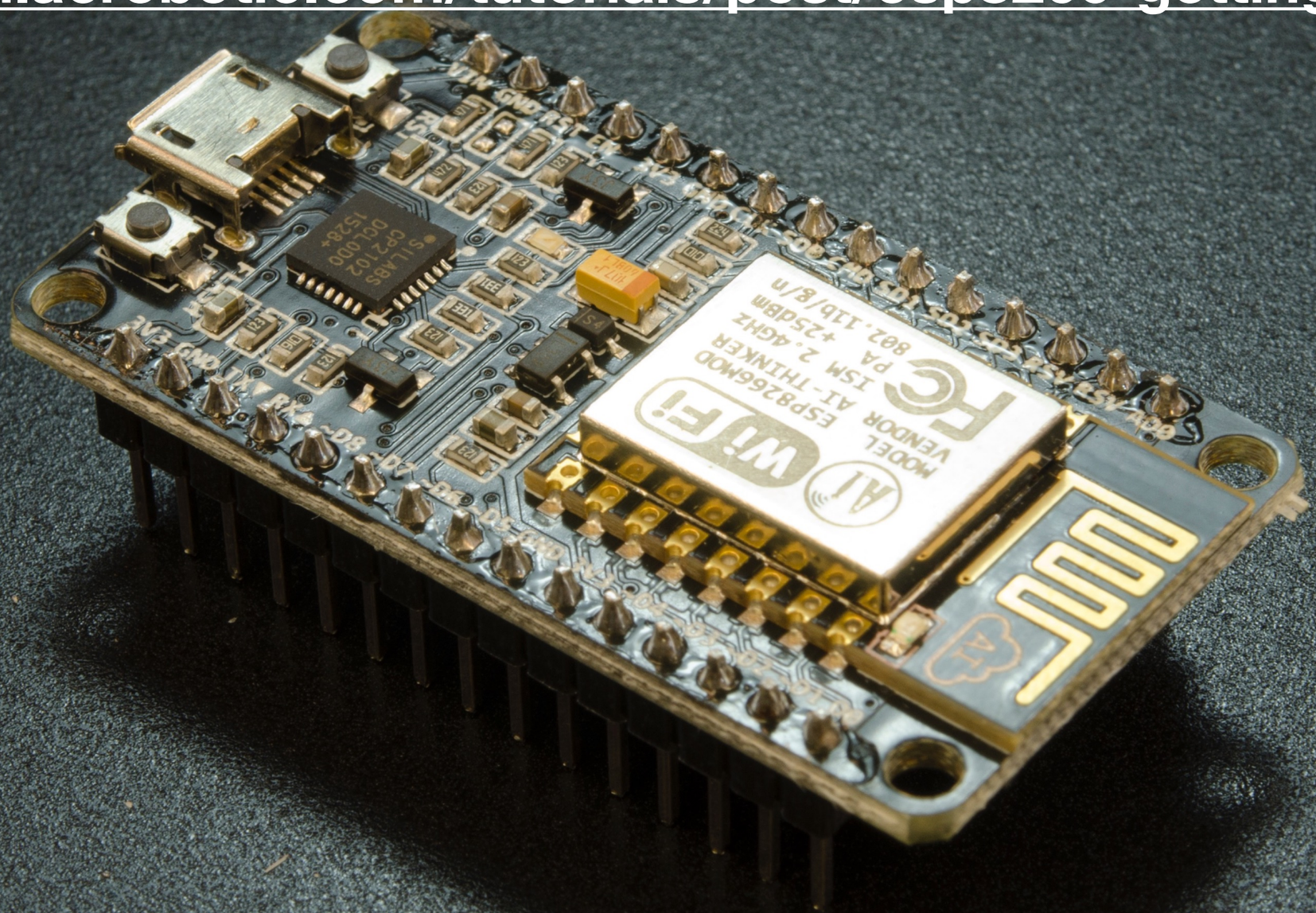
The ESP8266

Serial to Wi-Fi adapter by Expressive Systems (Summer ‘14)

ESP-NN ‘Breakout’ Modules

Development Board for ESP-12E Module More info:

http://learn.acrobotic.com/tutorials/post/esp8266-getting-started

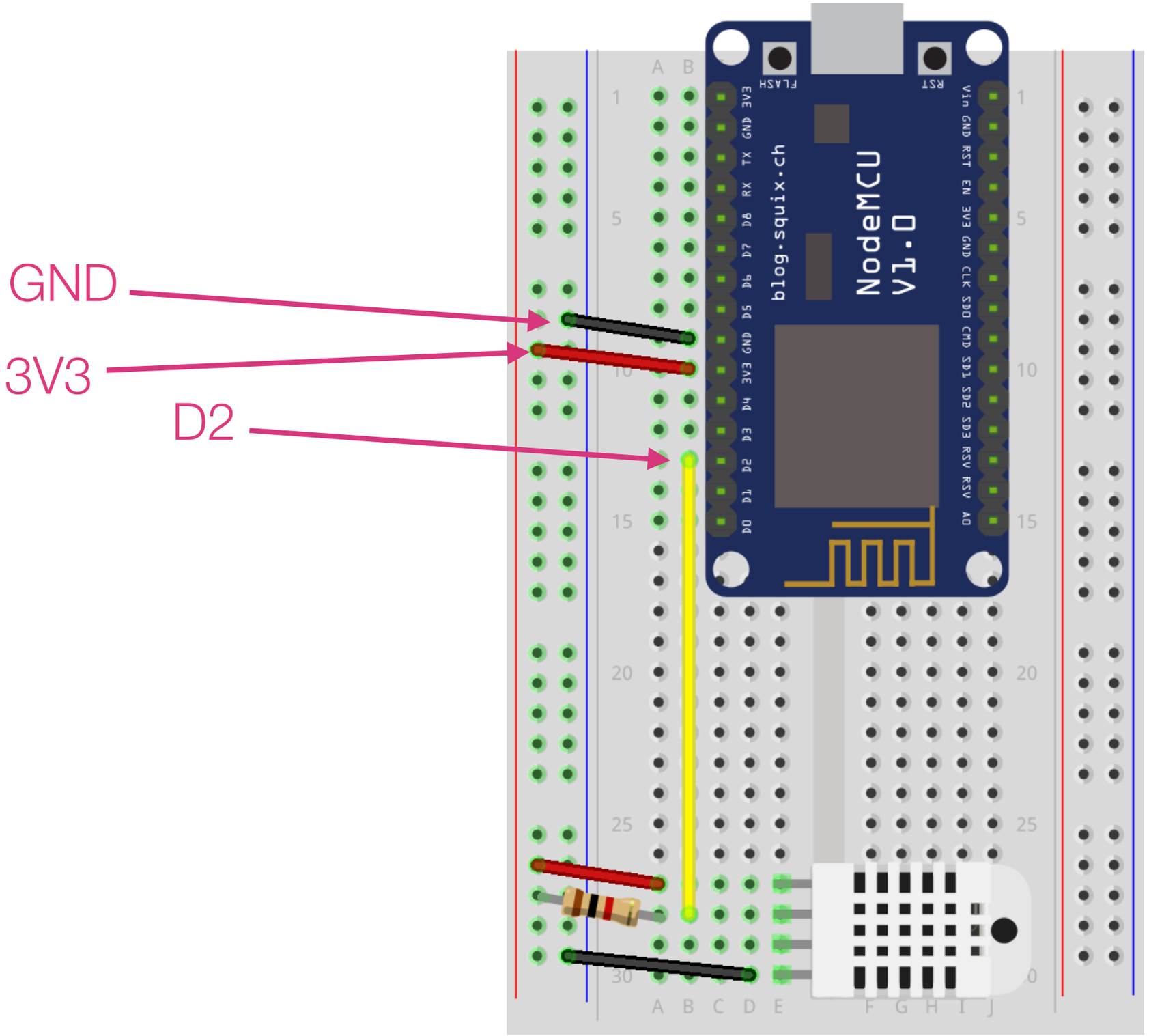


Setting Up the Hardware

Wiring a temp/humidity sensor

Setting Up the Hardware

Wiring up the circuit



Getting Started with Arduino

Installing the Arduino IDE—a computer application to edit, compile, and upload our programs, as well as communicate via USB with the ESP8266 development board (and others)

Windows

Mac (OSX 10.5+)

Linux (32-bit, 64-bit)

Getting Started with the Arduino IDE

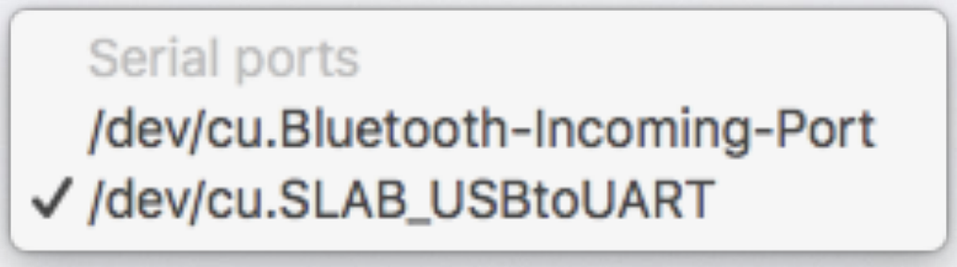
Configuring the Arduino IDE to support the ESP8266

Download and install the USB drivers:

http://j.mp/ESP8266-driver

In the Arduino IDE check under:

Tools > Port

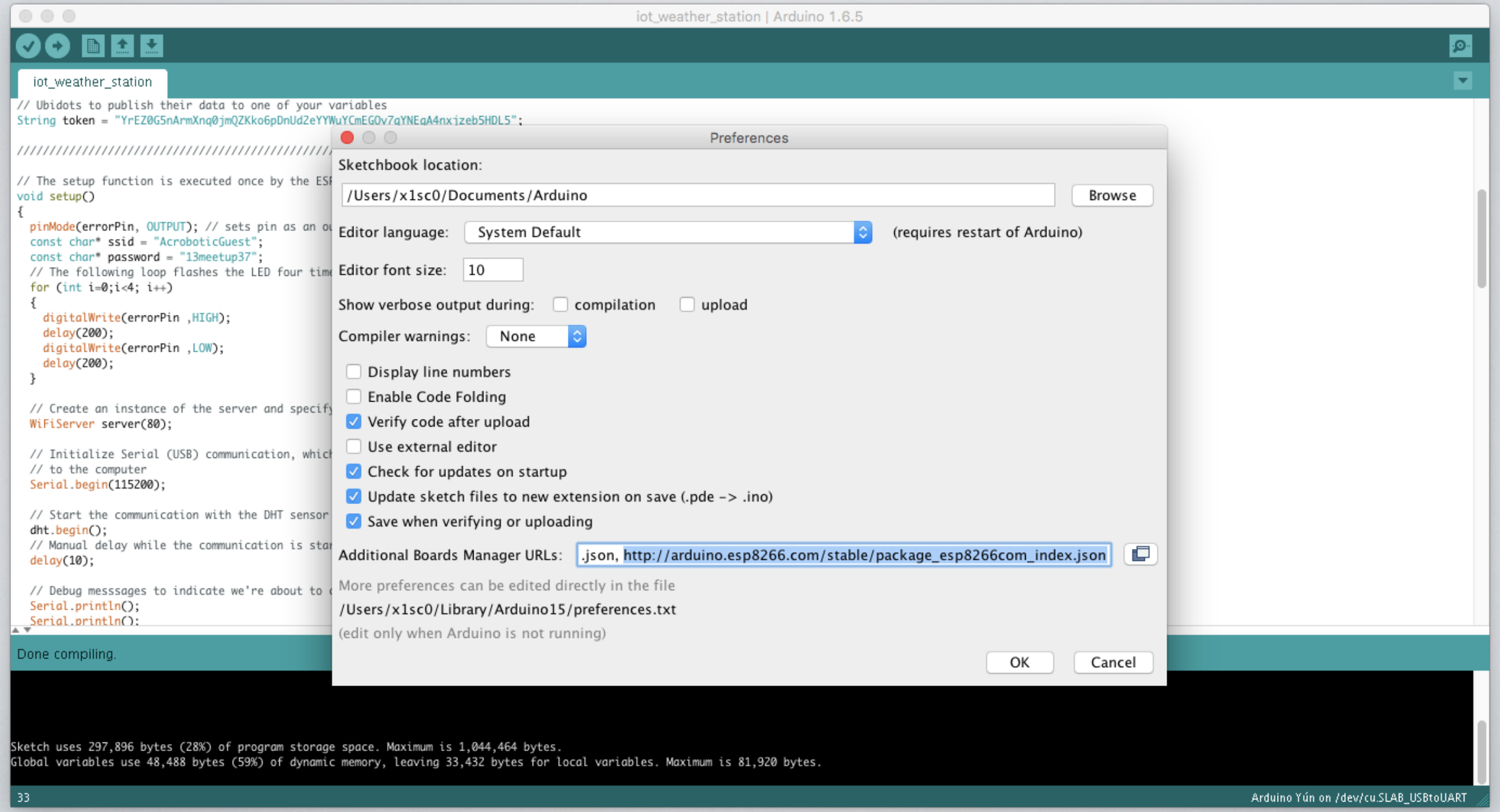


Getting Started with the Arduino IDE

Configuring the Arduino IDE to support the ESP8266

Navigate to “Preferences” and under Additional Board Manager URLS enter:

http://arduino.esp8266.com/stable/package_esp8266com_index.json



Getting Started with the Arduino IDE

Configuring the Arduino IDE to support the ESP8266

Install the ESP8266 boards under:

Tools > Board > Boards Manager

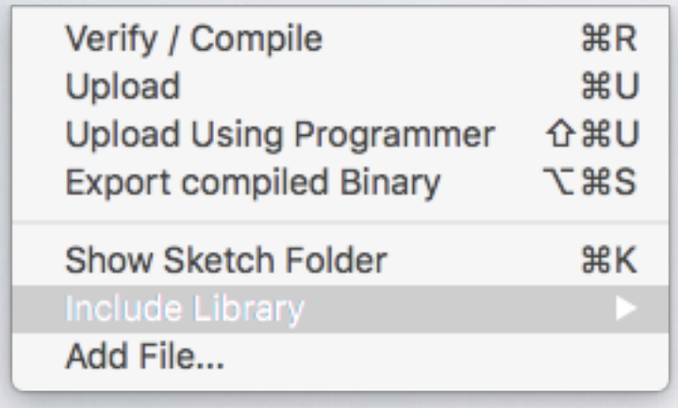
In the Arduino IDE select:

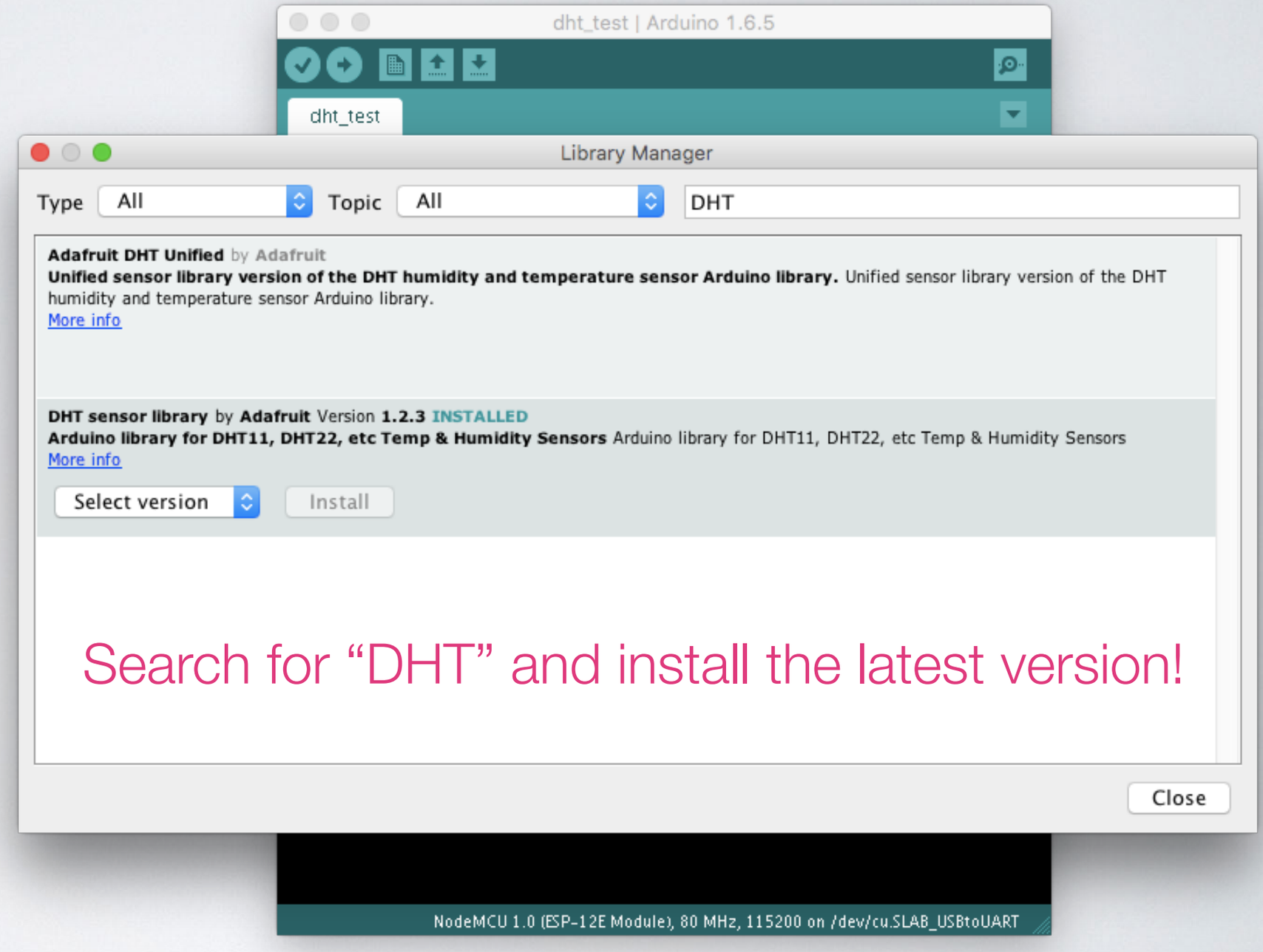
Tools > Board > NodeMCU 1.0 (ESP-12E)

Getting Started with the Arduino IDE

Configuring the Arduino IDE to support the DHT-22 sensor

Open the Library Manager from the menu: *Sketch* > *Include Library* > *Manage Libraries…*

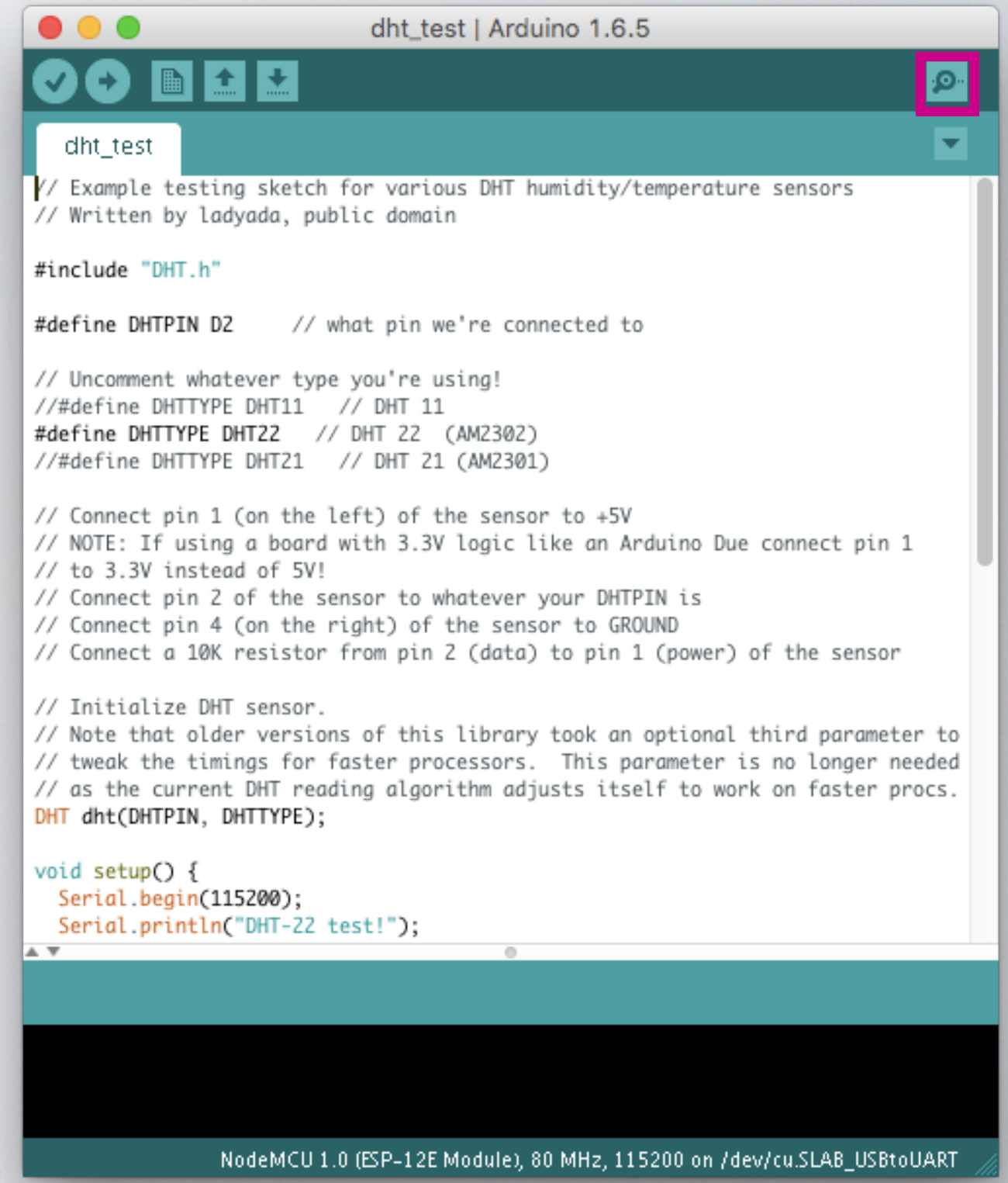




Setting Up the Hardware

Testing our connections [iot_weather_station/dht_test/dht_test.ino]

Upload dht_test.ino to the board and open the Serial Monitor to check the measured data!



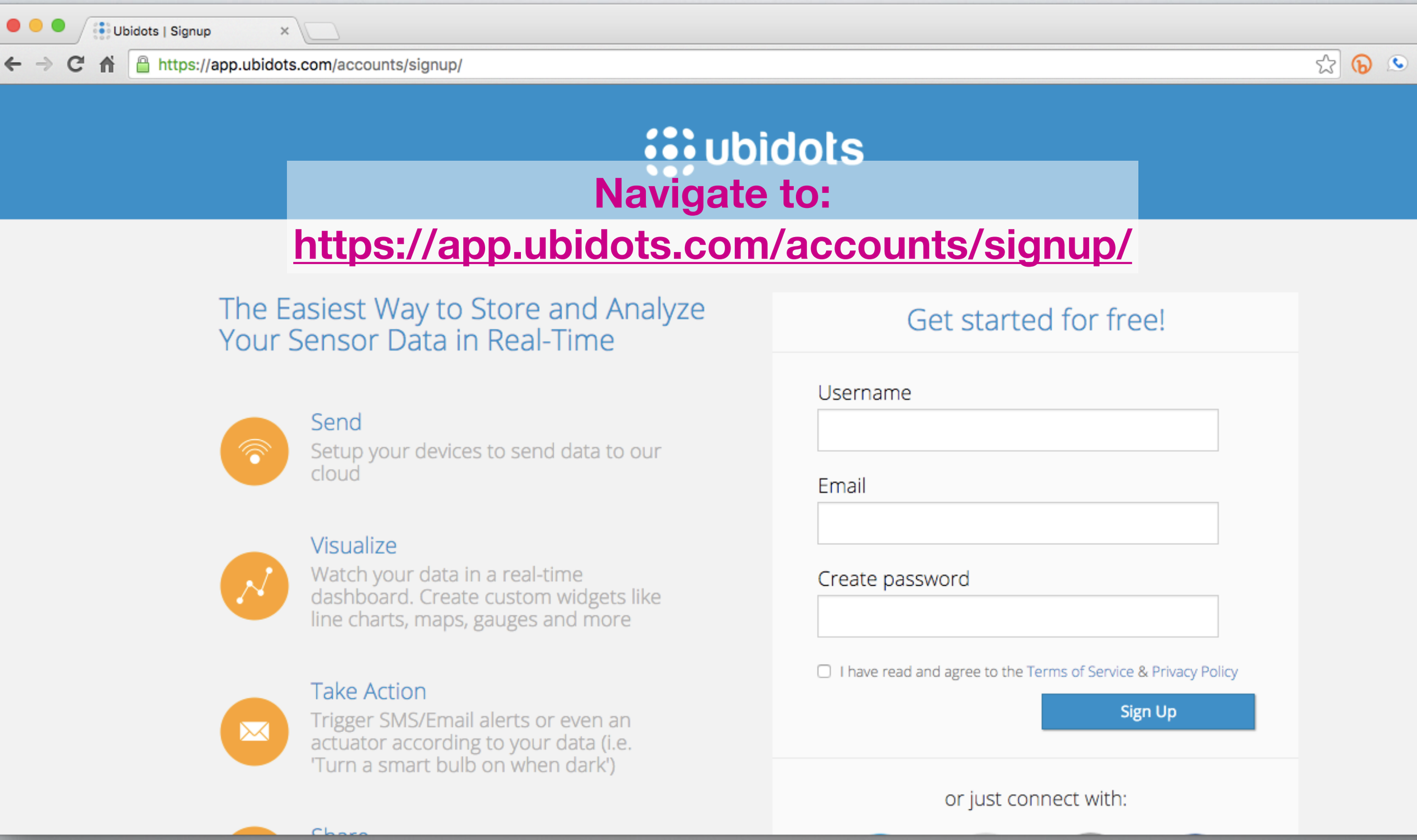
Configuring the Backend and Frontend

Using Ubidots API for posting data (backend)

Using Ubidots Widgets for displaying data (frontend)

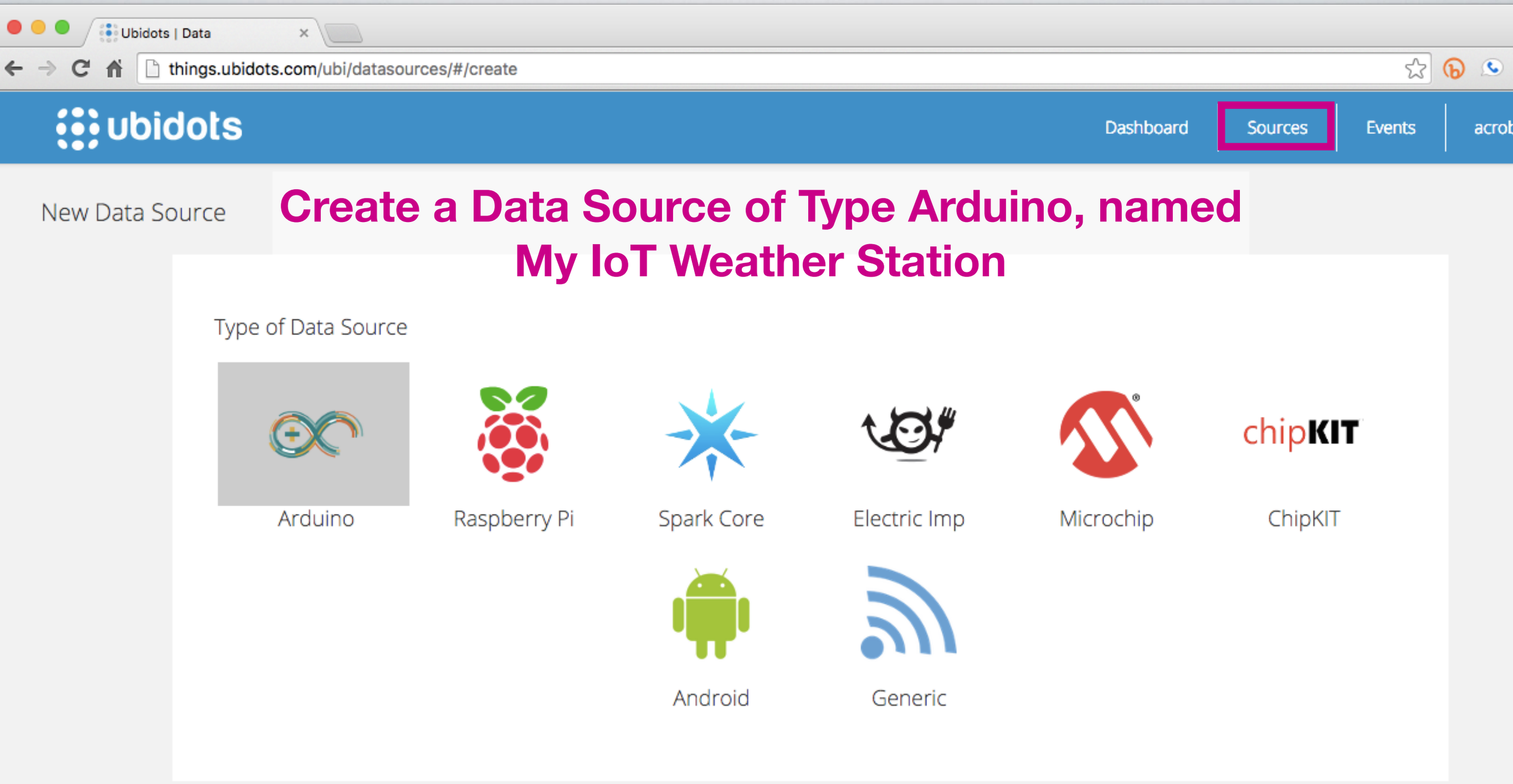
Configuring the Backend

Getting started with Ubidots



Configuring the Backend

Setting up a Data Source



Configuring the Backend

Setting up variables for the Data Source

Ubidots | Data

things.ubidots.com/ubi/datasources/#/detail/56364747762542284d39dd56

ubidots

Dashboard | Sources | Events | acrob

Back to Data Sources

My ESP8266's Variables

Create two “normal” variables: Temperature (F) and Humidity (%)

Add Variab

My ESP8266

0 Variables
No Last Activity

No tags

Name:
My ESP8266

Description:

Tags:
No tags

Context:
_icon: arduino

URL Endpoint:

Add Variable

$f(x)$

Add Derived Variable

What are Variables?

A variable is a series of data containing values that change ove time.

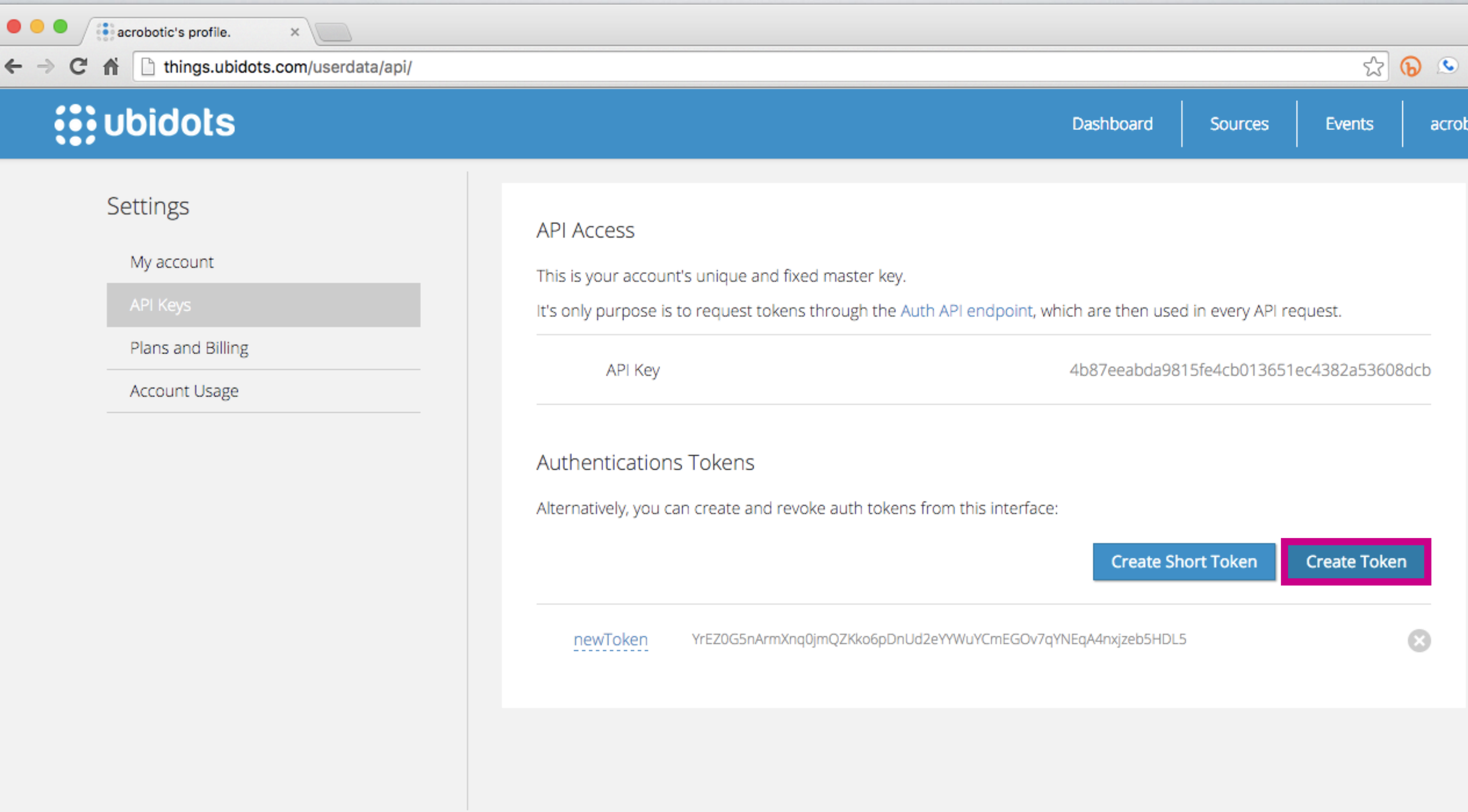
In our case, variables refer to sens that measure something from the real world, such as:

- Temperature
- Humidity
- Pressure
- Position
- Speed
- Acceleration
- Sound Level
- Light Level
- Amount of people
- Volume
- Water Level
- Motion Presence
- Disk Usage
- .. or any time-stamped data!

Ubidots accepts any type of time

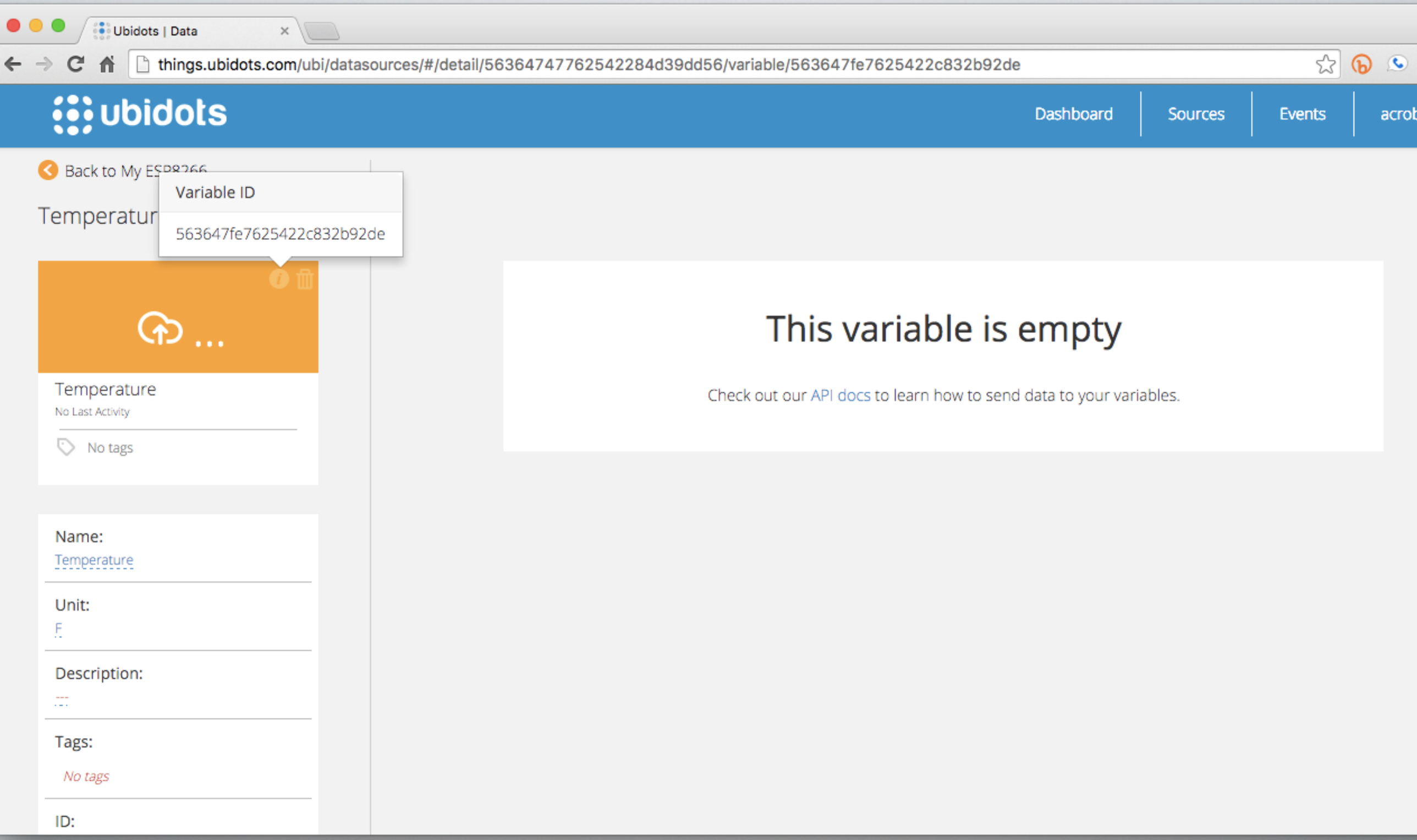
Configuring the Backend

Getting your Ubidots access token



Configuring the ESP8266

Change the Wi-Fi settings, Token, and Variable IDs

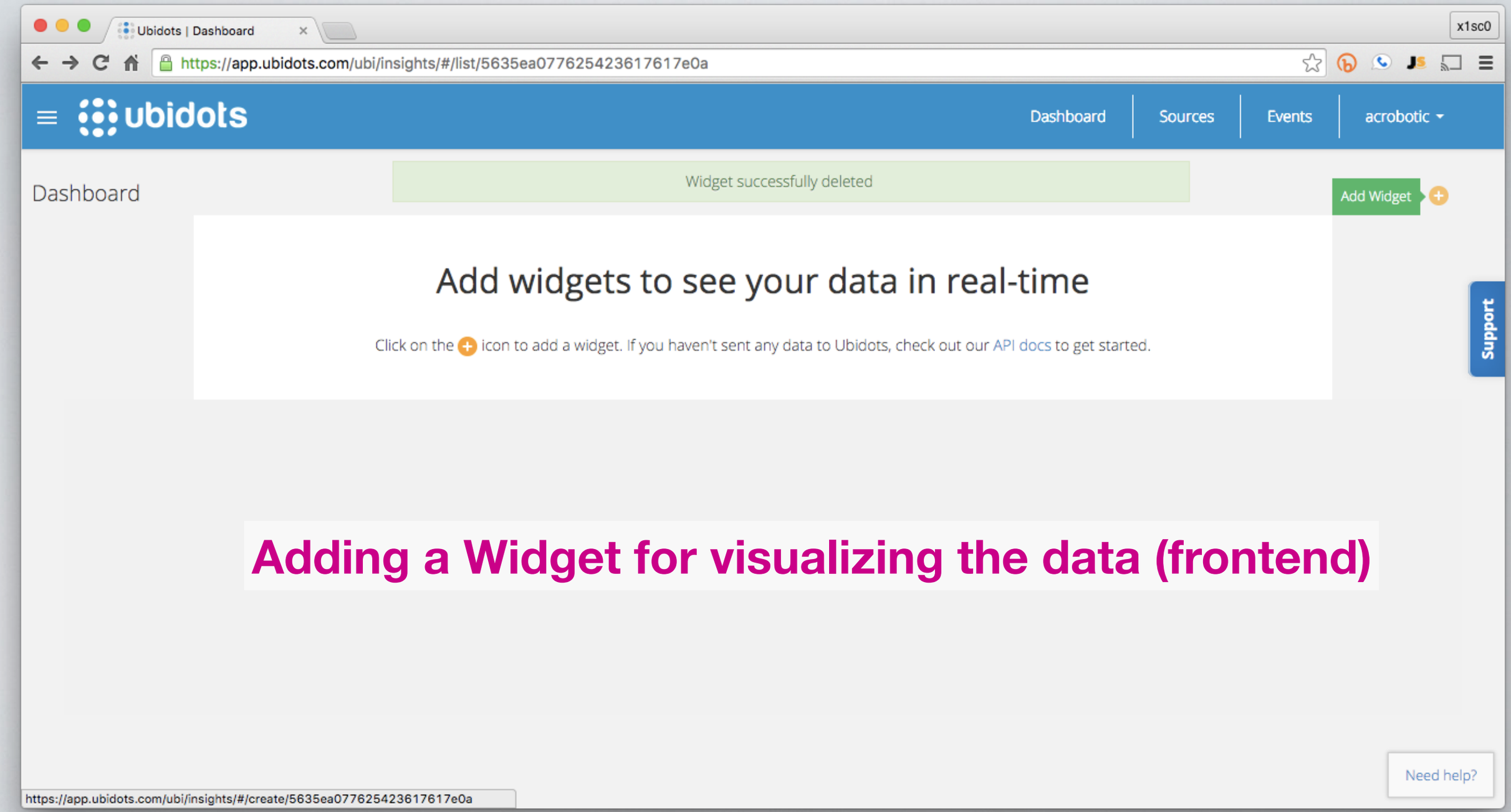


Configuring the ESP8266

Change the Wi-Fi settings, Token, and Variable IDs

Visualizing the Data

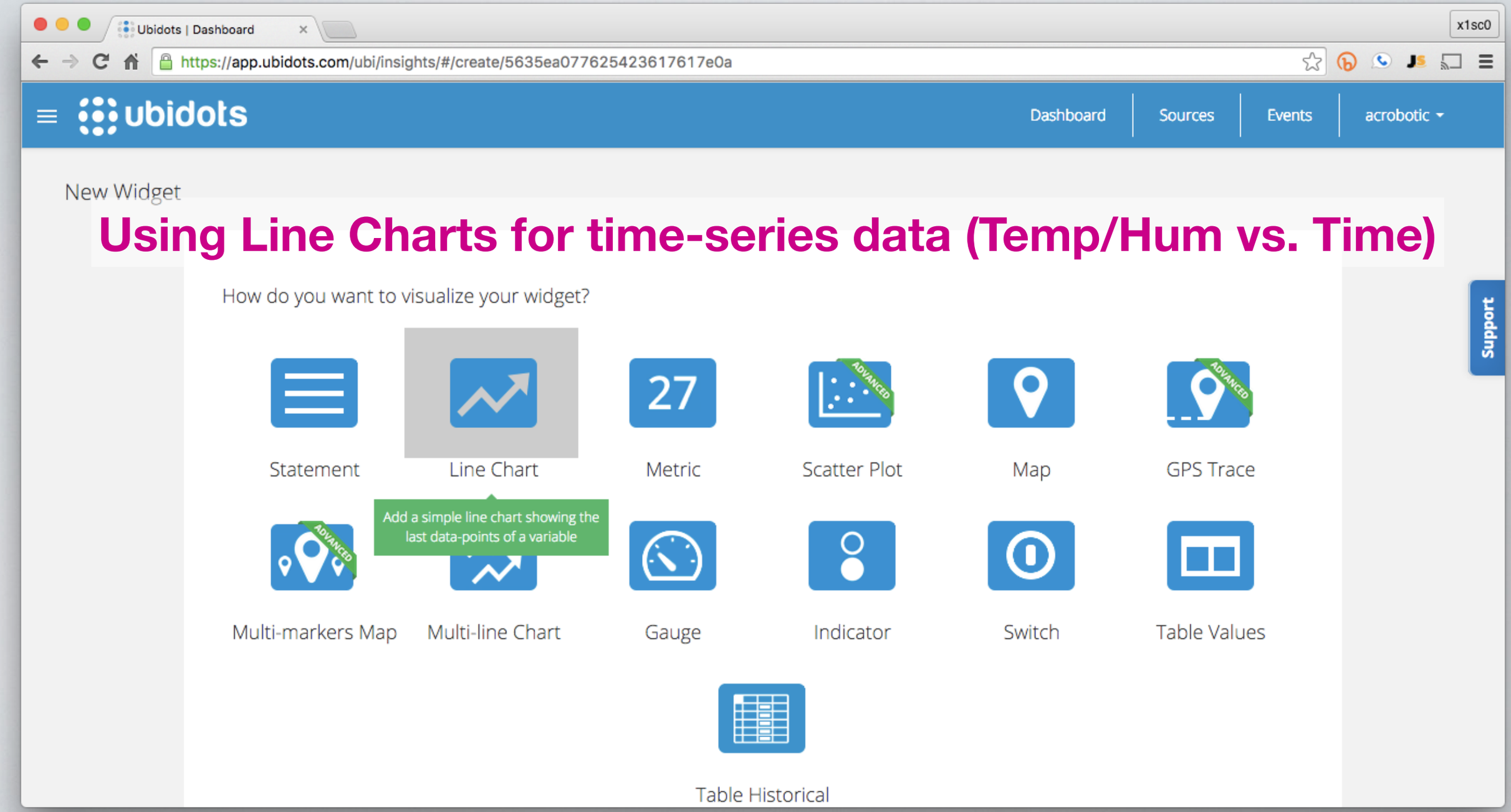
Configuring the Ubidots fronted widgets for data visualization



Adding a Widget for visualizing the data (frontend)

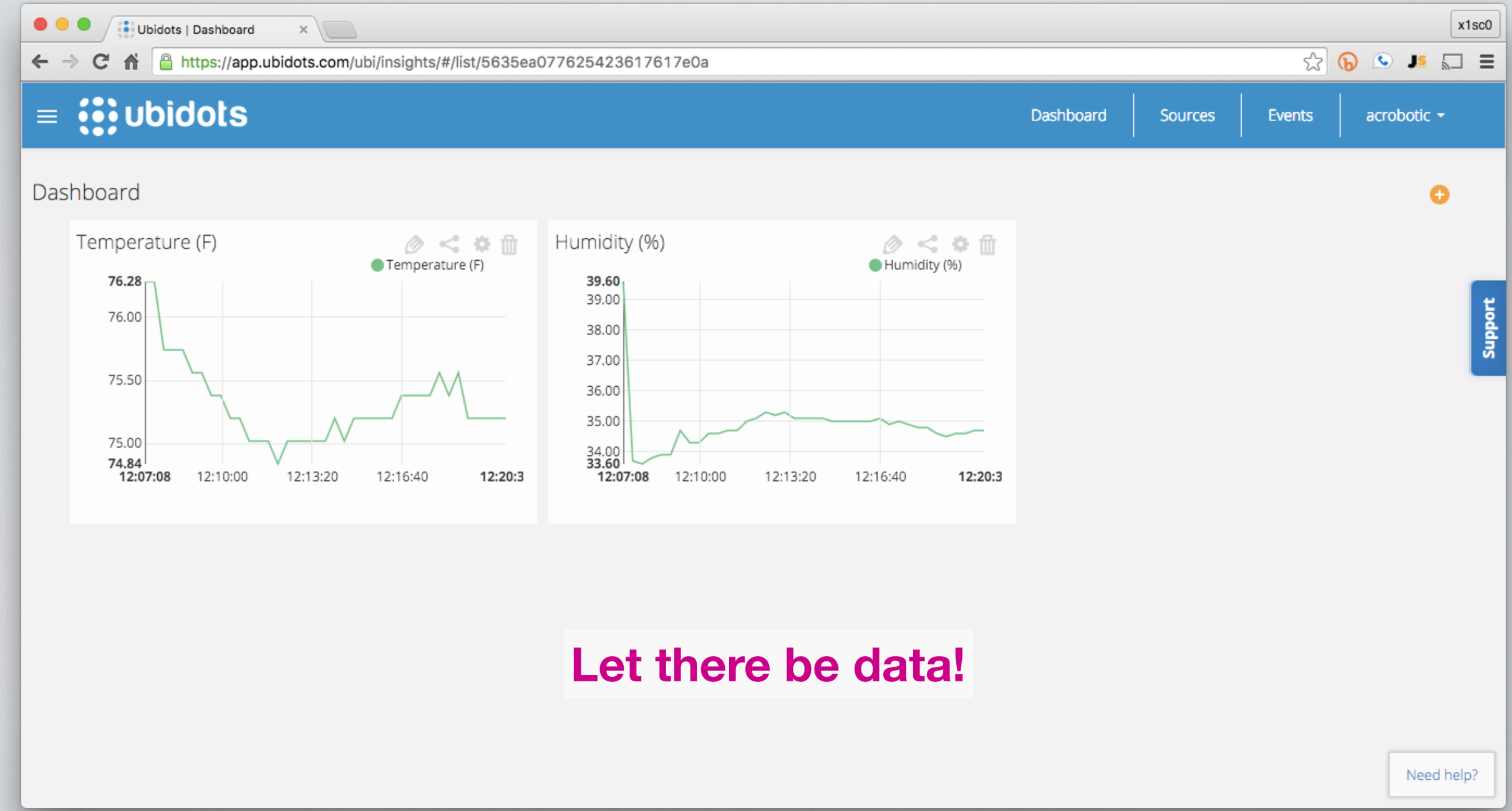
Visualizing the Data

Configuring the Ubidots fronted widgets for data visualization



Visualizing the Data

Configuring the Ubidots fronted widgets for data visualization



Visualizing the Data

Accessing the Ubidots fronted widgets for data visualization