

CLASS 1  
INTERNET OF THINGS

IOT CONCEPT

Modelix Robotics

## Index

1.0. Introduction .....	2
2.0. First Project - LED Trigger.....	4
2.0.1. <i>What do you need for this project?</i> .....	4
2.0.2. <i>Controller code</i> .....	5
2.0.3. <i>Connecting the controller to the internet</i> .....	8
2.0.4. <i>Communicating with the controller through the internet</i> .....	9

## 1.0. Introduction

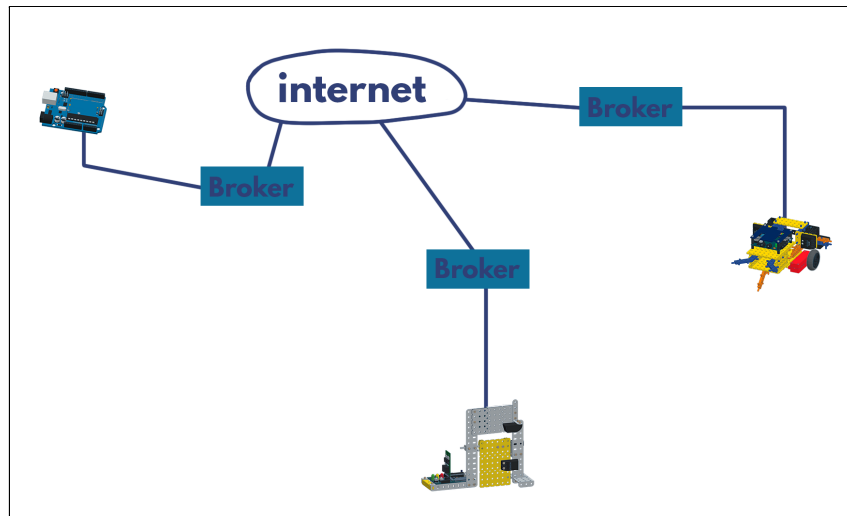
Internet of Things (IoT) is a concept that describes the intercommunication of everyday objects through the internet. The theory illustrates a system capable of communicating and executing tasks only through Internet access, without the need for human interaction.

In order to better conceptualize the idea, the main elements of the IoT system are described below.

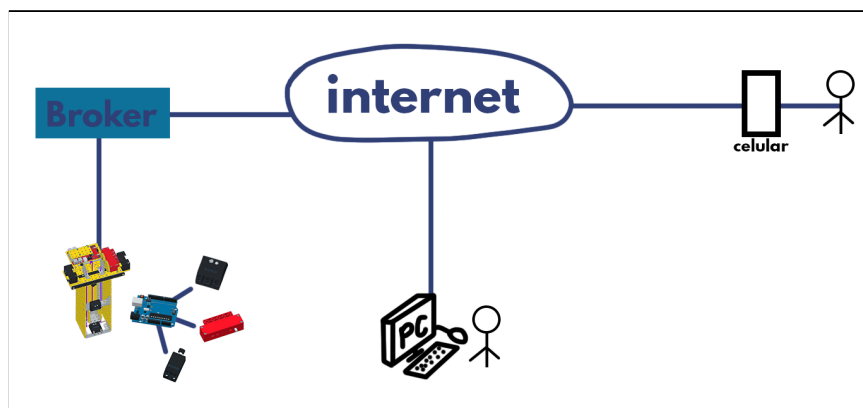
1. **Physical Parts** - Physical electronic devices that connect to the internet through cables and signals. A microcontroller (Arduino) fits in this category.
2. **Coding** - Before the system runs without the need for human interaction, it's required to code the microcontroller. For that, a programmer creates a code based on the desired functionalities using a coding software, and then uploads it to the controller.
3. **TI Softwares** - Apps in the virtual workspace that make the system communication possible. Examples are Web Apps, APIs, centralized platforms and Brokers. We will learn more about these softwares later on.

Some possible IoT system structures are:

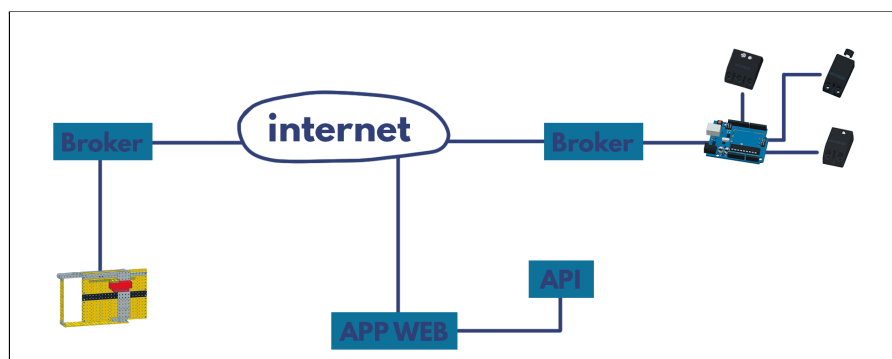
Example A



Example B



Example C



## 2.0. First Project - LED Trigger

Now, in order to understand the IoT concept in a practical exercise, we will create a simple project. We will create a code to trigger a LED through the Internet.

### 2.0.1. *What do you need for this project?*

1. A computer with an internet connection. Windows operating system is required.
2. A microcontroller with an USB cable for connection to the computer. For this class, we will be using Arduino Uno, but any microcontroller will work.
3. One LED actuator.
4. A generic C++ IDE (Integrated Development Environment) software. For this class, we will be using Arduino IDE.
5. Having registered in the Modelix IoT Platform, as well as having downloaded the Modelix IoT Broker. To access to both, go in the link ([www.MyIoT.space](http://www.MyIoT.space))

Now let's get to work!

## 2.0.2. Controller code

1. Enter the link <https://www.arduino.cc/en/software>, and download the software.

### Downloads



**Arduino IDE 1.8.16**

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board.

Refer to the [Getting Started](#) page for Installation instructions.

**SOURCE CODE**

Active development of the Arduino software is [hosted by GitHub](#). See the instructions for [building the code](#). Latest release source code archives are available [here](#). The archives are PGP-signed so they can be verified using [this](#) gpg key.

**DOWNLOAD OPTIONS**

- Windows** Win 7 and newer
- Windows** ZIP file
- Windows app** Win 8.1 or 10 [Get](#)
- Linux** 32 bits
- Linux** 64 bits
- Linux** ARM 32 bits
- Linux** ARM 64 bits
- Mac OS X** 10.10 or newer

[Release Notes](#) [Checksums \(sha512\)](#)

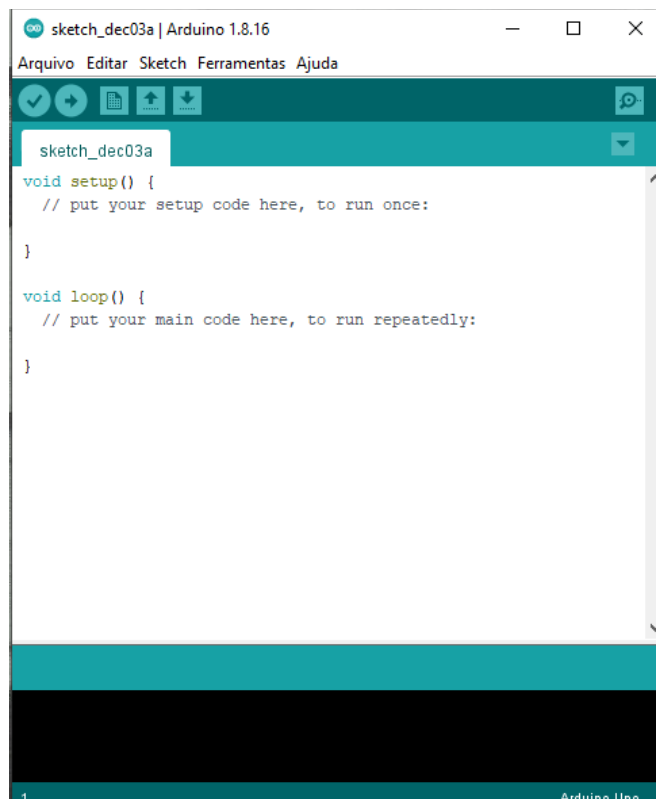
**Hourly Builds**

Download a **preview of the incoming release** with the most updated features and bugfixes.

**Previous Releases**

Download the previous version of the current release, the classic 1.0.x, or old beta releases.

2. After completing the download, click on the file and go through the installation steps. After the installation is complete, open the program.



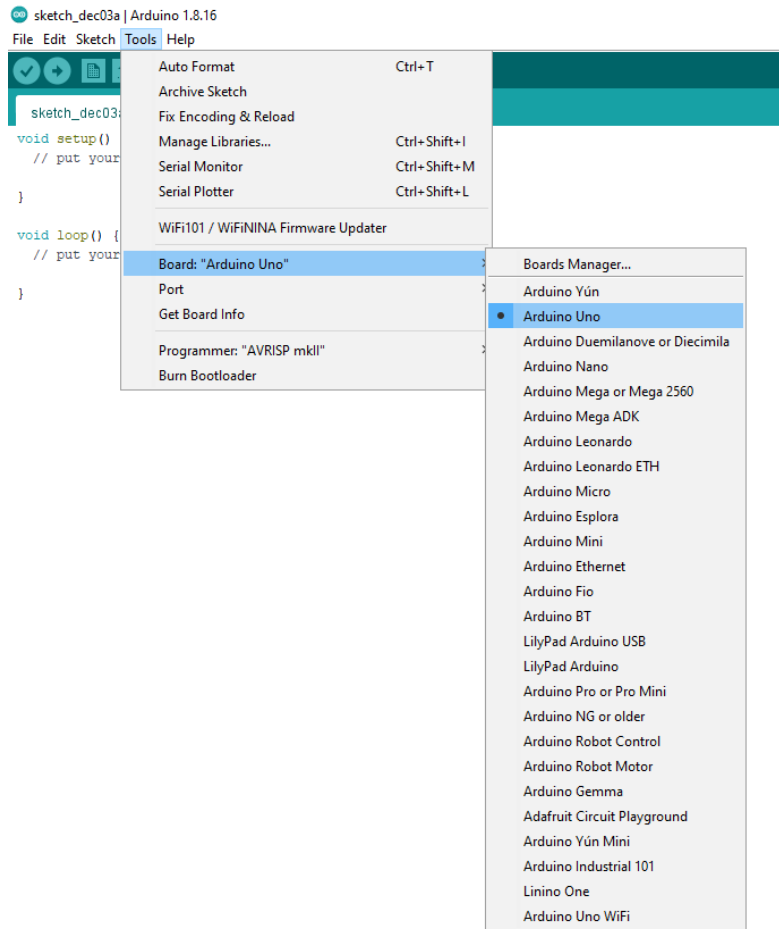
```
sketch_dec03a | Arduino 1.8.16
Arquivo  Editar  Sketch  Ferramentas  Ajuda

sketch_dec03a
void setup() {
  // put your setup code here, to run once:
}

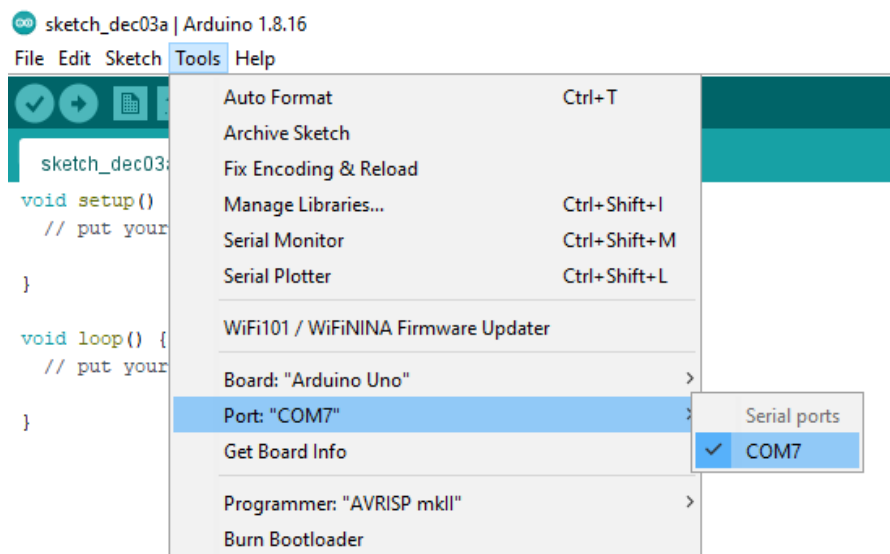
void loop() {
  // put your main code here, to run repeatedly:
}

1 Arduino Uno
```

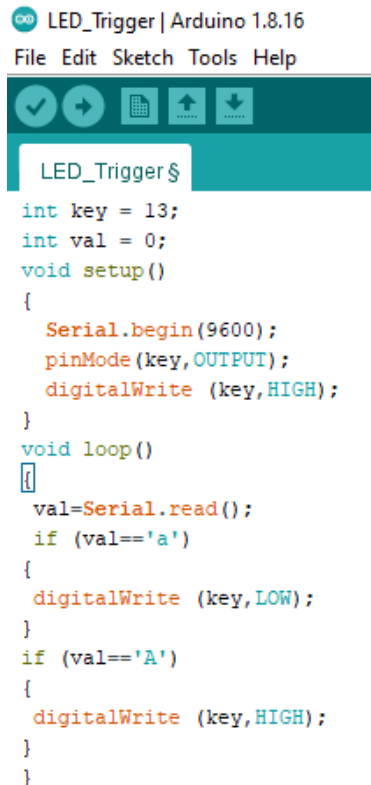
3. Plug-in your microcontroller to the computer. In the Arduino program, go on “Tools”, “Board” and select your arduino.



4. Go to “Tools”, “Port” and select the available port, corresponding to your plugged-in microcontroller.



5. Insert the code as follows, and click on the right-pointing arrow on the top left corner (upload button).



```
LED_Trigger | Arduino 1.8.16
File Edit Sketch Tools Help
LED_Trigger$
int key = 13;
int val = 0;
void setup()
{
  Serial.begin(9600);
  pinMode(key,OUTPUT);
  digitalWrite (key,HIGH);
}
void loop()
{
  val=Serial.read();
  if (val=='a')
  {
    digitalWrite (key,LOW);
  }
  if (val=='A')
  {
    digitalWrite (key,HIGH);
  }
}
```

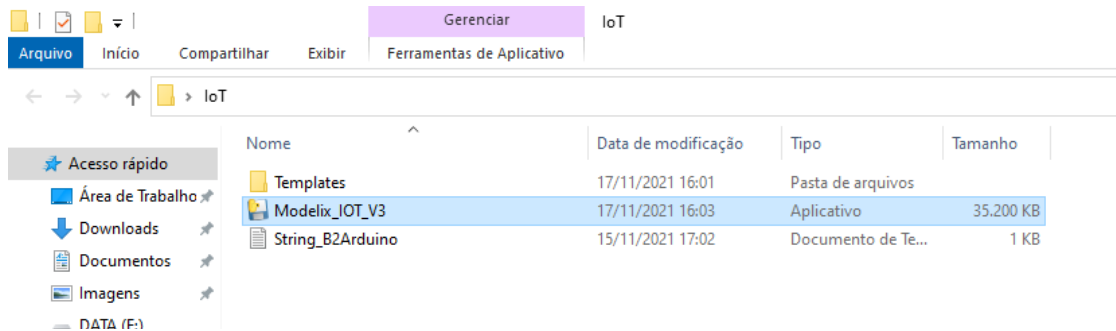
6. After the “Done uploading” message appears, close the Arduino IDE software.



7. All set! The first part of your project is ready. Now let’s connect the system to control it through the internet.

### 2.0.3. Connecting the controller to the internet

1. With your Arduino still connected, go to the downloaded IoT folder. In case you don't have this folder, go to item 5 of section 2.0.1.
2. Extract the files from the compacted folder, and open "Modelix\_IOT\_V3". This file is the Modelix IoT Broker.



3. When the app opens up, insert the same email and password registered on the Modelix IoT Platform (Item 5 of section 2.0.1). Select your microcontroller, and click "Start".

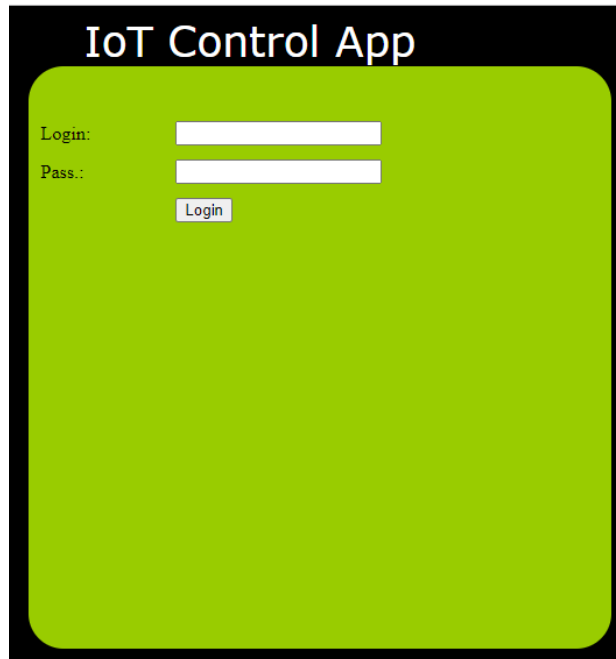


4. When the screen clears, you will be connected to the internet.



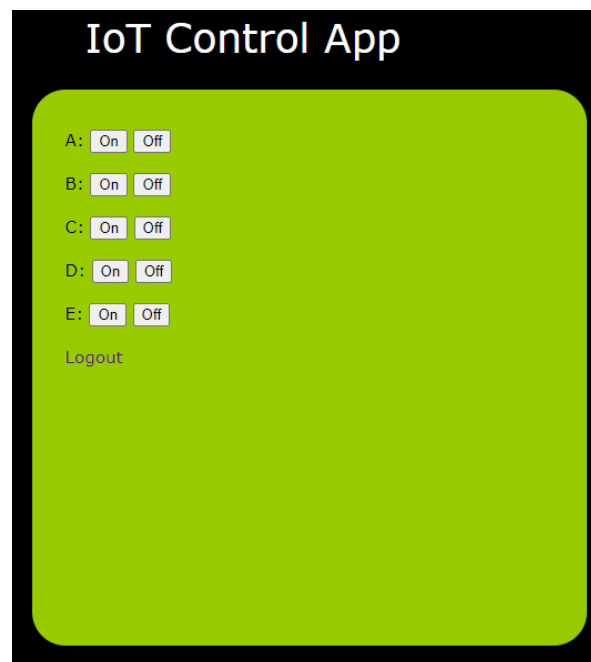
#### 2.0.4. Communicating with the controller through the internet

1. Log in to the communication platform, on the link [www.leomar.com.br/iot](http://www.leomar.com.br/iot) . With the Modelix IoT Broker running, you can do this step from any place on the planet with internet!
2. You will see the IoT Control App page. Log in with **the same** email and password used to log in into the broker. Otherwise, the connection will not work.



The screenshot shows the login interface of the IoT Control App. It features a black header with the text "IoT Control App" in white. Below the header is a light blue rounded rectangle containing the login form. The form includes two input fields: "Login:" and "Pass:.". Below the password field is a "Login" button.

3. The following screen will appear:



The screenshot shows the main control interface of the IoT Control App. It features a black header with the text "IoT Control App" in white. Below the header is a light blue rounded rectangle containing the control interface. The interface includes five rows of controls, each with a label (A, B, C, D, E) and two buttons labeled "On" and "Off". Below the controls is a "Logout" button.

4. Before pressing any button, it's interesting to understand how each one is communicating with the microcontroller. Each button sends a letter from your computer straight to the controller. For every letter port (A, B, C, D and E) the ON and OFF buttons send a signal equivalent to that letter in uppercase and lowercase, respectively. The button values can be seen below.

Letter	ON Button	OFF Button
A:	A	a
B:	B	b
C:	C	c
D:	D	d
E:	E	e

5. Understanding the signals, we can start communicating with our controller! Only the letters included in the code will be able to be used. With that in mind, we will only be using A buttons in this project, since we programmed these values (A and a) on step 5 of section 2.0.2. The project is ready! Now the ON and OFF buttons of the A port will turn our LED on and off.