





# Workshop - Introdução a IoT

A criação de novos modelos de negócio a partir da transformação da interação entre consumidor, produto e indústria.

# Ricardo Lins Mota, Eronides da Silva Neto e Heitor Fonseca de Araújo

Instituto SENAI de Inovação para Tecnologias da Informação e Comunicação





### Introdução a Internet das Coisas



Curso de Introdução a IoT (EAD)

ferramentas da Indústria 4.0.

Curso introdutório para a Introdução apresentação da IoT como umas ferramentas

Introdução a algumas das ferramentas práticas utilizadas em projetos de IoT.

**Workshop Prático** 

Introdução a IoT

Capacitação em IoT

Uso de temas relacionados a IoT em aulas e projetos





O que aprendemos até agora sobre a Internet das Coisas?



## Conteúdos abordados no curso EAD - SENAI/Joy Street

- ☐ IoT conceitos e aplicações
- ☐ Tecnologias digitais da Indústria 4.0.
- Camadas da Internet das Coisas
- ☐ Tecnologias de comunicação
- ☐ Ferramentas de visualização (HMI)



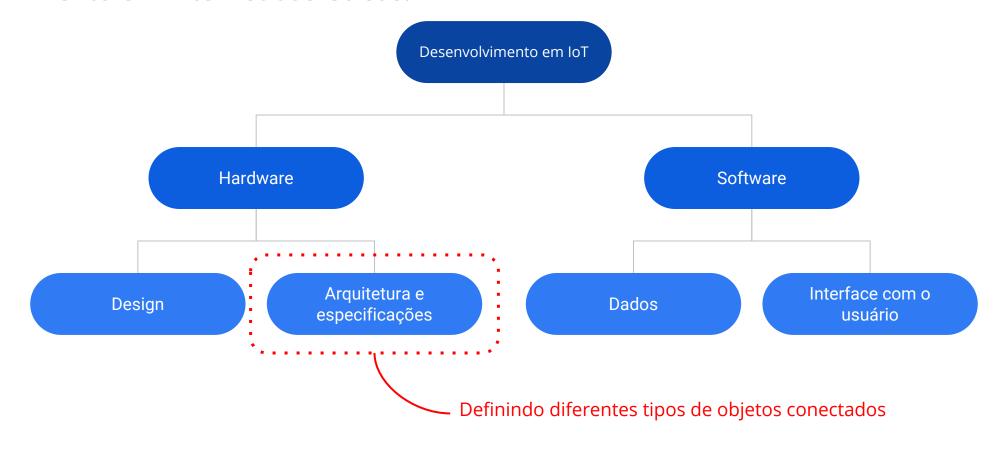
## Agenda do Workshop

- Metodologia do workshop
- IoT stack
- Plataforma Arduino
- ESP8266 e NodeMCU
- Conexão de sensores e atuadores
- Desenvolvimento de projetos embarcados
- Conexão com a nuvem



#### Metodologia do nosso workshop

Uso de ferramentas educacionais de baixo custo para o aprendizado de etapas do desenvolvimento em Internet das Coisas.





#### IoT Stack: um sistema de ... sistemas

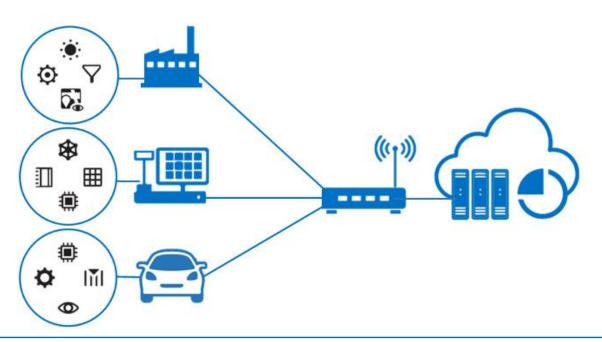




#### IoT stack

#### Como podemos ver, temos diferentes áreas:

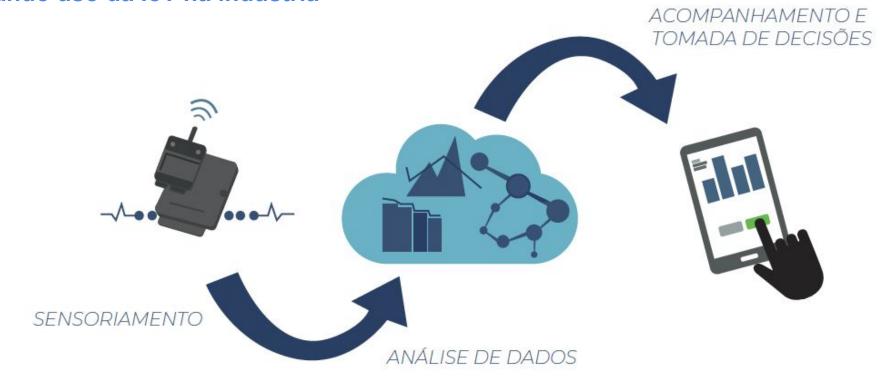
- 1. As coisas
- 2. Tecnologias de comunicação
- 3. Software: as etapas de processamento, armazenamento e análise dos dados.





#### As camadas da IoT

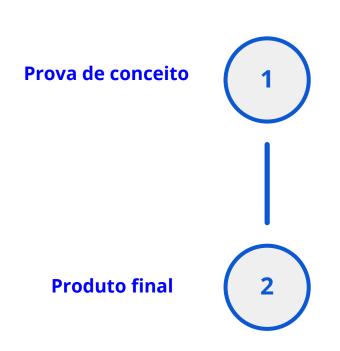
#### Considerando uso da IoT na indústria

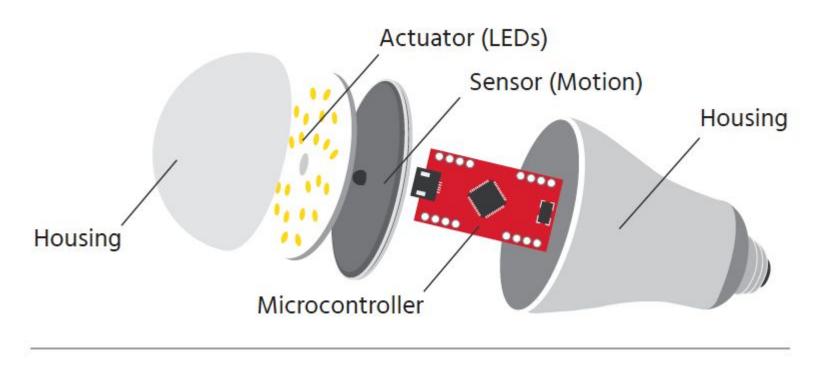




#### Como construir um objeto conectado?

#### Parte do desenvolvimento

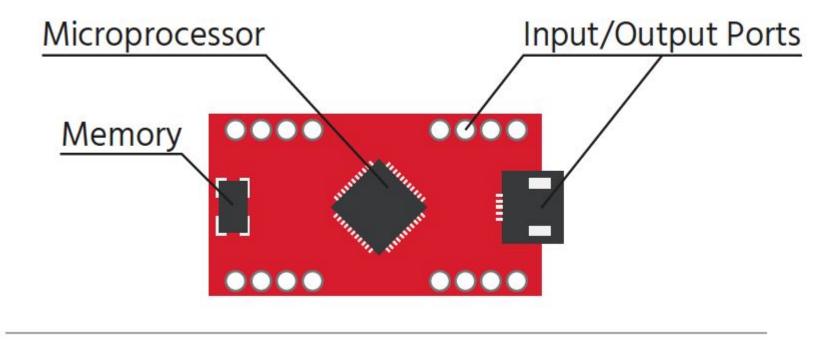




A arquitetura de uma "coisa"



#### A unidade de controle das diferentes coisas



O funcionamento de um microcontrolador



# **PARTE I - Conceitos**



# **Arduino**

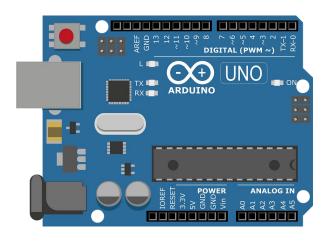


#### A plataforma Arduino

Arduino é uma plataforma open-source hardware. \*\* open source hardware.



Qualquer um pode modificar, utilizar e adaptar o ambiente de programação.



A placa é composta por um **microcontrolador**, circuitos de entrada/saída e que pode ser facilmente conectada à um computador e programada via IDE (Integrated Development Environment, ou Ambiente de Desenvolvimento Integrado) utilizando uma linguagem baseada em C/C++, sem a necessidade de equipamentos extras além de um cabo USB.



## A plataforma Arduino

O Arduino possui uma gama de módulos, sensores e atuadores, que auxiliam na montagem de uma prova de conceito ou projeto.

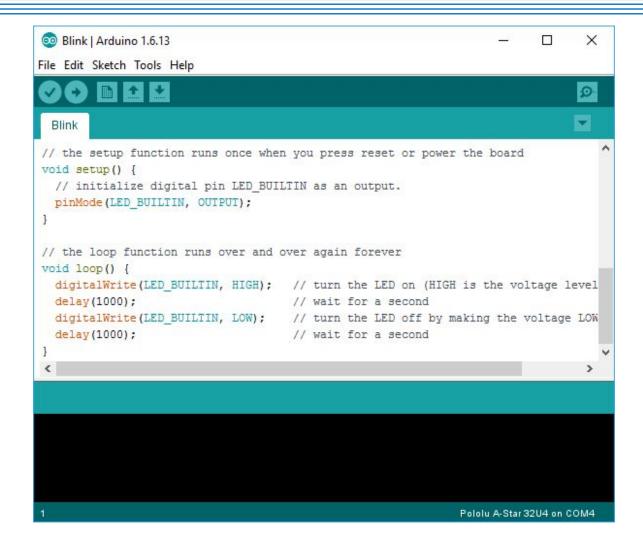
☐ Também existem placas auxiliares chamadas de "shield" que auxiliam o arduíno em tarefas mais complexas, como uma pilha Ethernet.







#### **Arduino IDE**



- → Programação para diversas placas Arduino, utilizando "bibliotecas" (conjunto de códigos) para sensores e dispositivos específicos.
- Possibilidade de verificar erros.
- Gratuita e compatível com Windows, MAC e Linux.



#### **Arduino e Internet das Coisas**

#### Módulo ESP8266



#### Placa de desenvolvimento NodeMCU



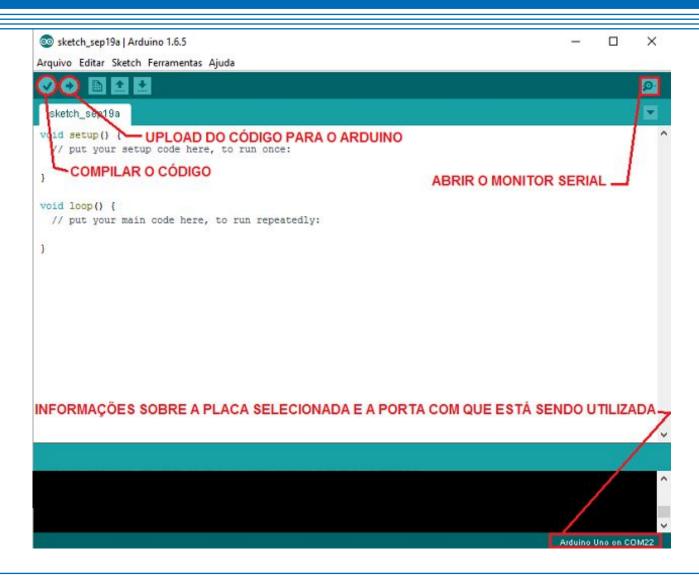
- Entre os inúmeros módulos que surgiram recentemente para explorar a onda da Internet das Coisas (IoT), o que mais se destaca é o **ESP8266**, da empresa **Espressif**.
- Com diversos modelos diferentes, o módulo ESP8266 pode conter apenas o transceiver ou também operar de maneira standalone, contendo o microcontrolador responsável para comandar diferentes tarefas.
- Conexão Wi-Fi em frequência de 2.4GHz com suporte a WPA e WPA2;



Como começar a programar e fazer projetos com Arduino?



#### **Conceitos básicos**





## O ambiente de programação Arduino

```
ForLoopIteration | Arduino 1.6.8 Hourly Build 2016/02/19 11:12
                                    ForLoopIteration
                                  10 created 2006
                                   11 by David A. Mellis
                                   12 modified 30 Aug 2011
                                   13 by Tom Igoe
                                   15 This example code is in the public domain.
                                   17 http://www.arduino.cc/en/Tutorial/ForLoop
                                   18 */
Setup do programa
                                   20 int timer = 100:
                                                               // The higher the number, the slower the timing
                                   22 void setup() {
                                   23 // use a for loop to initialize each pin as an output:
                                                                                                                    Executado apenas uma vez
                                   for (int thisPin = 2; thisPin < 8; thisPin++) {</pre>
                                        pinMode(thisPin, OUTPUT);
                                   26 }
                                   27 }
          Loop
                                   29 void loop() {
                                   30 // loop from the lowest pin to the highest:
                                   31 for (int thisPin = 2; thisPin < 8; thisPin++) {
                                        // turn the pin on:
                                        digitalWrite(thisPin, HIGH);
                                        delay(timer);
                                        // turn the pin off:
                                         digitalWrite(thisPin, LOW);
                                                                                                                              Execução infinita
                                   37 }
                                       // loop from the highest pin to the lowest:
                                   40 for (int thisPin = 7; thisPin >= 2; thisPin--) {
                                       // turn the pin on:
                                        digitalWrite(thisPin, HIGH);
                                         delay(timer);
                                        // turn the pin off:
                                         digitalWrite(thisPin, LOW);
```



#### Estrutura de programação

#### Documentação de Referência da Linguagem Arduino

A linguagem de programação do Arduino pode ser dividida em três partes principais: estruturas, valores (variáveis e constantes) e funções.

# Disponível em <a href="https://www.arduino.cc/reference/pt/">https://www.arduino.cc/reference/pt/</a>

Para controlar a placa Arduino e realizar computações.

Entradas e Saídas Digitais	Funções Matemáticas	Números Aleatórios
digitalRead()	abs()	random()
digitalWrite()	constrain()	randomSeed()
pinMode()	map()	
	max()	Bits e Bytes
Entradas e Saídas Analógicas	min()	bit()
analogRead()	pow()	bitClear()
analogReference()	sq()	bitRead()
analogWrite()	sqrt()	bitSet()
		bitWrite()
Apenas Zero, Due e Família MKR	Funções Trigonométricas	highByte()
analogReadResolution()	cos()	lowByte()
analogWriteResolution()	sin()	
	tan()	Interrupções Externas
Entradas e Saídas Avançadas		attachInterrupt()
noTone()	Caracteres	detachInterrupt()

#### Leitura recomendada

#### <u>Programação com Arduino, Simon Monk</u>



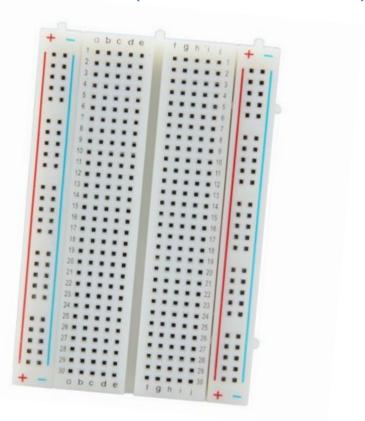


Conceitos básicos de eletrônica

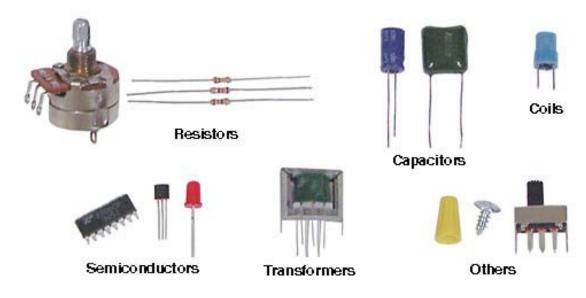


#### Básico de eletrônica

#### Protoboard (matriz de contato)



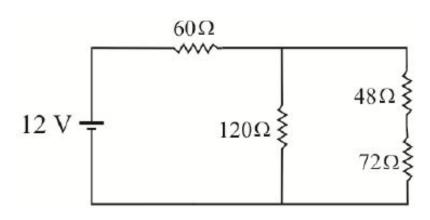
### Componentes de circuitos eletrônicos



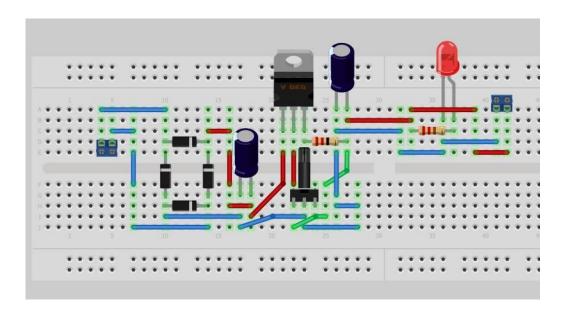


#### O funcionamento de circuitos na protoboard

#### Circuito elétrico



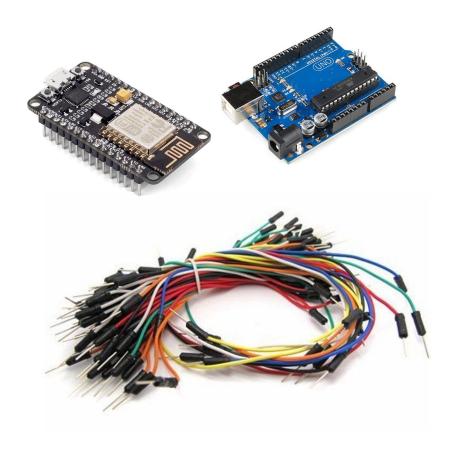
#### Representação na protoboard



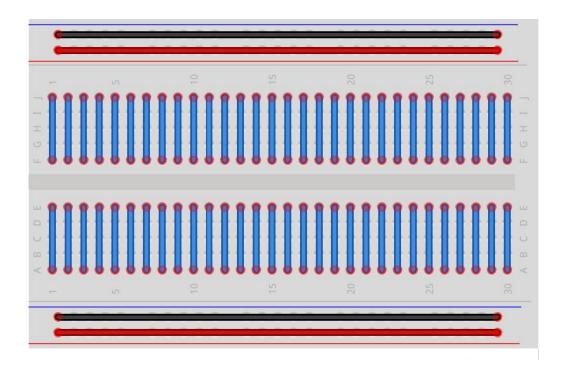
Obs: Comparativo ilustrativo, são dois circuitos diferentes



#### Funcionamento da protoboard

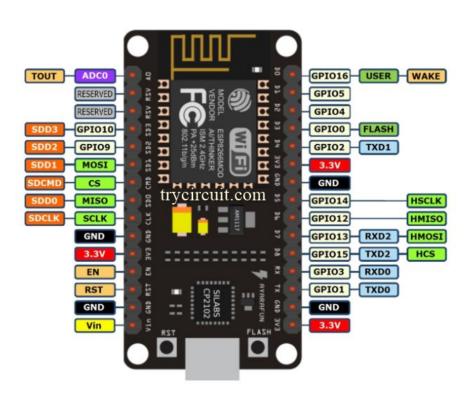


#### Trilhas com diferentes potenciais elétricos





### Interfaceamento com a placa NodeMCU

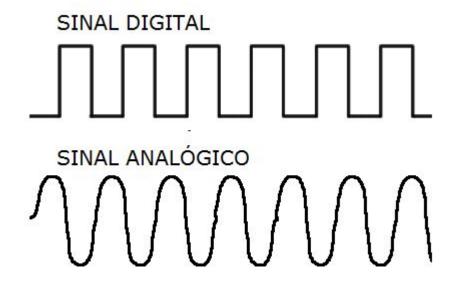


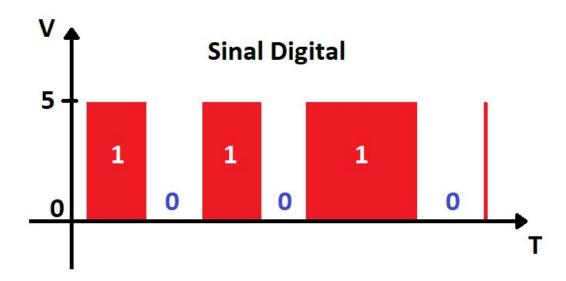
- Cada pino (interface) da placa NodeMCU contém um propósito diferente.
- As interfaces podem ser consideradas digitais e analógicas.
- □ Também temos pinos de comunicação serial (ex: UART, SPI, I2C)



## Sinais elétricos digitais e analógicos

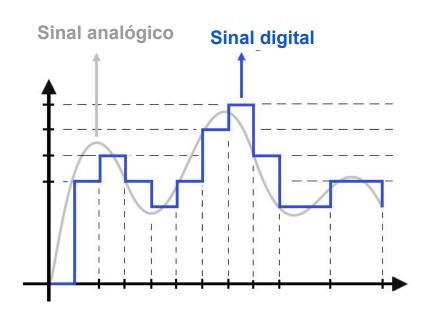
☐ A representação de sinais elétricos pode ser feita de duas maneiras:







## Sinais elétricos digitais e analógicos



Sinal analógico: é composto por um sinal contínuo, que varia em função do tempo. É possível representá-lo com uma curva, que apresenta intervalos com valores que variam entre um possível intervalo definido;

**Exemplos**: sinais "reais" como temperatura, umidade, tensão da rede elétrica e velocidade.

Sinal digital: um digital tem valores discretos, com números descontínuos no tempo e na amplitude. Enquanto o formato analógico apresenta variações infinitas entre cada um de seus valores, o digital assumirá sempre os valores discretos (ex: representação binária), diminuindo a faixa de frequência entre eles e a oscilação.

**Exemplos**: LEDs, buzinas (buzzers), sensor de presença, acionamento de rele.



# **PARTE II - Atividades**



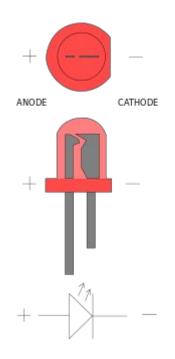
# Repositório com os códigos das atividades

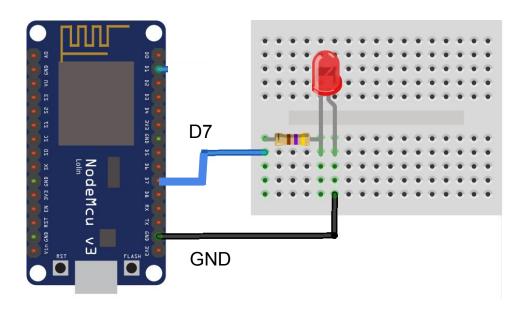
Todos os códigos estão disponíveis em <a href="https://github.com/kakalins/SENAI-IOT">https://github.com/kakalins/SENAI-IOT</a>



#### **Atividade 01: acionamento de um LED**

- Uso do código <u>blink.ino</u>
- Identificar os pinos para a conexão







#### Missão do curso:

A loT é, entre tantas coisas, a conexão entre objetos físicos e tecnologias digitais. Projete um objeto, para uso cotidiano, que se utilize da loT.

## Objetos do cotidiano com IoT

Unidade de controle de ar/sala



Sensor de presença

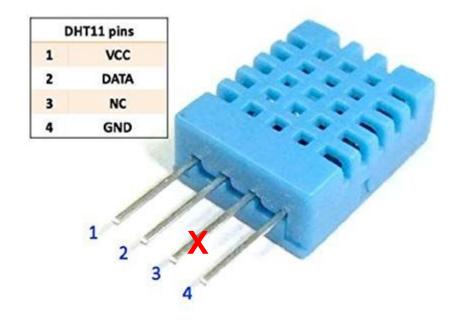


Escolha um dos dois objetos para fazer parte da Internet das Coisas!



## Sensores - Conectando objetos na IoT

#### Sensor de temperatura e umidade



#### Sensor de presença





## Metodologia para a conexão do objeto a loT

#### Teste individual do sensor

Conexão do sensor com o microcontrolador para verificar o funcionamento

# Configuração da plataforma de monitoramento

Configuração da conexão do sistema embarcado com a Internet através de plataforma de monitoramento

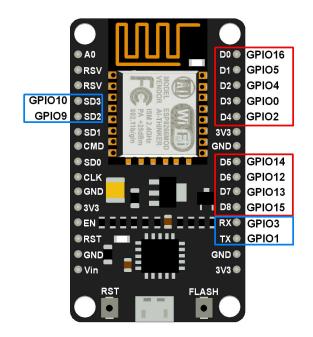
#### Estabelecimento de um objeto conectado

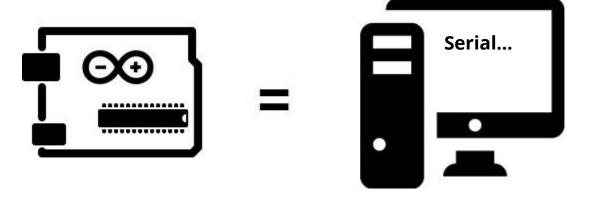
Uso de um objeto do cotidiano com novas funcionalidades devido a conectividade.



#### Atividade 02: teste do sensor

- ☐ Códigos: <u>test-DHT.ino</u> <u>test-PIR.ino</u>
- Visualização de funcionamento via Serial Monitor

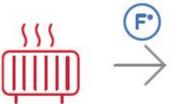






## Aplicações com objetos conectados na loT

Envio dessas informações para nuvem











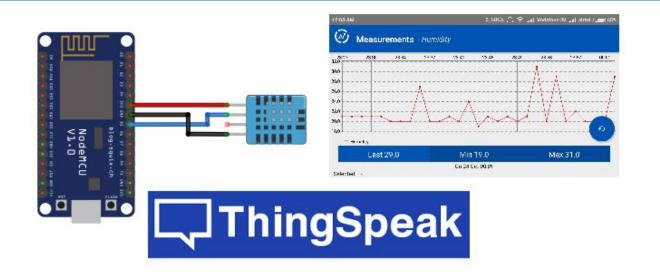


Atuação

Sensor detecta calor ou frio

Tomada de decisão

### IoT - Plataformas gratuitas para monitoramento e controle

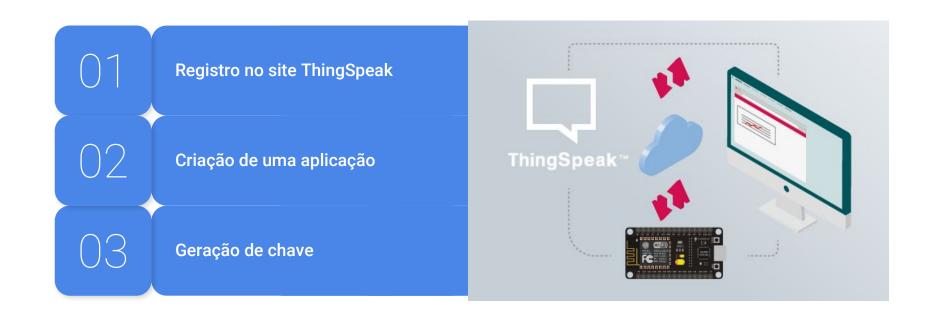






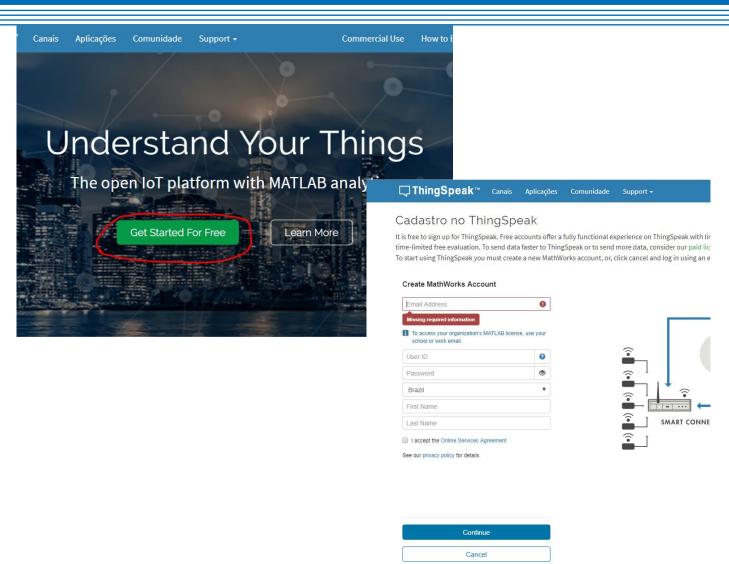
## **Atividade 03: ThingSpeak**

Código: ThingDHT.ino ThingPIR.ino





#### Configuração ThingSpeak

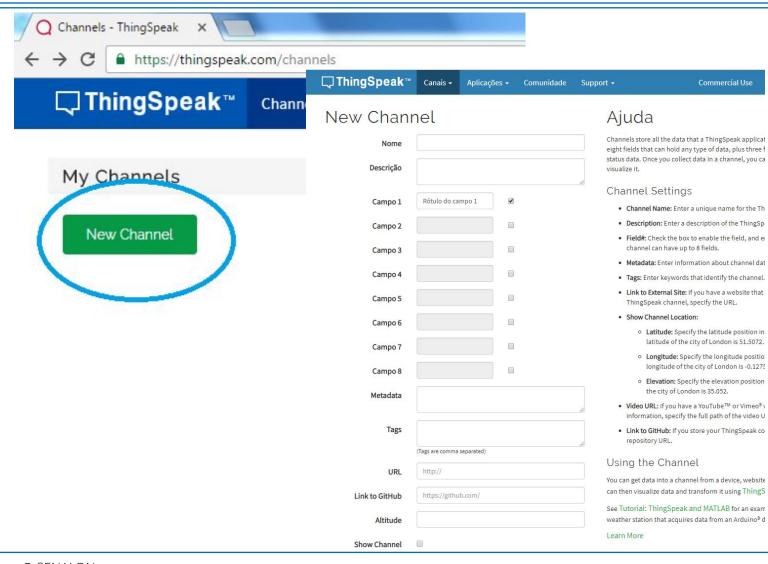


# Siga os seguintes passos para configurar:

- 1. Entre no site ThingSpeak.com.
- 2. Clique em Get Started For Free.
- 3. Na página de cadastro, preencha seus dados para criar sua conta.



#### Configuração ThingSpeak

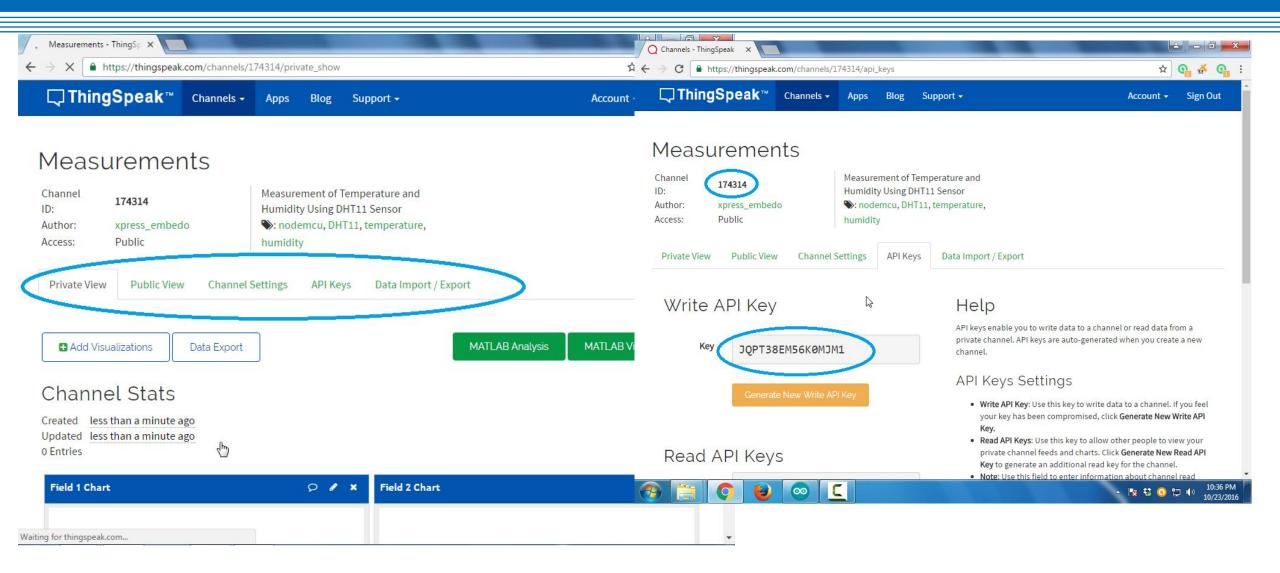


# Siga os seguintes passos para configurar:

- 1. Após o cadastro, clique em New Channel na página de canais.
- 2. Preencha o formulário com os dados do canal.
- 3. Clique em salvar canal no final da página.



#### Configuração ThingSpeak





# **Obrigado!**

https://github.com/kakalins/SENAI-IOT

https://isitics.com

