



## Na Aula 2: (link: <a href="https://goo.gl/3JpX4S">https://goo.gl/3JpX4S</a>)



#### Aula 3

- Broker MQTT Mosquitto
- MQTT PubSubClient
- Versão Final do código do Arduino
- Iniciar Aplicação Android



# Broker MQTT Mosquitto

Mosquitto é um broker de mensagens que implementa o protocolo MQTT. MQTT é uma ferramenta robusta que permite a troca de mensagens usando o modelo publish/subscribe. Isso se encaixa perfeitamente no tema "Internet das Coisas" em dispositivos de sensoriamento, smartphones, computadores e microcontroladores como o Arduino.

Mostrar funcionando!



### Exemplo 12 - MQTT no Arduino

- Baixar o projeto: <a href="https://goo.gl/pFMwSH">https://goo.gl/pFMwSH</a>
- Ler e entender como funciona a API MQTT
- Executar
  - Digitar texto na porta serial
  - Verificar resposta na porta serial
  - Verificar resposta na telona

Aulas 1 e 2: (link: <a href="https://goo.gl/3JpX4S">https://goo.gl/3JpX4S</a>)



#### Exemplo 13 - Versão Final

```
const char topico[] = "v4tech/sylvio/iot"; // use seu nome
// funções
void atualiza led();  // usada somente dentro do loop(). ligar, desligar ou piscar led vermelho
void atualiza status(); // usada somente dentro do loop(), chama enviaJson() em intervalos fixos de 20s
// envia status completo no formado:
// {"led":ESTADO, "temperatura":VALOR} onde ESTADO: "ligado", "desligado", "piscando"
void enviaJson();
// usada dentro do callback() do MQTT para ler extrair a mensagem que chegou
// {"status":0} -> Deve chamar enviaJson() para informar status da placa.
// {"seta led":ESTADO} -> Seta estado do LED. ESTADO: "ligado", "desligado", "piscando". Chama
enviaJson() também.
// {"pisca led":VALOR} -> Seta tempo e pisca led. VALOR: Milisegundos. Chama enviaJson() também.
void analisaJson(String json);
void ligaLed(int porta);
void desligaLed(int porta);
void inverteLed(int porta);
void conectaMqtt();
```

VENTURUS TECH
JANEIRO/16

#### Exemplo 13 - Versão Final

Projeto: <a href="https://goo.gl/mDcKJf">https://goo.gl/mDcKJf</a>

