

Configurações iniciais de MQQT

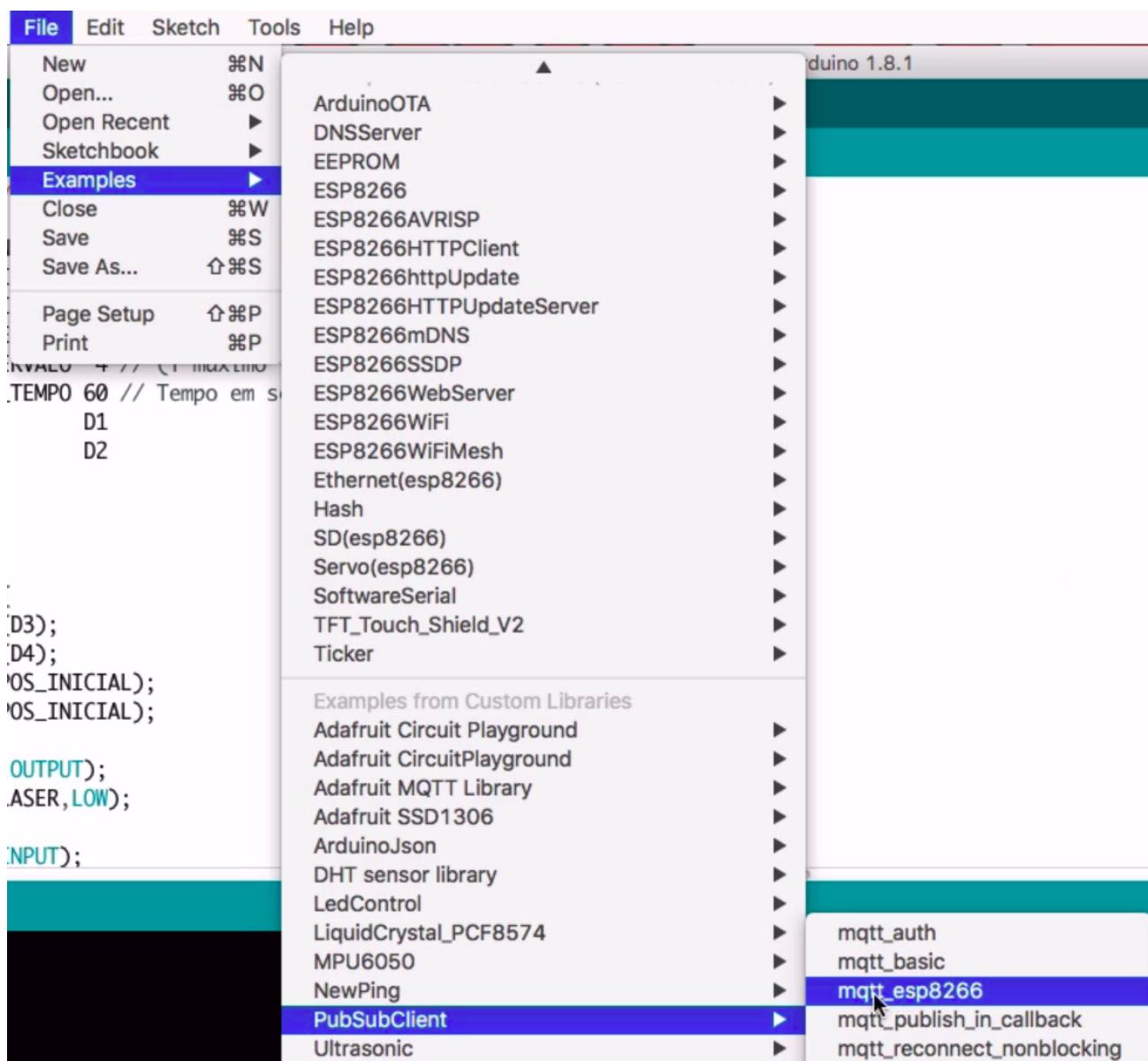
Transcrição

A proposta inicial do nosso projeto já está finalizada, porém, para estendermos um pouco, estudaremos uma parte que envolve IoT que também está presente em outros cursos:

- [Internet das Coisas: Medindo e publicando dados com NodeMcu e MQTT](https://cursos.alura.com.br/course/iot-com-nodemcu) (<https://cursos.alura.com.br/course/iot-com-nodemcu>).

Queremos verificar remotamente, utilizando o sensor PIR, se houve detecção de movimento ou não, via MQTT.

Começaremos abrindo o exemplo padrão `mqtt_esp8266` da biblioteca `PubSubClient` como vemos abaixo:



Faremos algumas adaptações no código exemplificado para evitar escrever tudo do zero, antes, criaremos uma conta no [CloudMQTT](https://www.cloudmqtt.com/) (<https://www.cloudmqtt.com/>) - que será nosso *broker* para envio de mensagens. Após a criação da conta, teremos uma lista que informa quais instâncias foram criadas.

CloudMQTT Instances

The screenshot shows the CloudMQTT Instances dashboard. At the top, there are links for 'Account (Edit)', 'PayPal (Change)', and 'Invoices (All)'. Below that, it says 'Not set up.' and 'rmerces LABS'. In the bottom right corner, there is a blue button labeled '+ Create'.

Clicaremos em "Create" e no formulário que se abre, preencheremos o nome da instância, que pode ser da sua preferência. Neste caso, chamamos de *Robotic Laser CAT*. O plano deixaremos como está (*que é o gratuito - Cute Cat*) e a localização também.

Create new CloudMQTT Instance

PayPal not enabled. Please enable PayPal if you want to subscribe to a paid plan

Name

Robotic Laser CAT

Data center

Amazon US East (Northern Virginia)

Plan

Cute Cat

To learn more about the different plans please visit: www.cloudmqtt.com/plans.html

Create

Cancel

Feito isso, seremos levados a listagem de instâncias novamente. Então clicaremos em **Details** para ver os detalhes da instância que criamos.

Name	Plan	Region	
Robotic Laser CAT	Cat	US	

Nesta página serão encontrados endereços, usuário e senha para utilizar o serviço de *broker* do CloudMQTT. São estes dados que iremos inserir em nosso código. No arquivo da biblioteca que carregamos anteriormente, podem ser encontradas três configurações básicas:

```
const char* ssid = "SEU SSID DE REDE";
const char* password = "SUA SENHA";
const char* mqtt_server = "Disponivel no CloudMQTT";
```

Atualize-as de acordo com as configurações da sua rede e da sua instância do CloudMQTT. Além disso, precisaremos das informações de porta, usuário e senha, isso porque o CloudMQTT precisa da autenticação para permitir a comunicação. Então abaixo das três linhas anteriores, complete as demais abaixo com os dados da sua instância.

```
int mqtt_port = PORTA_DA_INSTANCIA;  
const char* mqtt_user = USUARIO_DA_INSTANCIA;  
const char* mqtt_password = SENHA_DA_INSTANCIA;
```

Lembre-se de salvar esse *sketch* em um diretório acessível.