



Projeto 08

Coisas da Casa – Prática

Jan K. S. – janks@puc-rio.br

ENG4051 – Projeto Internet das Coisas

Testes Iniciais

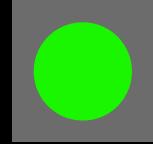
Observe o código inicial de configuração do Matter.

1

Implemente o controle de luz RGB com o LED do ESP 32.

↪ DICA: você tem que fazer um if/else para o acender.

Alexa, acende a
lâmpada 4!

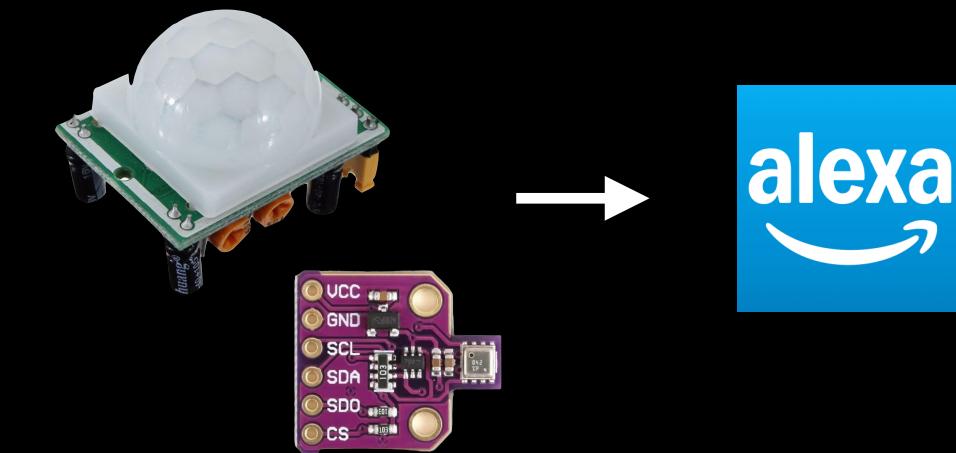


Acenda, apague e mude a cor da luz
usando comandos de voz na Alexa com o número da sua placa.

Ao detectar movimento ou inércia no sensor de movimento,
emita esses eventos para o Matter.

2

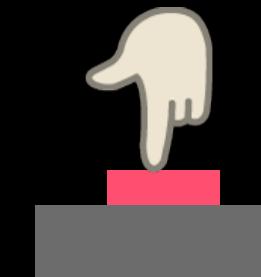
A cada 10 segundos, meça a temperatura e umidade
e envie para o Matter.



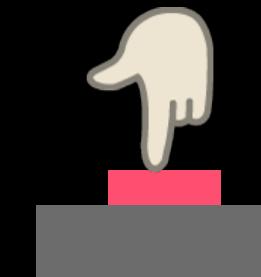
Verifique os resultados no app Alexa do professor.

3

Ao apertar o botão no pino 5,
aumente um contador e imprima na Serial.



Cont: 1



Cont: 2

(reinicia ESP)



Cont: 3

Salve essa contagem no Preferences
ao clicar e carregue na setup.

Comente o código Matter na setup e na loop para desativá-lo.

4

Crie uma rede WiFi AP com o seu nome e uma senha de 8 caracteres.



Crie um servidor com uma rota que retorne o seu nome.

Conecte seu celular nessa rede e acesse a rota pelo navegador.

Testes Iniciais

192.168.4.1/nome

Jan K. S.

Protótipo

1

Copie o código dos Testes Iniciais.

No começo do programa,
exiba na tela e-paper a mensagem a seguir,
com o QR Code para a conexão no WiFi AP.

Conecte-se ao WiFi no QR Code abaixo



2

Monitore o número de conexões no WiFi AP.

Quando alguém se conectar,
exiba 1 vez a mensagem a seguir na tela,
com o QR Code para a página de ajustes.

Agora acesse a página no QR Code abaixo



<http://192.168.4.1/ajustes>

3

Crie variáveis globais para nome da rede WiFi AP e a senha,
começando com texto vazio.

Faça o Build e Upload do LittleFS no ESP32.

Crie uma rota "/ajustes" que renderize o arquivo `ajustes.html`,
substituindo `{{{nomeRedeWiFi}}}` e `{{{senhaWiFi}}}`
pelos textos nas variáveis globais.



4

Ao receber os dados POST da página de ajustes, salve o nome da rede do usuário e sua senha nas variáveis globais.

Salve o nome da rede e a senha também nas Preferências. Carregue-as no começo do programa.

Nome da Rede	Senha
Projeto



```
nomeRedeWiFi = "Projeto";  
senhaWiFi = "2022-11-07";
```

5

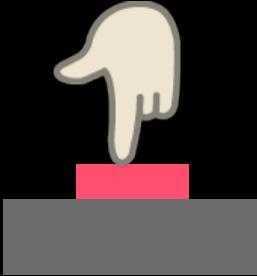
Crie uma variável booleana global modoSetup.

Na setup, carregue ela das Preferências com o valor inicial true.

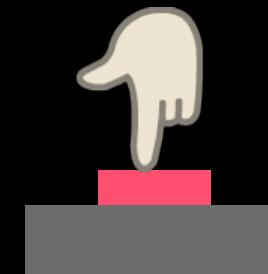
Ao segurar o botão (pino 5), alterne o estado dela, salve nas Preferências e reinicie o ESP32 com o comando `esp_restart()`.

↪ DICA: use a `botao.setHoldHandler`.

Ative o WiFi AP e o servidor somente se a variável for true; caso contrário, ative o Matter e conecte-se na rede WiFi escolhida.



```
modoSetup = true;
```



```
modoSetup = false;
```



Protótipo: Dados do WiFi e Modo Setup

Integração

1

Quando acontecer um evento de mudança de luz, envie um JSON com o número da sua placa e o estado da luz (true/false) para o MQTT no tópico "rgb/NUMERO_DA_SUA_PLACA".

Ao receber esse tópico do Node-RED, insira os dados na tabela luzes.

rgb/NUMERO_DA_SUA_PLACA

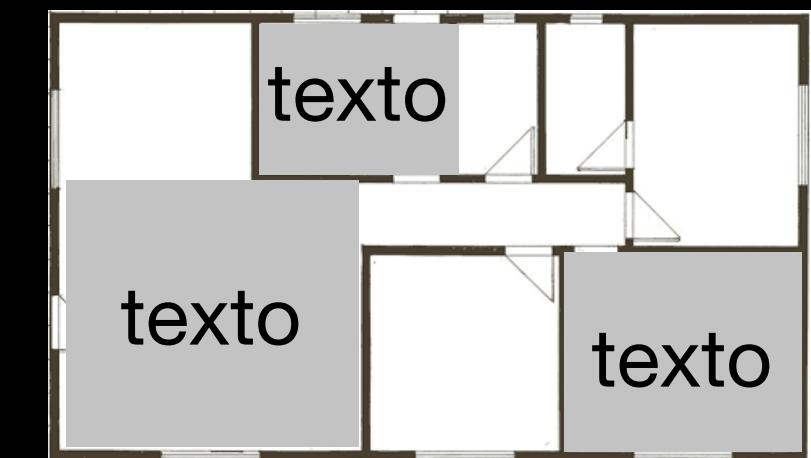
{"numero": SEU_NUMERO, "aceso": true}



No Grafana, crie uma visualização tipo Canva.

2

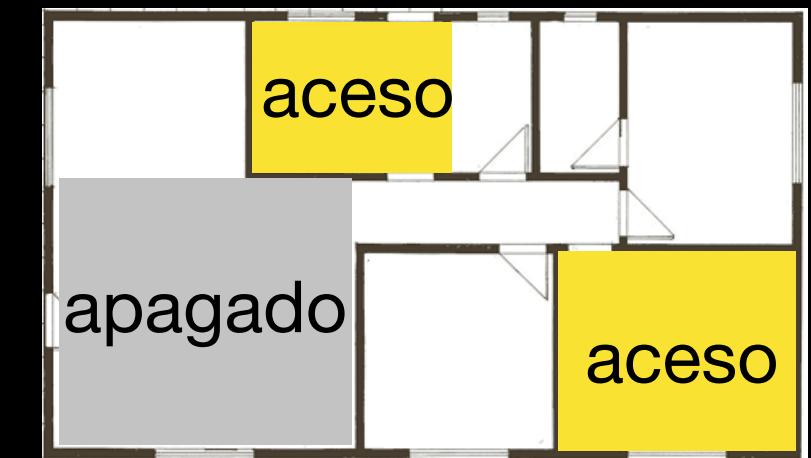
Configure a imagem de fundo para <https://janks.link/iot/casa.png>.



Adicione 3 elementos tipo "texto", representando as lâmpadas da casa.

3

Adicione 3 queries SQL separadas para 3 placas da sala, renomeando aceso para aceso1, aceso2 e aceso3, com ordem descrescente de data/hora e limite de 1 dado.



Faça um Value Mapping para renomear true/false para aceso/apagado, com as cores amarelo/cinza.

Selecione cada elemento e associe seu texto e sua cor de fundo ao valor da lâmpada respectiva. Defina a cor do texto como preto.

Integração: Dashboard com as Lâmpadas da Casa

Desafio

Crie um servidor Flask com uma rota que retorne a string com a versão atual do seu aplicativo ESP (ex: "1.0.1").

1

Crie outra rota que retorne o arquivo `firmware.bin` do build do projeto.

↳ DICA: pesquise sobre a `send_file`.

Rode o Pinggy para gerar um URL público e teste as rotas no navegador.

2

No ESP32, crie uma variável global para a versão do seu app (ex: "1.0.1").

Imprima essa variável no começo do programa.

3

A cada 5 segundos na loop, verifique se há uma nova versão do seu app, fazendo uma requisição ao seu servidor.

Se houver, baixe e atualize o firmware.

4

Aumente a versão (ex: "1.0.2") e faça o build do programa sem upload.

Altere a versão do servidor e teste a atualização automática no ESP32.



janks.link/iot/projeto08.zip

Material do Projeto 08