

# Projeto 05

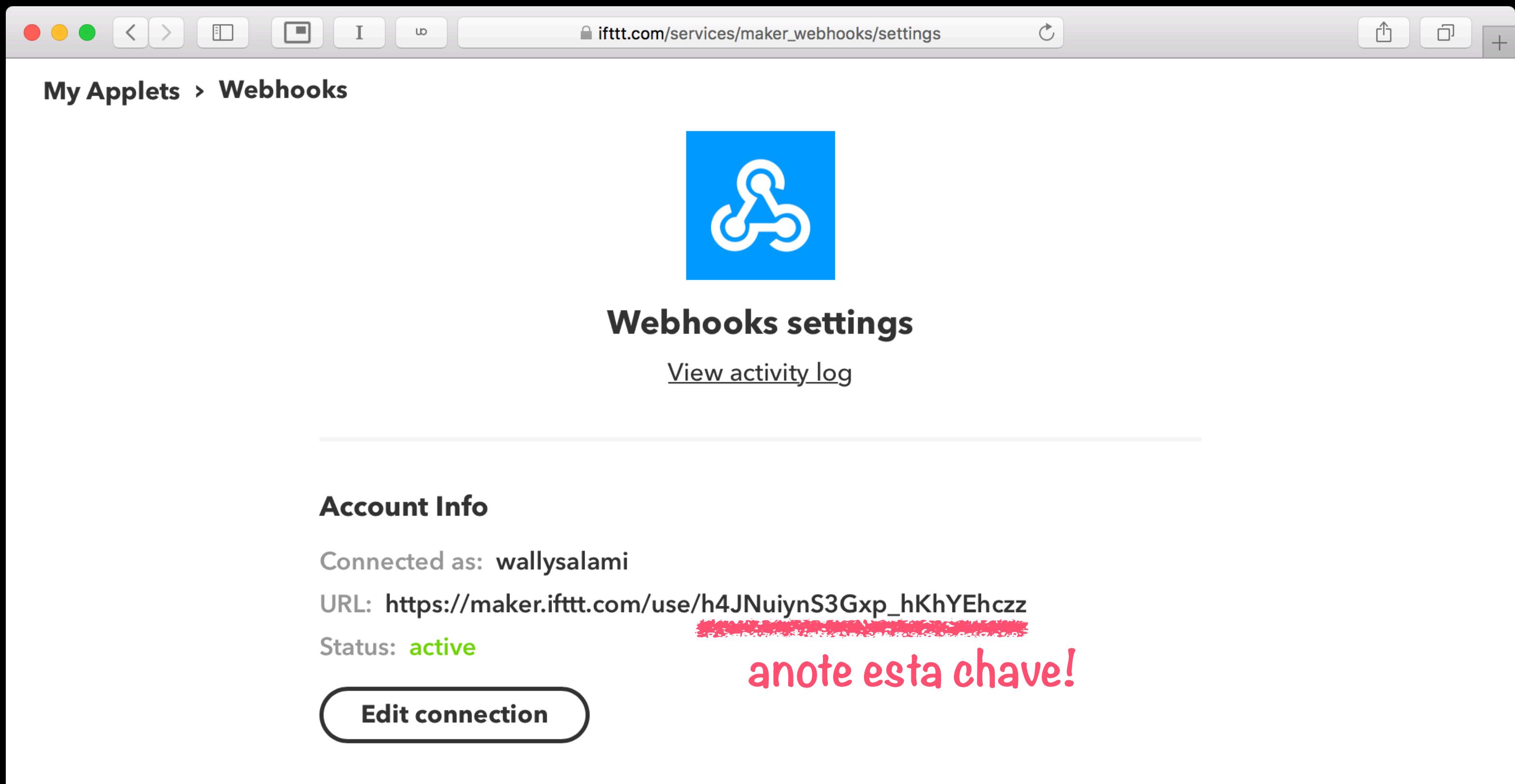
## Controle Automático – Prática

Jan K. S. – janks@puc-rio.br

ENG1419 – Programação de Microcontroladores

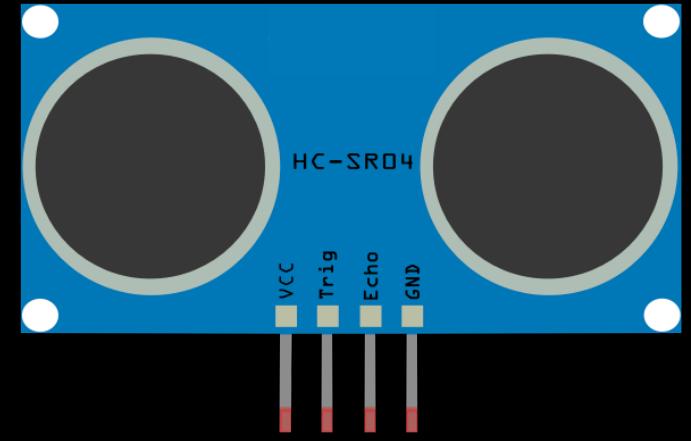
# Testes Iniciais

[https://ifttt.com/services/maker\\_webhooks/settings](https://ifttt.com/services/maker_webhooks/settings)



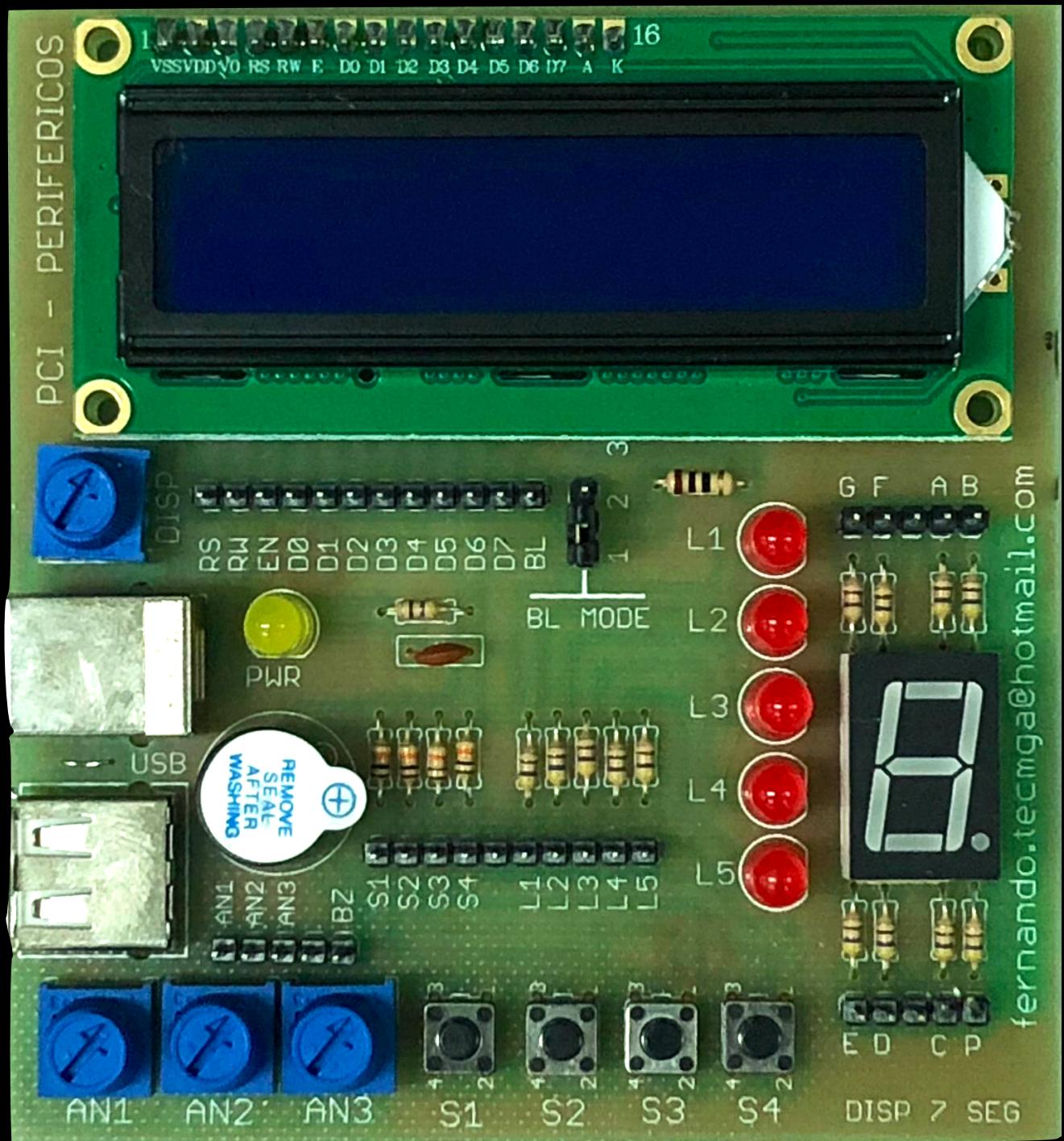
The screenshot shows a web browser window with the URL [https://ifttt.com/services/maker\\_webhooks/settings](https://ifttt.com/services/maker_webhooks/settings) in the address bar. The page title is "My Applets > Webhooks". A blue icon representing a webhook is displayed. The main heading is "Webhooks settings". Below it is a link "View activity log". A horizontal line separates this from the "Account Info" section. In the "Account Info" section, it says "Connected as: wallysalami", "URL: [https://maker.ifttt.com/use/h4JNuiynS3Gxp\\_hKhYEhczz](https://maker.ifttt.com/use/h4JNuiynS3Gxp_hKhYEhczz)" (the URL is redacted), and "Status: active". There is a button labeled "Edit connection". To the right of the redacted URL, there is a red annotation in Portuguese: "anote esta chave!" (note this key!).

Chave Secreta do IFTTT



GPIO 17, 18

GPIO 16



GPIO 11, 12, 13, 14

GPIO 21  
GPIO 22  
GPIO 23  
GPIO 24  
GPIO 25

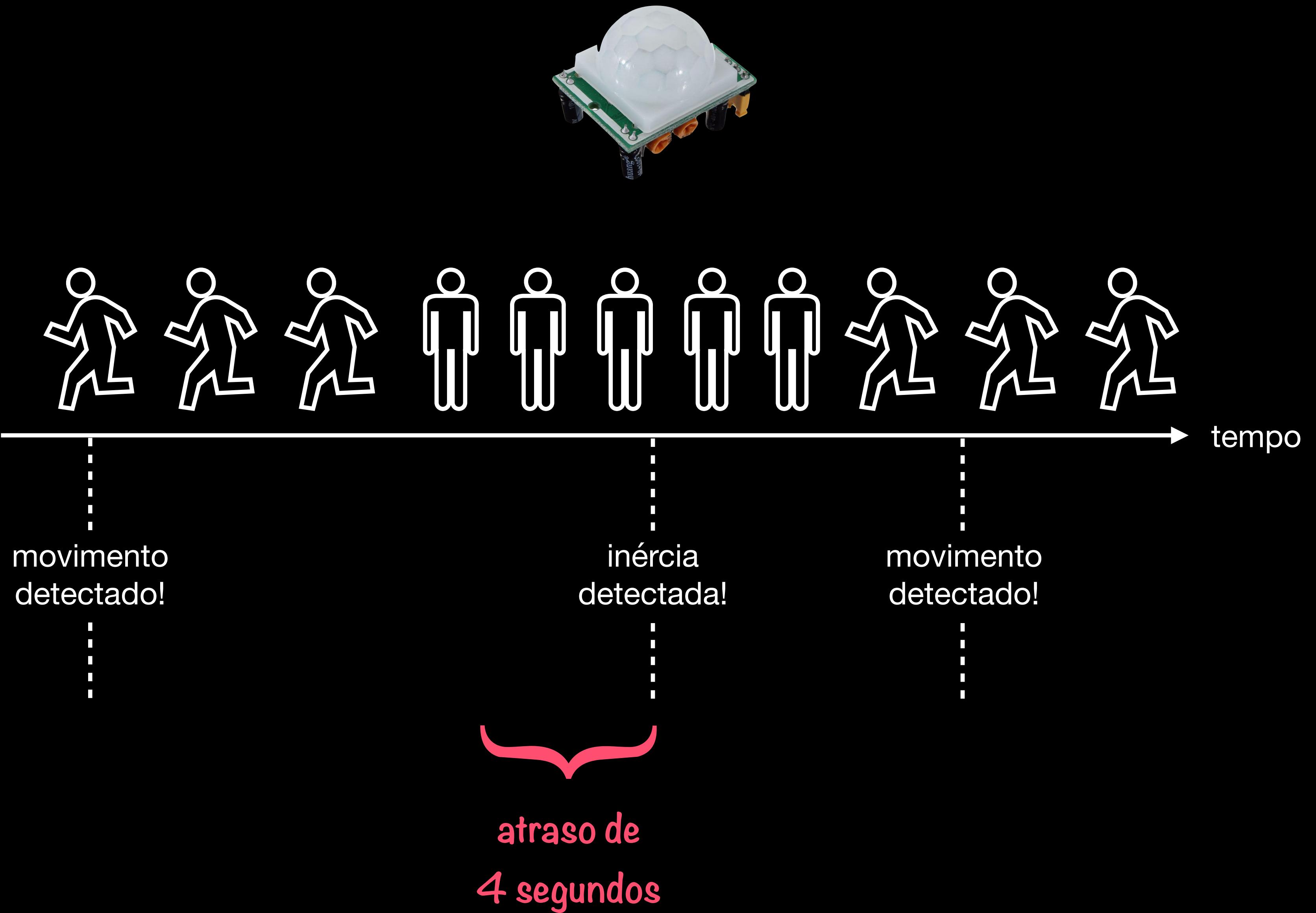


GPIO 8

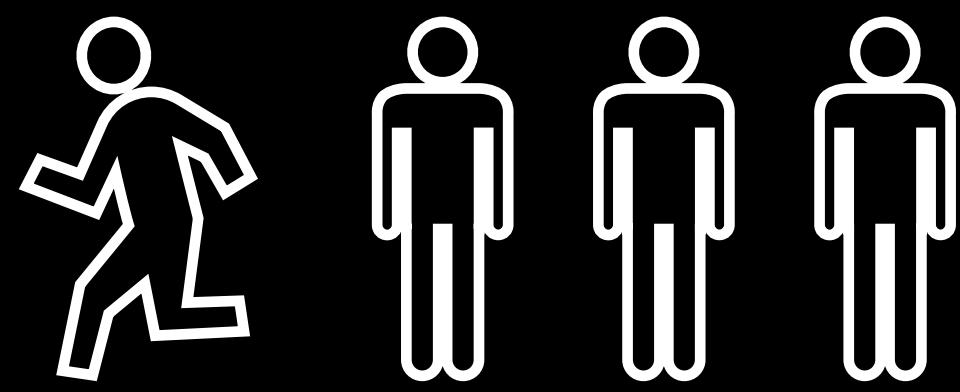


GPIO 27

Conexões com as Portas da GPIO



Atraso de Detecção nos Sensores de Movimento



Como faz para  
esperar mais tempo?



:  
inéria  
detectada!

atraso de  
4 segundos

atraso de mais alguns segundos

ao detectar inéria



inicie um Timer  
para esperar mais tempo

ao detectar movimento



cancele o Timer

Acréscimo no Tempo de Espera da Inéria



## Testes Iniciais

Acenda o LED 1 ao **detectar um movimento**, e apague-o ao **detectar a inércia**.

↪ DICA: use `when_motion` e `when_no_motion`.

Acenda o LED 2 ao detectar um movimento e apague-o **somente se não houver movimento por 8 segundos**.

↪ DICA: veja o algoritmo no slide anterior. O LED 2 nunca pode estar apagado se o LED 1 estiver aceso.

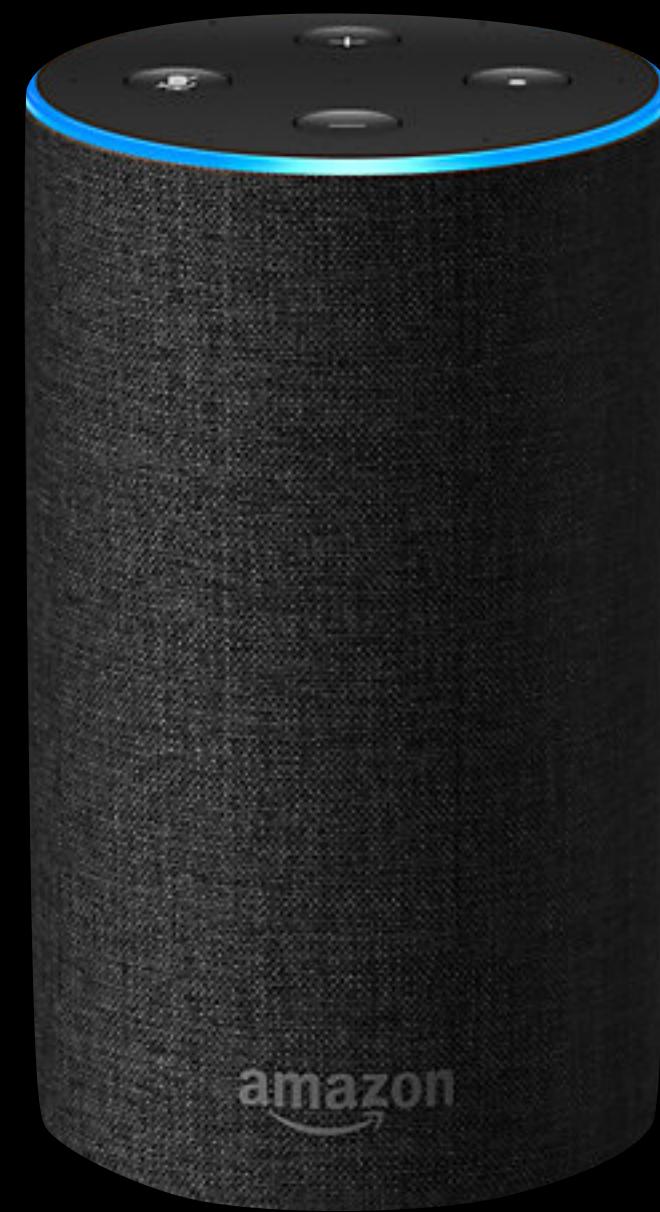
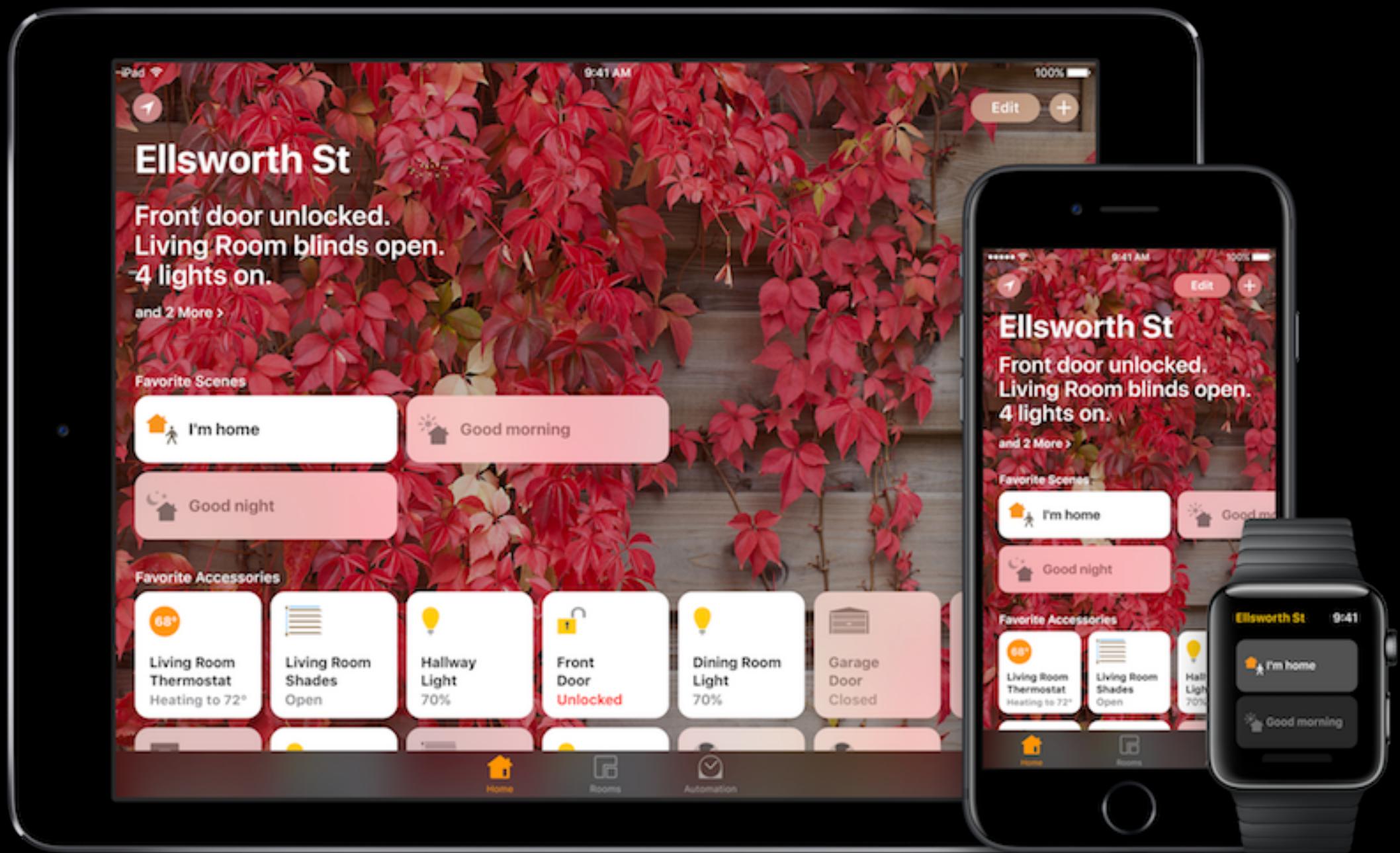
Crie um **applet** no site do IFTTT que, ao receber um evento tipo Webhook, acrescente os 2 valores fornecidos em uma linha no documento "Sensores" do Google Docs, no formato "`{{Value1}}` % de luz / `{{Value2}}` cm".

↪ DICA: depois de criar o applet, copie a sua chave em [https://ifttt.com/services/maker\\_webhooks/settings](https://ifttt.com/services/maker_webhooks/settings).

Ao apertar o Botão 1, **chame o evento do applet acima**, enviando o valor do sensor de luz (em porcentagem) e o do sensor de distância (em cm).

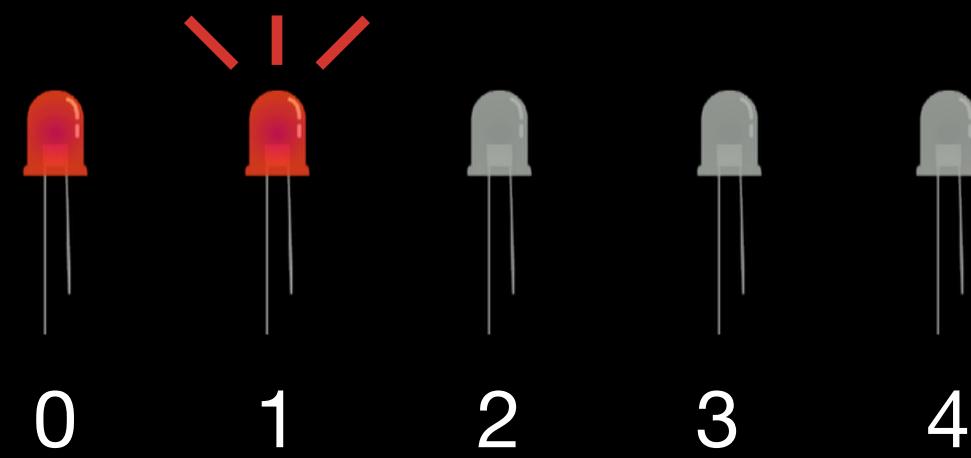
↪ DICA: imprima o `resposta.text`, para verificar se o pedido foi enviado corretamente para o IFTTT.

# Implementação

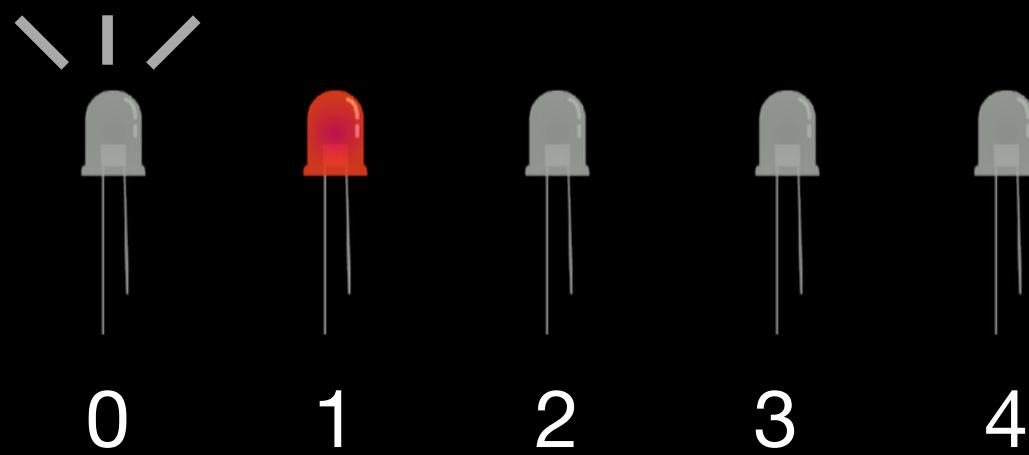


Smart Home

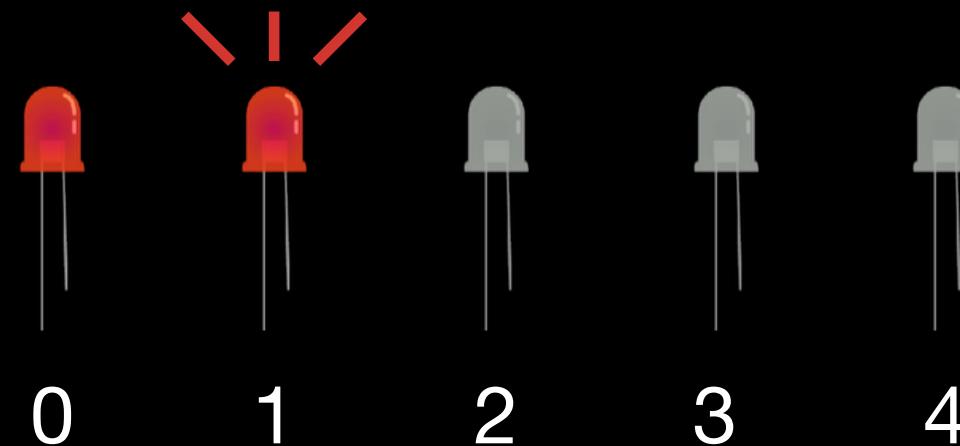
`atualiza_led(1, True)` →



`atualiza_led(0, False)` →

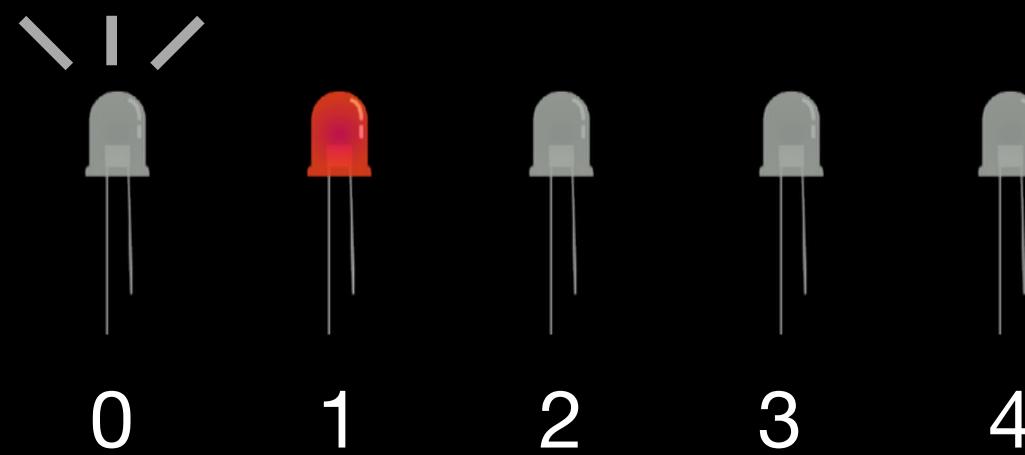


atualiza\_led(1, True) →



```
{  
    "data": datetime(2020, 8, 20, 15, 30, 45),  
    "estado_dos_leds": [True, True, False, False, False]  
}
```

atualiza\_led(0, False) →



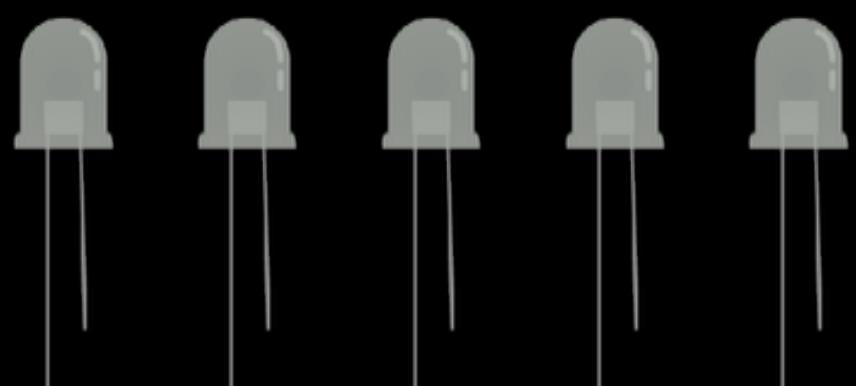
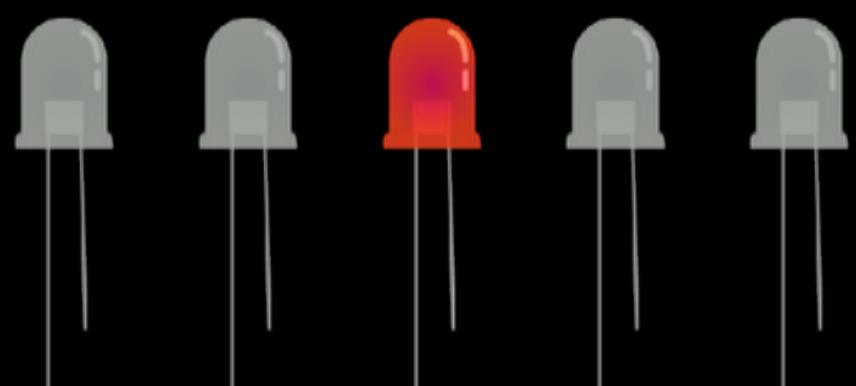
```
{  
    "data": datetime(2020, 8, 20, 15, 52, 27),  
    "estado_dos_leds": [False, True, False, False, False]  
}
```

turn **on** light number 3

turn **off** light number 3

<http://.../luz/3/on>

<http://.../luz/3/off>





## Implementação

Crie a função `atualiza_led` que recebe o índice do LED + o novo estado (True = aceso, False = apagado) e altere o estado desse LED. Chame a função para testar.

Dentro da função anterior, crie e insira um documento com a data/horário atual + lista com o estado (True/False) de cada LED em uma coleção do banco de dados. Teste a função novamente e verifique o MongoDB Compass.

↪ DICA: preencha a lista usando `is_lit` dentro de um `for`.

Inicie um servidor Flask com uma página que receba o número do LED + o estado desejado (ex: /led/1/on) e atualize o LED solicitado. Teste a página em localhost:5000 e depois com o ngrok.

↪ DICA: use os parâmetros `int` e `string` na rota.

Integre o Google Assistant com o IFTTT para controlar as luzes por voz pelo app, usando comandos do tipo "turn on light number 2".

↪ DICA: verifique **atentamente** se o endereço do webhook está sem espaços em branco. E não esqueça de colocar o idioma do aplicativo em inglês.

# Aperfeiçoamento



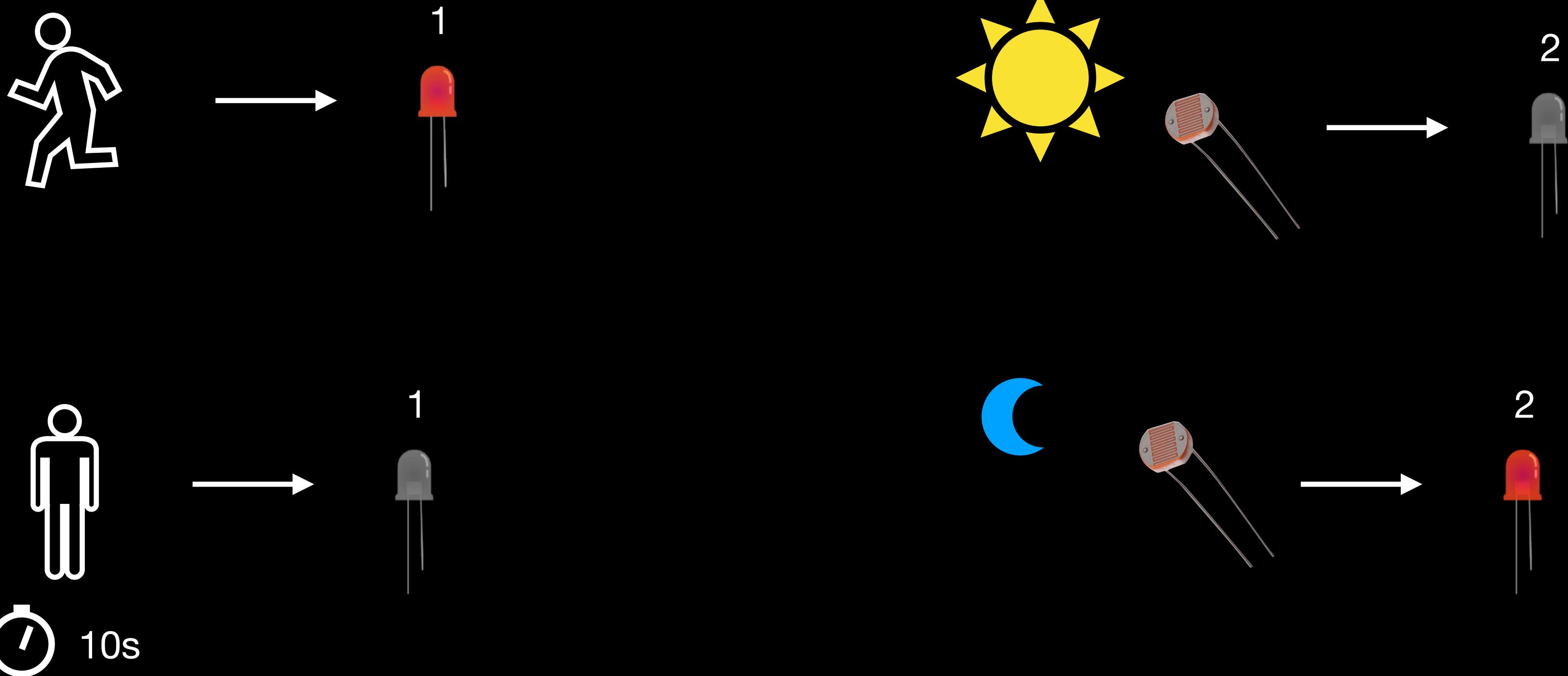
05b\_implementacao.py

cópia  
----->



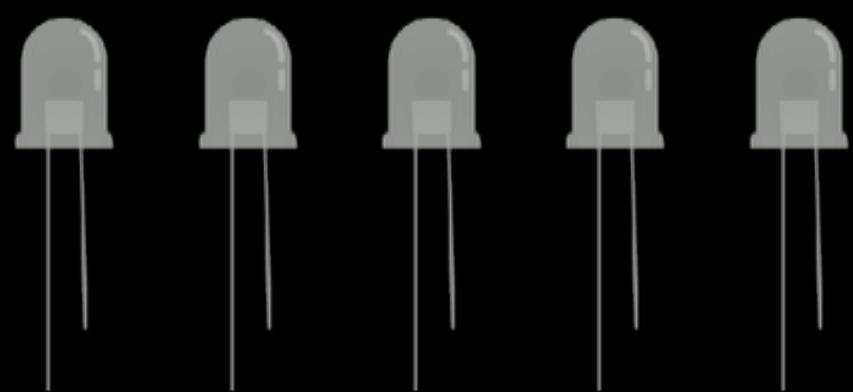
05c\_aperfeicoamento.py

Cópia do Código da Implementação para o Aperfeiçoamento



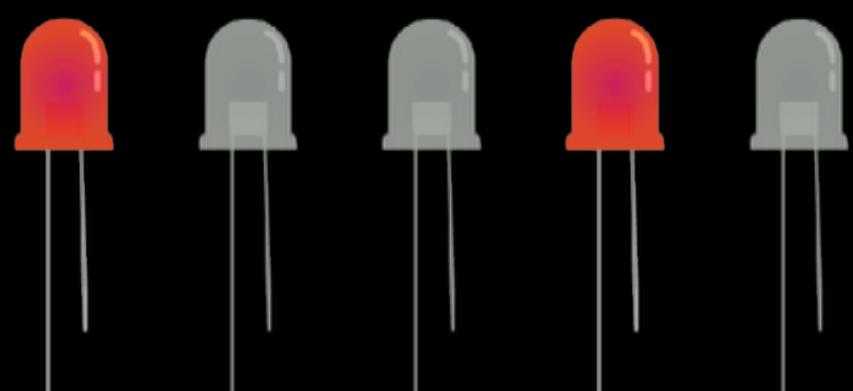
Controle dos LEDs 1 e 2 de Acordo com os Sensores

# Início do Programa

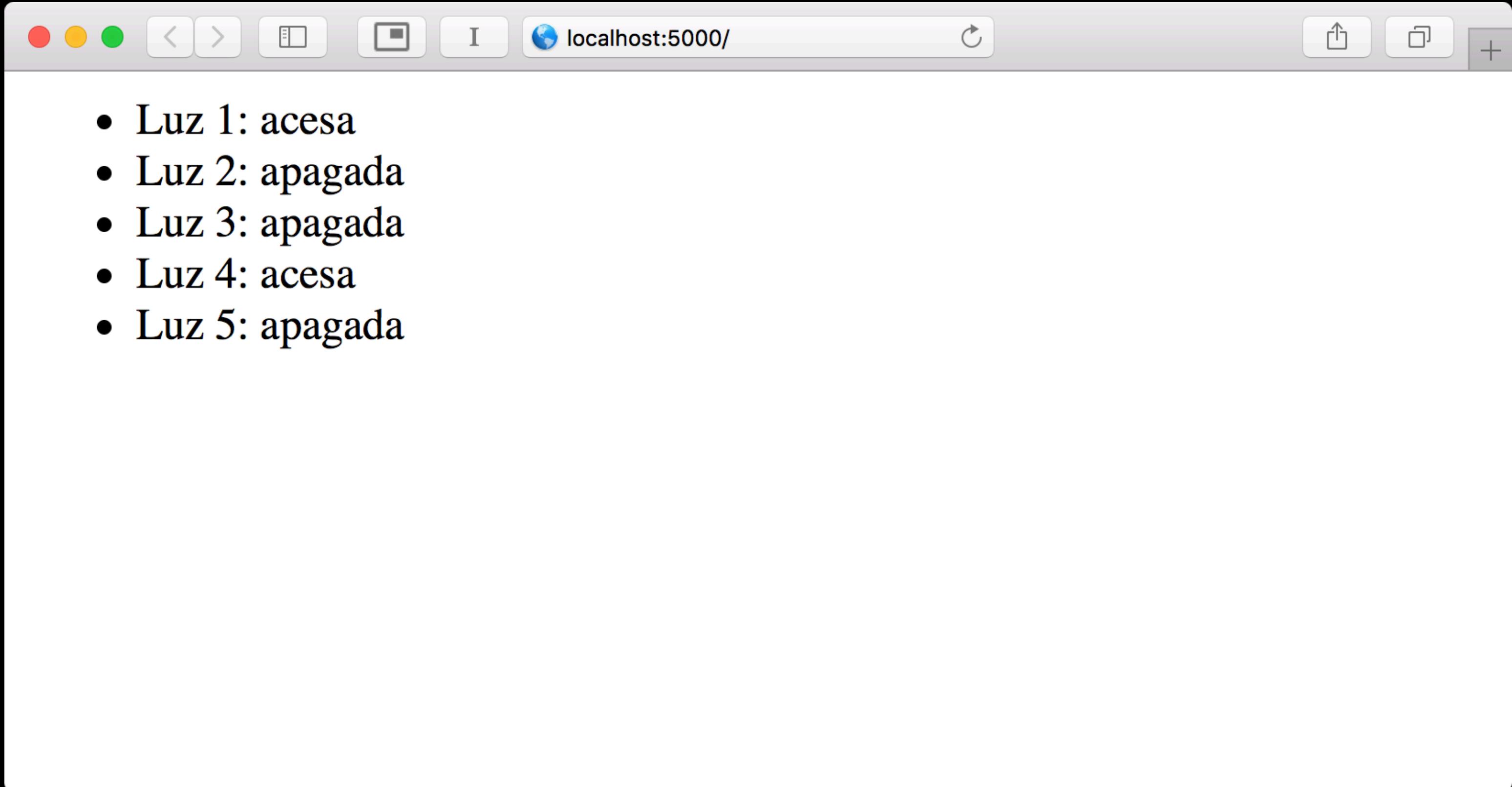


Último Dado:

```
{  
    "data": datetime(2020, 8, 20, 15, 30, 45),  
    "estado_dos_leds": [True, False, False, True, False]  
}
```



Restauração do Estado Anterior dos LEDs



- Luz 1: acesa
- Luz 2: apagada
- Luz 3: apagada
- Luz 4: acesa
- Luz 5: apagada

Página com o Estado de Todos os LEDs



## Aperfeiçoamento

Controle o LED 1 também de acordo com sensor de movimento: acenda-o quando houver movimento, e apague depois de 10 segundos sem movimento, chamando a função `atualiza_led` da Implementação.  
↪ DICA: copie e adapte o código dos Testes Iniciais.

Controle o LED 2 de acordo com a luz captada: acenda-o caso esteja escuro e apague-o caso esteja claro, chamando a função `atualiza_led` da Implementação.  
↪ DICA: use `when_dark` / `when_light`.

Quando o programa iniciar, busque o dado mais recente no banco e restaure o estado anterior dos LEDs.  
↪ DICA: você pode passar um critério de busca vazio + ordenação para a `find_one`.

Crie uma página que verifique e exiba o estado de todos os LEDs, seguindo o formato HTML dos slides anteriores.  
↪ DICA: gere o HTML dentro da função, concatenando strings dentro de um `for`.

# Desafio Extra



05c\_aperfeicoamento.py

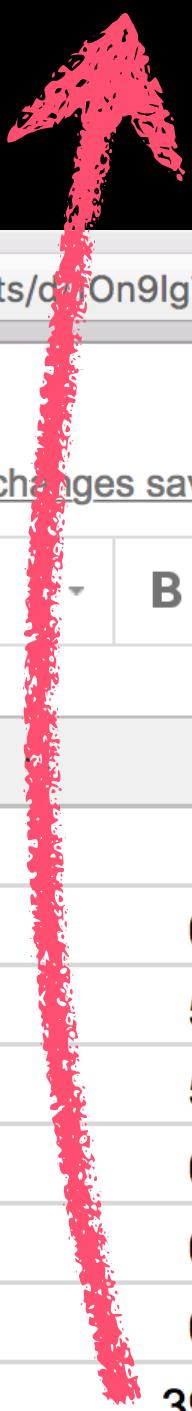
cópia  
----->



05d\_desafio.py

Cópia do Código do Aperfeiçoamento para o Desafio

entre 12:18:00 até 12:18:59,  
Luz 3 ficou acesa durante 39 segundos

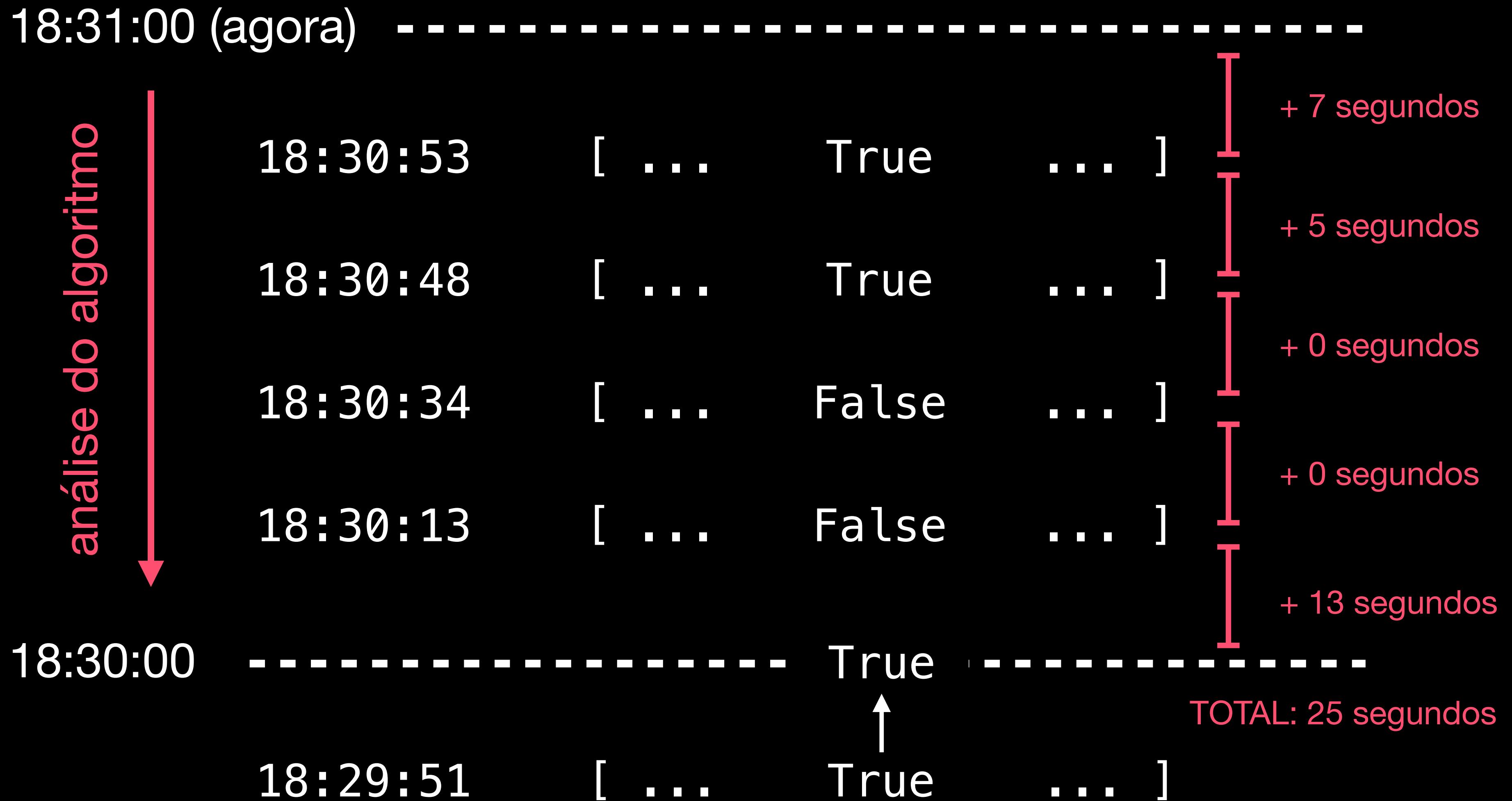


A screenshot of a Google Sheets document titled "IFTTT\_Maker\_Webhooks\_Events". The document has a single sheet named "Página1". The data is organized into columns: Column A contains dates and times; Columns B through F contain numerical values for five different lights (Luz 1 to Luz 5). The data shows that Luz 3 was active for 39 seconds at 12:18:59.

	A	B	C	D	E	F
5	Data	Luz 1	Luz 2	Luz 3	Luz 4	Luz 5
6	September 17, 2018 at 12:12PM	0	0	0	0	0
7	September 17, 2018 at 12:13PM	8	5	4	4	0
8	September 17, 2018 at 12:14PM	8	5	4	8	3
9	September 17, 2018 at 12:15PM	0	0	0	0	0
10	September 17, 2018 at 12:16PM	0	0	0	0	0
11	September 17, 2018 at 12:17PM	0	0	10	13	0
12	September 17, 2018 at 12:18PM	51	39	18	0	6
13	September 17, 2018 at 12:19PM	1	1	0	0	3
14	September 17, 2018 at 12:21PM	0	0	0	0	0
15	September 17, 2018 at 12:21PM	29	0	0	0	0
16	September 17, 2018 at 12:22PM	29	0	0	3	2
17	September 17, 2018 at 12:23PM	0	0	0	3	2
18	September 17, 2018 at 12:24PM	0	0	0	0	0

Planilha com Resumo do Tempo Total de Iluminação de Cada Luz

## LED N



Exemplo de Contagem do Tempo a Partir de Vários Dados



## Desafio Extra

Crie uma função que receba o número de um LED e uma data/horário, **retornando total de segundos em que o LED ficou aceso após aquela data**. Teste a função no shell do Thonny.

↪ DICA: use as funções `find` / `find_one`, os operadores `$lt` / `$gt` e a ordenação decrescente.

Crie uma função que obtenha o tempo aceso dos 5 LEDs desde 1 minuto atrás e **escreva esses valores + a data em uma linha na planilha do Google Sheets**.

Teste a função no shell do Thonny.

↪ DICA: lembre-se que existe um limite no número de parâmetros que podem ser passados para o IFTTT.

**Crie um Timer** que chame a função acima a cada 30 segundos.



[janks.link/micro/projeto05.zip](https://janks.link/micro/projeto05.zip)

Material do Projeto 05