

MYIOT.SPACE

AULA 2

ALARME DE

PRESENÇA

Sustentabilidade e IOT

Visão geral

TÓPICOS

- 1 • Materiais necessários
- 2 • Proposta educativa e questionamentos iniciais
- 3 • Tutorial de programação e conexão
- 4 • Introdução a IoT



Resumo: Esta aula aborda a criação de um sistema de alarme inteligente, integrando a montagem física com a internet para acompanhamento remoto.

1. MATERIAIS

CONTROLADOR

Arduino Uno
Arduino Sensor Shield
V5.0

SENSORES E ATUADORES

1 x Sensor mini
1 x Módulo Buzzer

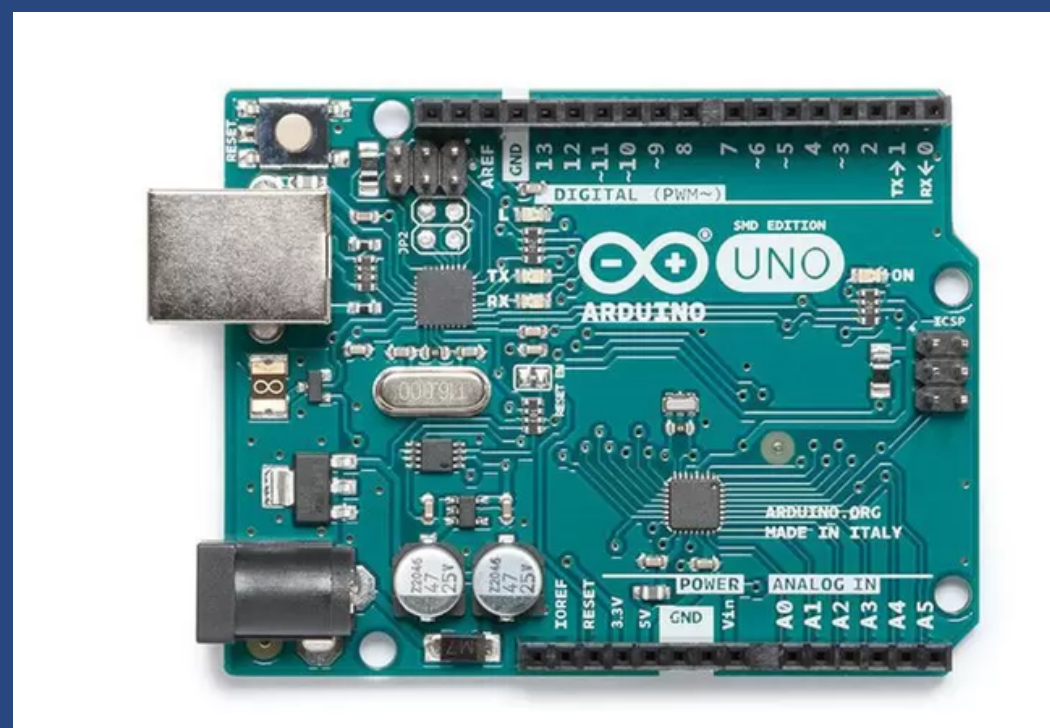
CONECTORES

Jumpers fêmea-fêmea
1 x Cabo para conexão
do controlador (USB AB)

ESTRUTURA (OPCIONAL)

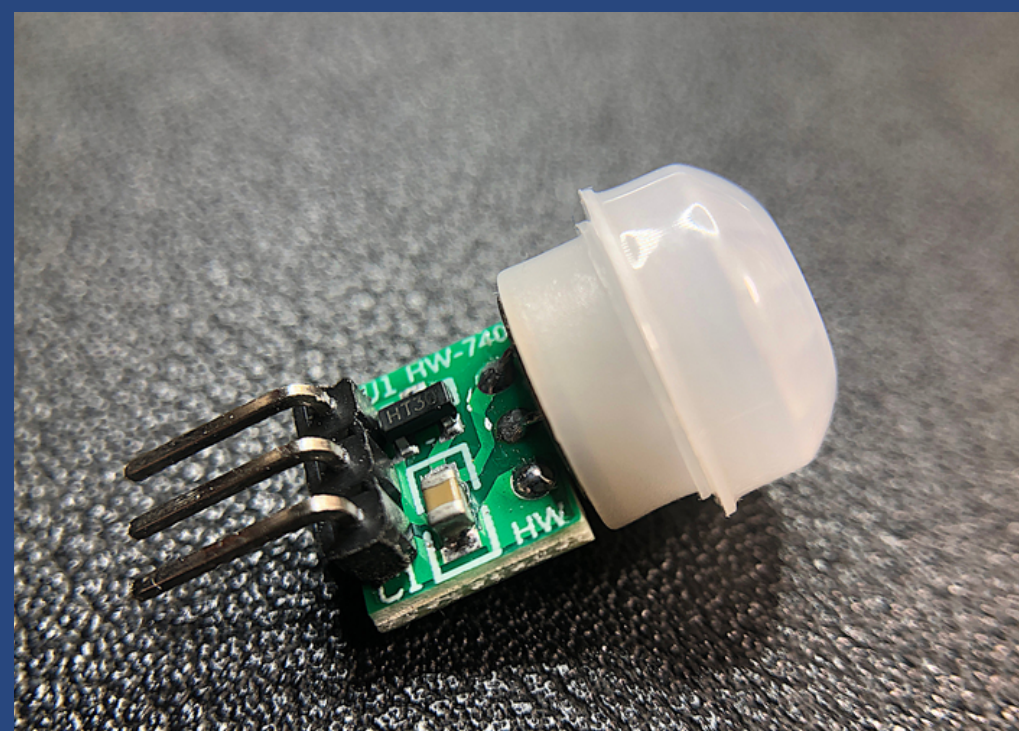
Neste exemplo não
haverá estrutura física.

CONHECENDO MELHOR OS MATERIAIS



MICROCONTROLADOR

O microcontrolador é um pequeno computador, e o cérebro do projeto. Nele definimos instruções, estas que ele segue automaticamente.



MINI PIR

Ele detecta a presença de seres vivos se movimentando pelo ambiente e envia sinais digitais para o microcontrolador.



MODULO BUZZER

O módulo buzzer é um atuador digital. Quando ativado emite um aviso sonoro estridente, o que é ótimo para o projeto do alarme.

2. Proposta educativa e questionamentos iniciais

Para começar a aula

Nesta aula iremos trabalhar com sensores e atuadores digitais para projetar um sistema de alarme de presença inteligente.

Usaremos eletrônica básica e programação por blocos, além dos sistemas de IoT, para realizar o projeto.

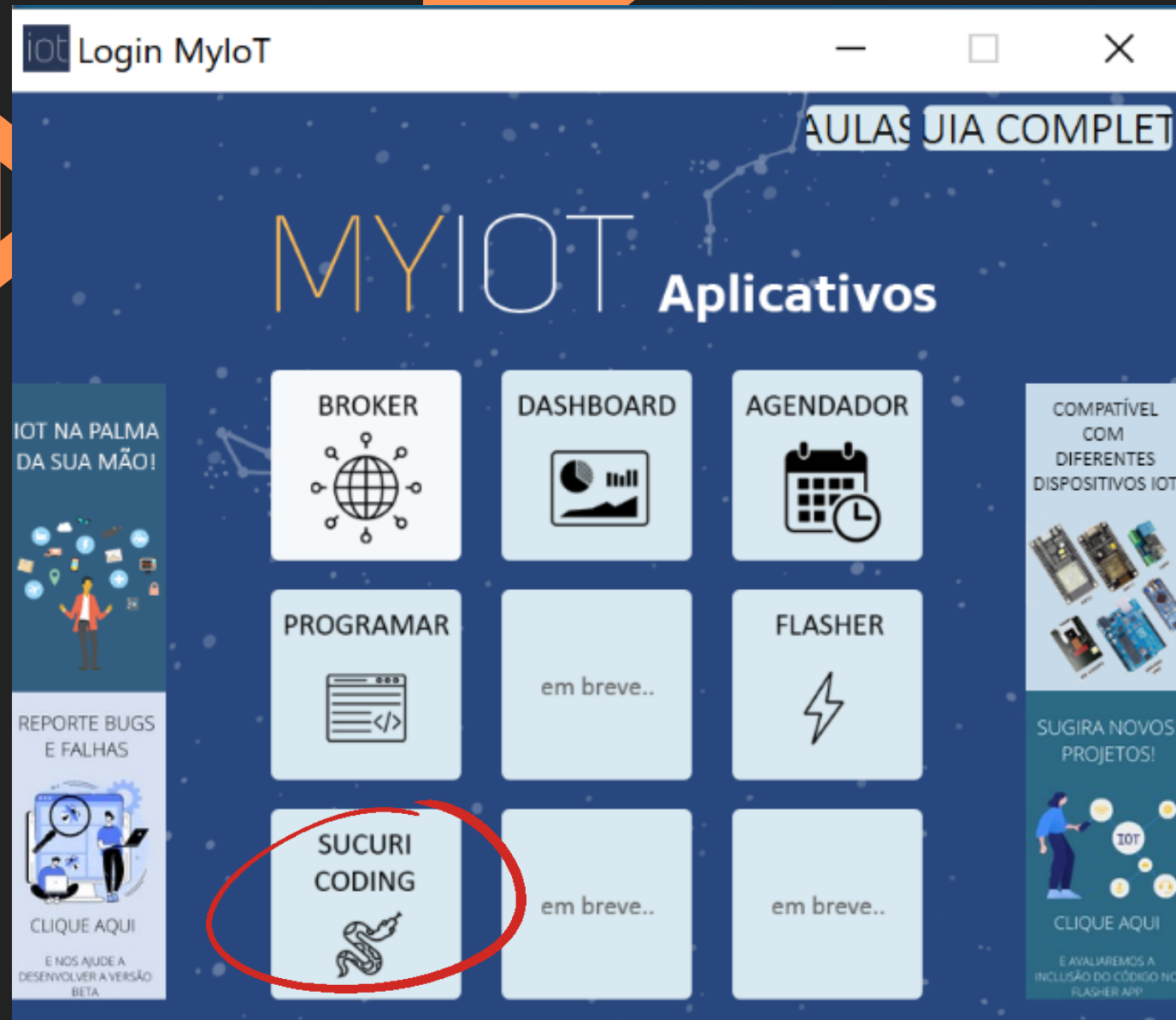
Alarme Residencial Wi-Fi

Este alarme é um exemplo comercial de alarme de segurança conectado a internet, nele é possível fazer o acompanhamento remoto por meio de um aplicativo



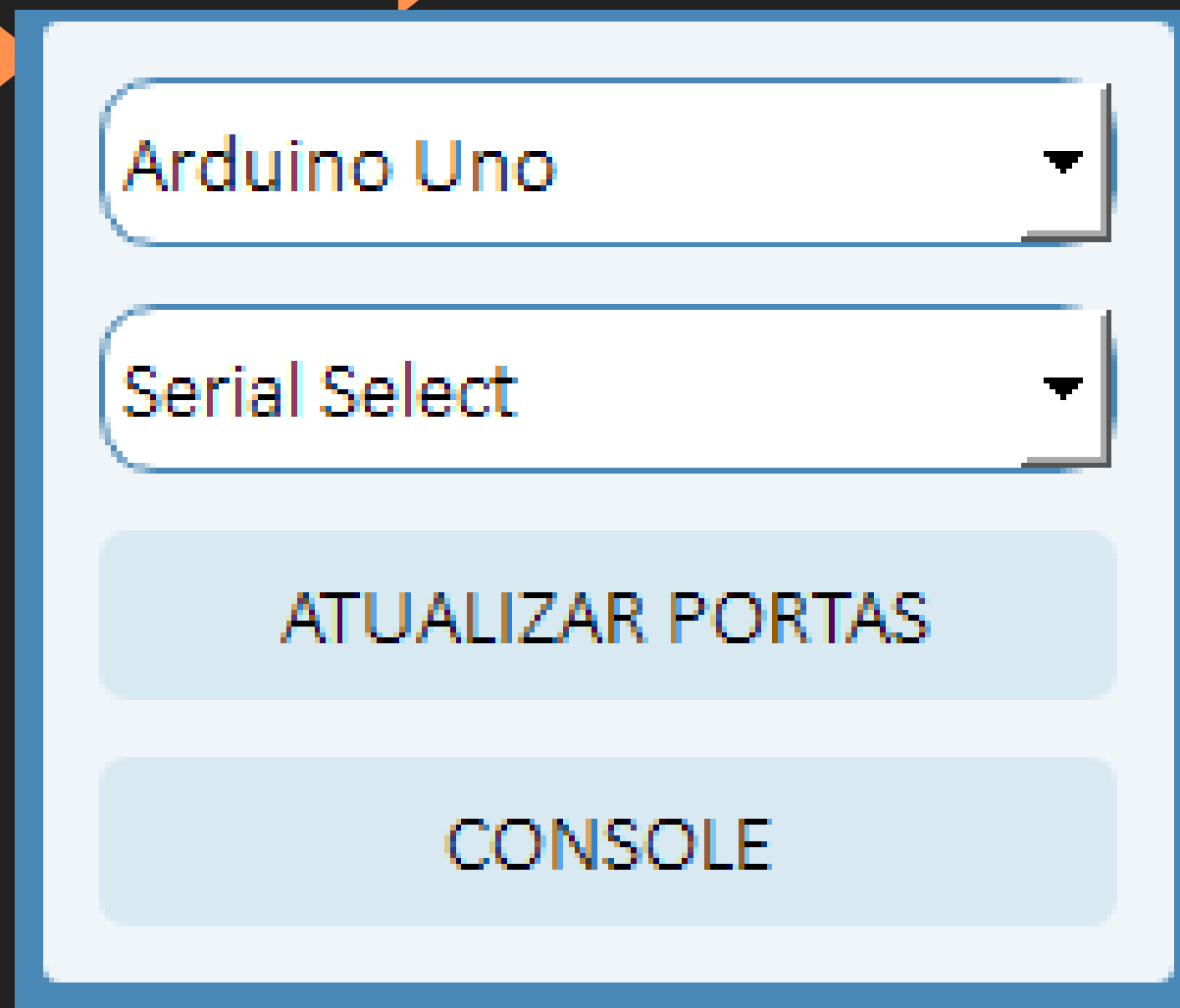
Sensor de presença

É um exemplo comercial muito usado no ambiente doméstico. Ele aciona dispositivos conectados a ele quando detecta a presença de seres vivos em movimento.



Iniciando o programa

Abra “SUCURI CODING” para iniciar a programação por blocos.



Criando a programação

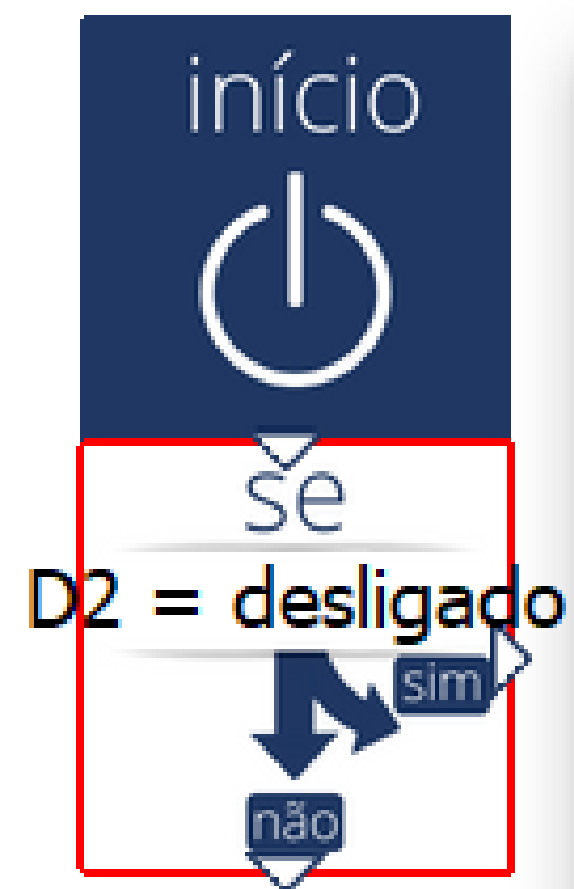
Já no ambiente de programação por blocos, vamos inicialmente pensar na lógica do programa. Queremos que o alarme seja acionado toda vez que haja presença de alguém no ambiente, ou seja, toda vez que o sensor PIR retornar nível lógico alto.

Selecione Arduino Uno como microcontrolador para iniciar a programação.



Definindo a condição

Para ler o sinal do sensor vamos usar o bloco "SE". Este bloco dá a opção de escolher entradas a serem analisadas e comparadas com outros parâmetros. Conecte com o bloco de início e clique 2x para abrir os parâmetros.

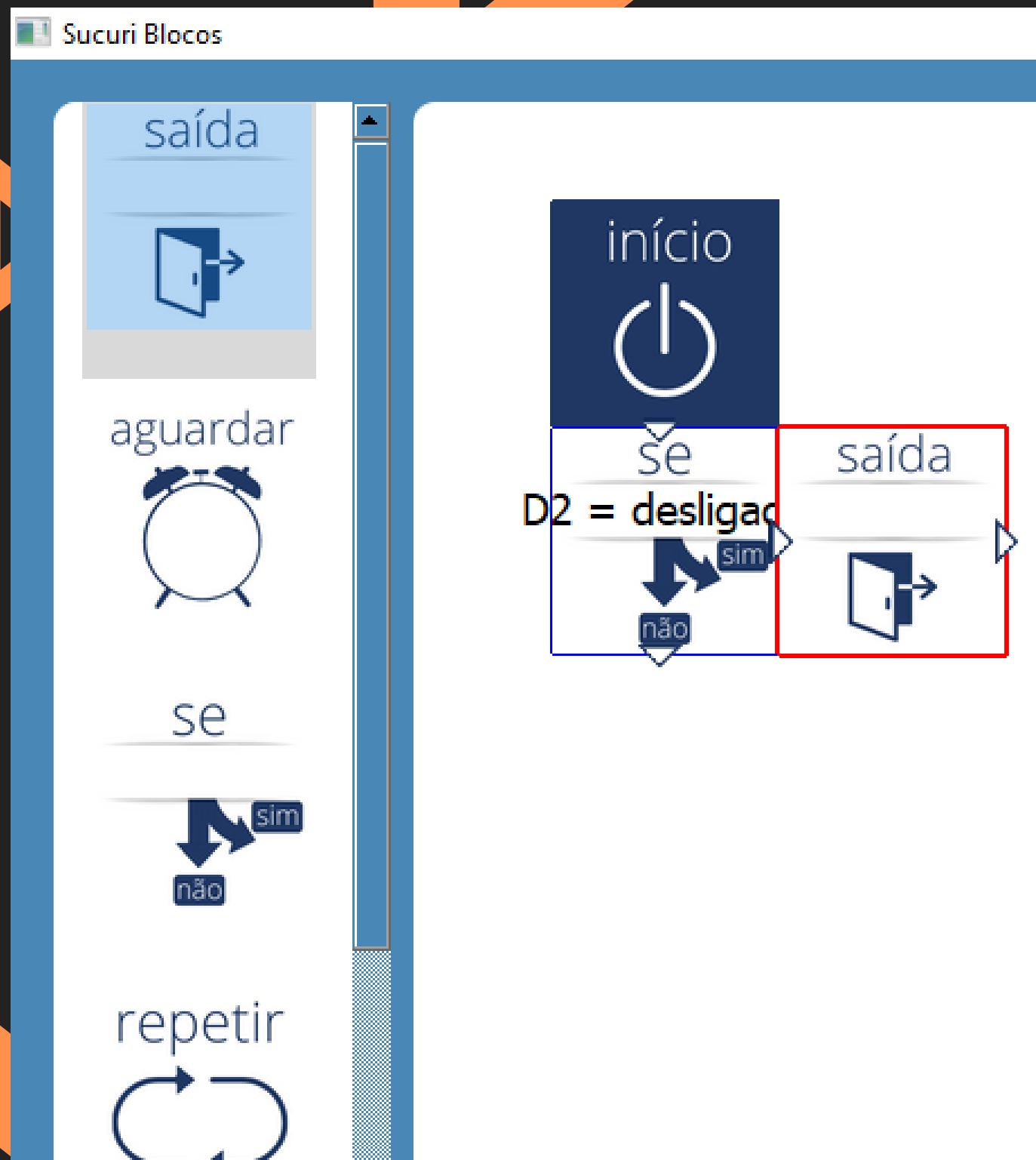


The dialog box titled 'Parâmetros Blo...' has a blue background. It contains three dropdown menus: the first is set to 'D2 Digital', the second to 'Iqual (=)', and the third to 'desligado'. At the bottom is an 'Ok' button.

Definindo a condição

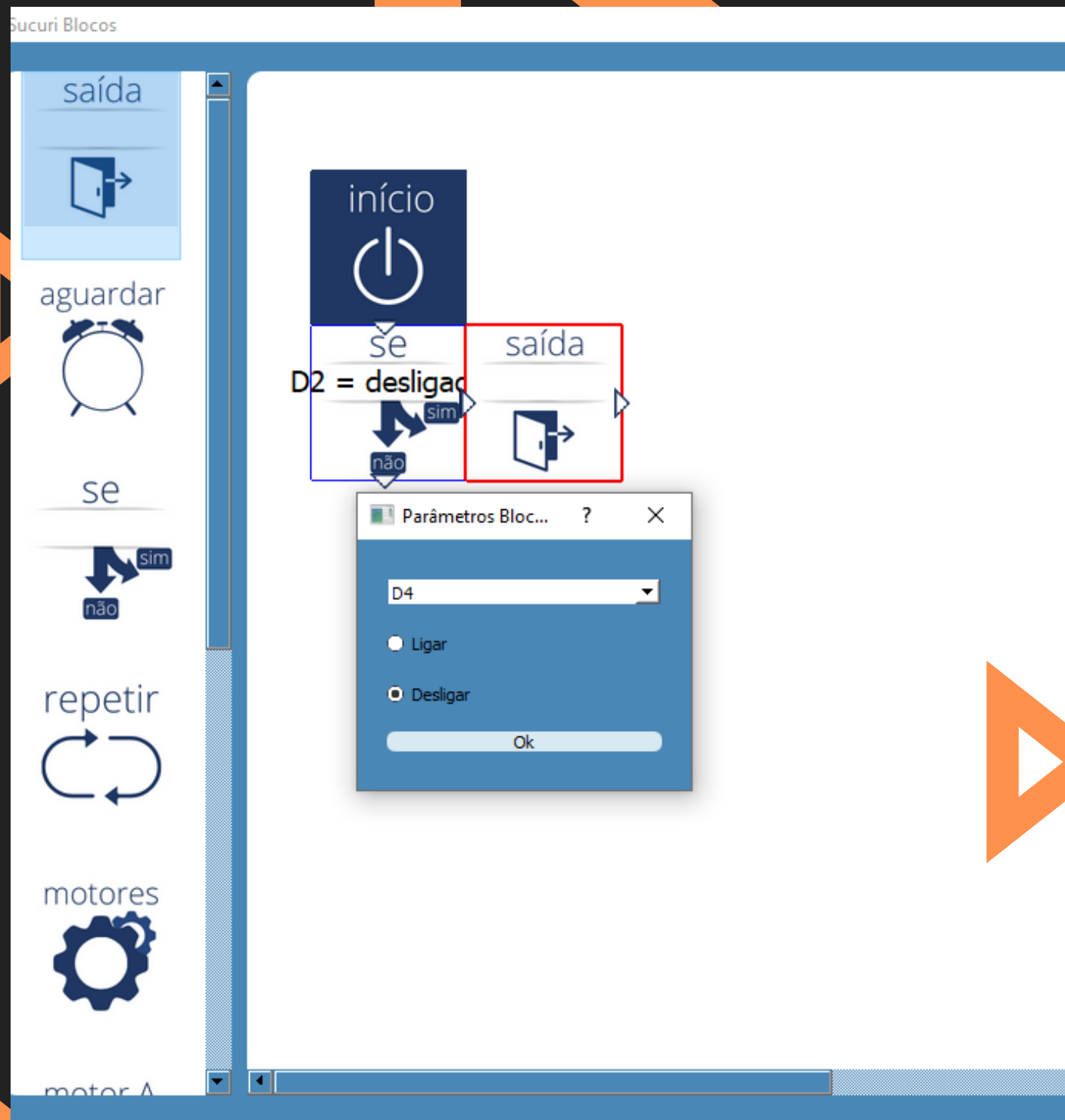
Em parâmetros, escolha a entrada digital D2 e selecione "DESLIGADO".

Toda vez que o sensor enviar sinal lógico baixo (desligado) a condição irá ser verdadeira, portanto sairá para "SIM".



Definindo a saída

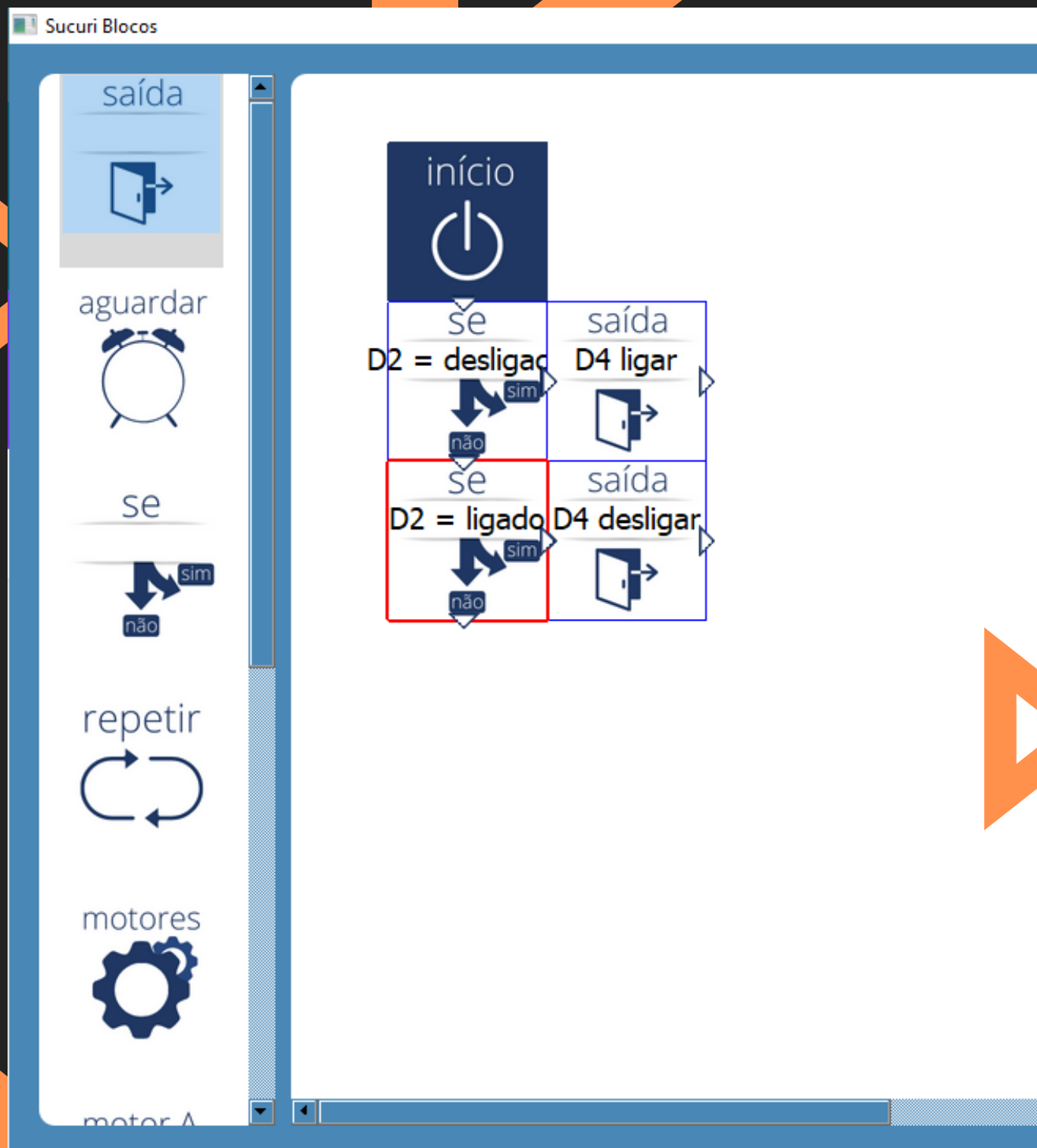
Agora vamos definir a porta que será ligado o Buzzer. Arraste o bloco SAÍDA para o lado direito do SE, clique 2x para abrir seus parâmetros.



Definindo a saída

Em parâmetros, escolha a saída digital D4 e selecione DESLIGAR.

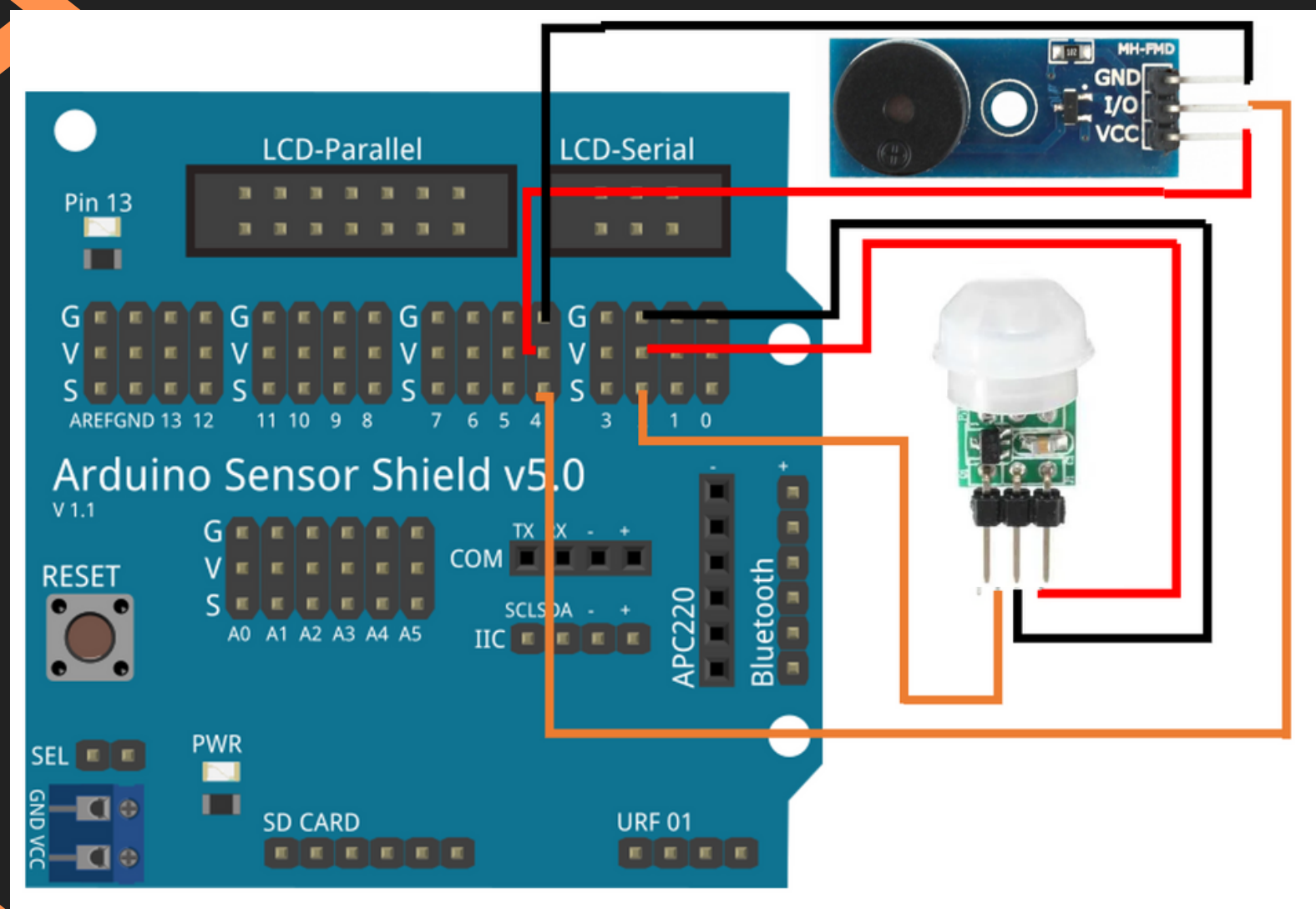
Toda vez que a condição do bloco SE for verdadeira, a saída D4 será desativada.



Criando a rotina

Repita o procedimento dos dois primeiros blocos porém agora ligando a saída D4 quando o sensor estiver LIGADO. Para isso, selecione os 2 blocos anteriores e pressione "Control + C". Arraste os blocos para baixo e conecte-os com os anteriores.

Clique 2x no novo bloco de SAÍDAe SE e altere para "LIGAR".



Conexões físicas

Neste projeto, será necessário conectar o módulo Buzzer e o mini PIR no Arduino Uno utilizando o Arduino Sensor Shield para facilitar as conexões. Utilizando seis jumpers fêmea - fêmea. Faça as seguintes conexões:

Buzzer:

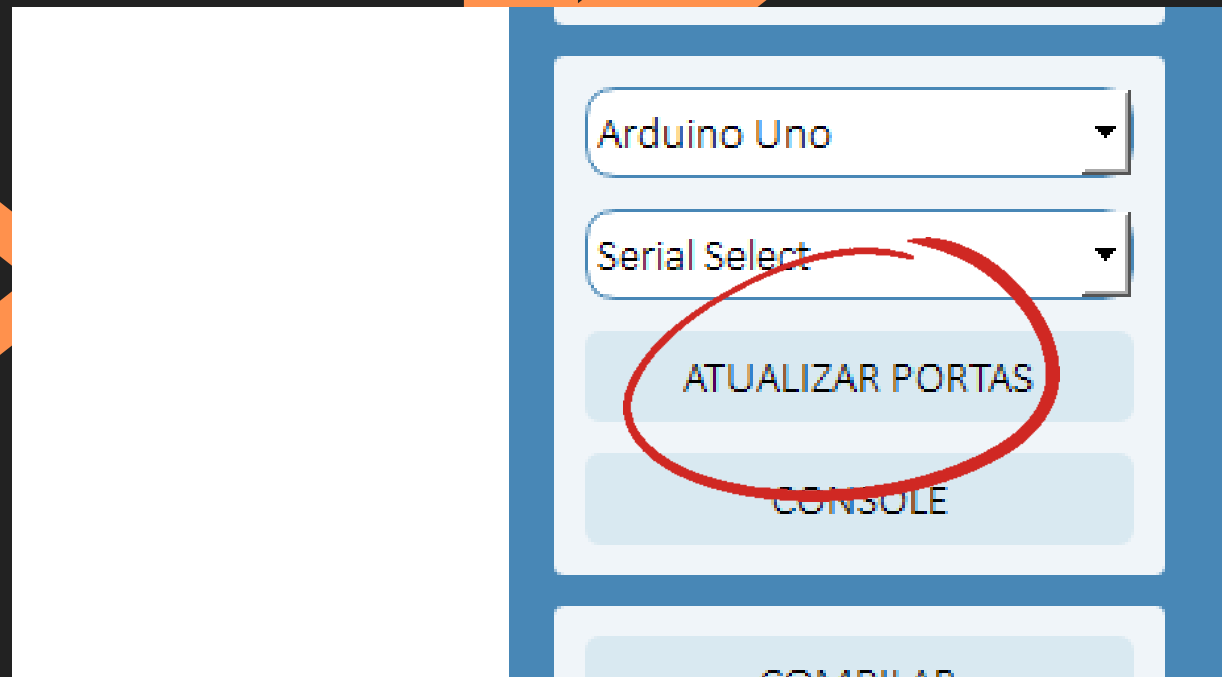
- GND - G4
- I/O - S4
- VCC - V4

PIR:

- VCC - V2
- GND - G2
- OUT- S2

Pinos podem variar dependendo do sensor utilizado, sempre verificar as conexões antes

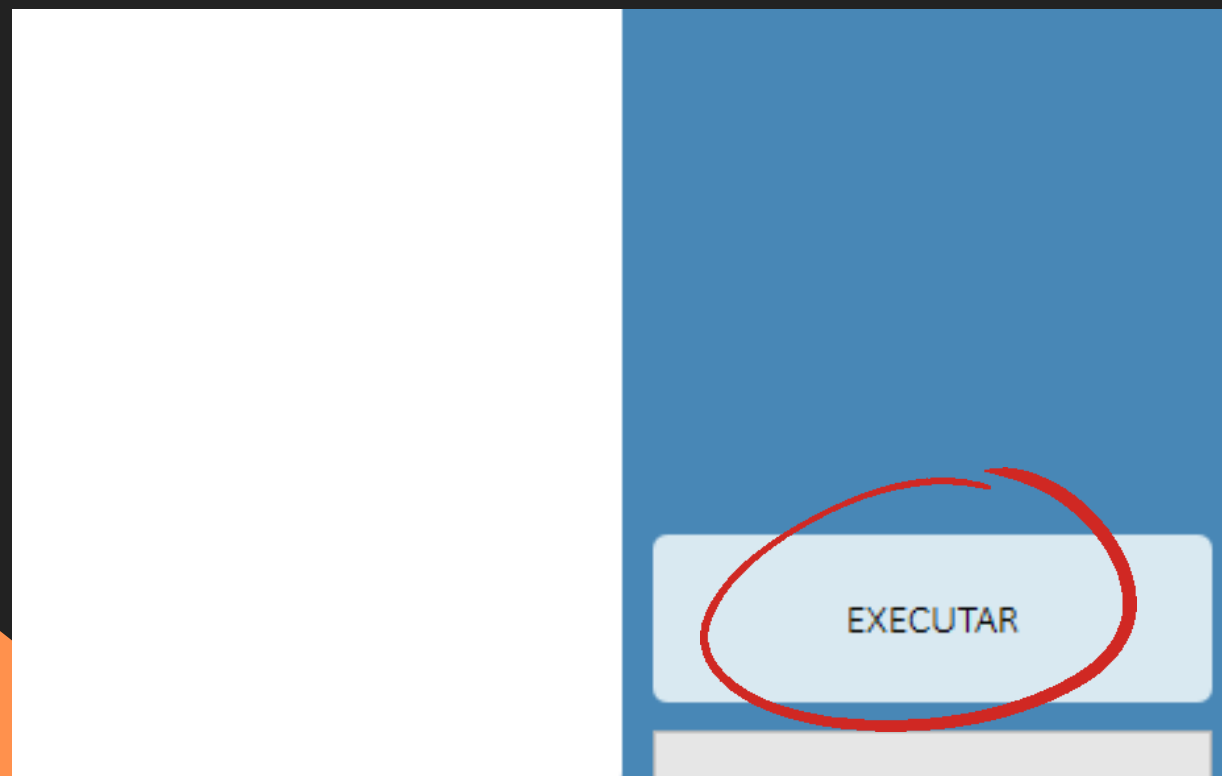
1



Conectando o microcontrolador e baixando a programação

Nossa programação está pronta. Conecte o microcontrolador no computador e clique em atualizar portas. Em seguida, compile e execute o código.

2

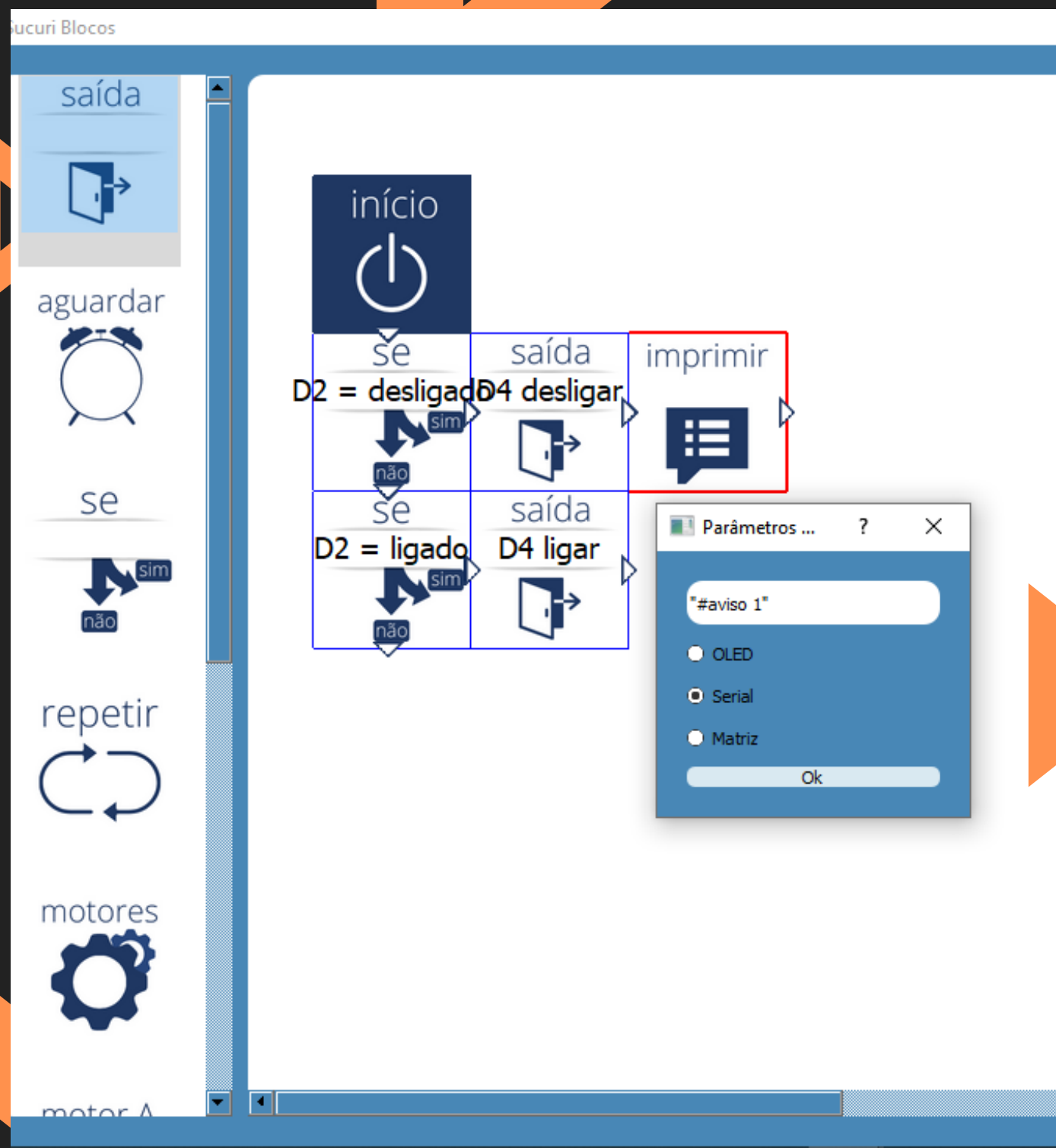


4. Utilizando as funções de IoT

Para dar início nos conceitos de IoT, vamos fazer um exemplo de alarme Wi-Fi.

Volte para o Sucuri Coding para inciar a programação.

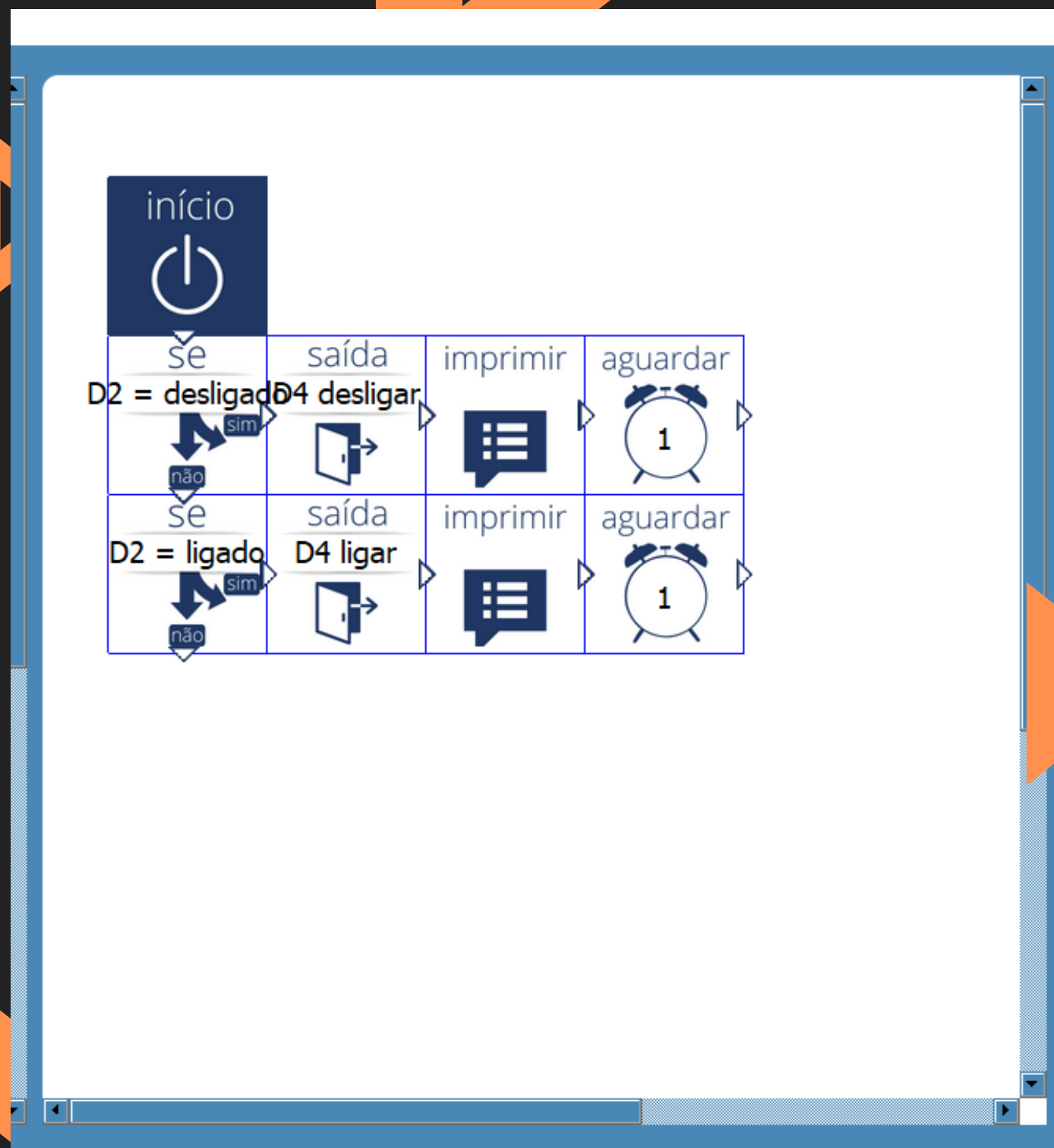




Comunicação com o Broker

Adicione o bloco imprimir após a ativação da saída, clique nele 2x. Nos seus parâmetros digite "#aviso 1" e marque a serial.

Repita o processo na linha de desligar porém com o comando imprimindo "#aviso 2".



Controlando a velocidade do envio

Para não sobrecarregar o boker com muitos avisos devemos adicionar comandos de AGUARDAR depois dos blocos de IMPRIMIR.

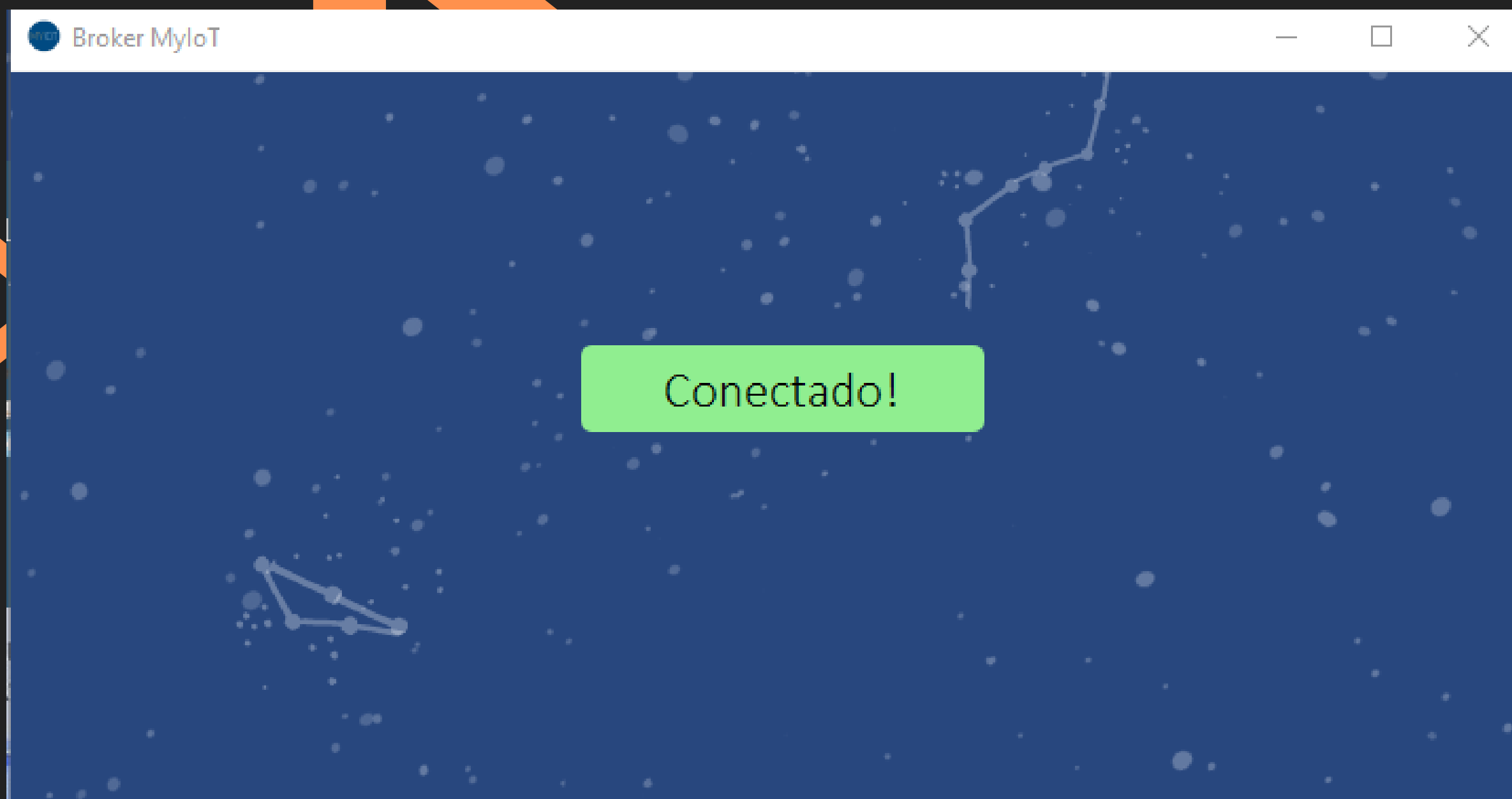
1 segundo já é suficiente para um funcionamento adequado.

Compile e execute o código.



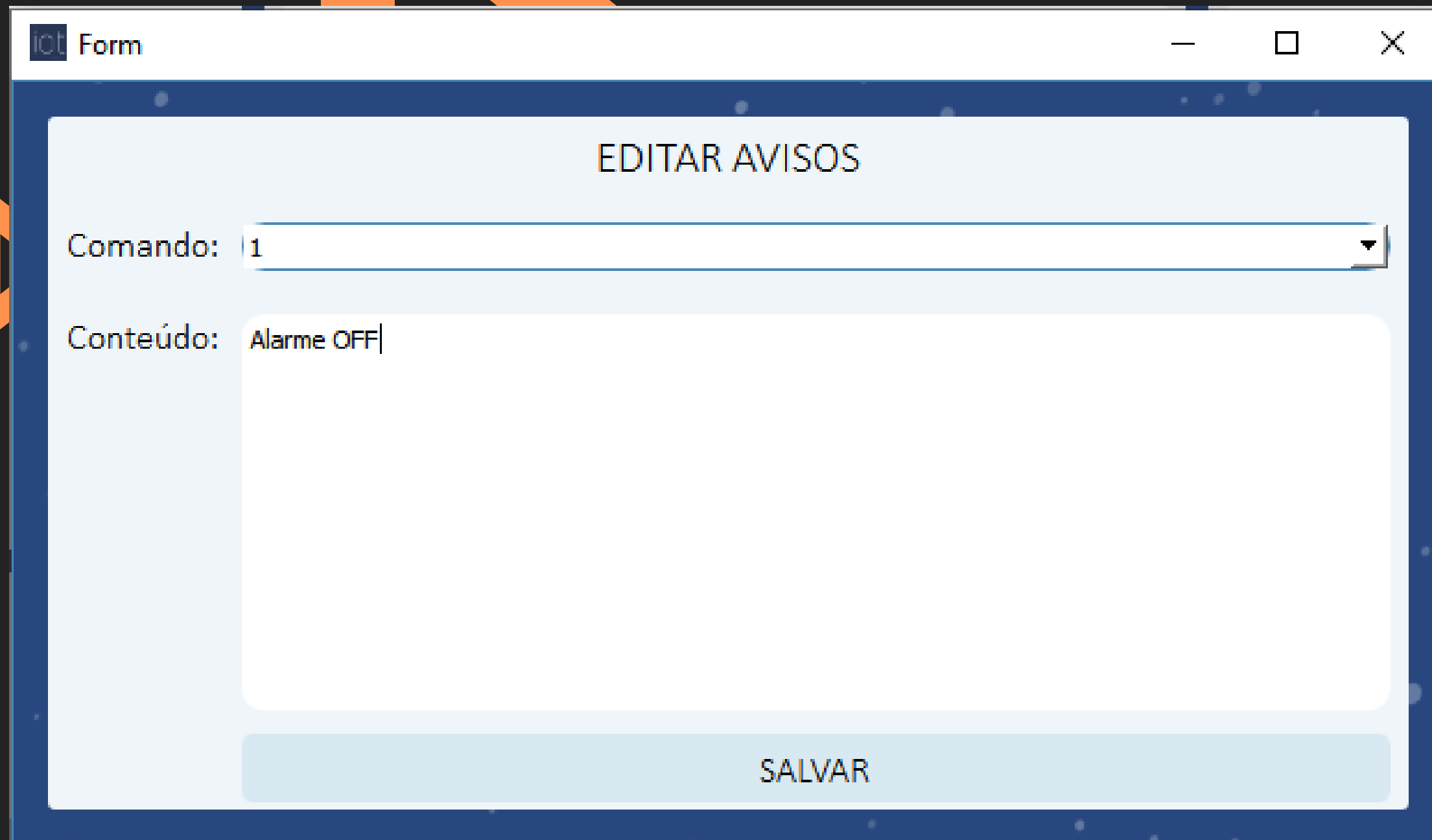
Conectando no BROKER

Abra o aplicativo Broker e conecte-o.



Conectando no Broker

Em seguida abra o volte novamente ao menu MYIOT para abrir o DASHBOARD.



Form

EDITAR AVISOS

Comando: 1

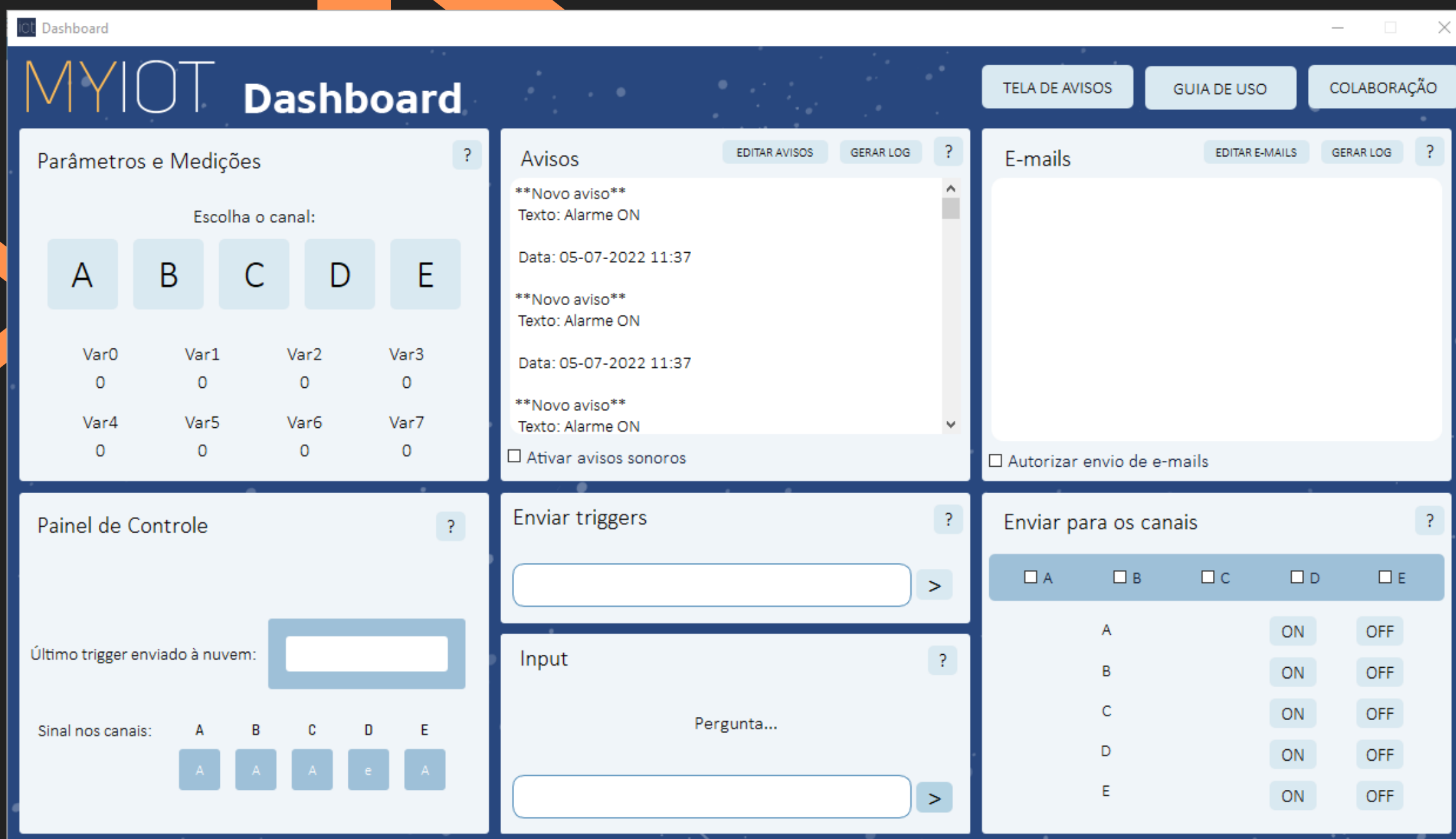
Conteúdo: Alarme OFF

SALVAR

Editando os avisos

Ao abrir o DASHBOARD, clique em EDITAR AVISOS. Selecione o aviso 1 e digite a mensagem que será mostrada na tela quando o alarme estiver desarmado.

Faça o mesmo com o aviso 2, quando o alarme for acionado.



Acompanhando os avisos

Tente se manter imóvel e em seguida se movimentar próximo ao sensor epetitivamente, para assim acompanhar o funcionamento do alarme.

Quando movimento for detectado, o buzzer deve soar e o aviso será enviado ao DASHBOARD