

FIAP – FACULDADE DE INFORMÁTICA E ADMINISTRAÇÃO PAULISTA

Alberto R. Peruchi – RM99706 - 2TDSR
Angelo Augusto Pelluci - RM98310 - 2TDSR
Gabrielle da Silva Stanguini RM98786 - 2TDSA
Maria Fernanda Ribeiro Melo RM98818 - 2TDSA
Nicolas Reis do Espírito Santo RM98819 - 2TDSA

GLOBAL SOLUTION - DISRUPTIVE
ARCHITECTURES: IOT, IOB & GENERATIVE IA
Documento da Global Solution 2ºSemestre

São Paulo
2024

SUMÁRIO

1. Descrição da ideia.....	03
2. Público Alvo.....	03
3. Benefícios da solução.....	04
4. Explicação com Imagens.....	05
5. Link GitHub.....	09
6. Link do projeto no Wokwi.....	09
7. Link do vídeo explicação.....	09

1- Descrição da Ideia:

A **SunWise** é uma solução inovadora desenvolvida para otimizar o controle de irrigação e iluminação em estufas agrícolas, promovendo eficiência e sustentabilidade. Através de sensores de umidade e controle automatizado de iluminação, a plataforma monitora as condições de umidade do solo e o horário do dia, ajustando a irrigação e as luzes de forma inteligente. Com interface web, os usuários têm acesso fácil e preciso às condições das suas estufas, garantindo a melhor utilização dos recursos disponíveis.

A solução foi projetada para atender a produtores agrícolas, gestores de estufas e agricultores que buscam reduzir o consumo de energia, otimizar os processos de irrigação e melhorar o controle das condições ambientais dentro das estufas. Pode ser utilizada em estufas de diversos tamanhos, tanto por pequenos produtores quanto por grandes empresas agrícolas.

2- Público Alvo:

O público alvo da solução é voltado para produtores agrícolas, especialmente pequenos e médios produtores que cultivam hortaliças, flores e outros vegetais em estufas, permitindo que tenham controle automatizado e eficiente sobre a irrigação e iluminação de seus cultivos. Gestores de estufas comerciais também se beneficiam, pois, o sistema facilita o monitoramento e a manutenção dos recursos hídricos e energéticos, simplificando o gerenciamento das operações diárias. A solução é ideal para empresas agrícolas sustentáveis que buscam integrar práticas mais ecológicas e econômicas em seus processos de produção, alinhando-se aos princípios da agricultura sustentável. Além disso, a solução atende agricultores urbanos, que cultivam produtos em estufas em centros urbanos e estão em busca de sistemas que otimizem a irrigação e o controle de iluminação, contribuindo para a sustentabilidade e o uso eficiente de recursos.

3- Benefícios da Solução:

A **SunWise** oferece uma série de benefícios para os consumidores, principalmente ao reduzir custos com operação. O sistema automatiza a irrigação, ativando-a apenas quando necessário com base na umidade do solo, evitando o desperdício de água e também de energia que é usada para fazer a bomba de água funcionar durante a irrigação. Além disso, o controle de iluminação ajusta o uso das luzes dependendo da hora do dia, garantindo que a energia elétrica seja utilizada de forma eficiente, evitando o uso desnecessário.

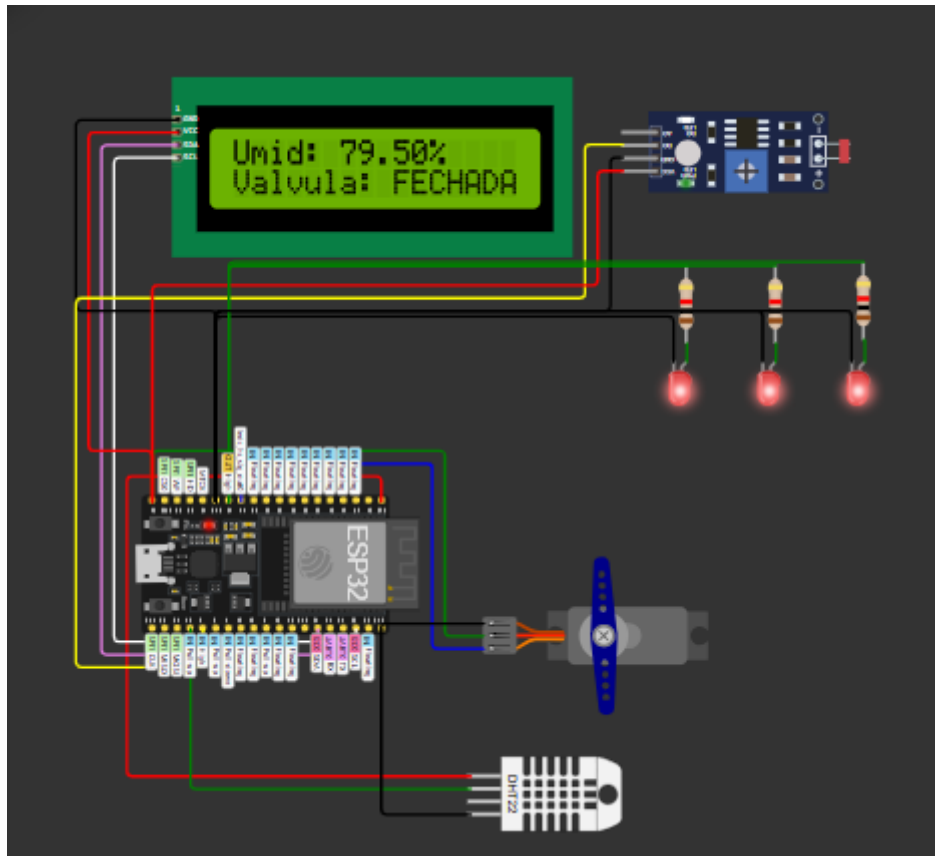
Outro benefício importante é a maior eficiência no uso de recursos. O sistema ajusta automaticamente a irrigação com base na umidade do solo, garantindo que as plantas recebam a quantidade ideal de água, o que promove um melhor crescimento e evita o uso excessivo de recursos. Com o monitoramento em tempo real, o usuário pode visualizar a umidade do solo, o status da irrigação e da iluminação, o que torna a gestão da estufa mais prática e eficiente. A solução oferece um dashboard fácil de usar, com gráficos de históricos de umidade, proporcionando aos usuários uma visão detalhada e atualizada de suas operações.

Além disso a solução possui um impacto significativo no meio ambiente, começando pela economia de água, ao controlar de forma inteligente a irrigação, evitando desperdícios. Também contribui para a redução das emissões de carbono por conta do controle da iluminação inteligente, que ajusta as luzes conforme a necessidade de luminosidade externa, reduzindo o consumo de energia elétrica e contribuindo para a diminuição das emissões de carbono associadas a produção de energia.

4- Explicação com imagens:

Para fazer esse projeto, nós utilizamos o simulador de circuito Wokwi.

Aqui está uma imagens do circuito montado:



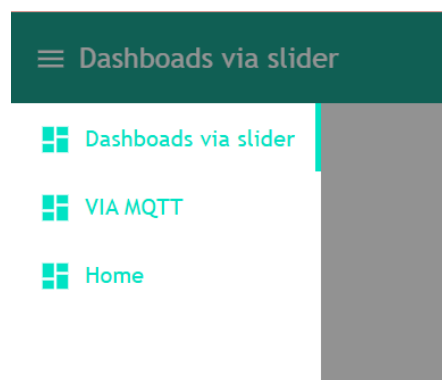
No código do circuito, nós colocamos o Servo Motor (a válvula), para representar o sistema de irrigação e a passagem ou não da água na irrigação. Para fazer a condição da válvula abrir ou fechar, nós colocamos esse sensor DHT22 que simula um valor de umidade e temperatura. A partir dos valores inseridos nele, nós criamos uma condição para que quando a umidade atinja um valor determinado de 50 ou mais, a válvula fecha e mostra o status da umidade e da válvula no monitor, representando a pausa na irrigação e na passagem de água. Caso a umidade atinja um valor inferior a 50, a válvula se abre e o status de irrigação muda no monitor, representando a passagem de água ativa na irrigação. Também colocamos um sensor de luminosidade, para caso o sensor detecte a presença de LUZ no ambiente, representando a luz solar durante o dia, os leds se apagam. Já, caso o sensor de luminosidade não detecte a presença de luz (solar), ele ativa os leds, que ficam ligados até as condições de luminosidade mudarem de estado.

```
{"temperatura":24,"umidade":79.5,"luminosidade":0}  
{"temperatura":24,"umidade":79.5,"luminosidade":4095}  
{"temperatura":24,"umidade":79.5,"luminosidade":4095}  
{"temperatura":24,"umidade":79.5,"luminosidade":4095}  
{"temperatura":24,"umidade":79.5,"luminosidade":4095}  
{"temperatura":24,"umidade":79.5,"luminosidade":4095}  
{"temperatura":24,"umidade":79.5,"luminosidade":4095}
```

No Monitor Serial, printamos um json com os números em tempo real, que se ajusta a cada mudança nos sensores, da temperatura, umidade e luminosidade (sobre esses valores, o número 4095 indica a ausência de luz, então os leds devem aparecer ligados. Caso o número esteja como 0 na luminosidade, isso representa a presença de luz, logo os leds devem aparecer desligados no circuito).

Buscamos informar esse json no monitor serial, pois ficaria mais fácil conectar com a parte do MQTT e posteriormente no NODE-RED. Infelizmente nós tivemos um problema significativo que comprometeu a conexão e envio de informações via MQTT. O servidor do Wokwi durante as semanas de desenvolvimento do projeto, estava com instabilidade e não conseguimos rodar o projeto com MTQQ, o que não fazia sentido pois o projeto sem a parte da conexão rodou, mesmo a compilação tendo demorado dias e erros na compilação de biblioteca aparecendo. Não conseguimos rodar para que os envios de informação via MQTT aparecem no Node-RED.

Mesmo esses erros acontecendo e os números não sendo enviados via MQTT do circuito no Wokwi, fizemos questão de montar dois fluxos no Node-RED com lógicas e dashboard para visualização dos dados. Criamos 3 páginas de demonstração:

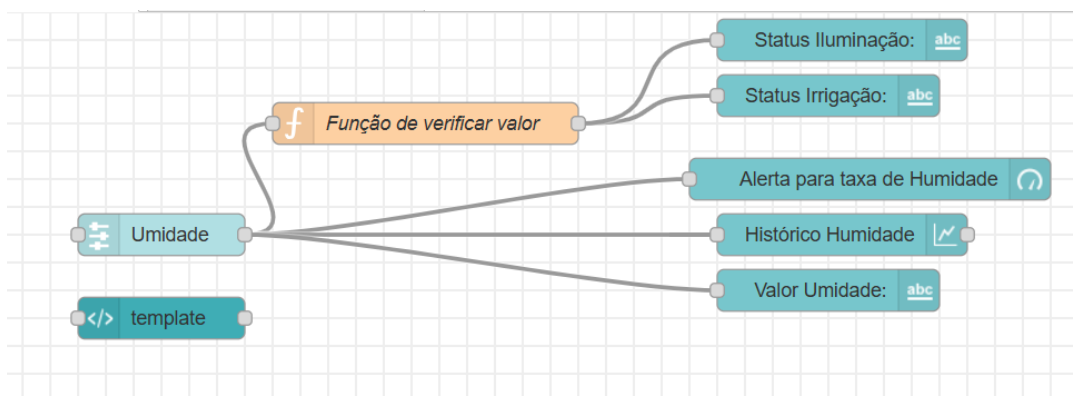


Na HOME está mostrando um pouco sobre o projeto de forma simples:

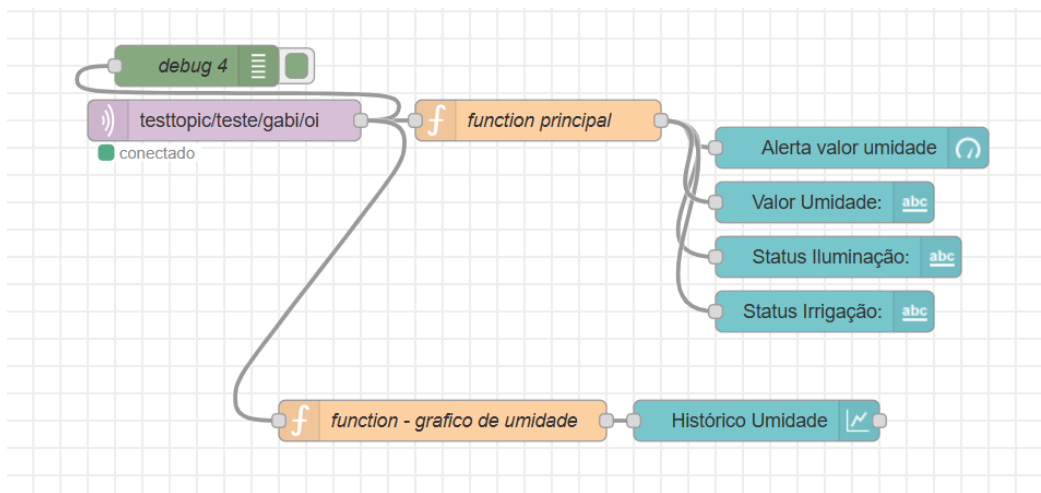


Fizemos 2 flows de demonstração:

1 flow: Simulamos nossa entrada de dados (teoricamente vindos do circuito e dos sensores) no node-red via SLIDDER. Com essas informações vindas do slider, que demonstravam o valor da umidade, fizemos uma função para mostrar os status de irrigação e iluminação. Para o status de irrigação, fizemos uma condição se o valor da irrigação for menor que 50, a mensagem de “Irrigação baixa, válvula aberta” é exibida. Se o valor for maior que 50, a mensagem de “Irrigação correta, válvula fechada” é exibida. Para o status da iluminação, pegamos o horário da entrada de dados e fizemos um condição onde se o horário for entre 05h e 17h30, a mensagem “Luzes apagadas” é exibida. Se o horário for entre 17h31 e 4h59, a mensagem “Luzes acesas” é mostrada. Para essa exibição colocamos dois textos conectados ao function node. Também colocamos um gauge para mostrar o valor da umidade e servir de alerta para o usuário. Como funcionalidade final colocamos um chart que mostra o histórico de umidade.



2 flow: Mandamos via MQTT o json igual a aplicação no wokwi mandaria. Com esse json vindo do mqtt, fizemos uma função que separa esses valores vindos e mostra os status de irrigação e iluminação. Para o status de irrigação, fizemos uma condição se o valor da irrigação for menor que 50, a mensagem de “Irrigação baixa, válvula aberta” é exibida. Se o valor for maior que 50, a mensagem de “Irrigação correta, válvula fechada” é exibida. Para o status da iluminação, pegamos o valor vindo do mqtt. Se o valor for 0 a mensagem “Luzes apagadas” é exibida. Se o valor for 4095 a mensagem “Luzes acesas” é mostrada. Para essa exibição colocamos dois textos conectados ao function node. Também colocamos um gauge para mostrar o valor da umidade e servir de alerta para o usuário. Como funcionalidade final colocamos um chart que mostra o histórico de umidade.



Json enviado via MQTT:

Connection
disconnected

Host

mqtt-dashboard.com

Port

8884

ClientID

clientId-NZ2qmTgPwj

Connect

Username

Password

Keep Alive

60

SSL

×

Clean Session

×

Last-Will Topic

Last-Will QoS

0

Last-Will Retain

Last-Will Message

Publish

Topic

testtopic/teste/gabi/oi

QoS

0

Retain

Publish

Message

[{"temperatura":24,"umidade":90.0,"luminosidade":0}]

Subscriptions

5- Link GitHub:

O código fonte exportado do Wokwi, juntamente com o link do simulador Wokwi localizado no ReadMe, e o arquivo do Flows do Node-Red, se encontram no repositório do GitHub.

<https://github.com/AngPF/GS-SunWise-IOT.git>

6- Link do projeto no Wokwi:

Link do código e do circuito funcional no Wokwi:

<https://wokwi.com/projects/414393427034769409>

Link do projeto com MQTT que não está rodando pelos problemas no WOKWI.
Enviando somente para validação:

<https://wokwi.com/projects/414636232313826305>

7- Link do vídeo explicação:

Vídeo da explicação do código para a Global Solution:

<https://youtu.be/r-TcEO-0Kc>