

SENSOR DE UMIDADE DO SOLO PARA CIDADES INTELIGENTES

Bruno Arial Ramos – 251237
Giovane Nunes Malagola – 251510
Matheus Bonatti – 251262

Hélio Filho

Pedro Rafael Matunaga– 250827
Victor Henrique Soares Baleeiro– 252366

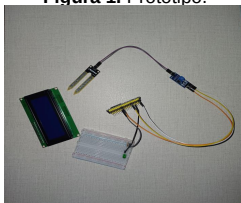
INTRODUÇÃO

O projeto tem como foco o desenvolvimento e uso de sensores de umidade do solo para aplicação em plantações verticais dentro de cidades inteligentes. Esses sensores permitem monitorar com precisão a umidade do solo, ajudando no controle eficiente do uso da água e na manutenção da saúde das plantas.

Em um cenário urbano, onde o espaço é limitado e os recursos naturais precisam ser utilizados de forma racional, o sensor de umidade torna-se uma ferramenta essencial. Ele possibilita o acompanhamento em tempo real das condições do solo, evitando desperdícios e promovendo práticas agrícolas mais sustentáveis.

Além disso, o projeto reforça a importância da agricultura urbana como alternativa para reduzir a dependência de alimentos vindos de áreas rurais, contribuindo para cidades mais resilientes, verdes e autossustentáveis. Ao integrar a tecnologia ao cultivo urbano, buscamos incentivar a inovação, a preservação ambiental e o uso consciente dos recursos naturais.

Figura 1. Protótipo.



Fonte:autor,2025.

JUSTIFICATIVA

Pensando nas cidades inteligentes, chegamos a conclusão que além de automatizar linhas de produções era necessário ter uma automatização na produção de alimentos dentro da cidade, tornando-a mais auto sustentável diminuindo a carga de produção no campo.

OBJETIVOS e ODS

Observar a variação da umidade do solo em ambientes urbanos. Informar automaticamente a necessidade de irrigação por meio de sensores. Conscientizar sobre o uso racional da água em plantações urbanas. Construir uma solução tecnológica simples e eficiente para aplicação em cidades inteligentes.

ODS trabalhadas no projeto: ODS 6 - Água potável e saneamento: promover o uso eficiente da água na agricultura urbana. ODS 11 - Cidades e comunidades sustentáveis: contribuir para a sustentabilidade urbana com tecnologias inteligentes. ODS 12 - Consumo e produção responsáveis: estimular o consumo consciente de recursos naturais.

ORÇAMENTO

Um total gasto de R\$141,28 (unidade), itens utilizados: ESP-32; Jumper FxM; Jumper FxM; display LCD 20x4; Potenciômetro 20komhs; LEDs e resistores(5V); um protoboard; o sensor de umidade do solo e um módulo serial I2C.

Tabela 1

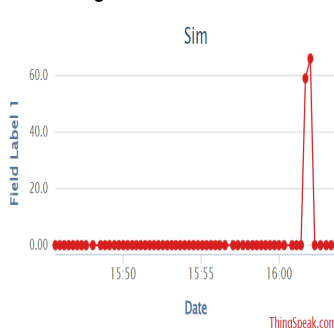
Produto:	Valor:
ESP-32	R\$ 62,40
Jumper FxM	R\$ 2,46
Jumper FxM	R\$ 5,92
Display LCD	R\$ 35,56
Potenciômetro	R\$ 2,00
LEDs+Resistores	R\$ 7,41
Protopboard	R\$ 5,78
Sensor de umidade	R\$ 14,00
Módulo serial I2C	R\$ 5,75
Total	R\$ 141,28

RESULTADOS E VALIDAÇÃO

O sensor de umidade foi testado com sucesso em diferentes condições de solo, coletando dados em tempo real sobre os níveis de umidade. Os valores captados foram enviados automaticamente para a nuvem por meio da plataforma ThingSpeak, onde foram registrados e exibidos em gráficos contínuos.

A análise dos gráficos permitiu observar com precisão as variações da umidade ao longo do tempo, especialmente em momentos de irrigação e secagem do solo. Essa coleta contínua validou a eficiência do sensor em identificar mudanças nas condições do solo.

Figura 2. Gráfico da nuvem.



Fonte: ThingSpeak,2025.

Tabela 2. Código.

```
1 #include <WiFi.h>
2 #include <Wire.h>
3 #include <ESP8266.h>
4 #include <ESP8266WiFi.h>
5 #include <ESP8266WebServer.h>
6 #include <ESP8266HttpClient.h>
7 #include <ESP8266HttpClient.h>
8 #include <ESP8266HttpClient.h>
9 #include <ESP8266HttpClient.h>
10 #include <ESP8266HttpClient.h>
11 #include <ESP8266HttpClient.h>
12 #include <ESP8266HttpClient.h>
13 #include <ESP8266HttpClient.h>
14 #include <ESP8266HttpClient.h>
15 #include <ESP8266HttpClient.h>
16 #include <ESP8266HttpClient.h>
17 #include <ESP8266HttpClient.h>
18 #include <ESP8266HttpClient.h>
19 #include <ESP8266HttpClient.h>
20 #include <ESP8266HttpClient.h>
21 #include <ESP8266HttpClient.h>
22 #include <ESP8266HttpClient.h>
23 #include <ESP8266HttpClient.h>
24 #include <ESP8266HttpClient.h>
25 #include <ESP8266HttpClient.h>
26 #include <ESP8266HttpClient.h>
27 #include <ESP8266HttpClient.h>
28 #include <ESP8266HttpClient.h>
29 #include <ESP8266HttpClient.h>
30 #include <ESP8266HttpClient.h>
31 #include <ESP8266HttpClient.h>
32 #include <ESP8266HttpClient.h>
33 #include <ESP8266HttpClient.h>
34 #include <ESP8266HttpClient.h>
35 #include <ESP8266HttpClient.h>
36 #include <ESP8266HttpClient.h>
37 #include <ESP8266HttpClient.h>
38 #include <ESP8266HttpClient.h>
39 #include <ESP8266HttpClient.h>
40 #include <ESP8266HttpClient.h>
41 #include <ESP8266HttpClient.h>
42 #include <ESP8266HttpClient.h>
43 #include <ESP8266HttpClient.h>
44 #include <ESP8266HttpClient.h>
45 #include <ESP8266HttpClient.h>
46 #include <ESP8266HttpClient.h>
47 #include <ESP8266HttpClient.h>
48 #include <ESP8266HttpClient.h>
49 #include <ESP8266HttpClient.h>
50 #include <ESP8266HttpClient.h>
51 #include <ESP8266HttpClient.h>
52 #include <ESP8266HttpClient.h>
53 #include <ESP8266HttpClient.h>
54 #include <ESP8266HttpClient.h>
55 #include <ESP8266HttpClient.h>
56 #include <ESP8266HttpClient.h>
57 #include <ESP8266HttpClient.h>
58 #include <ESP8266HttpClient.h>
59 #include <ESP8266HttpClient.h>
60 #include <ESP8266HttpClient.h>
61 #include <ESP8266HttpClient.h>
62 #include <ESP8266HttpClient.h>
63 #include <ESP8266HttpClient.h>
64 #include <ESP8266HttpClient.h>
65 #include <ESP8266HttpClient.h>
66 #include <ESP8266HttpClient.h>
67 #include <ESP8266HttpClient.h>
68 #include <ESP8266HttpClient.h>
69 #include <ESP8266HttpClient.h>
70 #include <ESP8266HttpClient.h>
71 #include <ESP8266HttpClient.h>
72 #include <ESP8266HttpClient.h>
73 #include <ESP8266HttpClient.h>
74 #include <ESP8266HttpClient.h>
75 #include <ESP8266HttpClient.h>
76 #include <ESP8266HttpClient.h>
77 #include <ESP8266HttpClient.h>
78 #include <ESP8266HttpClient.h>
79 #include <ESP8266HttpClient.h>
80 #include <ESP8266HttpClient.h>
81 #include <ESP8266HttpClient.h>
82 #include <ESP8266HttpClient.h>
83 #include <ESP8266HttpClient.h>
84 #include <ESP8266HttpClient.h>
85 #include <ESP8266HttpClient.h>
86 #include <ESP8266HttpClient.h>
87 #include <ESP8266HttpClient.h>
88 #include <ESP8266HttpClient.h>
89 #include <ESP8266HttpClient.h>
90 #include <ESP8266HttpClient.h>
91 #include <ESP8266HttpClient.h>
92 #include <ESP8266HttpClient.h>
93 #include <ESP8266HttpClient.h>
94 #include <ESP8266HttpClient.h>
95 #include <ESP8266HttpClient.h>
96 #include <ESP8266HttpClient.h>
97 #include <ESP8266HttpClient.h>
98 #include <ESP8266HttpClient.h>
99 #include <ESP8266HttpClient.h>
100 #include <ESP8266HttpClient.h>
```

Fonte: Autoral,2025.

CONCLUSÃO

O projeto apresentado demonstra a viabilidade e a importância da automação no cultivo urbano por meio de um sistema de irrigação inteligente baseado no ESP-32 e o sensor de umidade do solo. Utilizando componentes de baixo custo e fácil acesso, foi possível desenvolver um protótipo funcional capaz de monitorar diferentes níveis de umidade e pode vir a atuar automaticamente com base nas condições do solo. A simplicidade da implementação, aliada à possibilidade de escalabilidade, torna este sistema ideal tanto para pequenas plantações urbanas quanto para estruturas maiores, como fazendas verticais em edifícios.

PERSPECTIVAS (OPCIONAL)

Inclusão do display LCD para o monitoramento local, sem necessitar estar conectado em uma rede. Automatização do sistema de irrigação, otimização do código, utilizar um sistema de nuvem melhor, que possa apresentar mais dados.

AGRADECIMENTOS

Agradeço ao meu grupo, a WJ componentes eletrônicos e aos professores Thiago Prini e Hélio Filho que nos auxiliaram des do início do projeto.