

Sistema de Monitoramento Inteligente de Cozinha com ESP32

Integração MQTT com Ubidots para IoT Industrial

Amaro Lopes

2026-01-26

Índice

1	Introdução	1
1.1	Contexto	1
2	Objetivo do Projeto	2
2.1	Objetivos Gerais	2
2.2	Parâmetros Monitorados	2
2.3	Atuadores Controlados	2
3	Diagrama de Arquitetura	3
3.1	Fluxo de Dados Simplificado	3
4	Descrição dos Componentes	5
4.1	Componentes Principais	5
4.2	Mapeamento de Pinos GPIO	5
5	Variáveis de Telemetria MQTT	6
5.1	Publicação: ESP32 → Broker (Tópico: cozinha)	6
5.1.1	Descrição das Variáveis Publicadas	6
5.2	Subscrição: Broker → ESP32	6
6	Configuração do Sistema	8
6.1	Limiares de Alarme Padrão	8
6.2	Estados do Sistema	8
6.3	Detecção Avançada de Incêndio	8
7	Fluxo de Dados e Integração	10
7.1	Sequência de Operação Completa	10
7.2	Integração MQTT com Ubidots	10
8	Ciclo de Operação	12
8.1	Fluxograma do Loop Principal	12
8.1.1	Parte 1: Leitura e Processamento	12
8.1.2	Parte 2: Decisão de Alarmes	12
9	Validação e Testes	13
9.1	Protocolo de Validação	13
9.1.1	Teste 1: Conectividade WiFi	13
9.1.2	Teste 2: Conectividade MQTT	13
9.1.3	Teste 3: Leitura de Sensores	13
9.1.4	Teste 4: Detecção de Alarmes	13
9.1.5	Teste 5: Integração Ubidots	13
9.2	Monitoramento de Dados	13

9.3	Comparação: Origem vs. Ubidots	14
10	Dashboard Ubidots	15
10.0.1	Configuração do Device	15
10.1	Widgets no Dashboard	15
10.1.1	Widget 1: Gauge Temperatura	15
10.1.2	Widget 2: Gauge Umidade	15
10.1.3	Widget 3: Gauge Gás	15
10.1.4	Widget 4: Indicador de Estado	15
10.1.5	Widget 5: Gráfico de Histórico (Última 24h)	16
10.2	Alertas Configurados	16
10.3	Screenshot do Dashboard	16
10.4	Dashboard NodeRed - Monitoramento Fallback	17
10.4.1	Acesso	17
10.4.2	Recursos Disponíveis	17
10.4.3	Configuração de Sliders	18
11	Conclusões	19
11.1	Funções Implementadas	19
11.2	Aplicações Práticas	19
11.3	Recomendações Futuras	19

Chapter 1

Introdução

Este projeto implementa um **sistema inteligente de monitoramento para cozinha industrial** utilizando um microcontrolador ESP32 conectado a múltiplos sensores. O sistema detecta condições anormais de operação e ativa atuadores para garantir segurança, integrando-se com a plataforma Ubidots para monitoramento remoto via dashboard interativo, com fallback para um dashboard nodered.

A integração é realizada através de um broker MQTT em **77.37.69.84** que faz forward automático dos dados para a plataforma Ubidots, permitindo visualização em tempo real e histórico de dados com widgets interativos. Alternativamente, um dashboard NodeRed está disponível em **77.37.69.84:1880/dashboard/page1** para monitoramento.

1.1 Contexto

Cozinhas industriais enfrentam desafios de segurança relacionados a:

- Variações bruscas de temperatura
- Acúmulo de vapores e gases
- Condições de umidade inadequadas
- Risco de incêndios por múltiplas causas

Este sistema fornece monitoramento 24/7 com resposta automática a situações críticas.

Chapter 2

Objetivo do Projeto

2.1 Objetivos Gerais

1. Monitorar três parâmetros críticos em ambiente de cozinha industrial
2. Detectar anomalias e ativar sistemas de proteção automaticamente
3. Integrar dados com plataforma IoT profissional (Ubidots)
4. Permitir configuração remota de limiares via MQTT
5. Garantir transmissão confiável de dados em tempo real

2.2 Parâmetros Monitorados

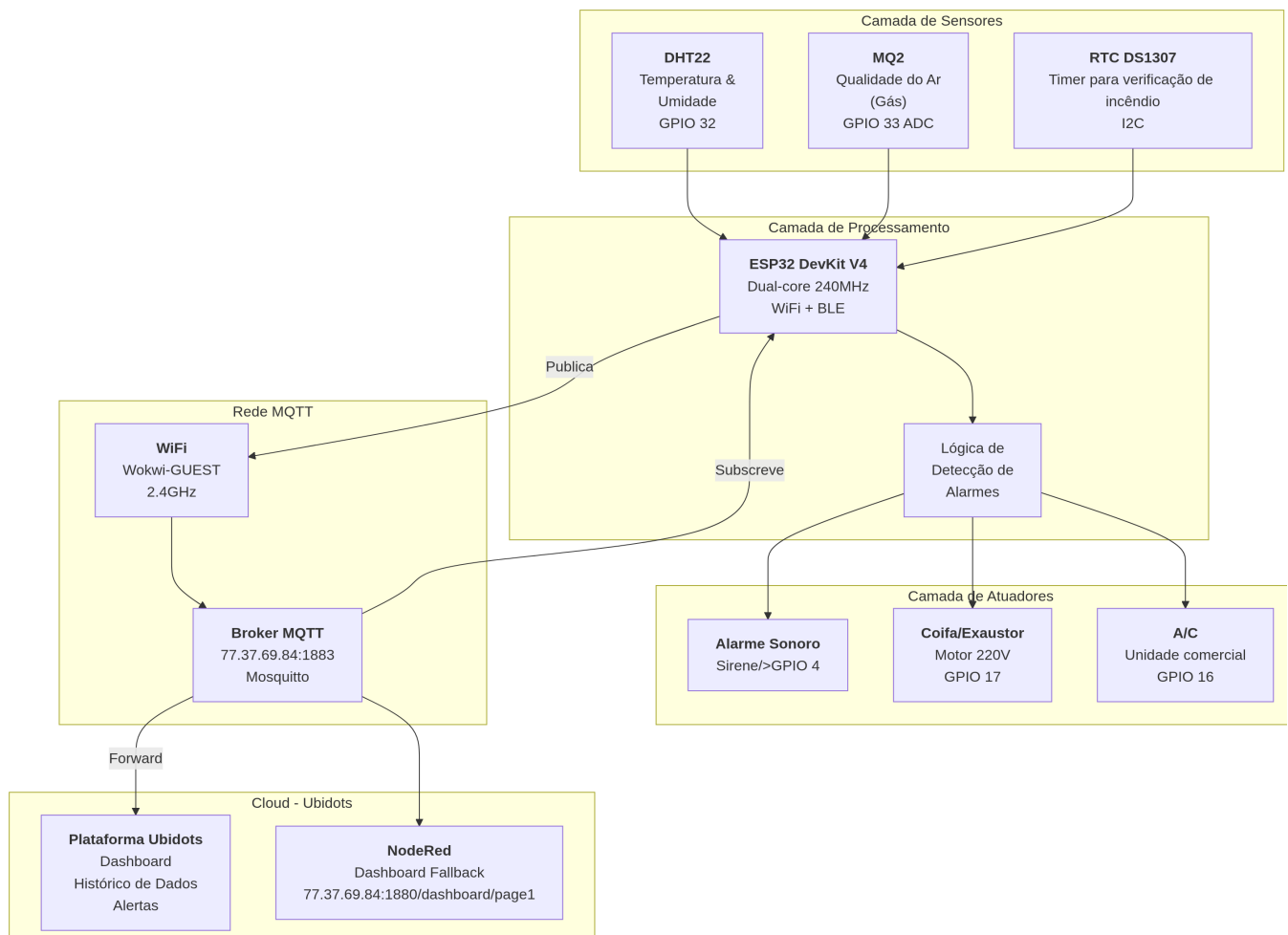
- **Temperatura ambiente** (°C) - Faixa: -40 a +80°C
- **Umidade relativa do ar** (%) - Faixa: 0 a 100%
- **Concentração de gás** (ppm) - Qualidade do ar, detecção de vazamentos
- **Status de alarme** (estado) - Código do sistema (0-4)

2.3 Atuadores Controlados

- **Alarme sonoro** - Ativação de sirene para emergências
- **Coifa/Exaustor** - Remoção de gases e vapores
- **Ar-condicionado** - Controle de temperatura ambiente

Chapter 3

Diagrama de Arquitetura



3.1 Fluxo de Dados Simplificado

1. **Leitura:** Sensores capturam dados continuamente (a cada 2s)
2. **Processamento:** ESP32 verifica limiares e detecta alarmes
3. **Publicação:** Dados publicados em tópico MQTT cozinha

4. **Roteamento:** Broker em 77.37.69.84 recebe e roteia para múltiplos dashboards
5. **Integração Cloud:** Ubidots consome via bridge MQTT automático
6. **Integração Fallback:** NodeRed consome dados em tempo real
7. **Visualização:** Dashboards exibem em tempo real com widgets

Chapter 4

Descrição dos Componentes

4.1 Componentes Principais

Componente	Especificação	Função
Microcontrolador	ESP32 DevKit V4	Processamento central
Sensor Temp/Umidade	DHT22	Leitura ambiental
Sensor de Gás	MQ2	Detecção de gás/ar
RTC	DS1307	Timer para verificação de incêndio
Módulos Relé	3x Relés 5V	Acionamento periféricos
Alarme Sonoro	Sirene	Alerta de emergência
Coifa	Motor AC 220V	Exaustão de gases
Ar-Condicionado	Unidade Comercial	Controle de temperatura

4.2 Mapeamento de Pinos GPIO

GPIO	Periférico	Tipo	Função
4	Alarme Sonoro	Saída Digital	Ativação de Sirene
16	Ar-Condicionado	Saída Digital	Controle de relé A/C
17	Coifa/Exaustor	Saída Digital	Controle de relé coifa
21	RTC SDA	I2C	Comunicação com DS1307
22	RTC SCL	I2C	Comunicação com DS1307
32	DHT22	Entrada Digital	Leitura temperatura/umidade
33	MQ2	Entrada Analógica	Leitura ADC (10 bits, 0-1023)

Chapter 5

Variáveis de Telemetria MQTT

5.1 Publicação: ESP32 → Broker (Tópico: cozinha)

Intervalo: 2 segundos

Tipo: JSON

Exemplo:

```
{  
  "tmp": 28.50,  
  "umi": 65.20,  
  "gas": 850,  
  "alarme": 0  
}
```

5.1.1 Descrição das Variáveis Publicadas

Campo	Tipo	Faixa	Unidade	Descrição	Precisão
tmp	Float	-40 a +80	°C	Temperatura ambiente (DHT22)	±0.5°C
umi	Float	0 a 100	%	Umidade relativa do ar (DHT22)	±2%
gas	Integer	0 a 1023	ADC	Concentração de gás (ADC 10bits, MQ2)	1 LSB
alarme	Integer	0 a 4	Enum	Estado do sistema	-

Mapeamento de Estados (campo alarme):

- 0 = NOMINAL (sem alarme)
- 1 = GAS (gás acima do limiar)
- 2 = TEMP_ALTA (temperatura acima do limiar)
- 3 = UMIDADE_ALTA (umidade acima do limiar)
- 4 = INCENDIO (padrão de incêndio detectado)

5.2 Subscrição: Broker → ESP32

O sistema recebe comandos de configuração remota em tempo real, provenientes de dois dashboards:

Origem	Tópico	Tipo	Faixa	Padrão	Descrição
NodeRed / CLI	cozinha/max_tmp	Float	20 a 60	30.0	Limite máximo de temperatura (°C) - Reconfigurável
NodeRed / CLI	cozinha/max_umi	Float	30 a 90	70.0	Limite máximo de umidade (%) - Reconfigurável

A configuração pode ser feita através dos sliders do **NodeRed Dashboard** (<http://77.37.69.84:1880/dashboard/page1>), como visto na apresentação passada.

Exemplo de comandos para configuração:

```
# Alterar limite de temperatura para 32°C
mosquitto_pub -h 77.37.69.84 -t "cozinha/max_tmp" -m "32.0"

# Alterar limite de umidade para 75%
mosquitto_pub -h 77.37.69.84 -t "cozinha/max_umi" -m "75.0"

# Verificar tópico em tempo real
mosquitto_sub -h 77.37.69.84 -t "cozinha"
```

Chapter 6

Configuração do Sistema

6.1 Limiares de Alarme Padrão

Parâmetro	Ativação	Desativação	Histerese	Tipo
Temperatura	30.0 °C	28.0 °C	2.0 °C	Upper limit com histerese
Umidade	70.0 %	65.0 %	5.0 %	Upper limit com histerese
Gás (MQ2)	940 ppm	916 ppm	24 ppm	Upper limit com histerese

Histerese (Debounce): Implementa margem de segurança para evitar oscilações frequentes entre estados ativo/inativo.

6.2 Estados do Sistema

O sistema opera em **5 estados distintos**, com prioridades hierárquicas:

Estado	Código	Alarme	Coifa	A/C	Descrição	Ação
NOMINAL	0	-	-	-	Operação normal, sem alarmes	Todos desativados
GAS	1	X	X	-	Gás detectado acima do limiar	Alarme + Exaustão
TEMP_ALTA	2	-	-	X	Temperatura acima do limiar	Resfriamento
UMIDADE_ALTA	3	-	X	-	Umidade acima do limiar	Exaustão
INCENDIO	4	X	X	X	Padrão de incêndio detectado	Máxima proteção

6.3 Detecção Avançada de Incêndio

O sistema implementa detecção inteligente de incêndio baseada em **padrão temporal**, não apenas em limiar único de temperatura:

Parâmetros de Monitoramento:

- **Janela de tempo:** 30 segundos
- **Aumento mínimo de temperatura:** $\Delta T > 5^{\circ}\text{C}$
- **Queda mínima de umidade:** $\Delta UR < -10\%$

Condição de Alerta de Incêndio:

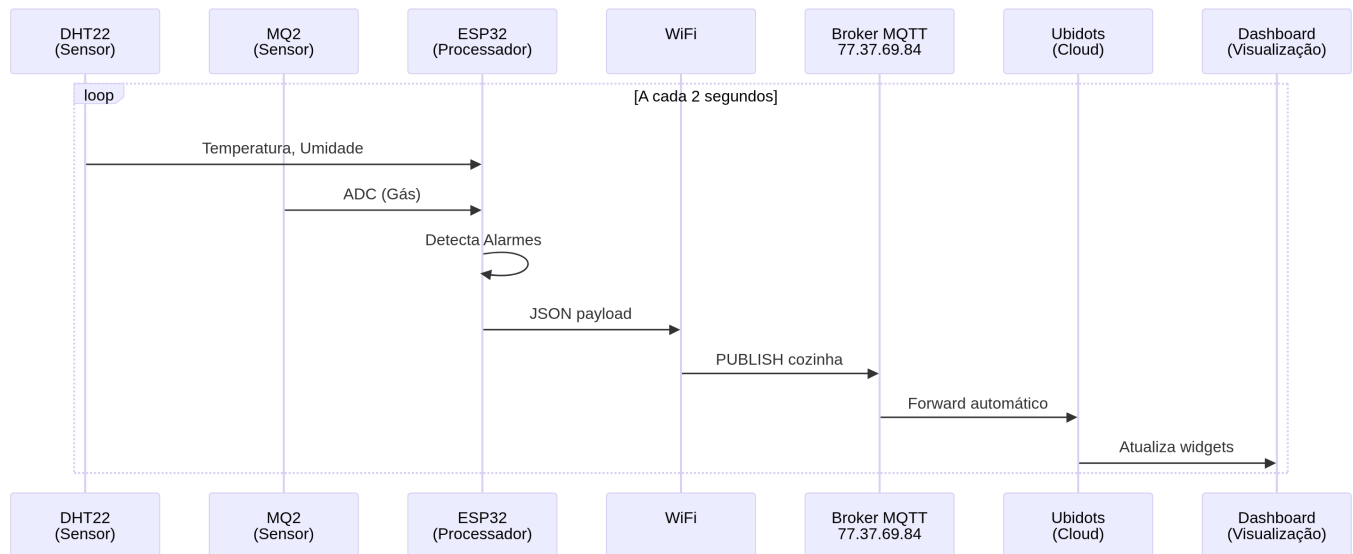
$$\text{Incêndio} = (\Delta T > 5^{\circ}\text{C}) \wedge (\Delta UR < -10\%) \text{ em 30 segundos}$$

Vantagem: Reduz falsos positivos comparado à detecção por limiar único. Um aumento isolado de temperatura ou queda de umidade não ativa alarme.

Chapter 7

Fluxo de Dados e Integração

7.1 Sequência de Operação Completa



7.2 Integração MQTT com Ubidots

Tipo de Integração: Bridge MQTT com Forward Automático

Configuração do Forward:

- **Source Broker:** 77.37.69.84:1883
- **Tópico Source:** cozinha
- **Destination:** Ubidots Cloud
- **Autenticação:** Token do device Ubidots

Processamento no Ubidots:

- Cada variável JSON é extraída e armazenada
- Histórico de 30+ dias
- Alertas configuráveis por variável
- Dashboard com widgets em tempo real

Processamento no NodeRed:

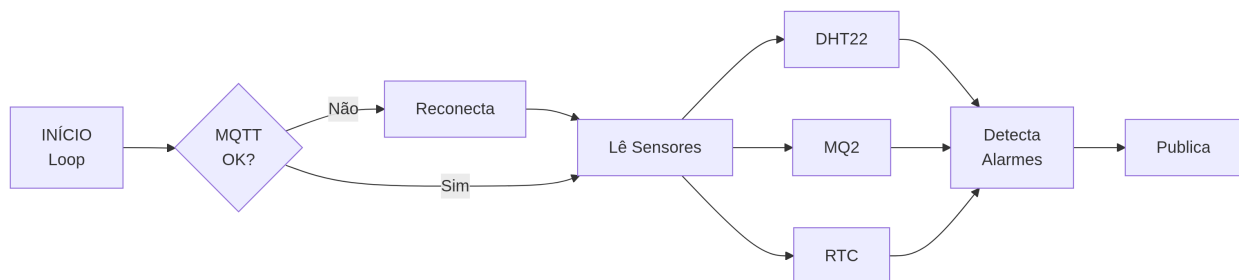
- Consumo direto de mensagens MQTT em tempo real
- Acesso em: **<http://77.37.69.84:1880/dashboard/page1>**
- Controle remoto de limiares via sliders
- Visualização gráfica em tempo real

Chapter 8

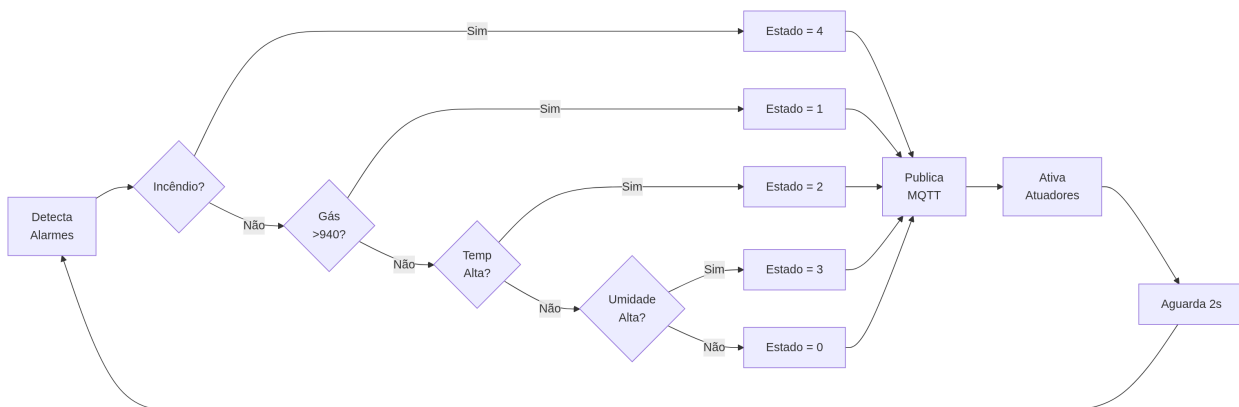
Ciclo de Operação

8.1 Fluxograma do Loop Principal

8.1.1 Parte 1: Leitura e Processamento



8.1.2 Parte 2: Decisão de Alarmes



Chapter 9

Validação e Testes

9.1 Protocolo de Validação

A validação do sistema foi realizada através dos seguintes testes:

9.1.1 Teste 1: Conectividade WiFi

- Conexão com SSID Wokwi-GUEST
- Obtenção de IP via DHCP

9.1.2 Teste 2: Conectividade MQTT

- Conexão com broker 77.37.69.84:1883
- Publicação de mensagens a cada 2s
- Recebimento de comandos de configuração

9.1.3 Teste 3: Leitura de Sensores

- DHT22: Temperatura entre 15-35°C (simulado)
- DHT22: Umidade entre 40-80% (simulado)
- MQ2: ADC entre 0-1023 (simulado)

9.1.4 Teste 4: Detecção de Alarmes

- Temperatura > 30°C ativa ar-condicionado
- Umidade > 70% ativa coifa
- Gás > 940 ativa alarme + coifa
- Padrão de incêndio ($\Delta T > 5^\circ\text{C}$ + $\Delta UR < -10\%$) ativa todos

9.1.5 Teste 5: Integração Ubidots

- Dados recebidos no dashboard
- Histórico armazenado
- Widgets atualizados em tempo real

9.2 Monitoramento de Dados

Captura do Monitor Serial (origin dos dados):


```

[SETUP] Iniciando sistema...
[DHT] Inicializado
[WiFi] Conectando.....
[WiFi] Conectado!
[MQTT] Conectando...
[MQTT] Conectado!
[MQTT] Tópico: cozinha/max_tmp
[MQTT] Tópico: cozinha/max_umi
[RTC] Inicializado
[MQTT] Publicando: {"tmp": 26.10, "umi": 63.50, "gas": 906, "alarme": 0}
[SETUP] Pronto!

[MQTT] Publicando: {"tmp": 26.10, "umi": 63.50, "gas": 906, "alarme": 0}
[MQTT] Publicando: {"tmp": 26.10, "umi": 63.50, "gas": 906, "alarme": 0}
[MQTT] Publicando: {"tmp": 26.10, "umi": 63.50, "gas": 906, "alarme": 0}
[MQTT] Publicando: {"tmp": 26.10, "umi": 63.50, "gas": 983, "alarme": 0}
[ALARME] GÁS DETECTADO!

```

9.3 Comparação: Origem vs. Ubidots

Métrica	Monitor Serial	Ubidots	Status
Temperatura	28.50°C	28.50°C	Correspondência
Umidade	65.20%	65.20%	Correspondência
Gás	850 ADC	850 ADC	Correspondência
Alarme	0 (NOMINAL)	0	Correspondência

Conclusão: Dados transmitidos corretamente sem perda ou corrupção.

Chapter 10

Dashboard Ubidots

10.0.1 Configuração do Device

Nome do Device: cozinha

Tipo: Sensor IoT

Variáveis Configuradas:

Variável	Tipo	Unidade	Limites	Status
temperatura	Float	°C	-40 a +80	Ativa
umidade	Float	%	0 a 100	Ativa
gas	Integer	ppm	0 a 1023	Ativa
alarme	Integer	Enum	0 a 4	Ativa

10.1 Widgets no Dashboard

10.1.1 Widget 1: Gauge Temperatura

- **Variável:** temperatura
- **Mín:** 15°C | **Máx:** 40°C
- **Alerta:** > 30°C (vermelho)

10.1.2 Widget 2: Gauge Umidade

- **Variável:** umidade
- **Mín:** 30% | **Máx:** 90%
- **Alerta:** > 70% (laranja)

10.1.3 Widget 3: Gauge Gás

- **Variável:** gas
- **Mín:** 500 ppm | **Máx:** 1000 ppm
- **Alerta:** > 940 ppm (vermelho)

10.1.4 Widget 4: Indicador de Estado

- **Variável:** alarme
- **Estados:**

- 0 = Verde (NOMINAL)
- 1 = Vermelho (GAS)
- 2 = Laranja (TEMP_ALTA)
- 3 = Amarelo (UMIDADE_ALTA)
- 4 = Vermelho (INCENDIO)

10.1.5 Widget 5: Gráfico de Histórico (Última 24h)

- **Variáveis:** temperatura, umidade, gas
- **Tipo:** Linha com pontos
- **Intervalo:** Últimas 24 horas

10.2 Alertas Configurados

Evento	Condição	Ação
Temperatura Alta	tmp > 30°C	Email
Umidade Alta	umi > 70%	Nada
Gás Detectado	gas > 940	SMS
Incêndio	alarme == 4	SMS + Email

10.3 Screenshot do Dashboard

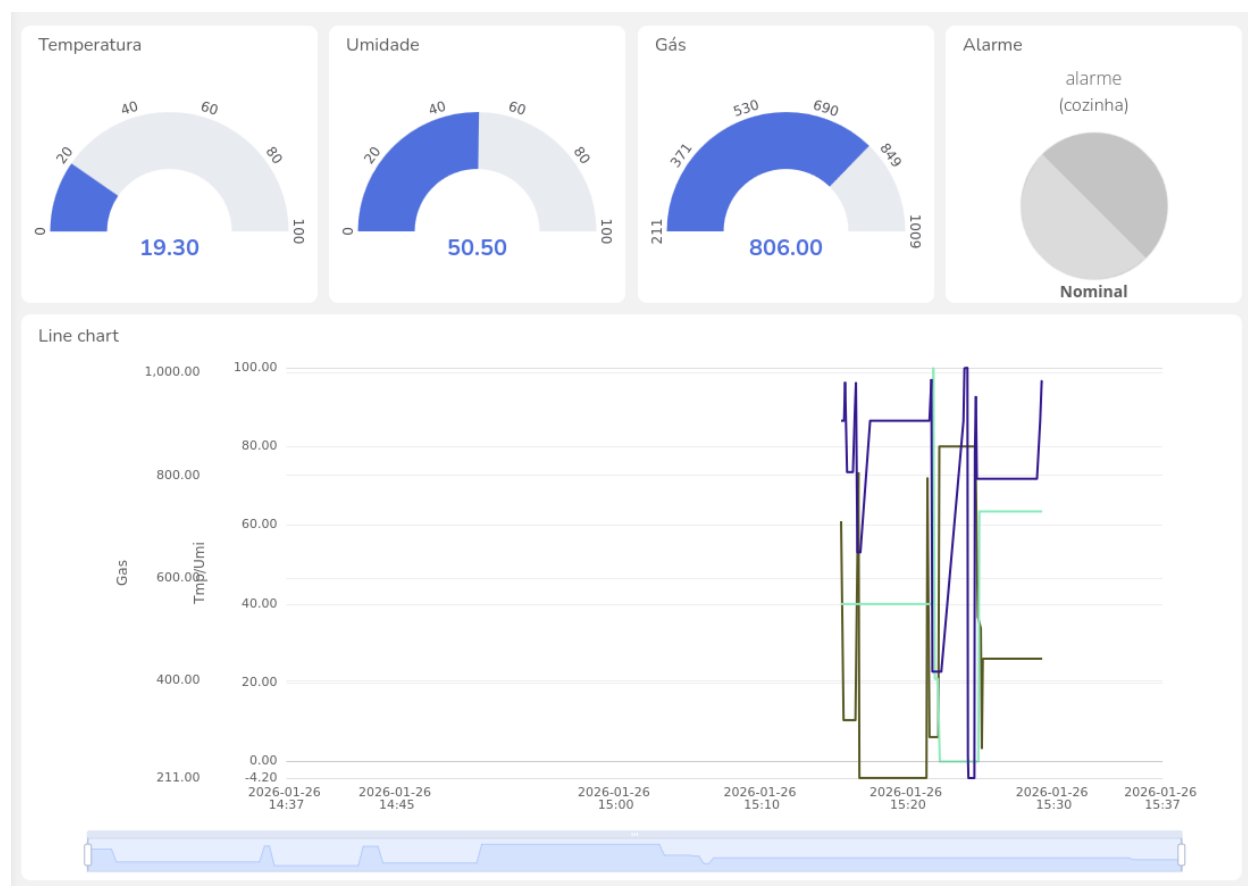


Figura 10.1: Dashboard no Ubidots

Conteúdo Esperado:

- 4 widgets numéricos com valores em tempo real
- Gráfico histórico de 24 horas
- Status indicador colorido
- Últimas leituras: data/hora

10.4 Dashboard NodeRed - Monitoramento Fallback

10.4.1 Acesso

O sistema também disponibiliza um dashboard NodeRed para monitoramento alternativo ao Ubidots:

URL: <http://77.37.69.84:1880/dashboard/page1>

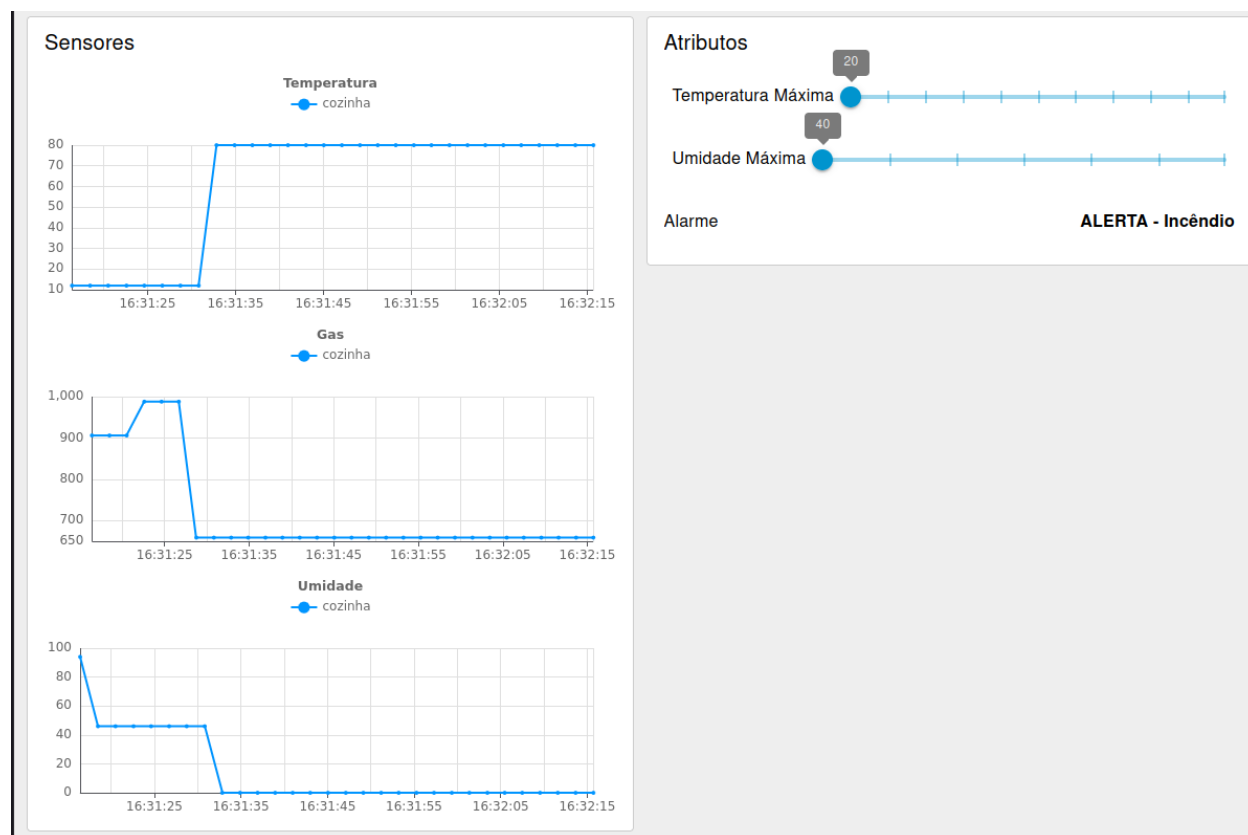


Figura 10.2: Dashboard no NodeRed

10.4.2 Recursos Disponíveis

Recurso	Descrição	Funcionalidade
Indicadores Numéricos	Temperatura, Umidade, Gás	Valores em tempo real
Status do Alarme	Código 0-4	Indicador colorido (verde/vermelho/amarelo)
Gráfico em Tempo Real	Histórico 2 minutos	Visualização de tendências
Slider Temperatura	Input de controle	Alterar limite máximo de temp
Slider Umidade	Input de controle	Alterar limite máximo de umidade

10.4.3 Configuração de Sliders

Os sliders do dashboard NodeRed publicam diretamente nos tópicos MQTT:

Slider Temperatura → cozinha/max_tmp

Slider Umidade → cozinha/max_umi

Operação:

1. Mover slider de temperatura
2. NodeRed publica valor em cozinha/max_tmp
3. ESP32 subscreve e recebe o comando
4. Limite é atualizado imediatamente

Chapter 11

Conclusões

11.1 Funções Implementadas

1. **Detecção de Incêndio em 2D:** Combinação de ΔT e ΔUR em janela temporal
2. **Configuração Remota:** Limiares ajustáveis via MQTT em tempo real
3. **Histerese Dinâmica:** Evita oscilações entre estados
4. **Priorização de Alarmes:** Sistema hierárquico de estados
5. **Integração Cloud:** Dashboard profissional com Ubidots

11.2 Aplicações Práticas

Este sistema pode ser estendido para:

- Outras ambientes: restaurantes, indústrias alimentícias
- Múltiplos sensores: CO₂, fumaça, luminosidade
- Integração com sistemas SCADA
- Análise preditiva com machine learning
- Mobile app de alertas

11.3 Recomendações Futuras

- ☐ Adicionar certificado SSL/TLS para segurança
- ☐ Implementar backup local em SD card
- ☐ Criar relatórios mensais de histórico
- ☐ Adicionar IA para detecção de padrões anormais

Última Atualização: 26 de janeiro de 2026

Versão: 1.0

Status: Finalizado