



Especificação Técnica e Funcional

Enterprise Challenge - Sprint 2 - Reply

FIAP

Introdução

Este documento tem como objetivo, definir a solução e permitir a validação antes da sua execução, evitando retrabalhos e frustrações com o resultado.

Integrantes

Integrante	RM
Vinícius Pereira Santana	RM564940
Vitor Augusto Prado Guisso	RM562317
Ryan Carlos Sousa Alves Cunha	RM561677

Histórico de Tarefas

Data	Versão	Descrição	Autor / RM
29/05/2025	01	Criação do documento de requisitos referente a primeira documentação e criação da lógica	Vinícius Santana RM: 564940
05/06/2025	02	Análise de requisitos e criação do circuito na plataforma wokwi	Vitor Guisso RM: 562317
07/06/2025	03	Extração de dados simulados.csv e criação de gráfico	Ryan Carlos RM: 561677
10/05/2025	04	Elaboração da Especificação Técnica e Funcional	Vinícius Santana RM: 564940

A Solicitação do Cliente

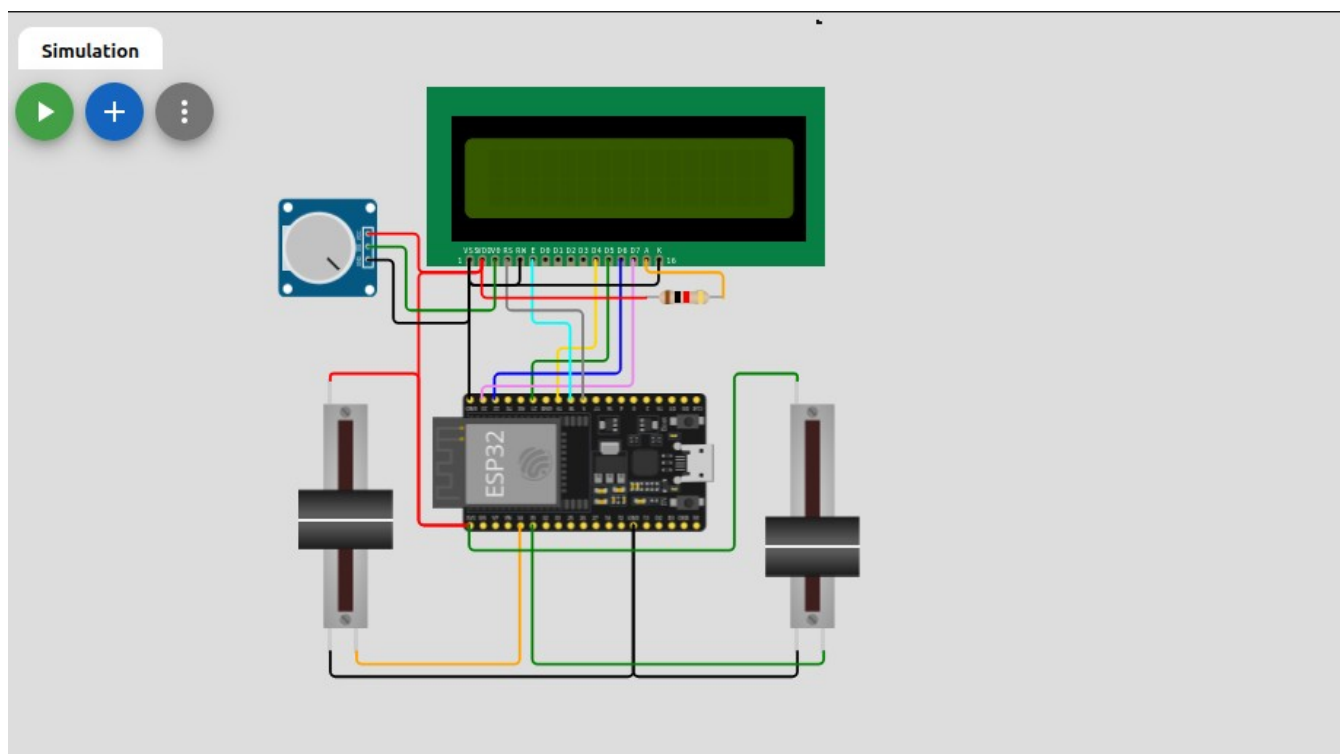
Necessidades

O cliente busca obter uma economia no consumo de energia excessivo causado pelos **motores trifásicos de grande porte**, especialmente durante horários de pico, e evitar multas por ultrapassagem da demanda contratada, o que tem ocasionado em perdas financeiras diretas. Com isso o cliente gostaria de obter visibilidade em tempo real e previsibilidade do desligamento dos motores em horários de pico

Solução

Monitoramento:

Conforme descrito na Especificação técnica da fase 3, para essa primeira etapa prática na fase 4 desenvolvemos uma solução utilizando o WOKWI e ESP32 afim de simular o monitoramento do consumo elétrico de motores industriais simulando a corrente elétrica dos motores trifásicos com potenciômetros



Projeto	Solução
Link Wokwi	https://wokwi.com/projects/433339124946444289

Arquitetura de componentes e suas Funções

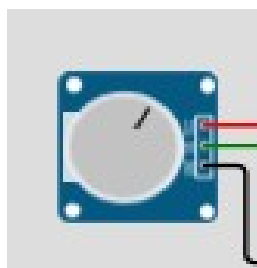
1. ESP32 DevKit

- **Função:** Microcontrolador principal do projeto.
- **Responsável por:** Ler os sinais analógicos dos potenciômetros (simulando consumo dos motores trifásicos), processar os dados e exibir informações no display LCD.
- **Pinos utilizados:**
 - GND (terra), 3V3 (alimentação),
 - Pinos analógicos (ex: GPIO 32, 33, 34 etc.),
 - Pinos digitais conectados ao LCD.



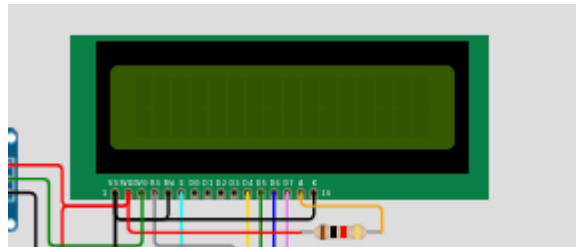
2. 3x Potenciômetros

- **Função:** Simulam a corrente elétrica dos três motores trifásicos.
- **Como funcionam:** Cada potenciômetro varia um valor analógico de 0 a 4095 (no ESP32) conforme gira o botão.
- **Conectados a:** Pinos analógicos do ESP32 (ex.: GPIO 32, 33, 34).
- **Finalidade:** Gerar uma leitura proporcional ao “consumo” de cada fase do motor trifásico.



3. Display LCD 16x2 (com módulo I²C ou sem)

- **Função:** Exibe informações em tempo real como:
 - Potência total estimada (kVA),
 - Alertas de pico de consumo,
 - Ações como “Desligamento de motor”.
- **Conexões típicas:**
 - VSS → GND
 - VDD → 5V
 - RS, E, D4-D7 → pinos digitais do ESP32
 - RW geralmente ligado ao GND
 - Potenciômetro para ajuste de contraste.



4. Resistor (ligado ao LCD)

- **Função:** Limitador de corrente, geralmente usado no pino do **LED de backlight** do display (ou no pino de contraste em alguns casos).
- **Valor típico:** 220 Ω ou 330 Ω .

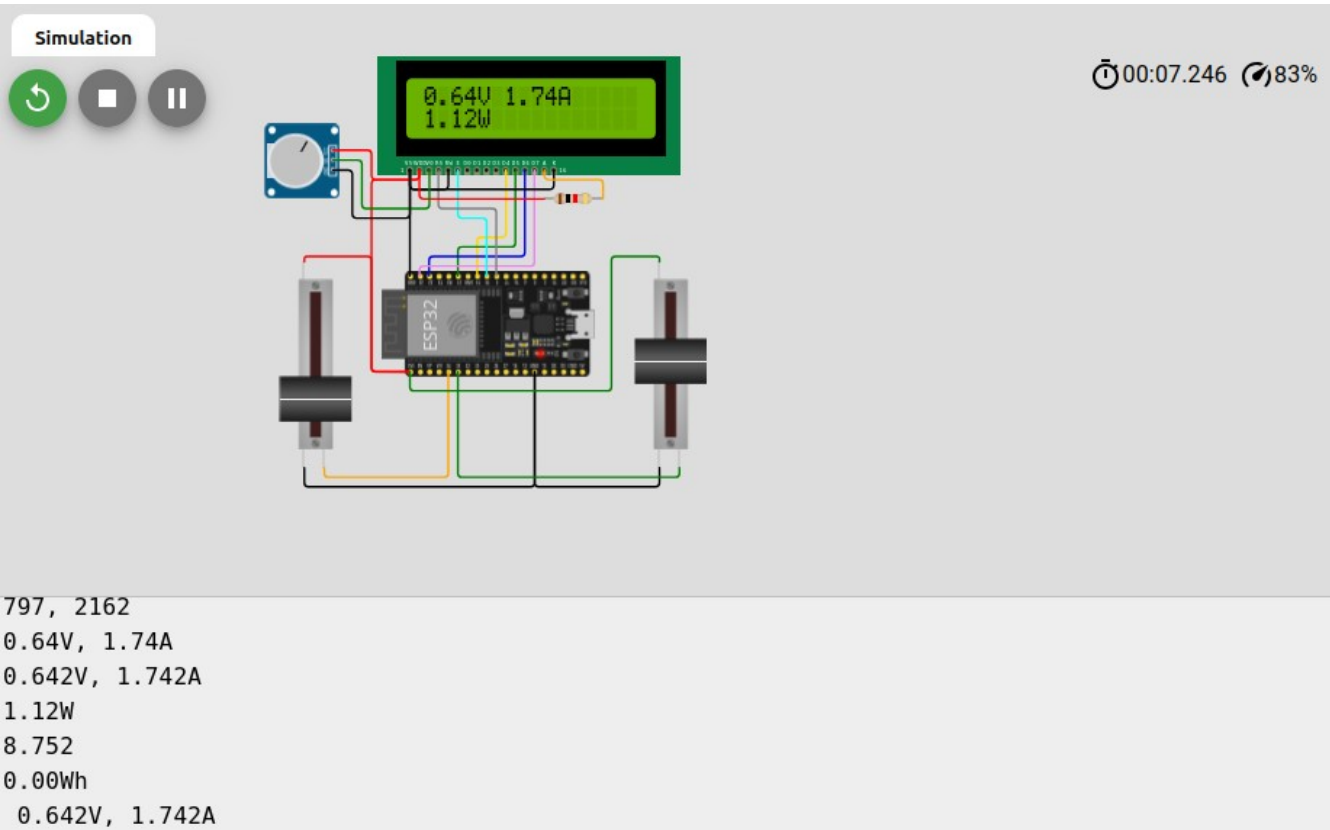


5. Fios de conexão

- **Função:** Interligam todos os componentes eletricamente.
- **Cores variadas** para facilitar a identificação:
 - Vermelho: alimentação (3.3V ou 5V)
 - Preto: terra (GND)
 - Outras cores: sinais digitais ou analógicos.

Resumo da lógica do projeto

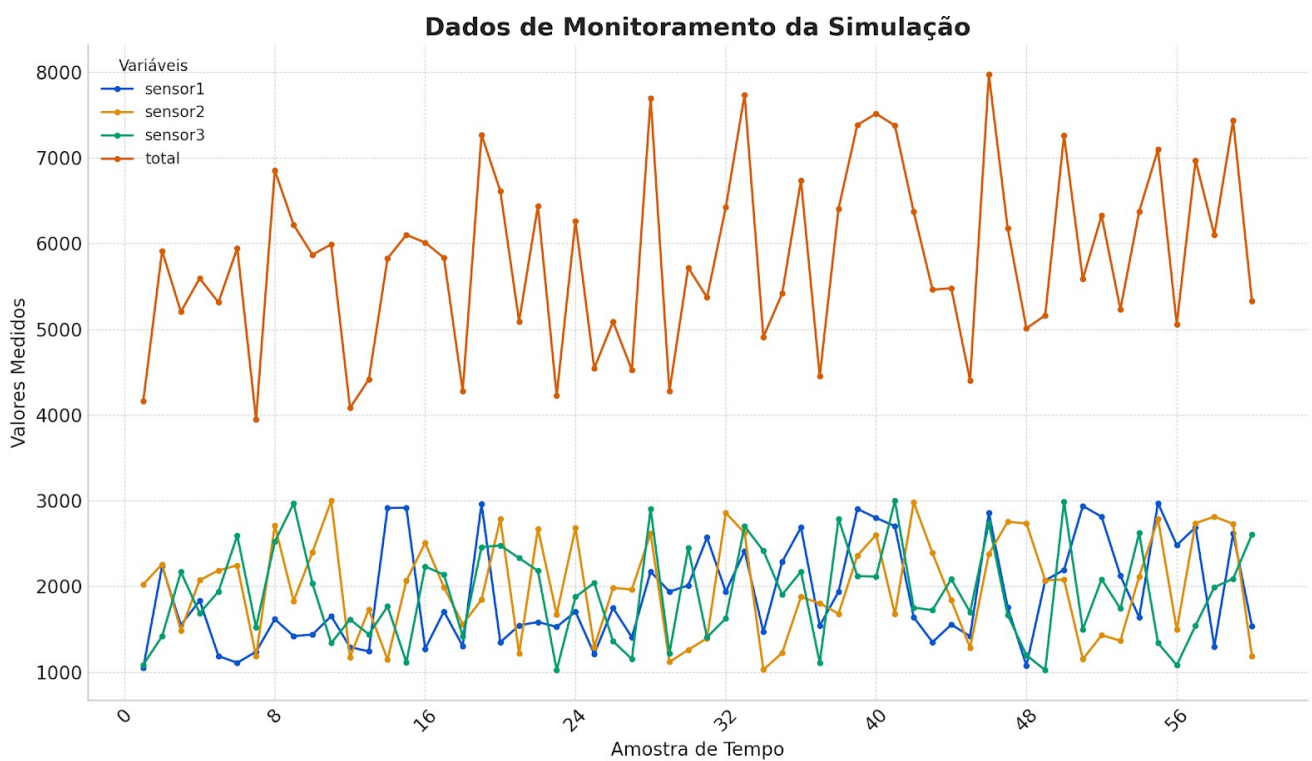
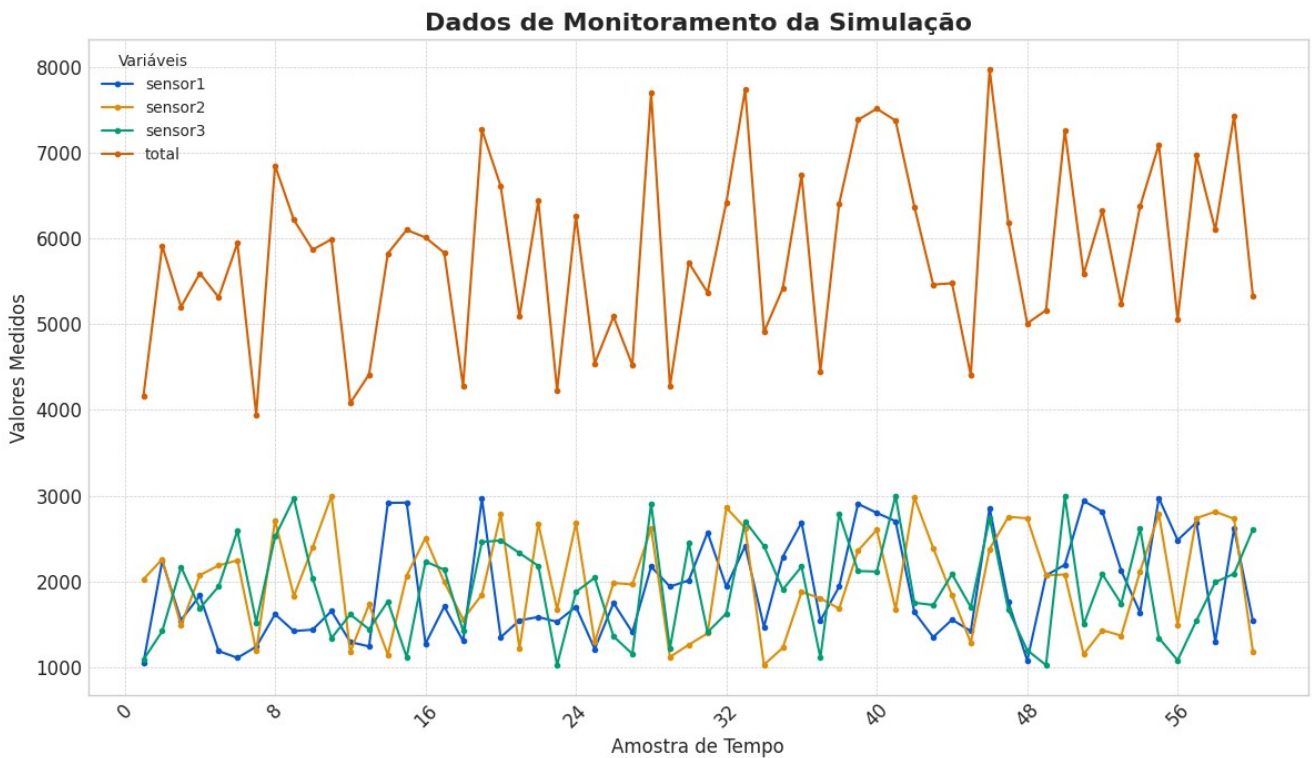
- 1. O **ESP32** lê os valores analógicos dos **3 potenciômetros**, simulando o consumo de cada fase de um motor trifásico.
- 2. Esses valores são **processados** para calcular uma potência total.
- 3. O **LCD** mostra os valores de consumo e mensagens em tempo real.



Dados Coletados

1	timestamp	sensor1	sensor2	sensor3	total
2	2025-06-10T02:23:18,134	1051	2024	1085	4160
3	2025-06-10T02:23:19,134	2237	2258	1419	5914
4	2025-06-10T02:23:20,134	1548	1486	2168	5202
5	2025-06-10T02:23:21,134	1834	2072	1687	5593
6	2025-06-10T02:23:22,134	1187	2186	1940	5313
7	2025-06-10T02:23:23,134	1108	2245	2592	5945
8	2025-06-10T02:23:24,134	1238	1186	1522	3946
9	2025-06-10T02:23:25,134	1617	2712	2521	6850
10	2025-06-10T02:23:26,134	1420	1829	2968	6217
11	2025-06-10T02:23:27,134	1437	2397	2035	5869
12	2025-06-10T02:23:28,134	1653	2999	1338	5990
13	2025-06-10T02:23:29,134	1292	1175	1614	4081
14	2025-06-10T02:23:30,134	1241	1731	1441	4413
15	2025-06-10T02:23:31,134	2915	1145	1766	5826
16	2025-06-10T02:23:32,134	2918	2067	1115	6100
17	2025-06-10T02:23:33,134	1272	2507	2232	6011
18	2025-06-10T02:23:34,134	1707	1990	2136	5833
19	2025-06-10T02:23:35,134	1303	1557	1419	4279
20	2025-06-10T02:23:36,134	2964	1845	2459	7268
21	2025-06-10T02:23:37,134	1348	2786	2477	6611
22	2025-06-10T02:23:38,134	1545	1216	2330	5091
23	2025-06-10T02:23:39,134	1584	2673	2181	6438
24	2025-06-10T02:23:40,134	1528	1674	1024	4226
25	2025-06-10T02:23:41,134	1704	2681	1876	6261
26	2025-06-10T02:23:42,134	1211	1290	2042	4543
27	2025-06-10T02:23:43,134	1748	1981	1359	5088

Gráficos de Monitoramento da Simulação



Matriz de Responsabilidades

Integrante	RM	Função Principal	Descrição
Vinícius Pereira Santana	RM564940	Líder Técnico e Documentação	Responsável pela organização geral do projeto, versionamento no GitHub, README, integração entre os módulos e entrega final.
Vitor Augusto Prado Guisso	RM562317	Desenvolvedor Backend / API	Desenvolverá a API em Python (FastAPI) para recepção e envio de dados dos sensores, além de auxiliar na comunicação com o Odoo.
Ryan Carlos Sousa Alves Cunha	RM561677	Especialista IoT e Coleta de Dados	Simulação e definição da coleta de dados via sensores (PZEM-004T, Modbus), incluindo comunicação via MQTT.