

Especificação de Requisitos de Software

Projeto: *Sistema de Monitoramento de Energia Elétrica (SMEE)*

Versão: 2.0

Discentes: Gustavo Henrique Martins, Ícaro Caldeira Botelho, Maruan Biasi El Achkar, Rafael Pereira, Ricardo Falcão Schlieper

Docente: Luís Camargo

CamarVolt101

Sumário

Histórico de Atualização

Sumário

1 Introdução

1.1 Objetivo deste documento

1.2 Escopo do produto

1.3 Visão geral deste documento

2 Descrição geral do produto/protótipo

3 Contexto para Produto/Protótipo

4 Requisitos do Produto/Protótipo

4.1 Requisitos Funcionais

4.2 Requisitos Não Funcionais

4.3 Requisitos de Hardware

4.4 Requisitos de Software

5 Artefatos do Produto/Protótipo

6 Integrações e Extensões

6.1 Bluetooth Sniffer

6.2 Integração com Skill da Alexa

7 Considerações Finais

8 Apêndices (Evidências de Implementação)

1 Introdução

1.1 Objetivo deste documento

Este documento apresenta a especificação de requisitos do Sistema de Monitoramento de Energia Elétrica (SMEE). Seu objetivo é detalhar as funcionalidades, os requisitos técnicos e os artefatos necessários para o desenvolvimento do protótipo, permitindo o acompanhamento em tempo real do consumo de energia, a geração de relatórios e a emissão de alertas.

1.2 Escopo do produto

Produto: Sistema de Monitoramento de Energia Elétrica (SMEE)

Componentes principais: Sensores de corrente e tensão, microcontrolador ESP32, comunicação MQTT, servidor backend (Node-RED, InfluxDB, Mosquitto), dashboard de visualização (Grafana ou aplicação web).

Missão: Promover o uso eficiente e consciente da energia elétrica, permitindo monitoramento em tempo real, identificação de desperdícios, redução de custos e aumento da segurança elétrica.

1.3 Visão geral deste documento

A estrutura segue: descrição geral (Seção 2), contexto (Seção 3), requisitos (Seção 4), artefatos (Seção 5), integrações e extensões (Seção 6), considerações finais (Seção 7) e apêndices (Seção 8).

2 Descrição geral do produto/protótipo

O SMEE é um protótipo IoT que coleta dados de tensão, corrente e potência, transmitindo-os via ESP32 e rede Wi-Fi para um servidor central. Os dados são processados em Node-RED, armazenados em InfluxDB e exibidos em dashboards Grafana, com possibilidade de relatórios e alertas configuráveis.

3 Contexto para Produto/Protótipo

Ambiente de uso: residências, empresas e indústrias.

Usuários: moradores, administradores, engenheiros de manutenção.

Fluxo de interação: Sensores → ESP32 → Wi-Fi → MQTT Broker → Node-RED → InfluxDB → Grafana/App → Usuário.

4 Requisitos do Produto/Protótipo

4.1 Requisitos Funcionais

- **RF01:** Medir tensão, corrente, potência e fator de potência a cada 10 segundos.
- **RF02:** Calcular e armazenar o consumo total em kWh.
- **RF03:** Exibir gráficos e relatórios no dashboard.
- **RF04:** Gerar alertas em caso de sobrecarga ou consumo anormal.
- **RF05:** Permitir exportação dos dados em CSV/JSON.
- **RF06:** O aparelho deve ser constituído de um "adaptador" de tomada, servindo de ponte entre o final do cabo e a tomada.
- **RF07:** O aparelho deve conter apenas 1 plug macho compatível com o padrão brasileiro (2P+T).
- **RF08:** O aparelho deve conter no mínimo 1 plug fêmea compatível com o padrão brasileiro (2P+T).
- **RF09:** O aparelho deve ter capacidade de se conectar a redes Wi-Fi.
- **RF10:** O aparelho deve ter a capacidade de enviar dados para um servidor central via Wi-Fi.
- **RF11:** O aparelho deve ser capaz de funcionar utilizando apenas a eletricidade da tomada em que foi instalado.
- **RF12:** O aparelho deve enviar os dados de forma segura.
- **RF13:** O servidor deve ser capaz de receber dados via Wi-Fi.
- **RF14:** O servidor deve ser capaz de receber dados de múltiplos aparelhos.
- **RF15:** O servidor deve ser capaz de armazenar dados.
- **RF16:** O servidor deve ter um dashboard visual.
- **RF17:** O servidor deve calcular valores gastos em energia elétrica a partir de dados de custo inseridos pelo usuário.
- **RF18:** O servidor deve ter "presets" de custos elétricos (Exemplo: Preço da energia elétrica em Joinville).

4.2 Requisitos Não Funcionais

- **RNF01:** O sistema deve ter disponibilidade mínima de 99%.
- **RNF02:** O tempo de resposta dos dados deve ser inferior a 5 segundos.
- **RNF03:** Todos os dados transmitidos devem ser criptografados (TLS).
- **RNF04:** O sistema deve armazenar dados históricos por pelo menos 12 meses.
- **RNF05:** O aparelho deve ser semelhante a adaptadores tradicionais, tanto em aparência quanto em funcionamento.
- **RNF06:** O aparelho não pode protuberar mais do que 10cm da tomada, medindo a partir de ângulo reto à superfície utilizável da tomada.

- **RNF07:** O aparelho não pode protuberar mais do que 5cm da tomada nas direções paralelas à superfície utilizável da tomada.
- **RNF08:** O aparelho deve usar materiais que inibam ou diminuam o risco de fogo.
- **RNF09:** O aparelho deve ter design que iniba ou diminua o risco de fogo.
- **RNF10:** O aparelho não deve dar choque elétrico no usuário.
- **RNF11:** O aparelho não pode ser nomeado "Ultimate Electricity Measuring Device Pegatron 5000".
- **RNF12:** O aparelho não pode ter aparência ofensiva.
- **RNF13:** O servidor não pode desintegrar dados.
- **RNF14:** O servidor não pode ser nomeado "Ultimate Electricity Measuring Server Pegatron 5000".
- **RNF15:** O servidor deve receber e tratar os dados de forma segura.

4.3 Requisitos de Hardware

- Sensores de corrente (SCT-013).
- Módulo medidor de energia (PZEM-004T).
- Microcontrolador ESP32.
- Fonte de alimentação estabilizada.
- Módulo Wemos ESP8266 com CH340G
(<https://www.autocorerobotica.com.br/modulo-wemos-esp8266-com-ch340g>).
- ZMCT103C 5A
(https://shopee.com.br/product/343910579/19225078896?gads_t_sig=VTJGc2RHVmtYMTlxTFVSVVRrdENkVHQ3ZkZSUTMrR3pBWmZZNzdrcnRBMXZ4MUZGQWFkQmZudWt0K3BnbDRBdEhyYXV2WnNCcUVyVEhmdjJScndqeTJ0M1RQVDAySnpp eGZIVUhNcitxSGkzZEhhc216NWtrdC8wSVFDRkZ6ZDJiTkE0NlpCZmpObTdJWGVhOUQzZE5RPT0&utm_source=chatgpt.com).
- Fonte de tomada 5V 1A (adaptada e integrada).
- Diodo de proteção.
- Protoboard para prototipagem.
- Fios diversos.
- Jumper Cables Macho-Macho, Macho-Fêmea e Fêmea-Fêmea.
- Equipamento de solda.
- Estanho de solda.
- Computador para deixar o servidor rodando ou alugar um via nuvem (AWS, Azure, Cloud, Oracle, etc.).

4.4 Requisitos de Software

- Firmware em C++ (Arduino/ESP-IDF).
- Mosquitto (broker MQTT).
- Node-RED (processamento de dados).
- InfluxDB (banco de dados de séries temporais).
- Grafana (visualização).
- Computador com VSCode e NodeJS para programar o servidor.

CamarVolt10T

5 Artefatos do Produto/Protótipo

Visão de Usabilidade (Aplicação): Dashboard web exibindo gráficos de consumo em tempo real, histórico e alertas.

Visão Arquitetural: Sensores → ESP32 → Wi-Fi → MQTT Broker → Node-RED → InfluxDB → Grafana/App.

Visão Estrutural (Diagrama de Blocos IoT): Blocos principais: sensoriamento, processamento local, transmissão de dados, armazenamento, visualização.

Visão Comportamental (Diagrama de Sequência):

1. Sensor mede valores.
2. ESP32 processa e envia via MQTT.
3. Servidor armazena e processa.
4. Dashboard exibe dados ao usuário.

6 Integrações e Extensões

6.1 Bluetooth Sniffer

Um Bluetooth sniffer é uma ferramenta de hardware ou software que captura e monitora comunicações Bluetooth entre dispositivos, funcionando de forma similar a um analisador de tráfego de rede, mas focado em Bluetooth. Ele detecta pacotes de anúncios ou trocas de dados, decodifica-os e apresenta em formato legível. Permite:

- Descobrir dispositivos BLE próximos;
- Monitorar pacotes de anúncios BLE;
- Analisar intensidade do sinal (RSSI);
- Depurar aplicações BLE;
- Fazer engenharia reversa de serviços BLE.

6.2 Integração com Skill da Alexa

O SMEE poderá ser integrado a uma skill personalizada da Alexa, permitindo comandos de voz como: "Alexa, qual é o consumo atual de energia?" ou "Alexa, me avise se houver sobrecarga". Essa integração melhora a acessibilidade e aumenta a usabilidade do sistema em ambientes residenciais e corporativos.

7 Considerações Finais

Durante o desenvolvimento do SMEE foi possível compreender como as tecnologias IoT contribuem para eficiência energética. As principais dificuldades incluíram a calibração dos sensores, integração de módulos de software e segurança elétrica. O protótipo demonstrou potencial real de aplicação prática, com extensões futuras como monitoramento via Bluetooth e integração com assistentes virtuais.

8 Apêndices (Evidências de Implementação)

- Link do repositório remoto: <https://camarvolt.github.io/>

CamarVolt10T