MÓDULO 1:

SOFTWARE MOCKUP PARA MEASUREMENT UNITS (MUs) DE SUBESTAÇÕES DE ENERGIA

Documento TR-01
Versão 1.0 de janeiro de 2024
Projeto da disciplina STR
www.feelt.ufu.br

Execução:

LRI - LABORATÓRIO DE REDES INTELIGENTES
www.lri.ufu.br
UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Alan Petrônio Pinheiro
Coordenador do projeto – UFU/LRI

Execução e pesquisa:

Nome 1
Nome 2
Nome 3

1. DATA VERSÃO ORIGINAL 08-1-2024	**2. DATA ÚLTIMA ATUALIZAÇÃO** 21-2-2024	**3. DATA COBERTA** JAN/24 ATÉ MAI/24
4. TÍTULO DESTE DOCUMENTO REPORTE TÉCNICO DA APLICAÇÃO MU		**5a. PROCESSO SEI DO P&D** -
		5b. NÚMERO PROJETO P&D 00123456789....
6. AUTOR(ES) ALAN PETRÔNIO PINHEIRO (UFU)		**5c. ETAPA DO PROJETO** TODAS
		5d. TIPO DE PRODUTO DOCUMENTAÇÃO TÉCNICA DE SOFTWARE DE DISCIPLINA STR
7. ENDEREÇO AV. JOÃO NAVES DE ÁVILA, 2121, BLOCO 3N - UBERLÂNDIA - MG		**8. NÚMERO DO DOCUMENTO** **TR-01**
9. DISTRIBUIÇÃO DESTE DOCUMENTO DISTRIBUIÇÃO ABERTA A TODOS OS INTERESSADOS.		
10. NOTAS COMPLEMENTARES -		
11. RESUMO ESTE DOCUMENTO DESCREVE A MODELAGEM DOS ELEMENTOS DA APLICAÇÃO MU QUE GERA PACOTES DE DADOS PARA OUTROS ELEMENTOS DE SOFTWARE RETRATADO POR OUTROS REPORTES TÉCNICOS. TODOS MÓDULOS CONSTITUEM UM SISTEMA SUPERVISÓRIO PARA O SETOR ELÉTRICO.		
12. PALAVRAS-CHAVE P&D; IOT; SISTEMA EM TEMPO REAL, MERGE UNIT, SISTEMA SUPERVISÓRIO, MEDIÇÃO EM SUBESTAÇÕES DE ENERGIA.		
13. CLASSIFICAÇÃO SEGURANÇA: **ABERTA**	**14. NÚMERO DE PÁGINAS** -	**15. NOME DO RESPONSÁVEL PRINCIPAL E CONTATO** ALAN PETRÔNIO PINHEIRO. EMAIL: alan_petronio@yahoo.com.br. TELEFONE: (34)3239-4701

HISTÓRICO DE VERSÕES DESTE TR

Tabela 1 – *Histórico de versões deste reporte técnico.*

Versão	Data	Modificações
1.0	janeiro/2024	• Principais elementos de projeto • Requerimentos básicos • Modelagem de pacotes e fluxo de pacotes • Interfaces básicas

SUMÁRIO

RESUMO GERAL 4
1 – INTRODUÇÃO: VISÃO GERAL DA SOLUÇÃO 4
1.1 – PROPÓSITO E ESCOPO 4
1.2 – PRODUTO: PERSPECTIVAS E FUNÇÕES 5
1.3 – RESTRIÇÕES DO PRODUTO E CONSIDERAÇÕES 6
2 – REQUISITOS 7
2.1 – CENÁRIOS DE USO 7
2.2 – REQUISITOS E VALIDAÇÃO 8
2.3 – VERSIONAMENTO 9
2.4 – ELEMENTOS DE PROJETO 10
2.4.1 – Módulo de medição 10
2.4.2 – Máquina de estados 10
2.4.3 – INTERFACES DE USUÁRIO 11
2.5 – INTERFACES DE COMUNICAÇÃO E INFRA DE TIC 12
3 – MODELAGEM-MÉTODOS 13
3.1 – BLOCOS DE ELEMENTOS PRINCIPAIS 13
3.2 – TABELA GERAL DE OBJETOS IPSO E RECURSOS DE URI 15
3.3 – MODELAGEM DE RECURSOS 15
3.3.1 – Eventos de carregamento da tela principal 15
3.4 – PLANOS DE TESTES ..... **ERRO! INDICADOR NÃO DEFINIDO.**

RESUMO GERAL

Este reporte técnico aborda os elementos do sistema de software "mockup" que compõe a solução de sistema supervisório para o setor elétrico. Em específico, ele também dá suporte a aplicação de proteção de sistemas elétricos de potência e automação de processos. Todas as aplicações voltadas exclusivamente para o setor elétrico. Em específico, o software aqui posto é uma versão digitalizada do "measurement unit" ou "merge unit" prevista na norma IEC61850 responsável por fazer medições de corrente e tensão diretamente da rede elétrica e injetá-las na rede de computadores, através de pacotes IP, provendo a outros elementos (também ligados a mesma rede) indicação contínua dos valores lidos. Vale destacar que este sistema envolve comunicação tanto com outros softwares quanto hardwares e ambos constituem um sistema maior: o supervisório. Outros reportes técnicos da solução podem ser consultados para informações sobre a solução como um todo.

1 – Introdução: visão geral da solução

1.1 – Propósito e escopo

O propósito do sistema é fazer medições constantes de corrente e tensão e provê-las a uma rede ethernet para outros elementos de tal maneira que estes outros elementos possam identificar alguns eventos na rede elétrica geral (quedas de energia, curtos-circuitos, transitórios, etc). Logo, o foco desta aplicação é gerar medidas elétricas segundo: (i) a necessidade de outros módulos e (ii) o que está acontecendo na rede elétrica. Tal sistema, será chamado designado doravante como MU (measurement unit).

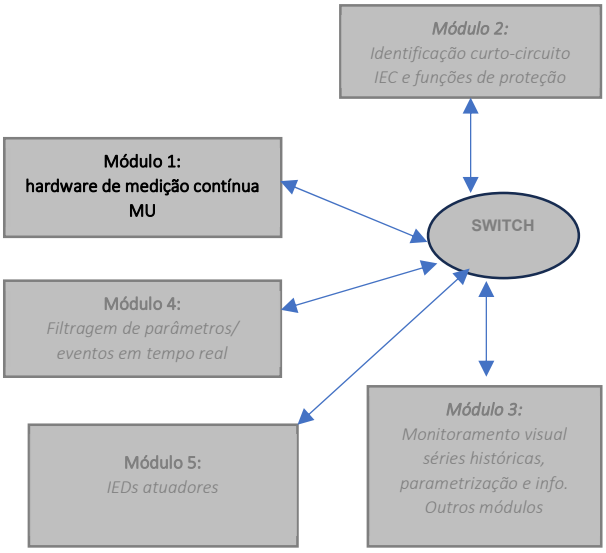

Figura 1.1.1: Visão geral de escopo.

Desta maneira, fica evidenciado que mesmo embora a principal função deste "modulo 1" seja a de fornecer medidas elétricas para o 'meio', ele deve fazer isto para atender as características funcionais de outros subsistemas, a saber:

1) Módulo de monitoramento visual de séries históricas;
2) Módulo de identificação de curto circuito e "IEDs atuadores", que trabalham em complemento;
3) Módulo de filtragem de parâmetros e eventos em tempo real.

Desta maneira, muitos dos requisitos deste sistema são para atender aos requisitos funcionais dos subsistemas listados anteriormente.

Ainda, vale entender que dentro de uma mesma subestação de energia (SE), podem haver vários MUs conforme se observa na figura da sequência. Cada um responsável por medir uma determinada porção do circuito (ver diagrama unifilar) de uma subestação. Todavia, existe para todos estes MUs apenas 1 instância de sistema supervisório e serviços (outros módulos) que devem ser capazes de processar todas as informações de todos MUs presentes dentro de uma ou mais SEs.

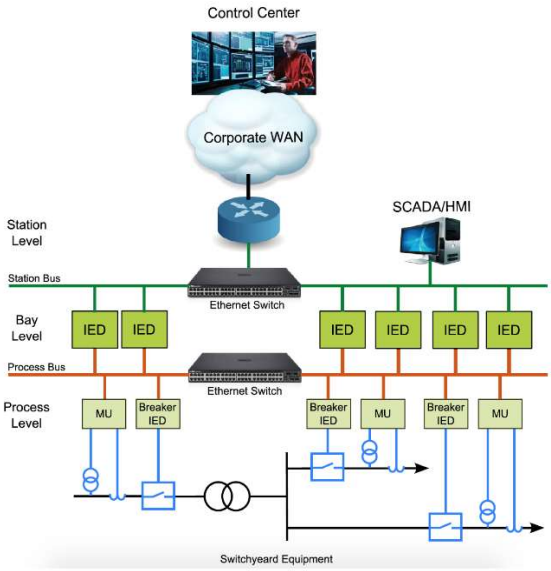

Figura 1.1.2: Visão geral de cenário.

1.2 – Produto: perspectivas e funções

Este sistema tem como principal função informar a rede de comunicação sobre as medições que faz com a maior fidelidade possível de medição e prover, para os outros sistemas que dependem dele, condições de funcionamento destes atendendo aos requerimentos de tempo para correta atuação de todo o sistema em circunstâncias específicas de eventos elétricos (como falhas, oscilações, quedas, curtos, etc). Para isto, tem como principais funções:

1) Fazer medição de 3 correntes e 3 tensões ;
2) Enviar as medidas por rede ethernet em IP (confiável);
3) Pode ser parametrizado remotamente pelo operador;
4) Responder as demandas dos requerimentos de tempo da norma IEC61850.

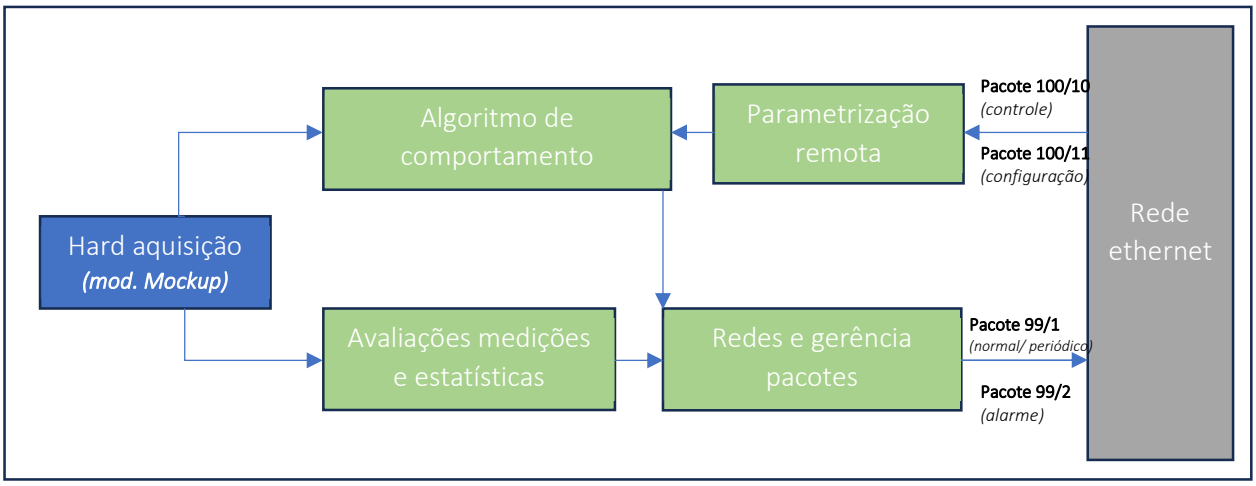

Figura 1.2.1: *Principais elementos de projeto de 1 MU.*

Para entender o sistema, comecemos a análise observando a Figura 1.3.1. Com base nisto, descreve-se os elementos:

- **Módulo hardware aquisição:** faz as medidas analógicas, digitaliza e entrega para a memória os valores aquisicionados em frequência de aquisição adequada.
- **Módulo algoritmo de comportamento:** roda em "paralelo" com o bloco de "avaliações medições e estatísticas" e tem como função verificar se existe algum indício de acontecimento de evento na rede. Sua função é determinar a taxa de envio de pacotes, quando na suspeita de existência de evento na rede.
- **Módulo 'avaliações das medições e estatísticas':** pega as medidas atuais, e de posse das medidas passadas (que ele buferizou) faz as avaliações estatísticas (média, RMS, desvio padrão) e avalia se a medição não é um outlier que possa prejudicar as estatísticas gerando valores incoerentes que causem problemas de 'falso positivo'.
- **Módulo 'redes e gerência de pacotes':** este módulo pega os dados estatísticos já calculados e monta os pacotes de rede despachando eles. Ele faz isto com a frequência de envio apontada pelo módulo 'algoritmo de comportamento'. Esta frequência varia segundo a ocorrência – ou não ocorrência – de evento na rede.
- **Módulo 'parametrização remota':** faz com que as parametrizações do algoritmo de comportamento possam ser mudadas remotamente por algum operador sem que ter fazer com que ele precise ir até o hardware para fazer estas mudanças.

1.3 – Restrições do produto e considerações

A solução geral aqui prevista foi testada para condições específicas e nestas, foram identificadas as seguintes restrições ou limitações para os quais o sistema proposto não foi projetado para atuar. Estas restrições e limitações são mostradas na tabela da sequência.

Tabela 1.3.1: *Restrições e limitações previstas para sistema.*

Nº	Restrição/limitação	Descrição/detalhamento

1	O sistema é voltado apenas para SE de energia que seguem o padrão IEC 61850	Ele se aplica somente as normas de SE de média tensão da IEC, especialmente a IEC 61850, para tempos de atuação, arquitetura, configuração e formatação de dados.
2	O sistema é focado apenas para bay de processos	Sua medição e normalização é para atuação em transformadores de potência, disjuntores AT/MT, chaves seccionadoras e cubículos de medição.
3	O sistema precisa de um computador industrial instalado dentro da SE	Para execução é necessário que no ambiente da SE exista um computador industrial apto a operar neste tipo de ambiente e com os mecanismos de proteção corretos. Ainda, ter interfaces de comunicação com latência desprezível.
4	Sem ausência de redundância ou WDT	Caso o software falhe, não existe nenhum mecanismo de contingência do mesmo.
5	O sistema emula no máximo 10 MU	O sistema proposto consegue emular só até 10 unidades de 'measuremente units'.
6	Aplicável somente a medições indiretas	Pode ser usado com faixas de tensão
7	Switch ethernet L3 com configurações específicas	Deve ser necessário um switch com recursos de L3, especialmente VLAN, e com boa capacidade de tráfego e, preferencialmente, controle de fluxo de seus buffers para evitar instabilidades na rede.

2 – Requisitos

2.1 – Cenários de uso

Os seguintes cenários foram identificados para este sistema.

a) **Cenário 1 – operação em condições normais**: nele quando não há nenhum evento, o sistema opera em condições normais seguindo a sequência de passos indicada na figura da sequência. Ela ilustra como deve ser seu comportamento.

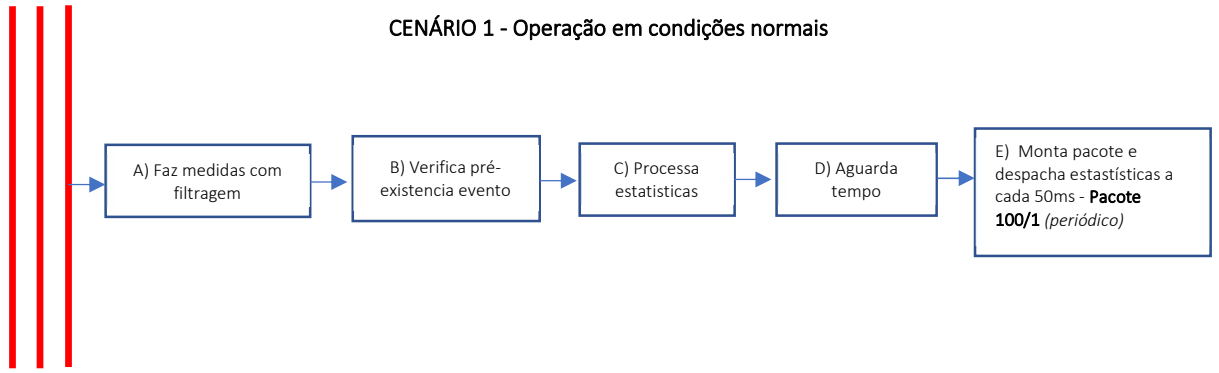

Figura 2.1.1: *Cenário de aplicação.*

b) **Cenário 2 – operação em transitórios elétricos**: nele, o "algoritmo de comportamento" verificou que houve uma mudança abrupta na rede elétrica. Pode ser várias coisas. Uma delas é um principio de curto circuito (ainda a confirmar), ou um transitório ou qualquer coisa que aponte para uma mudança

da rede, mas que ainda não seja possível indicar que de fato é um evento da rede. Ai ele entre me modo "alerta" e faz medções mais frequências, rápidas e avalia continuamente se mantém neste cenário ou vai pra outro. A figura na sequência ilustra como deve ser seu comportamento.

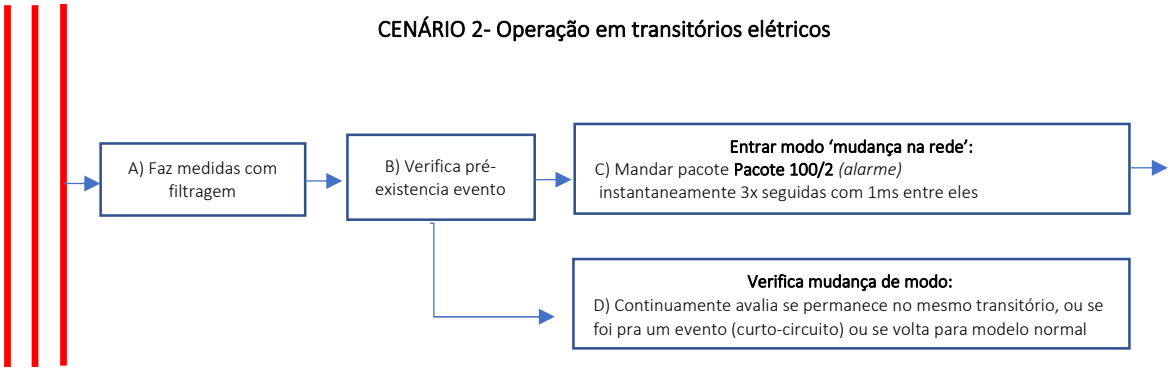

Figura 2.1.2: *Cenário de aplicação.*

c) **Cenário 3 – operação em curto-circuito**: nele, o sistema deve entrar no modo 'mudança na rede' onde foi confirmada que está provavelmente acontecendo algo na rede. A decisão do curto-circuito deve ser feita por outro módulo externo a este sistema. Contudo, a identificação aqui feita serve apenas para entrar neste modo onde a intensidade de envio de pacotes com medidas instantânea é a maior possível com a menor latência, mantida o funcionamento do serviço. A figura na sequência ilustra como deve ser seu comportamento.

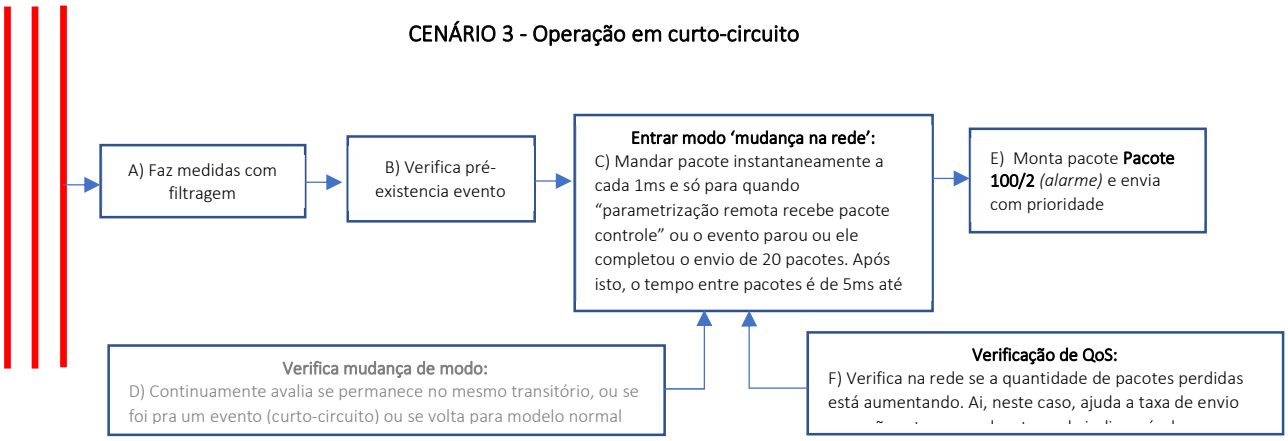

Figura 2.1.3: *Cenário de aplicação.*

2.2 – Requisitos e validação

Com base nas entrevistas com os clientes, equipe de engenharia e avaliações de cenário de uso, desenvolveu-se na sequência a seguinte lista de requerimentos, vista na tabela da sequência.

Tabela 2.2.1: *Mapa de requerimentos.*

Classe/ Componente	Nº req.	Requisito	Origem requisito	Prio	Tipo validação

1 - Robustez	1.1	Ter alimentação auxiliar para não cair (bateria)	Não parar de funcionar durante eventos elétricos extremos	1	Testar em cenário comutando desligamento das fontes por X vezes e analisar disponibilidade do serviço.
	1.2	Ter redundância alimentação		1	Usar uma fonte AC e outra fonte DC.
2 - Funcional	2.1	Fornecer medidas de corrente e tensão usando rede ethernet em taxas aceitáveis para normas proteção	Norma IEC	1	Implatar um módulo de analog front end e validar medidas usando maleta de relé para geração de sinais.
	2.2	Em condições normais, evitar consumir a rede para não prejudicar QoS da rede (taxa perda < 20%)	Evitar que um MU em alarme atrapalhe os outros MUs	1	Testar eventos usando um switch 10/100Mb com 10 MUs operando em regime máximo e ver a taxa de erro de pacotes
	2.3	Na possibilidade de evento, mandar informações o' mais rápido possível' visando garantir um tempo de atuação de sistema (MU + identificação) inferior a 2 ciclos 60Hz (<32ms)	Norma IEC	1	Fazer um recurso no software que aumenta o valor de corrente de modo que os outros módulos identifiquem este aumento, mandem as medidas instanteamente e criamos uma rotina para medir tempo de envio e recebimento que deve ser inferior a 10% do tempo indicado como limite (~3ms)
	2.4	O sistema deve ter até 10 pontos de medição (MUs) em regime trifásico.	Comercialmente é melhor ter 1 só sistema que faz várias unidades	2	Fazer um outro módulo de software de teste que identifique os pacotes na rede e seja capaz de contabilizar a quantidade de MUs
	2.5	O registro de resultados deve levar em conta até a 50º harmônica	ABNT e concessionárias	2	Estimar a quantidade de medidas dentro de 1 segundo de captura
3 – Não funcionais	3.1	Alto MTBF (superior a 1 falha/ano com DEC < 2 mim)	Alta disponibilidade	2	• Hardware: câmera ambiental • Software: múltiplas instâncias em container
	3.2	Sistema deve contribuir para decrementar DEC e FEC evitando falsos positivos	ANEEL	2	Avaliações feitas pelos usuários com base em cenários estatísticos contabilizando-se o tempo (e frequencia) de desligamentos desnecessários em que o sistema aqui proposto poderia atuar evitandos tais desligamentos

2.3 – Versionamento

Os recursos do software são distribuídos em versões conforme estimado pela tabela na sequência.

Tabela 2.3.1: *Tabela de recursos do sistema e versão.*

Versão	Recurso
1.0 *(fev/24)*	■ O sistema deve ter até 10 pontos de medição (MUs) em regime trifásico □ Fornecer medidas de corrente e tensão usando rede ethernet em taxas aceitáveis para normas proteção □ Na possibilidade de evento, mandar informações o mais rápido possível visando garantir um tempo de atuação de sistema (UM + identificação) inferior a 2 ciclos 60Hz (<32ms)
1.1 *(mar/24)*	□ Ter alimentação auxiliar para não cair (bateria) □ Ter redundância alimentação □ Em condições normais, evitar consumir a rede para não prejudicar QoS da rede (taxa perda < 20%) □ Estimador de QoS □ WDT das MUs

2.4 – Elementos de projeto

2.4.1 – Módulo de medição

A arquitetura básica do hardware de medição é registrada na figura da sequência. Observe a presença de 2 fontes para medição.

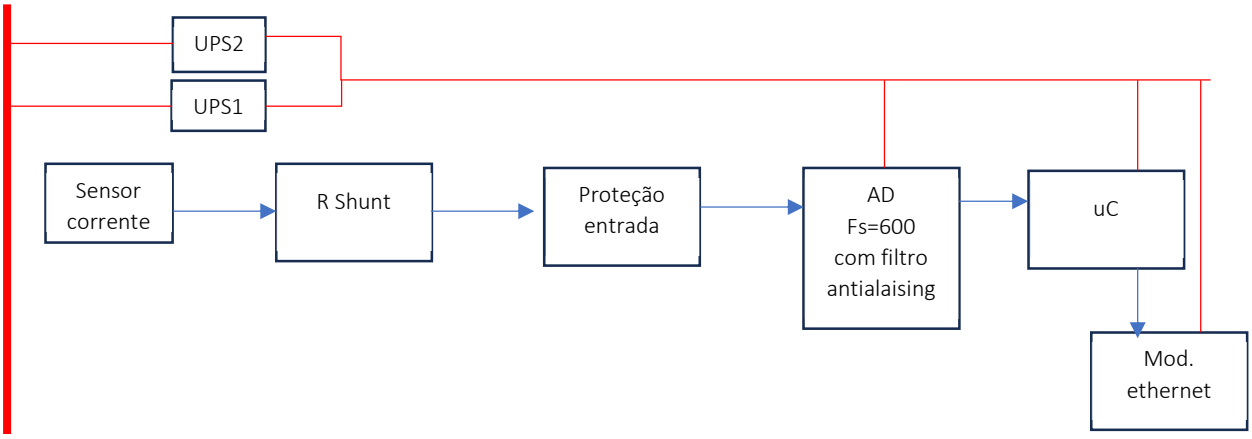

Figura 2.4.1.1: *Elementos do hardware*

Para cada um destes elementos, tem-se algumas referências de implementação ou "aplications notes". Estes são mostrados na tabela da sequência.

2.4.2 – Máquina de estados

Baseado nos cenários identificados e requerimentos construídos, tem-se a seguinte proposição para a máquina de estados de uma MU.

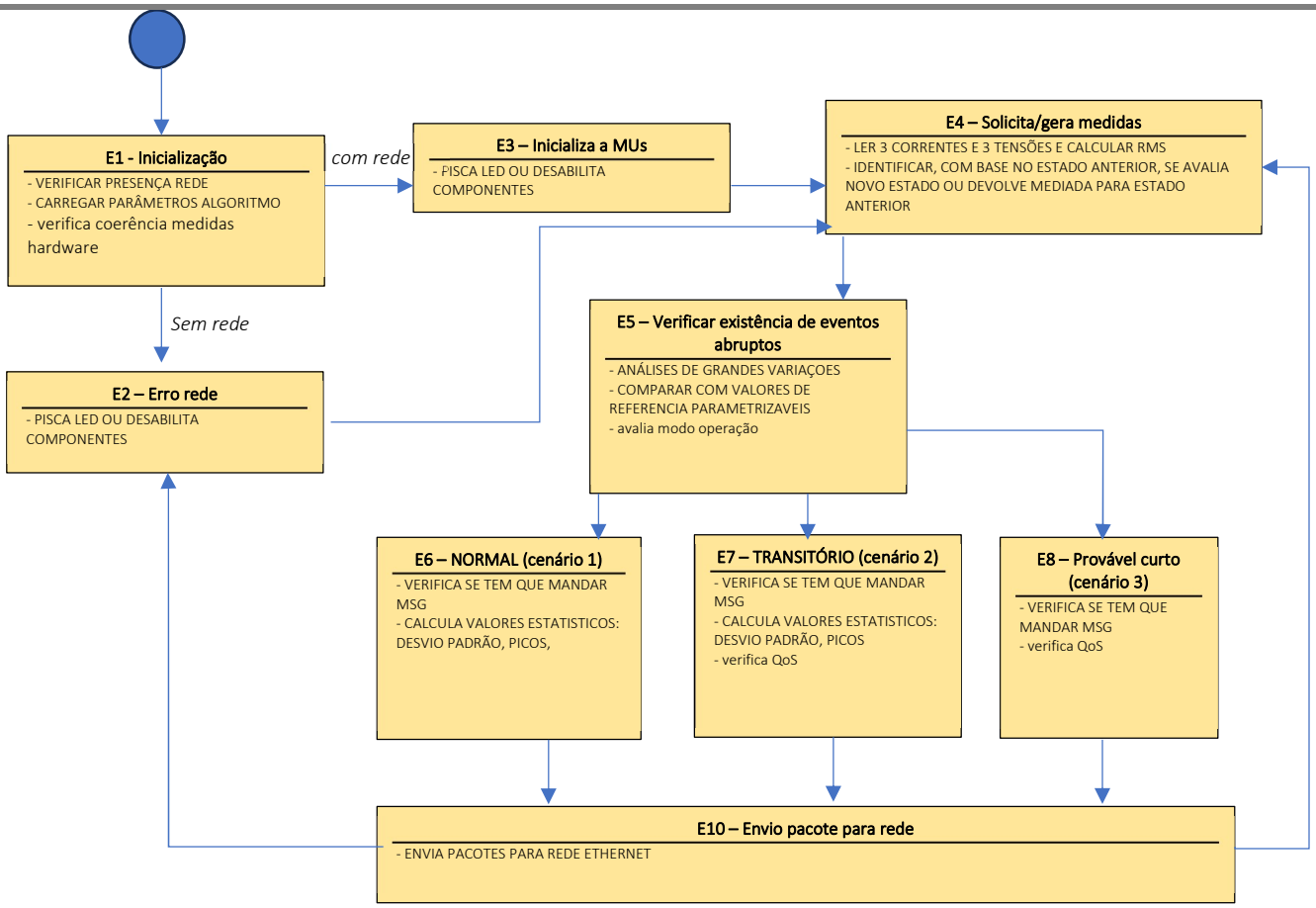

Figura 2.4.2.1: *Máquina de estados de uma MU.*

Todavia, há de se lembrar que estas MUs são executadas dentro de uma aplicação que simula diferentes MUs. Logo, o sistema MU também deve ter estes recursos:

a) Criação dinâmica de MUs
b) Recepção de dados broadcast da rede (estimar o QoS)
c) Estimador de QoS e métricas de tempo
d) Whatch dog timer das MUs
e) Avaliação de disponibilidade de rede

Tais recursos devem ser objeto de modelagem posterior.

2.4.3 – Interfaces de usuário

Para fins de caracterização do sistema, a figura na sequência ilustra a interface desta aplicação indicando alguns de seus recursos previstos.

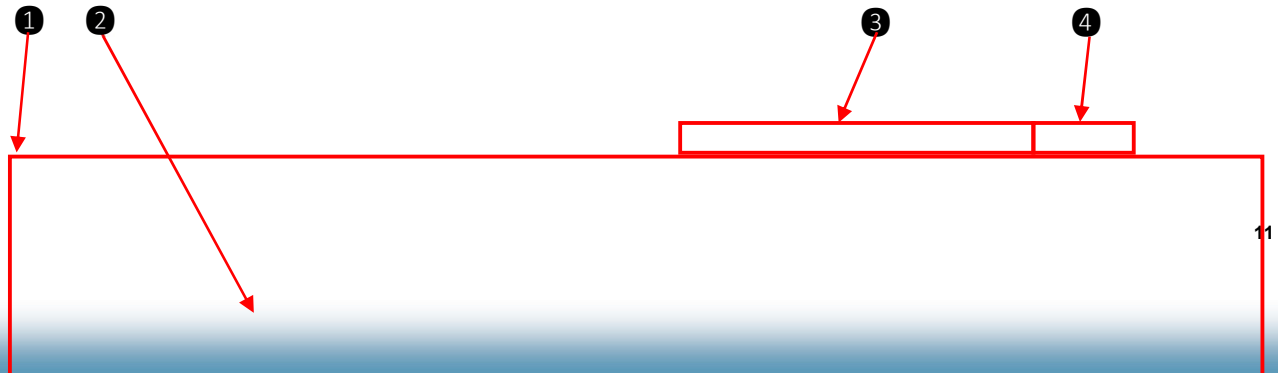

Figura 1.x: *Interface principal da aplicação*

Na sequência, uma breve descrição destes principais elementos:

1. xxxx
2. xxxxx.
3. xxxxx.
4. xxx
5. xxxx
6. xxxxx
7. xxxx
8. xxxx

2.5 – Interfaces de comunicação e infra de TIC

O sistema poder ter duas interfaces de rede que podem ser usadas para dois cenários: (i) redundância ou (ii) cada interface alimenta separadamente um grupo de IEDs separados por VLANs.

No último caso, vale lembrar que o tráfego produzido pela interface 1 não chega aos dispositivos que estão ligados a VLAN 2. Logo, os tráfegos são separadas pelas VLANs, evitando degradação de QoS. Ao mesmo tempo, damos ao MU a capacidade de operar com múltiplos MUs e múltiplas medições elétricas dentro de uma SE, porém em um mesmo dispositivo, se observada a arquitetura definida na figura da sequência.

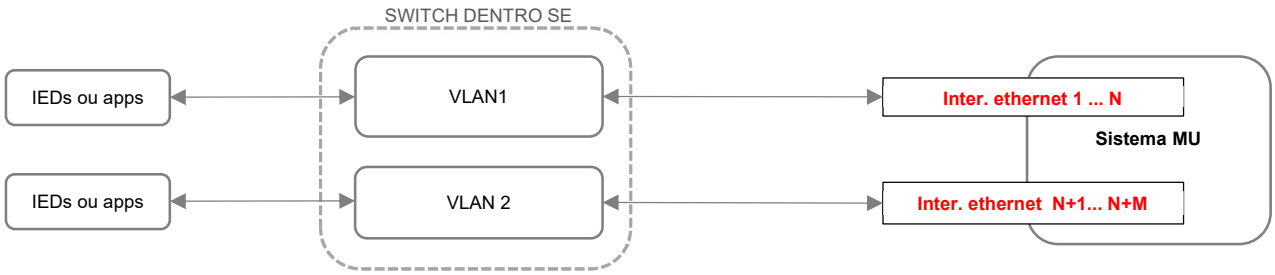

Figura 2.5.1 – Ligação básica das interfaces de rede do sistema MU.

3 – Modelagem

3.1 – Blocos de elementos principais

Na sequência é mostrado um conjunto de diagramas de blocos para exemplificar a arquitetar do sistema. Cada bloco é um objeto e estes são os principais objetivos previstos na solução. As setas indicam o fluxo das informações.

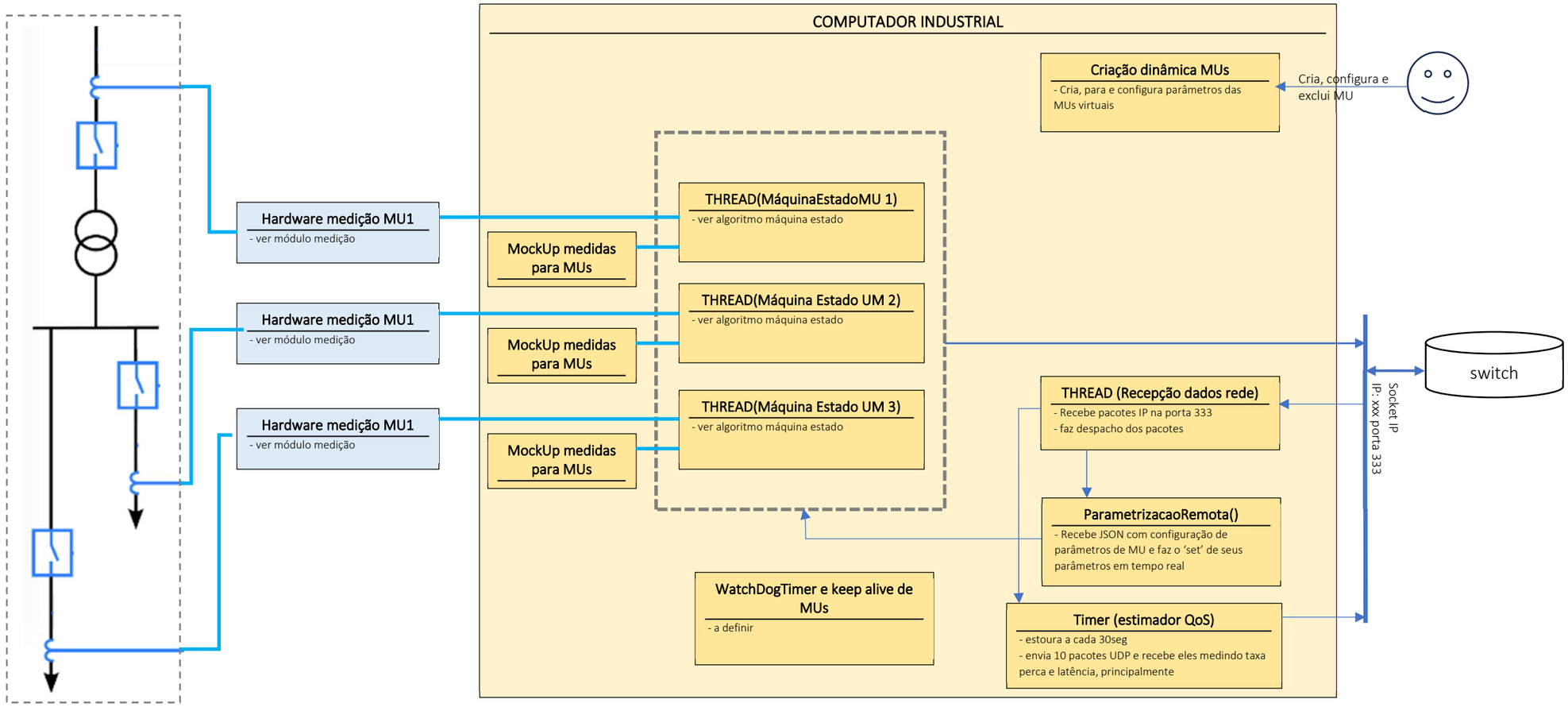

Figura 1.2.1: Componentes básicos do Digital Twins.

3.2 – Tabela geral de objetos IPSO e recursos de URI

Todos os recursos previstos entre as aplicações do TWINS são feitos por códigos IPSO[1] em padrão URI. Eles podem ser a designação de uma conexão para um evento, o formato de um pacote, o formato de um JSON específico dentre outras coisas. Ajudam a racionalizar e identificar facilmente o recurso e a estruturar o pensamento. Estes códigos estão dispostos na tabela da sequência.

Tabela x.: *Tabela IPSO de recursos do projeto.*

Objeto	Recurso		Significado	Link
	Nível 1	Nível 2		
99 *Mensagens com medidas*	**1** Medidas periódicas		Medidas elétricas trafegando na rede quando o MU está operando em condições normais.	
	2 Medidas em situação alteração		Medidas elétricas trafegando na rede quando o MU identificou uma grande mudança nos parâmetros e ai começa a transmitir mensagens com maior frequência.	
100 *Controle e configuração*	**10** Controle do protocolo MU		Mensagens onde um outro programa pode controlar alguns recursos do protocolo de medição do MU.	
	11 Configura parametrização MU		Mensagens onde um outro programa pode alterar remotamente alguns parâmetros do algoritmo do MU.	

3.3 – Fluxo geral de mensagens

A figura na sequência ilustra resumidamente as mensagens que são trocadas, em diferentes circunstâncias, entre o sistema MU e demais módulos.

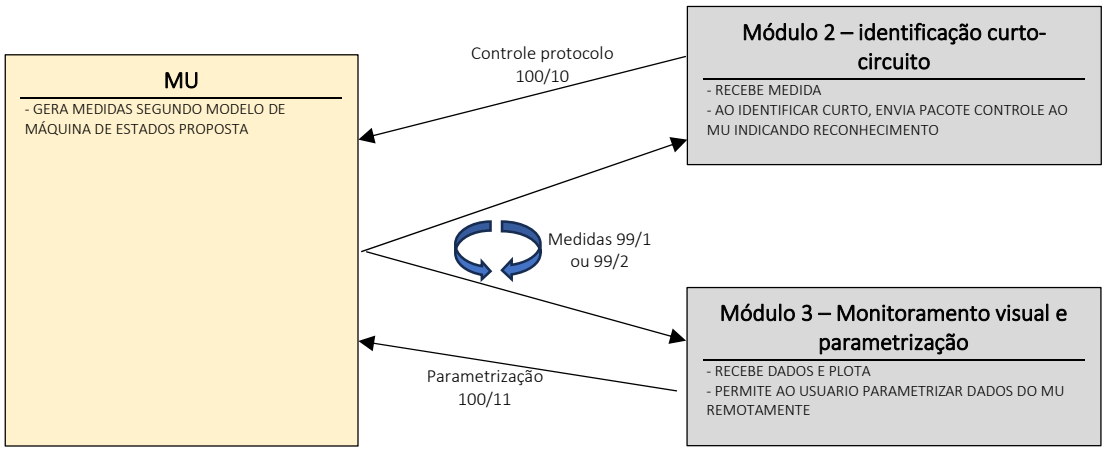

Figura 3.3.1: Diagrama ilustrando a ideia de como o front tem suas informações atualizadas dinamicamente

[1] https://www.avsystem.com/blog/IPSO-smart-objects/

Como observado, o MU envia para a rede em regime de broadcasting o pacote 99/1 (quando ele não percebe nenhuma alteração na rede significativa) ou o 99/2 (quando ele percebe uma modificação nos parâmetros elétricos e envia as medidas mais rapidamente).

Também, o MU recebe informações de outras aplicações. Uma delas é pacote de controle (pacote 100/10) que pode ser enviado por outro módulo solicitando a ele alguma mudança no seu comportamento de envio de mensagens, como por exemplo um comando avisando que ele pode parar de enviar 99/2 porque o evento já foi reconhecido e não faz mais sentido enviar tantos pacotes para a rede. Uma segunda opção seria um pacote de parametrização onde outra aplicação daria a oportunidade de configurar os parâmetros do algoritmo (usando o pacote 100/11) do MU remotamente por um operador.

Para entender melhor esta dinâmica, elas são tratadas individualmente com mais detalhes nas próximas subseções.

3.4 – Modelagem detalhada dos recursos

3.4.1 – Envio de mensagens de medição

A figura na sequência ilustra como deve ser o comportamento temporal da dinâmica de envio de pacotes de medidas elétricas do MU em dois diferentes cenários: (i) sem mudanças abruptas nas variáveis elétricas medidas e (ii) com mudanças abruptas.

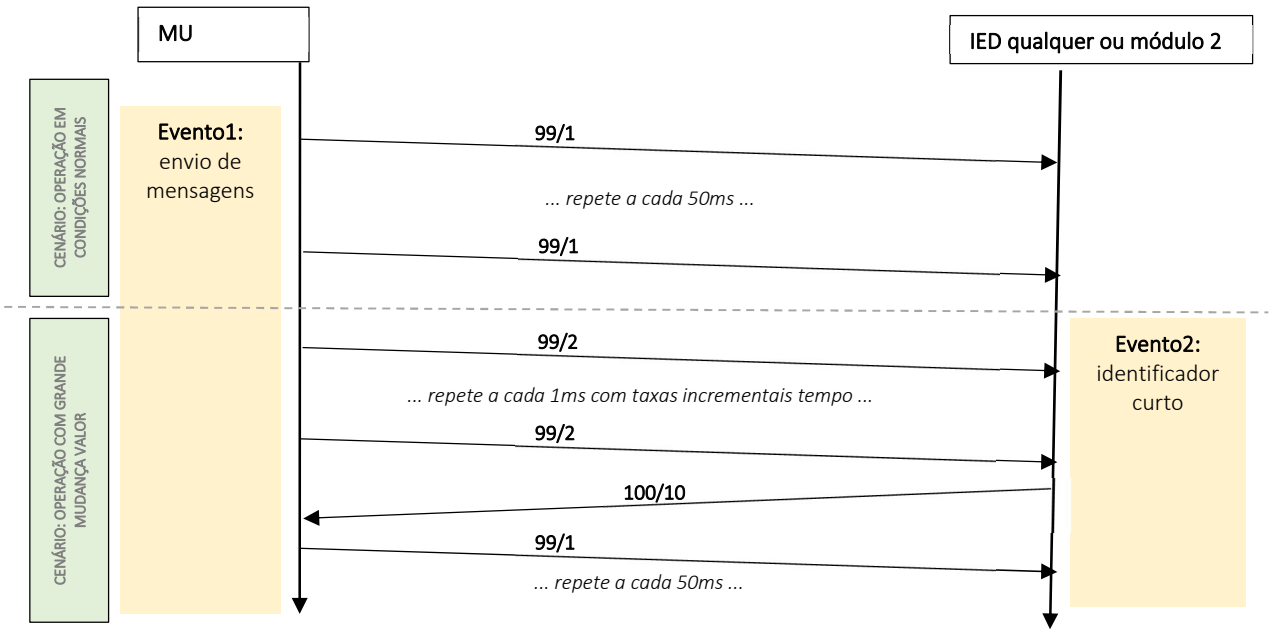

Figura 3.4.1.1: Diagrama pacotes e eventos associados ao envio de mensagens.

Com base no diagrama mostrado na figura anterior, descreve-se agora o comportamento dos principais eventos identificados.

Tabela 3.4.1.1: *Algoritmos equivalentes aos eventos vistos na Figura 3.4.1.1.*

Evento/Ação	Algoritmo

Evento 1 (envio de mensagens)	**Este evento deve implementar a máquina de estado mostrada na figura Figura 2.4.2.1. Algoritmo básico:** 1) Switch(MáquinaEstadoMU.Estado) a. Case Estado.E1: i. Implementar aqui os comandos vistos no estado E1 da Figura 2.4.2.1 ii. MáquinaEstadoMU.Estado = Estado.E2 (ou outro?) b. Case Estado.E2: i. Implementar aqui os comandos vistos no estado E1 da Figura 2.4.2.1 ii. MáquinaEstadoMU.Estado = Estado.E3 (ou outro?) .... c. Case Estado.E3: i. Implementar aqui os comandos vistos no estado E1 da Figura 2.4.2.1 ii. MáquinaEstadoMU.Estado = Estado.E1 (ou outro?) d.

Já o formato do pacote de dados é mostrado na sequência.

Tabela 3.4.1.2: *Formato do pacote de dados 99/1*

Campo	Valores	Significado
URI	**99/1**	Pacote de envio de medidas regulares
idMU	int	Identificador de qual MU está gerando este pacote
idAtivo	String (opcional)	Identificado de qual ativo o ID está monitorando.
numPct	inteito	Número do pacote gerado incrementalmente
timpeStamping	Data UMC	Data e hora em que foi gerado o pacote
freqEnvioMS	int	Taxa de aquisição e intervalo de tempo em que foram coletas as medidas e calculada a media.
medidas	[MedidasEletricas]	Contem um array com objetos do tipo "MedidasEletricas" **médias** que tem o seguinte formato: fase: String (A,B,C,N) tensao: float corrente: float angTensao: Float potApaVA: float potReatVAr: float potRealW: Float fatorP: float freq: float

E nas ocasiões onde acontecem grandes variações, gera-se no lugar do 99/1 o pacote 99/2 que tem mais informações e deve ter mais prioridade na rede.

Tabela 3.4.1.3: *Formato do pacote de dados 99/2*

Campo	Valores	Significado
URI	**99/2**	Pacote de envio de medidas urgentes com grande variação
idMU	int	Identificador de qual MU está gerando este pacote
idAtivo	String (opcional)	Identificado de qual ativo o ID está monitorando.
numPct	inteito	Número do pacote gerado incrementalmente
timpeStamping	Data UMC	Data e hora em que foi gerado o pacote
medidas	[MedidasEletricas]	Contem um array com objetos do tipo "MedidasEletricas" **instantâneas** que tem o seguinte formato: fase: String (A,B,C,N) tensao: float corrente: float angTensao: Float potApaVA: float potReatVAr: float potRealW: Float fatorP: float freq: float
variavelDescrepante	[String]	Indica qual é a variável que sofreu a grande variação e acarretou a ocorrência deste pacote.
faseDescrepante	[String]	Indica qual é a fase que sofreu a grande variação e acarretou a ocorrência deste pacote.

Temos também o pacote de controle que a MU pode receber e tomar alguma providência, segundo sua máquina de estados. Seu formato de dados é visto na sequência.

Tabela 3.4.1.4: *Formato do pacote de dados 100/10*

Campo	Valores	Significado
URI	**100/10**	Pacote de envio de controle do protocolo de medição.
idMU	int	Identificador de qual MU está gerando este pacote
idAtivo	String (opcional)	Identificado de qual ativo o ID está monitorando.
comando	int	Indica o comando de uma aplicação terceira está pedido para a MU executar. • 0 = indica que o objeto foi identificado e que ele pode voltar a sua condições de normalidade ou que providencias de tratamento de eventos já foram tomadas. • 1 = incrementar taxa de envio • 2 = decrementar taxa de envio segundo o valor do campo "parametro1"
parametro1	int	Usando para indicar algum parâmetro numero de algum comando.
parametro2	string	Usando para indicar algum parâmetro texto de algum comando.

3.4.2 – Parametrização remota

A figura na sequência ilustra como deve ser a parametrização remota.

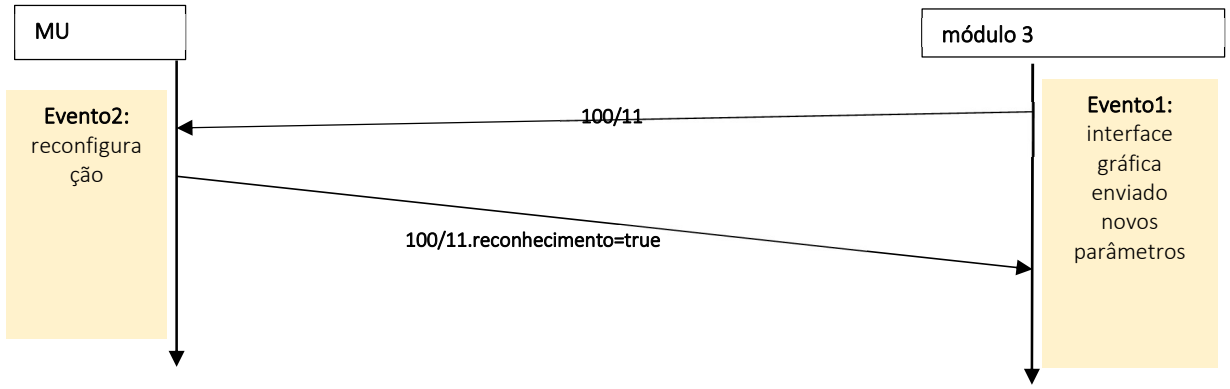

Figura 3.4.2.1: Diagrama pacotes e eventos associados à reconfiguração do MU

Com base no diagrama mostrado na figura anterior, descreve-se agora o pacote de reconfiguração.

Tabela 3.4.2.1: *Algoritmos equivalentes aos eventos vistos na Figura 3.4.1.1.*

Evento/Ação	Algoritmo
Evento 2 (envio de mensagens)	1) monta pacote 100/11 2) Verifica chegada de confirmação a. Aguarda 100ms b. Se pacotesEnviados < 3 i. Envia novo pacote ii. Incrementa "pacotesEnviados++" c. Senao i. Parametrização falhou. Mostra na interface gráfica falha da parametrizacao

Já o formato do pacote de dados é mostrado na sequência. Como pode ser visto, a principal limitação é que para cada parâmetro deve ser enviado 1 pacote. Uma modificação seria tornar os campos "parametro" e "valor" como vetores, permitindo assim múltiplas parametrizações em um único pacote.

Tabela 3.4.1.2: *Formato do pacote de dados 100/11*

Campo	Valores	Significado
URI	**100/11**	Pacote de envio de parametrização da MU
idMU	int	Identificador de qual MU está gerando este pacote

parametro	String	Nome do parâmetro
valor	float	Novo valor que a variável contida em “parâmetro” deve valer.
reconhecimento	bool	É o ACK da operação