



UNIVERSIDADE FEDERAL DO CEARÁ
CENTRO DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA
MESTRADO ACADÊMICO EM ENGENHARIA ELÉTRICA

KEVIN RABELO COSTA

**SISTEMA DE MONITORAMENTO E GERENCIAMENTO DE ENERGIA BASEADO
NO PROTOCOLO MQTT PARA UNIDADES COM GERAÇÃO DISTRIBUÍDA**

FORTALEZA

2021

KEVIN RABELO COSTA

SISTEMA DE MONITORAMENTO E GERENCIAMENTO DE ENERGIA BASEADO NO
PROTOCOLO MQTT PARA UNIDADES COM GERAÇÃO DISTRIBUÍDA

Dissertação apresentada ao Curso de Mestrado Acadêmico em Engenharia Elétrica do Programa de Pós-Graduação em Engenharia Elétrica do Centro de Tecnologia da Universidade Federal do Ceará, como requisito parcial à obtenção do título de mestre em Engenharia Elétrica. Área de Concentração: Engenharia Elétrica

Orientador: Prof. Dr. Fernando Luiz
Marcelo Antunes

Coorientador: Prof. Dr. Edilson Mineiro
Sá Júnior

FORTALEZA

2021

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Biblioteca Universitária
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

C873s Costa, Kevin Rabelo.
SISTEMA DE MONITORAMENTO E GERENCIAMENTO DE ENERGIA BASEADO NO
PROTOCOLO MQTT PARA UNIDADES COM GERAÇÃO DISTRIBUÍDA / Kevin Rabelo Costa. – 2021.
79 f. : il. color.

Dissertação (mestrado) – Universidade Federal do Ceará, Centro de Tecnologia, Programa de Pós-Graduação em Engenharia Elétrica, Fortaleza, 2021.

Orientação: Prof. Dr. Fernando Luiz Marcelo Antunes.

Coorientação: Prof. Dr. Edilson Mineiro Sá Júnior.

1. Sistemas de Monitoramento. 2. Internet das Coisa. 3. Casas Inteligentes. 4. Redes Inteligentes. 5. Eficiência Energética. I. Título.

CDD 621.3

KEVIN RABELO COSTA

SISTEMA DE MONITORAMENTO E GERENCIAMENTO DE ENERGIA BASEADO NO
PROTOCOLO MQTT PARA UNIDADES COM GERAÇÃO DISTRIBUÍDA

Dissertação apresentada ao Curso de Mestrado Acadêmico em Engenharia Elétrica do Programa de Pós-Graduação em Engenharia Elétrica do Centro de Tecnologia da Universidade Federal do Ceará, como requisito parcial à obtenção do título de mestre em Engenharia Elétrica. Área de Concentração: Engenharia Elétrica

Aprovada em:

BANCA EXAMINADORA

Prof. Dr. Fernando Luiz Marcelo
Antunes (Orientador)
Universidade Federal do Ceará (UFC)

Prof. Dr. Edilson Mineiro Sá Júnior (Coorientador)
Instituto Federal de Educação, Ciência e Tecnologia
do Ceará (IFCE)

Prof. Dr. Bruno Ricardo de Almeida
Universidade de Fortaleza (UNIFOR)

Prof. Dr. Fabricio Gonzalez Nogueira
Universidade Federal do Ceará (UFC)

“Aos meus pais, pelos valores ensinados e por mesmo na dificuldade, terem investido na minha educação”.

AGRADECIMENTOS

À Deus, por me propiciar a cada manhã uma nova oportunidade para ser melhor.

À minha família, por ter sido minha base durante essa caminhada, e por me reerguer nos momentos mais difíceis. Aos meus pais, Vladimir e Lêda, que são exemplos de valor e honestidade, e que mesmo com a dificuldade não mediram esforços para me proporcionar o que sou hoje. Aos meus irmãos, Saulo, Dylan e Ellen, por serem companhia, carinho e aprendizado.

À minha namorada, Dulce, por caminhar ao meu lado e por todo o incentivo, paciência e carinho durante a elaboração desse trabalho.

Ao Professor Fernando Antunes, pela oportunidade dada em desenvolver esse trabalho. Através dele venho adquirindo conhecimentos que levarei pra minha vida profissional e se Deus quiser, pro mundo. Agradeço também pela solicitude e acompanhamento durante esse trabalho.

Aos colegas do Departamento de Engenharia Elétrica, que de alguma forma, incentivaram ou contribuíram para o desenvolvimento desse trabalho. Algo que aprendi, e que levo pra vida é que não existe ciência se não houver discussão e aprovação da comunidade científica.

Ao povo brasileiro, que através do seu esforço financia toda a estrutura de pesquisa nas Universidades Públicas.

À todos os citados, o meu muito obrigado! Este trabalho não existiria se não fossem por vocês.

“I wish it need not have happened in my time”,
said Frodo. “So do I”, said Gandalf, “and so do
all who live to see such times. But that is not for
them to decide. All we have to decide is what to
do with the time that is given us.”

(J.R.R. Tolkien, *The Fellowship of the Ring*)

RESUMO

No Brasil, a geração distribuída foi implantada através da RN nº 482/2012, trazendo a possibilidade do consumidor gerar sua própria energia e injetar o excedente na rede. Hoje, nove anos depois, cerca de 80% da geração distribuída instalada no Brasil é feita de pequenas usinas de até 75 kWp. No entanto, o medidor bidirecional ao qual essas usinas são conectadas contabiliza apenas a diferença instantânea entre o que é consumido e o que é gerado, deixando o consumidor impossibilitado de acompanhar o consumo real da sua unidade através da fatura de energia. Vendo essa problemática, este trabalho propõe um sistema de monitoramento e gerenciamento para unidades com geração distribuída, capaz de apresentar dados em tempo real de consumo e geração, além de realizar o acionamento remoto de cargas. É proposto um hardware de medição independente baseado no módulo PZEM-004T e no microcontrolador ESP32, que envia os dados via WiFi utilizando o protocolo de comunicação leve MQTT. Os dados são recebidos em uma aplicação web, implementada utilizando o framework Ruby on Rails, onde são exibidos ao usuário e armazenados em um banco de dados PostgreSQL. O cenário da geração distribuída no Brasil e no mundo é apresentado, seguido de um estudo a cerca dos sistemas de medição inteligente e sua contribuição no novo paradigma da rede. O sistema desenvolvido foi instalado em um laboratório na cidade de Fortaleza, onde foi avaliado acerca da precisão e desempenho entregue ao usuário. O sistema proposto foi capaz de fornecer a informação necessária para identificar um potencial de redução na unidade e a necessidade de aumento na potência do gerador fotovoltaico instalado.

Palavras-chave: Sistemas de Monitoramento. Internet das Coisas. Casas Inteligentes. Redes Inteligentes. Eficiência Energética

ABSTRACT

In Brazil, the distributed generation was implanted in 2012 through the RN n° 482, bringing the possibility to the customer to generate your own energy and inject the excess on the grid. Today, nine year after, almost 80% of all the distributed generation installed is made of small power plants until 75 kWp. However, the bidirectional meter installed in these units accounts only for the instantaneous difference between what is consumed and generated, leaving the customer unable to track his real energy consumption through the energy bill. To address this issue, this work proposes a monitoring and management system for units with distributed generation, capable of presenting real time consumption and generation measures, besides load control. It is proposed an independent measurement hardware based on the PZEM-004T module and on ESP32 microcontroller, who sends data through WiFi using the lightweight MQTT protocol. The data are received in a web application, designed using the Ruby on Rails framework, where are exhibited to the user and stored in a PostgreSQL database. The distributed generation in Brazil and in the world is presented, followed by a study about the smart meters and its role on the new grid paradigm. The developed system was installed in a study laboratory in Fortaleza, where it was availed in precision and performance to the user. The proposed system was capable of bringing up a potential energy saving and showing a need to improve the photovoltaic generation capability.

Keywords: Monitoring Systems. Internet of Things. Smart Homes. Smart Grids. Energy Efficiency

LISTA DE FIGURAS

Figura 1 – Comparativo da tarifa branca e da tarifa convencional para a concessionária ENEL CE	22
Figura 2 – Fluxo da informação entre o broker MQTT e dispositivos conectados	29
Figura 3 – Estrutura de transmissão de dados no broker AMQP	30
Figura 4 – Visão macro do sistema proposto	32
Figura 5 – Protótipo experimental desenvolvido	34
Figura 6 – Arquitetura do sistema proposto	35
Figura 7 – Estrutura de dados da string enviada ao broker MQTT	37
Figura 8 – Estrutura de tópicos criados no broker MQTT	38
Figura 9 – Fluxograma do firmware desenvolvido	38
Figura 10 – Ciclo da resposta a requisição na arquitetura MVC	40
Figura 11 – Página criada para o acionamento das cargas	41
Figura 12 – Vista da aplicação web desenvolvida	42
Figura 13 – Protótipos instalados para teste experimental	46
Figura 14 – Curva de consumo da unidade em dias úteis	47
Figura 15 – Curva de consumo da unidade durante os fins de semana	47
Figura 16 – Energia gerada no sexto dia do período de teste	47
Figura 17 – Potência máxima do sistema fotovoltaico no sexto dia do período de teste	48
Figura 18 – Demanda registrada durante o período de teste	48
Figura 19 – Potência máxima do sistema fotovoltaico durante o período de teste	49
Figura 20 – Geração registrada durante todo o período de teste	50
Figura 21 – Consumo registrado durante todo o período de teste	50
Figura 22 – Variação de tensão ao longo do período de teste	51
Figura 23 – Lógica de acionamento da carga, com horário das mensagens recebidas	52
Figura 24 – Gráfico de corrente representando o acionamento da carga	53

LISTA DE TABELAS

Tabela 1 – Precisão do módulo PZEM-004T utilizado.	33
Tabela 2 – Comparação das medições obtidas pelo sistema de medição proposto e o multímetro Fluke 177	44
Tabela 3 – Custo associado a cada um dos componentes do medidor inteligente proposto.	53

LISTA DE ABREVIATURAS E SIGLAS

GD	Geração Distribuída
RN	Resolução Normativa
CPF	Cadastro de Pessoa Física
RO	<i>Renewable Obligation</i>
ANEEL	Agência Nacional de Energia Elétrica
PLC	<i>Power Line Carrier</i>
RTC	<i>Real Time Clock</i>
IoT	<i>Internet of Things</i>
GPRS	<i>General Packet Radio Services</i>
BT	Baixa Tensão
UART	<i>Universal Asynchronous Receiver/Transmitter</i>
NTP	<i>Network Time Protocol</i>

LISTA DE SÍMBOLOS

E_G	Energia gerada expressa em kWh/dia
HSP	Horas de Sol Pleno expresso em kWh/m ² .dia
P_{FV}	Potência do sistema fotovoltaico expresso em kWp
TD	Taxa de desempenho do sistema fotovoltaico

SUMÁRIO

1	INTRODUÇÃO	15
2	REVISÃO BIBLIOGRÁFICA	18
2.1	Geração Distribuída	18
2.1.1	<i>Panorama Mundial</i>	18
2.1.2	<i>Panorama Nacional</i>	20
2.2	Tarifação	21
2.2.1	<i>Grupo B</i>	22
2.2.2	<i>Grupo A</i>	23
2.3	Medição Inteligente	24
2.3.1	<i>Medidor Inteligente</i>	25
2.3.2	<i>Transmissão de dados</i>	26
2.3.3	<i>Servidor</i>	27
2.4	Protocolos de Comunicação	27
2.4.1	<i>Constrained Application Protocol (CoAP)</i>	27
2.4.2	<i>Message Queue Telemetry Transport (MQTT)</i>	28
2.4.3	<i>Advanced Message Queuing Protocol (AMQP)</i>	29
2.4.4	<i>Extensible Messaging and Presence Protocol (XMPP)</i>	30
2.5	Conclusão	30
3	SISTEMA PROPOSTO	32
3.1	Medidor inteligente desenvolvido	33
3.1.1	<i>Hardware de medição</i>	33
3.1.2	<i>Firmware desenvolvido</i>	35
3.2	Aplicação web desenvolvida	39
3.3	Conclusão	42
4	RESULTADOS	44
4.1	Teste de precisão do medidor proposto	44
4.2	Resultados experimentais	45
4.3	Validação do sistema de acionamento remoto	52
4.4	Custos do Sistema Desenvolvido	53
4.5	Conclusão	54

5	CONCLUSÕES E TRABALHOS FUTUROS	56
5.1	Trabalhos Futuros	57
	REFERÊNCIAS	59
	APÊNDICES	63
	APÊNDICE A – CÓDIGOS-FONTES UTILIZADOS NO HARDWARE	
	DE MEDIÇÃO	63
	ANEXOS	80

1 INTRODUÇÃO

Desde os primórdios, em que a energia elétrica era utilizada para iluminação até a atualidade em que surgem os veículos elétricos, sempre houve a necessidade de se mensurar e controlar a distribuição desse recurso. Pela característica indispensável que a energia tem para a sociedade, a medição dela vai mais além do que apenas fins de tarifação, tendo também impactos na provisão e na qualidade do fornecimento, além de influenciar diretamente no custo da energia elétrica.

Os primeiros medidores amplamente utilizados foram os eletromecânicos, que apesar de ainda serem vistos atualmente em unidades residenciais, possuem uma série de limitações como: a baixa precisão, a variação dos resultados de acordo com a temperatura e a medição apenas da energia ativa e unidirecional (GARCIA *et al.*, 2017). Além disso, existe ainda a necessidade de profissionais da concessionária de energia apenas para realizar a obtenção dos dados todos os meses, o que resulta em um aumento no custo de operação e eventualmente em erros de medição.

No cenário atual em que a demanda por energia elétrica vem crescendo e os consumidores são permitidos a gerar sua própria energia, existe, claramente, a necessidade por uma melhoria nos sistema de medição, e é nesse contexto, que a medição inteligente surge como alternativa. Segundo Garcia *et al.* (2017), que colabora com o que é dito por Barai *et al.* (2015), o medidor inteligente possui três características que não são vistas nos medidores convencionais, como: a possibilidade de enviar medições em alta frequência; a transferência de dados bidirecional, seja entre consumidor e medidor ou concessionária e medidor; e a publicação regular de dados, ou seja, o consumidor acompanha o crescimento da sua fatura em tempo real, sem a necessidade de aguardar até o fechamento da fatura.

No Brasil, a Geração Distribuída (GD) foi regulamentada pela Resolução Normativa (RN) nº 482/2012, sendo revisada ainda pela RN nº 687/2015 e pela RN nº 786/2017. Através destas, foram estabelecidas as condições gerais para o acesso a rede e para o sistema de compensação de energia. De forma geral, as unidades são divididas em microgeração, com limite de até 75 kWh e minigeração com limite de até 5 MW. Segundo dados da ANEEL (2020a), em Setembro de 2020 usinas com até 75 kWh representavam cerca de 80% de toda a GD no Brasil. Para realizar a conexão dessas usinas com a rede distribuição, o medidor unidirecional, na maior parte dos casos ainda eletromecânico, é substituído por um medidor bidirecional digital, que mede a diferença instantânea entre o que é consumido e o que é gerado na unidade. Nesse

modelo, em momentos em que é consumido mais do que é gerado é contabilizado consumo de energia, e inversamente, quando é gerado mais do que é consumido é contabilizada energia injetada na rede. No fechamento da fatura, a energia injetada é abatida do consumo e o cliente paga apenas a diferença entre esses dois valores. Caso, a energia injetada seja maior do que a energia consumida, o consumidor recebe créditos de energia que podem ser descontados nas próximas faturas ou no consumos de outras unidades cadastradas no mesmo Cadastro de Pessoa Física (CPF) do titular e na mesma zona de atuação da concessionária. Os créditos gerados podem ser utilizados pelo prazo de até 60 meses, antes de expirarem (ANEEL, 2015a).

Apesar da atualização dos medidores eletromecânicos para medidores eletrônicos, o que já representa um ganho na precisão das medições, os novos medidores bidirecionais não são inteligentes, o que significa que ainda é necessário um profissional da concessionária para o recolhimento das medições, além do cliente só receber os dados da medição dias antes de pagar a fatura. Não obstante, com a mudança do medidor unidirecional para o bidirecional, o consumidor fica impossibilitado de saber o consumo e a geração real da sua unidade, uma vez que o valores apresentados na fatura de energia representam apenas as medições diferenciais desses dois valores. O problema se intensifica uma vez que no Brasil 79,5% (ANEEL, 2017b) das unidades com GD estão em residências, onde o usuário não tem, muitas vezes, a ciência para interpretar esses resultados na fatura de energia.

Vendo essa necessidade, este trabalho desenvolve um sistema de monitoramento e gerenciamento de baixo custo para unidades com GD. O sistema proposto provê ao usuário dupla medição unidirecional para consumo e geração, independente do medidor da concessionária, além de permitir o acionamento remoto de cargas e informar ao usuário problemas como sobretensão, subtensão e falta de energia. O sistema de medição é baseado no medidor PZEM-004T e no microcontrolador ESP32 que recebe e envia os dados via WiFi para um *broker* MQTT. Os dados são então redirecionados e exibidos através de gráficos, atualizados em tempo real, ao usuário, em uma aplicação web desenvolvida no *framework* Ruby on Rails, e salvos em um banco de dados PostgreSQL. O uso do protocolo MQTT garante uma maior confiabilidade da informação e possibilidade de operação mesmo em redes de baixa velocidade.

O artigo desenvolvido a partir desse trabalho foi publicado no 13º *Seminar on Power Electronics and Control* (SEPOC), em formato virtual, sob o título *MQTT based Monitoring and Management System for Distributed Generation Units*.

O restante desse trabalho foi organizado como: o Capítulo 2 apresenta uma revisão

bibliográfica a cerca dos temas de interesse desse trabalho como a GD no Brasil e no Mundo, e os protocolos de comunicação de dados; O capítulo 3 apresenta os detalhes do sistema proposto; enquanto o Capítulo 4 apresenta os resultados experimentais obtidos além de traçar a curva de carga da unidade em que o sistema foi instalado; finalmente, o Capítulo 5 apresenta a conclusões a cerca dos resultados obtidos.

2 REVISÃO BIBLIOGRÁFICA

Neste capítulo serão apresentados aspectos teóricos importantes para o entendimento do trabalho, bem como as regulamentações e normativas ao qual as unidades com geração distribuída estão submetidas no Brasil e no Mundo.

2.1 Geração Distribuída

Apesar da tendência de globalização da GD, cada país estabeleceu normas particulares de acesso a rede e de incentivo as energias renováveis. Desse modo, será apresentado a seguir um panorama da GD em nível mundial, sendo apontados também os resultados positivos obtidos pelos países apresentados.

2.1.1 *Panorama Mundial*

Em muitos países desenvolvidos a geração distribuída aconteceu como forma de reduzir a liberação de gás carbônico e da poluição de forma geral. Segundo Jones *et al.* (2010), a geração distribuída foi implantada inicialmente em países como França, Reino Unido, Alemanha e Estados Unidos. Nesses países o modelo de geração distribuída foi instituído através de incentivos governamentais como descontos em tarifas, acesso a rede de distribuição e a garantia de pagamento pela energia gerada e injetada na rede.

De forma geral, o protocolo de Kyoto foi um dos agentes impulsionadores da GD na Europa, uma vez que esses mesmos países se comprometeram a reduzir 20% de energia de fontes renováveis, reduzir em 20% o desperdício de energia e reduzir as emissões de carbono em 20% até 2020 (SIMOES *et al.*, 2011).

No caso da Alemanha, por exemplo, que possuía uma matriz predominantemente nuclear e termoelétrica, foram instituídos incentivos pra compra de equipamentos, isenção de taxas e a inclusão das tarifa feed-in, que garante compensação pela energia renovável gerada, normalmente em contratos de longo prazo. As políticas lançadas inicialmente em 1991, foram rearranjadas ainda em 2004 e novamente em 2008. De modo que, hoje existe ainda a prioridade de acesso a rede de distribuição, a compra integral pelo operador da rede de toda a energia produzida, e a cobertura dos custos extras da concessionária (JONES *et al.*, 2010). Todas essas políticas fizeram da Alemanha líder na Europa em produção de energia fotovoltaica, eólica e solar-térmica. Segundo dados do ISE (2021), no país, pela primeira vez em 2020, a produção

solar e eólica juntas ultrapassaram a soma de toda a energia produzida por combustíveis fósseis, um total de 183 TWh contra 178 TWh, respectivamente.

A exemplo da Alemanha outros países da Europa também lançaram políticas para implementação da GD. No Reino Unido, foi lançada em Janeiro de 2000 a meta para que em 2010, 10% de toda a energia gerada viesse de fontes renováveis, para atingir esse marco foi instituída a *Renewable Obligation (RO)* que obriga as concessionárias de energia a adquirir pelo menos 7,9% da sua energia de fontes renováveis (JONES *et al.*, 2010).

Já na Espanha, a GD foi instituída através de uma serie de normas começando pela Ordem 5/9/1985 que estabelecia regras para conexão e operação de usinas hidroelétricas de até 5 MVA até as mais recentes como a RD 1663/2000 e a RD 2818/1998 que ditam sobre a cogeração e a interconexão de usinas fotovoltaicas na rede de distribuição. Em baixa tensão é permitida o limite de conexão de 100 kVA ou 50% da capacidade da subestação ao qual o gerador vai ser instalado. Em média e alta tensão são permitidos geradores assíncronos de até 5000 kVA e síncronos de até 10 MVA. São estabelecidos ainda medidores independentes para geração e consumo (MORENO-MUNOZ *et al.*, 2009), além de limites para variação de frequência e tensão para a conexão com a rede.

Apesar da rápida disseminação da fontes de energia renovável e da geração distribuída na Europa ainda existem muitos desafios a frente para serem explorados. O ultimo acordo climático da União Europeia objetivou atingir a neutralidade na emissão de gás carbônico até 2050 (COMMISSION, 2021). Com essa mudança toda a frota de veículos deverá se tornar elétrica e toda a geração de energia vinda de fontes renováveis, o que aumentará a necessidade por um maior controle de demanda e uma maior distribuição da geração.

Nos Estados Unidos, com o crescimento esperado da demanda em 260 GW para 2030, investimentos vem sendo realizados em fontes de energia renovável, na captura de carbono e em tecnologias de eficiência energética (JONES *et al.*, 2010). De modo geral, espera-se em 2030 atingir a marca de 24% de geração em fontes renováveis de energia (SURYANARAYANAN *et al.*, 2010).

Segundo Jones *et al.* (2010), muitos estados americanos possuem políticas próprias em relação a GD. No estado da Califórnia, por exemplo, vários programas foram criados para a construção de usinas fotovoltaicas em casas e empresas. No estado da Flórida existem créditos disponíveis para consumidores e empresários que queiram construir sua própria usina, além de créditos em tarifas de energia.

De forma geral pode-se perceber que a maioria das políticas de incentivo a GD renderam resultados positivos, como o aumento do percentual de energias renováveis na matriz energética nacional e no aumento no número de usinas de forma geral.

2.1.2 Panorama Nacional

A GD foi inicialmente regulamentada no Brasil em 17 de abril de 2012, pela RN nº 482/2012. Lá são estabelecidos os requisitos para instalação e acesso da micro e mini geração nos sistemas de distribuição, bem como o sistema de compensação de energia. Com a RN nº 482/2012, a GD cresceu de forma exponencial no Brasil passando de 373 usinas no final de 2014 para um pouco mais de 480 mil em abril de 2021 (ANEEL, 2021).

Segundo a RN nº 482/2012, para realizar o acesso a rede de distribuição é necessário que unidades com micro e minigeração realizem a substituição do medidor unidirecional pelo bidirecional, sendo a concessionária de energia responsável técnica e financeiramente pela substituição. A instalação do medidor bidirecional possibilita a contabilização da energia injetada na rede, em momentos em que está sendo gerado mais do que é consumido, e do consumo em momentos em que é consumido mais do que é gerado. Dessa forma, ao final do mês a energia injetada é abatida do consumo registrado e o cliente paga apenas o excedente do que consumiu. Caso o consumidor compense todo o seu consumo, o excedente de energia injetada é convertido em créditos de energia, podendo ser compensados nas próximas faturas (ANEEL, 2012).

Em 24 de Novembro de 2015, a Agência Nacional de Energia Elétrica (ANEEL) autorizou várias melhorias na GD através da RN nº 687/2015. As novas regras ampliaram o tempo de utilização dos créditos de energia de 36 pra 60 meses e permitiu que estes fossem compensados em outras unidades do mesmo titular, contanto que estivessem em uma mesma distribuidora. No entanto, mesmo que o consumidor gere tudo o que consumido é necessário ainda o pagamento da taxa de disponibilidade, valor referente ao consumo de 30kWh para unidades monofásicas; 50kWh para unidades bifásicas e 100 kWh para unidades trifásicas.

A prática de gerar em uma unidade para compensar em outra, foi categorizado pela norma como Autoconsumo Remoto. Outra forma de utilização criada foi a Geração Compartilhada que permitiu que vários consumidores se unissem em cooperativas para instalar um sistema que compensasse cada unidade participante em porcentagens definidas pelos próprios consumidores. A normativa estabeleceu o limite de até 75 kW para microgeração e 5 MW para

minigeração, além de reduzir o prazo de conexão no sistema de distribuição em 34 dias (ANEEL, 2015a).

O procedimento específico para conexão a rede de distribuição é determinado pelo módulo 3 do PRODIST, da mesma forma que os requisitos mínimos de proteção da usina. Para usinas de até 75 kW, por exemplo, se faz necessário um elemento de desconexão e interrupção, relé de sincronismo, dispositivo anti-ilhamento, além de proteção contra tensões e frequências excessivamente baixas ou altas (ANEEL, 2017a).

Além da publicação da regulação normativa, o governo brasileiro tem incentivado a geração distribuída através da isenção de impostos. Equipamento como inversores e placas tem isenção total de PIS e Cofins e recentemente a isenção de impostos de importação (BRASIL, 2020).

O sistema de compensação de energia instituído pela RN nº 482/2012 é conhecido mundialmente como *net metering* e é utilizado em vários países como EUA, Alemanha e Índia, oferecendo uma forma vantajosa do consumidor acessar a rede. Do ponto de vista do sistema elétrico a inclusão da GD proporciona uma postergação de obras de ampliação da rede, a diminuição no uso de fontes poluentes e um menor carregamento das redes de transmissão. Contudo, o aumento no número de novas usinas abriu espaço para a discussão do sistema de compensação atual. Segundo a ANEEL, as tarifas de fornecimento da rede de distribuição não são cobertos pelo sistema atual, de forma que esse valor tem que ser dividido entre todos os consumidores, mesmo os que não possuem geração distribuída. A proposta da ANEEL é que as unidades com GD que solicitarem acesso após a aprovação da normativa passem a pagar pelo custo da rede (TUSD Fio A e B) e por encargos imediatamente, enquanto os consumidor que já possuem GD só passariam para a nova regra em 2030 (ANEEL, 2019). Até o momento da escrita desse trabalho, apesar de terem sido realizadas várias consultas públicas, não foi apresentado ainda um parecer final a respeito da alteração.

2.2 Tarifação

Mesmo após a substituição pelo medidor bidirecional, unidades com GD continuam sendo tarifadas de acordo com a classe de consumo e o grupo tarifário que pertencem. Dessa forma, o estudo da tarifação se mostra útil também nesse contexto, sendo capaz de prover uma redução no consumo de energia.

2.2.1 Grupo B

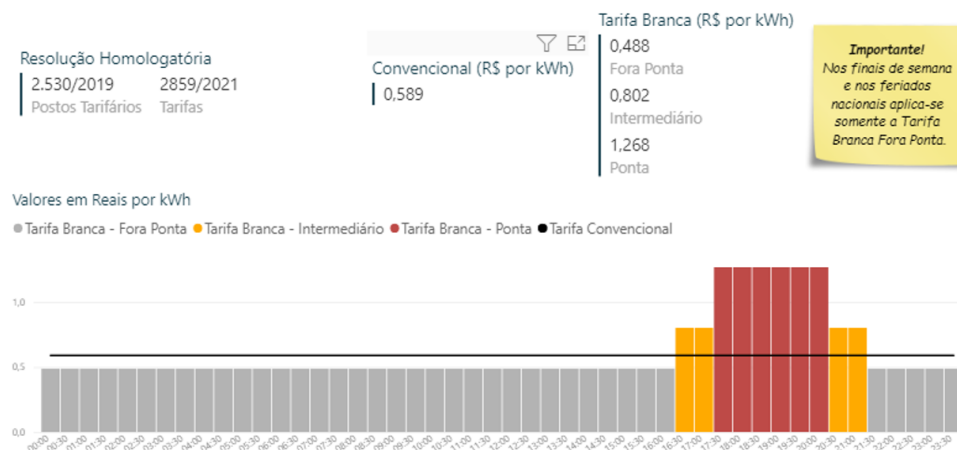
As unidades do grupo B são conectadas em baixa tensão e possuem essencialmente uma tarifa de energia monômnia, o que significa que o consumidor é tarifado unicamente pelo consumo de energia.

A partir de 1º de Janeiro de 2020 (ANEEL, 2020b), os consumidores com unidades do grupo B passaram a escolher entre duas modalidades distintas de tarifas: a convencional e a branca. Na tarifa convencional o usuário paga o mesmo valor de tarifa independente do horário do dia, ao passo que na tarifa branca são previstos três postos tarifários com diferentes tarifas cada.

No horário em que a rede está menos sobrecarregada ocorre o horário fora ponta, onde o consumidor paga também o menor valor de tarifa. Em momentos de pico na utilização da rede ocorre o horário de ponta, onde a tarifa é mais cara em relação ao demais postos tarifários, e entre 1h e 1h30min (a depender da concessionária) antes e depois do horário de ponta ocorre o horário intermediário. O horário fora ponta ocorre também durante todo o período do dia aos fins de semana e feriados.

A tarifa branca, dessa forma, surge como uma forma de reduzir a utilização da rede em momentos de pico, recompensando os consumidores que conseguirem transferir sua demanda para o horário de fora ponta com menores tarifas. Assim, quanto maior a diferença entre a tarifa convencional e a tarifa fora ponta maior a economia e conseqüentemente o incentivo para o usuário. A Figura 1 apresenta o gráfico comparativo com o posto horários e valores de tarifa para a tarifa branca e convencional na concessionária ENEL CE.

Figura 1 – Comparativo da tarifa branca e da tarifa convencional para a concessionária ENEL CE



Fonte: ANEEL (2015b)

2.2.2 Grupo A

As unidades do grupo A são conectadas a rede de média ou alta tensão, sendo tarifadas através de tarifas binômias, ou seja, através da demanda e do consumo de energia.

No sistema de tarifação atual cada unidade do grupo A possui valores de demanda contratada para os horários de ponta e fora ponta, medidas em kW. Dessa forma, a concessionária verifica se a unidade está dentro do valor contratado através da demanda medida, formada pela potência máxima registrada no intervalo de 15 minutos (ANEEL, 2010). Enquanto a demanda medida estiver dentro do valor contratado, a unidade paga apenas o referente a demanda contratada. No entanto, caso seja constatado ultrapassagem, o consumidor é onerado em cima da ultrapassagem em kW, por uma tarifa específica, normalmente mais cara. A fim de reduzir os custos com a demanda contratada, e ao mesmo tempo evitar ultrapassagens na mesma, se faz necessário um ajuste com base na curva de carga de cada unidade, ponto onde o sistema de monitoramento vem sendo bastante utilizado.

Segunda a RN nº 414/2012 as unidades pertencentes ao grupo A podem escolher entre duas modalidades tarifárias apresentadas a seguir.

Azul: Clientes que aderirem a modalidade tarifária azul tem essencialmente duas tarifas distintas de consumo, onde o consumo registrado em horário de pico da utilização da rede são caracterizados como hora ponta, e o horário complementar a esse como fora ponta. Assim como o consumo, nessa modalidade, existem também duas tarifas distintas para a demanda, podendo o usuário escolher também valores distintos de demanda contratada para cada posto tarifário.

Verde: A modalidade tarifária verde apresenta o mesmo padrão binômio que tarifa azul, havendo também a distinção entre hora ponta e fora ponta para consumo. Contudo, para a demanda essa distinção não acontece, sendo contratado apenas um único valor, independente do posto tarifário.

Além das duas modalidades tarifárias citadas, os clientes que possuem uma potência instalada de até 112,5 kW ou que estão em áreas de veraneio, independente da potência instalada, podem optar por serem tarifados como unidades do grupo B, não sendo necessário nesses casos o pagamento de demanda.

Além da tarifação por consumo e demanda os consumidores do grupo A são tarifados ainda em relação ao excedente de reativos. No escopo da RN nº 414 estabeleceu-se como limite o fator de potência 0,92, além de horários estabelecidos para que o fator de potência seja indutivo

ou capacitivo. Estabeleceu-se que dentre o horário de 23:30h e 06:30h as unidades deveriam possuir um fator de potência de até 0,92 capacitivo, enquanto no horário complementar a esse, um fator de potência de até 0,92 indutivo. Dessa forma, são verificadas medições a cada hora e caso o fator de potência da unidade esteja abaixo do valor preestabelecido são previstas tarifas de acordo com a quantidade de quilovolt-ampere-reativo-hora (kVARh) medido.

2.3 Medição Inteligente

Durante os últimos anos a demanda por energia vem crescendo em ritmo acelerado em vários países ao redor do mundo, seja pelo aumento da população mundial ou pela maior utilização de energia. Dessa forma, o conceito de *Smart Grid* surge como forma de usar a tecnologia para otimizar e proporcionar uma expansão adequada do sistema elétrico. Entre suas principais características podemos citar a inclusão de fontes renováveis de geração, além do fluxo bidirecional de energia e informação, ponto onde a medição inteligente aparece como fundamental.

Entende-se como medidor inteligente todo sistema capaz de medir quando e quanta energia é consumida, além de transmitir esse dado para uma plataforma de gerenciamento onde os dados são processados e tratados (BARAI *et al.*, 2015). Os dados coletados e exibidos de forma instantânea servem para a controle de faltas, prevenção de furtos de energia e controle de picos de demanda, promovendo uma maior eficiência energética e confiabilidade da rede (SREEDEVI *et al.*, 2020).

Do ponto de vista do usuário, a medição inteligente é capaz de proporcionar uma maior ciência a respeito do que é consumido, uma vez que se acompanha o crescimento dia a dia da curva de consumo ao longo do mês. Estudos publicados por Vassileva *et al.* (2013), Ueno *et al.* (2006) afirmam que unidades em que existe o conhecimento a respeito do que é consumido, tendem a ser mais efetivos para a redução de gastos.

Nos últimos anos os sistemas de medição inteligente vêm sendo utilizados nas mais diversas aplicações além de simplesmente a medição de dados. O trabalho proposto por Metering *et al.* (2017), por exemplo, propõe um medidor inteligente como forma de impedir as perdas por roubo de energia. Caso exista alguma diferença entre a corrente medida entre fase e neutro, o sistema interrompe a conexão e avisa a concessionária de energia via SMS, utilizando tecnologia GSM. Já em Shibu *et al.* (2018) os autores propõem um sistema para medição com capacidade de acionamento remoto de cargas e controle de uso de ar condicionados, através da medição de

um sensor de temperatura DHT11. Os dados são enviados via WiFi e visualizados pelo usuário utilizando uma aplicação Android. Semelhante a este, Aquino *et al.* (2018) apresenta um sistema de baixo custo para medição da potência ativa e energia consumida em cargas residenciais. Em Al-Ali *et al.* (2017) é proposto um sistema que além de medir o consumo, mede dados como umidade e temperatura do ambiente, os dados são exibidos de forma diferente para cada tipo de usuário: o consumidor só pode visualizar dados da sua própria unidade, ao passo que a concessionária pode visualizar todas as unidades em uma determinada área. Em Faisal *et al.* (2019) um medidor bidirecional é proposto. Nele os dados são adquiridos por sensores de tensão e exibidos em um *display* ao usuário. Um PIC16F877 é utilizado para tratamento dos dados e cálculo da fatura de energia.

Os trabalhos apresentados mostram como a medição inteligente e sistemas de monitoramento podem ser úteis à sociedade, contribuindo com desde uma redução no consumo de energia, até uma melhoria no sistema elétrico, facilitando a verificação de furtos de energia ou o faturamento de energia.

A arquitetura de um sistema de medição inteligente é definida por Kurosawa *et al.* (2015) em três partes: a primeira compreende o nó da rede a ser medido e o próprio medidor inteligente; a segunda parte é a tecnologia de transmissão de dados que leva os dados do hardware ao servidor, sendo normalmente construída com tecnologias como *Power Line Carrier* (PLC), Fibra Óptica, ou tecnologias wireless como LoRaWAN e WiFi, dependendo assim da distância e do ambiente que o sistema está inserido; a terceira parte compreende o servidor que recebe, valida e armazena as informações. De forma a elucidar o conteúdo, as Seções 2.3.1, 2.3.2 e 2.3.3 apresentam detalhes dessa estrutura.

2.3.1 Medidor Inteligente

Seguindo a divisão feita por Barai *et al.* (2015), o hardware de medição pode ser visto como possuindo uma parte analógica composta majoritariamente pelos componentes a seguir.

Sensores de tensão e corrente: realizam a conversão das variáveis elétricas em sinais lidos pelo microcontrolador, normalmente se utiliza um circuito divisor de tensão como sensor de tensão e um transformador de corrente como sensor de corrente;

Circuitos *Real Time Clock* (RTC): circuitos RTC garantem que a medição seja salva/enviada com a marcação correta de data e hora;

Carregadores de baterias: muitos medidores possuem baterias de backup para o caso de queda de energia, normalmente é necessário um circuito de carregamento para essas baterias;

Fontes de alimentação: nos medidores eletrônicos são responsáveis por adaptar a tensão da rede para a tensão de operação do microcontrolador e demais circuitos;

Conversores Analógicos-Digitais: Apesar de já virem integrados na maioria dos microcontroladores, algumas aplicações necessitam de uma exatidão maior na conversão dos dados analógicos em digitais, sendo utilizados assim conversores externos (CORREIA, 2019).

E por uma parte digital que inclui o microcontrolador, responsável principalmente por receber os dados, tratá-los e enviá-los ao servidor.

2.3.2 Transmissão de dados

Com o avanço do *Internet of Things* (IoT), tecnologias como LoRaWAN e WiFi ganharam mais destaque na transmissão de dados, principalmente por serem tecnologias wireless e pela distância que conseguem enviar dados. A tecnologia LoRaWAN, por exemplo, é capaz de enviar dados com uma distância de 12km a 15km em áreas rurais e de 3km a 5km em áreas urbanas (SAMUHASILP; PORA, 2018), dessa forma com algumas centenas de dispositivos é possível cobrir uma cidade inteira. O WiFi apesar de possui uma distância bem menor que o LoRaWAN, está atualmente integrado na grande maioria das residências com acesso a internet, podendo ser facilmente utilizado para tanto em medidores inteligentes domésticos. Outra tecnologia que vem ganhando destaque é a ZigBee. As redes Zigbee são baseadas no padrão sem fio IEEE 802.15.4, sendo utilizadas para criação de rede pessoais confiáveis, de baixo consumo e baixa taxa de transmissão, ideal para aplicações de monitoramento e controle (IRMAK *et al.*, 2016).

Em países em que o acesso a internet é limitado mas existe sinal de telefonia é recomendado o uso da tecnologia *General Packet Radio Services* (GPRS). Nela a informação percorre longa distância através de antenas, contudo, como a comunicação depende da estrutura de telefonia, existe um custo recorrente para a transmissão de dados.

Pode-se destacar também o PLC que utiliza a rede elétrica para enviar e receber dados. Uma frequência bem maior que a frequência da rede é inserida pelo codificador, enquanto do outro lado o decodificador recebe e interpreta os dados. Apesar da simplicidade de implementação, o PLC é restrito em alguns países por gerar interferência com outras tecnologias como rádio (CANO *et al.*, 2016). Em aplicações residenciais o PLC é limitado ainda pela distância,

sendo necessária a integração com outras tecnologias para se realizar o acesso remoto de dados (KURUPPU *et al.*, 2017). Apesar do baixo custo e a facilidade de implementação, o *bluetooth* também se torna restrito para aplicações locais por conta do alcance.

2.3.3 Servidor

Em aplicações em que vários nós são medidos, é gerada uma grande quantidade de dados a todos instante. Como esses dados devem ser organizados e armazenados se faz necessário a presença de um servidor. Segundo Kurosawa *et al.* (2015), o servidor pode ser dividido em duas estruturas complementares: o *head-End System* (HES) que aquisita, controla e envia dados aos medidores inteligentes, e pelo *Meter Data Management System* (MDMS) que processa, valida e armazena esses dados. O MDMS se comunica ainda com sistemas de faturamento ou supervisão ao usuário.

No trabalho de Correia (2019), por exemplo, uma *Virtual Private Server* (VPS) foi utilizada como servidor. A partir do medidor inteligente as medições eram enviadas via HTTP para uma aplicação web onde os dados eram armazenados em um banco de dados, organizados e exibidos ao usuário através de gráficos.

2.4 Protocolos de Comunicação

O protocolo de comunicação que os sistemas de medição inteligente e outros dispositivos de IoT utilizam é de fundamental importância para o sucesso da aplicação, tanto quanto os outros itens apresentados na seção anterior. A escolha entre um e outro protocolo depende da natureza da aplicação, sendo assim uma escolha particular a depender do ambiente, grau de flexibilidade do sistema, grau de confiabilidade da informação e a capacidade de processamento do dispositivo que realiza o envio dos dados. A seguir são apresentados os principais protocolos utilizados em aplicações de IoT. Espera-se que a explanação a seguir possa servir como tomada de decisão para trabalhos futuros.

2.4.1 *Constrained Application Protocol (CoAP)*

Apesar de ser diariamente utilizado na internet o protocolo HTTP não é recomendado para dispositivos de IoT. O grande pacote de dados é capaz de sobrecarregar redes restritas e dispositivos com baixa capacidade de processamento e disponibilidade de energia. Vendo isso o

grupo de trabalho *Constrained Restful Environments* (CoRE) desenvolveu o protocolo CoAP para ser mais leve e mais adequado a aplicações de IoT. Assim como o HTTP, o CoAP é baseado na arquitetura request-response onde o cliente solicita um recurso ao servidor e o mesmo responde. Além disso, o protocolo é baseado no *REpresentational State Transfer* (REST) podendo ser utilizados os verbos GET, POST, PUT and DELETE assim como no HTTP. O diferencial do CoAP está no uso do *User Datagram Protocol* (UDP) para o transporte, em vez de utilizar o *Transmission Control Protocol* (TCP), o que torna o protocolo mais adequado para pequenos dispositivos limitados em potência ou conexão (AL-FUQAHA *et al.*, 2015).

A protocolo é baseado em 4 modelos de mensagem: confirmáveis, *acknowledgement*, *reset* e não confirmáveis. As mensagens confirmáveis seriam as mensagens enviadas e que necessitam de resposta, caso uma mensagem confirmável seja recebida pelo destinatário, uma mensagem de *acknowledgement* ou de *reset* poderiam ser respondidas, a primeira indica que a mensagem chegou, enquanto a segunda que o destinatário não foi capaz de processar a mensagem confirmável; por último a mensagem não confirmável seria a mensagem que não necessita de resposta (SILVA, 2016).

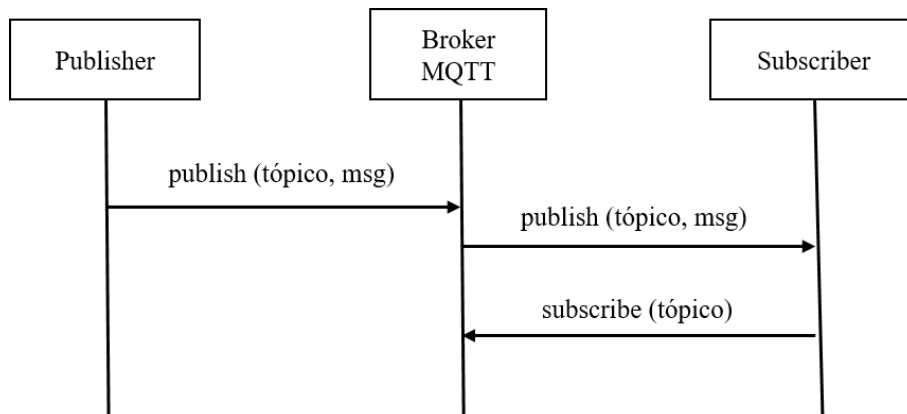
Uma mensagem CoAP utiliza basicamente um *header* de 4 bytes, seguido por um *token* que varia de 0 a 8 bytes. O restante da mensagem é composto por *options* e pelo *payload*, ficando uma mensagem típica entre 10 e 20 bytes (AL-FUQAHA *et al.*, 2015) .

2.4.2 *Message Queue Telemetry Transport (MQTT)*

O protocol MQTT é utilizado em aplicações de IoT principalmente pela sua mensagem reduzida e pelo seu mecanismo de comunicação muitos-para-muitos, um-para-muitos e um-para-um. Esse recurso é possível graças ao padrão *publish/subscribe* onde um determinado dispositivo publica em um tópico específico e todos os dispositivos autorizados e subscritos naquele tópico recebem a mensagem através do *broker*. O fluxo da informação pode ser ainda bidirecional, sendo cada dispositivo capaz de assumir o papel de publicar e/ou subscrever em determinado tópico. A Figura 2 apresenta o fluxo da informação em que uma mensagem é publicada pelo dispositivo n e recebida através do *broker* MQTT pelo dispositivo n+1.

A comunicação isolada através do *broker* possibilita vários recursos uteis ao protocolo, entre eles pode-se citar o *Will Message*, que possibilita a configuração do *broker* para realizar o envio de uma mensagem pré-configurada, em um determinado tópico, no momento em que algum dispositivo é subitamente desconectado (HUNKELER *et al.*, 2008).

Figura 2 – Fluxo da informação entre o broker MQTT e dispositivos conectados



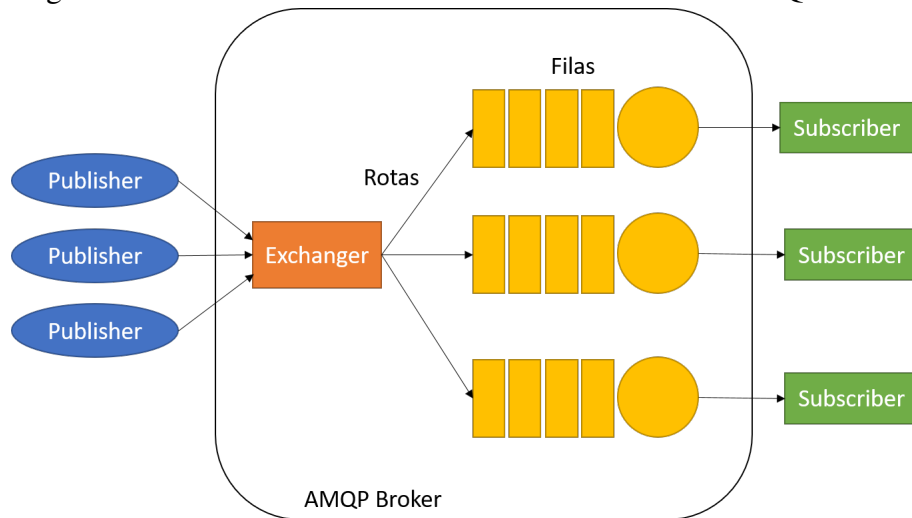
Fonte: O próprio autor

As mensagens são entregues em três níveis diferentes de QoS. Mensagens transmitidas com QoS 0 não recebem *acknowledgment* quando são publicadas, enquanto com QoS 1 o servidor retorna informando se a mensagem foi publicada e qual o identificador. Com QoS 2 quatro pacotes são transmitidos. Ao receber a mensagem, o servidor retorna informando que a mensagem foi recebida, então o cliente responde autorizando a publicação e por último o servidor responde informando que o dado foi publicado no tópico solicitado (KODALI; SORATKAL, 2016). As mensagens MQTT são compostas basicamente por um *header* fixo, um *header* variável e o *payload*, que contém a informação a ser transmitida.

2.4.3 Advanced Message Queuing Protocol (AMQP)

Baseado no mecanismo de *exchange* e *message queues*, o AMQP é um protocolo de comunicação confiável que possibilita a comunicação do tipo “no máximo uma vez”, “pelo menos uma vez” e “exatamente uma vez” entre dispositivos. A partir do momento que um dispositivo publica uma mensagem, *exchanges* são utilizados para separar e classificar cada mensagem em filas (*queues*), de acordo com regras preestabelecidas (AL-FUQAHA *et al.*, 2015), antes de serem enviadas ao destinatário. Apesar de suportar a comunicação *publish/subscriber* as mensagens podem ficar armazenadas nas filas enquanto o dispositivo que irá receber a informação está desconectado, ao contrário do protocolo MQTT (UY; NAM, 2019). A estrutura de transmissão de dados para o protocolo AMQP é apresentada na Figura 3.

Figura 3 – Estrutura de transmissão de dados no broker AMQP



Fonte: O próprio autor

2.4.4 Extensible Messaging and Presence Protocol (XMPP)

O XMPP é um protocolo aberto para mensagens instantâneas baseado no *Extensible Markup Language* (XML). O XMPP garante a troca de mensagens autenticadas, criptografadas e compatíveis com outros protocolos através de uma arquitetura cliente-servidor, onde cada dispositivo se conecta ao servidor através de conexões TCP, assim como servidores se conectam a outros servidores (WANG *et al.*, 2013).

A conexão de um dispositivo ao servidor é baseada em estrofes (*stanzas*), que podem ser divididas três partes: mensagem, presença e iq (*info/query*). Cada estrofe identifica o remetente e destinatário, além de preencher título e conteúdo de uma mensagem (AL-FUQAHA *et al.*, 2015).

2.5 Conclusão

Neste tópico foram introduzidos conceitos fundamentais para o desenvolvimento e entendimento do trabalho proposto. Partindo da premissa de que um sistema de monitoramento tem que estar alinhado com a forma como o consumidor é tarifado, foram apresentados as principais modalidades tarifárias para clientes do grupo A e B no Brasil. Não obstante, como este trabalho propõe a solução de um problema dos consumidores de geração distribuída, foi apresentado o panorama mundial e nacional nessa área, sendo destacados a forma de acesso a rede e resultados positivos obtidos. No Brasil, observou-se principalmente uma dependência na instalação do medidor bidirecional, que com o sistema de compensação e faturamento atual,

impossibilita o consumidor de saber exatamente quanto que é gerado e consumido na sua unidade.

Foi apresentado também o conceito de medição inteligente, junto com a sua importância para o novo paradigma de uso da rede, onde a figura de consumidor é substituída pelo prossumidor, que além de gerar a sua própria energia, injeta o excedente na rede de distribuição. Igualmente importante, foi visto as partes que compõem um sistema de medição inteligente, sendo necessário um estudo cuidadoso desde o hardware utilizado até o servidor que irá receber e armazenar as informações.

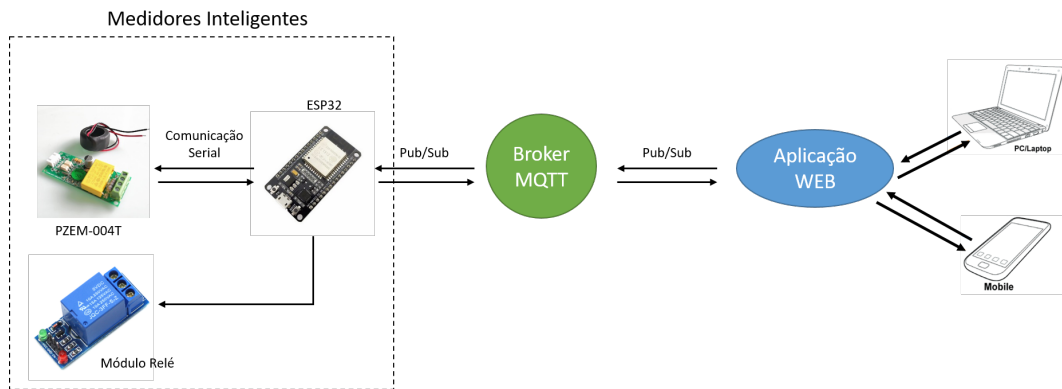
Quanto a transmissão dos dados, foram introduzidos os principais protocolos de comunicação utilizados atualmente. Dentre eles pode-se destacar os protocolos AMQP e MQTT, que possuem aplicação também na área de IoT, sendo esse último utilizado no presente trabalho.

3 SISTEMA PROPOSTO

Como apresentado na fundamentação teórica, o medidor bidirecional não contabiliza necessariamente todo o consumo nem toda a geração, haja visto, que uma parte do que é gerado é consumido pela própria unidade sem que chegue ao medidor. A fim de sobrepor essa problemática, o presente trabalho propõe a utilização de uma dupla medição unidirecional, sendo dedicado um medidor inteligente para consumo e outra para geração. Desse modo, as medições obtidas logo após o inversor e ao quadro de distribuição representam o perfil real da unidade, permitindo um melhor gerenciamento energético por parte do usuário.

Os dados obtidos são transmitidos através de uma conexão WiFi e apresentados ao usuário através da aplicação web desenvolvida. A transmissão dos dados é realizada com o protocolo MQTT, que proporciona uma comunicação isolada entre os dispositivos conectados e a aplicação web, além de proporcionar uma maior confiabilidade mesmo em redes com mais lentas. Dessa forma, o sistema proposto é composto de forma macro por medidores inteligente conectados e por uma aplicação web, apresentados em detalhe na Figura 4.

Figura 4 – Visão macro do sistema proposto



Fonte: O próprio autor

Neste capítulo são apresentados os principais pontos do sistema de monitoramento e gerenciamento proposto. A fim de facilitar a explanação, o presente capítulo foi dividido em duas partes: a primeira apresenta sobre o medidor inteligente desenvolvido, enquanto a segunda explica sobre a aplicação web para monitoramento dos dados.

3.1 Medidor inteligente desenvolvido

Considerando que a maioria dos medidores bidirecionais em Baixa Tensão (BT) não possuem protocolo de comunicação habilitado pela concessionária, optou-se por utilizar um sistema de medição independente. O sistema proposto possibilita ao usuário acompanhar separadamente a medição de consumo e geração da sua unidade.

3.1.1 Hardware de medição

Para realizar a medição da energia consumida e da energia gerada foi-se utilizado um módulo monofásico PZEM-004T da Peacefair. Nele a medição é realizada através de um transformador de corrente externo (TC), não invasivo, através do qual o módulo fornece dados de tensão, corrente, energia ativa, potência ativa, fator de potência e frequência da rede. O módulo possui uma precisão ideal conforme Tabela 1, e se comunica com o microcontrolador via *Universal Asynchronous Receiver/Transmitter* (UART). No PZEM-004T, a energia ativa é armazenada em um contador, incrementado até o valor máximo da Tabela 1, sendo resetado após esse valor.

Tabela 1 – Precisão do módulo PZEM-004T utilizado.

	Faixa de Medição	Precisão
Tensão	80~260V	0,5%
Corrente	0~100A	0,5%
Potência Ativa	0~23kW	0,5%
Fator de potência	0,00~1,00	1,0%
Frequência	45~65 Hz	0,5%
Energia ativa	0~9999,99 kWh	0,5%

Fonte: Dados retirados de: (PEACEFAIR, 2020)

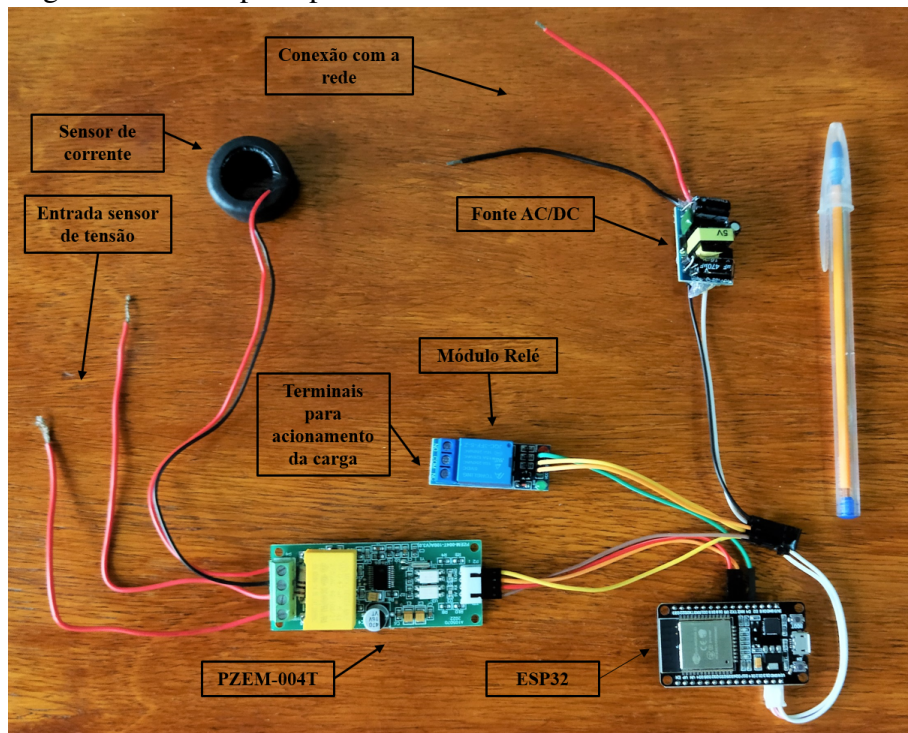
A fim de receber as medições do PZEM-004T e enviar os dados ao *broker* MQTT foi utilizado um microcontrolador ESP32. O ESP32 foi escolhido para essa aplicação por conter uma placa de comunicação WiFi nativa, não sendo necessária a instalação de um módulo extra para essa finalidade. O microcontrolador conta ainda com 18 canais analógico-digital, 10 pinos de sensor de toque, além de canais de comunicação UART, I2C, I2S, SPI, módulos PWM, entre outros recursos.

O ESP32 possui um módulo de programação *Over-The-Air* (OTA) que possibilita a atualização do *firmware* de forma remota, sem a necessidade de desconexão do microcontrolador.

Esse recurso é utilizado no sistema proposto por conta da praticidade e pela necessidade de atualização do *firmware* em locais de difícil acesso. Sobretudo pelo baixo custo e por todas as tecnologias citadas, o ESP32 vem sendo igualmente utilizado em várias aplicações de IoT.

De modo a realizar o acionamento das cargas conectadas em corrente alternada, foi utilizado um módulo com relé de estado sólido. Basicamente, o módulo conta com o circuito necessário para o acionamento do relé pelo ESP32, sem o risco de danificar o dispositivo. Cada medidor conta ainda com uma fonte de tensão que converte a tensão alternada da rede (220 VAC) na tensão necessária para alimentar o ESP32 e demais dispositivos (5 VDC). A Figura 5 apresenta a montagem dos componentes no protótipo desenvolvido.

Figura 5 – Protótipo experimental desenvolvido



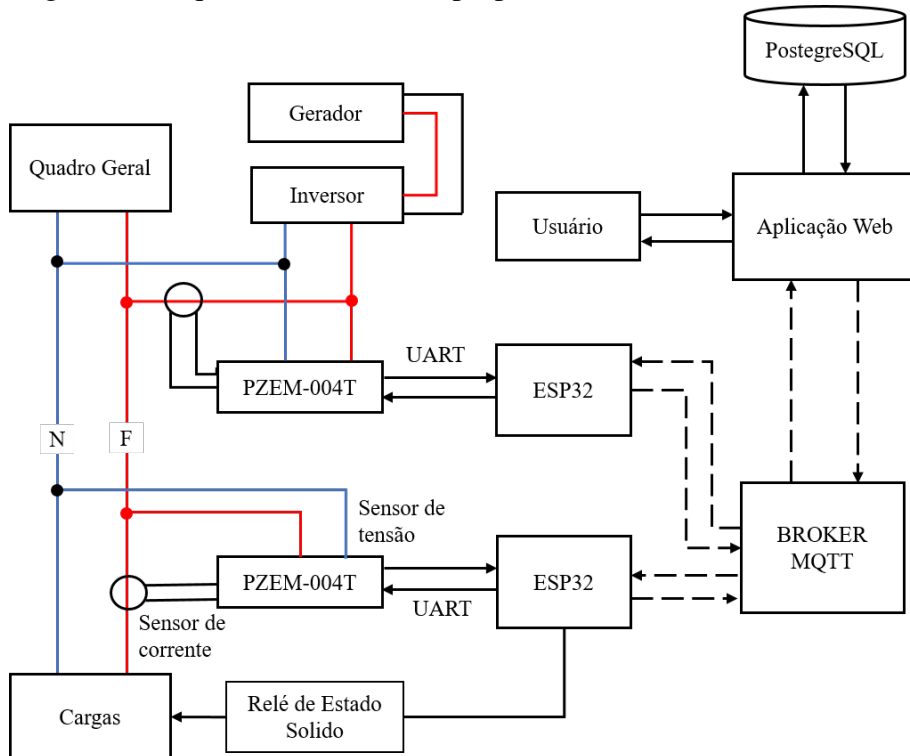
Fonte: O próprio autor

Como apresentado pela Figura 5, o sistema é construído de forma modular através da conexão simples entre dispositivos. Assim, em caso de falha, cada dispositivo pode ser substituído de forma individual, tornando o processo de manutenção bem mais simples e rápido.

Como consumo e geração devem ser monitorados, dois sistemas compostos por um ESP32 e um PZEM-004T cada devem ser instalados, um na saída do inversor e outro no quadro de distribuição geral da unidade consumidora. O acionamento de cargas pode ser utilizado em ambos os dispositivos ou apenas no de consumo, necessitando apenas do módulo com relé de estado sólido e a configuração adequada do tópico específico da carga. A arquitetura do sistema

com os medidores já posicionados na instalação é apresentado na Figura 6.

Figura 6 – Arquitetura do sistema proposto



Fonte: O próprio autor

Apesar do PZEM-004T ser um medidor monofásico, o sistema proposto pode ser utilizado também em unidades trifásica, através da conexão de três módulos PZEM-004T em um mesmo ESP32. A conexão é feita em uma mesma porta serial, onde cada medidor é identificado por um endereço. O ESP32 (mestre) requisita as medições de cada um dos módulos PZEM-004T (escravos) através do endereço e ao final a potência trifásica é calculada.

Para unidades de média e alta tensão se faz necessário a instalação dos medidores no lado de baixa tensão, devido a limitação de tensão do próprio PZEM-004T. Da mesma forma, deve-se observar a potência máxima, de 23 kW, suportada pelo mesmo.

O medidor inteligente pode ser utilizado ainda setorialmente para a medição de consumo em linhas de produção industriais ou para a medição específica de consumo de um determinado cômodo ou aparelho.

3.1.2 *Firmware desenvolvido*

O *firmware* da aplicação foi desenvolvido de modo a executar as principais funções de um medidor inteligente: aquisição dos dados do PZEM-004T, tratamento da informação,

verificação da conexão com a rede WiFi e *broker* MQTT, e envio dos dados.

A rotina do *firmware* inicia com a configuração dos recursos necessários como: WiFi, MQTT, OTA e a aquisição do horário real no servidor *Network Time Protocol* (NTP). Após a configuração, a conexão com WiFi e com o *broker* MQTT são iniciadas, seguida da configuração do horário interno ser realizada, com base no horário obtido no servidor NTP. Como não é previsto bateria de backup para o protótipo desenvolvido essa configuração é realizada sempre que o dispositivo é iniciado ou acontece falta de energia.

Ao realizar a conexão, o ESP32 envia uma mensagem de status ao *broker*, no tópico “*keep_alive*”, informando a aplicação web (subscrita neste tópico) que o dispositivo está ativo. Posteriormente, caso haja uma desconexão, uma *Will Message* é enviada pelo *broker* informando o novo estado da conexão.

É verificado ainda se existia alguma carga acionada antes do microcontrolador ser inicializado. Para isso, o último estado de acionamento das cargas é salvo na memória EEPROM em endereço único associado a uma única carga. Ao iniciar, o *firmware* procura por valores diferentes de zero na memória e caso algum valor seja notado, realiza-se o respectivo acionamento, não sendo necessário um reacionamento pelo usuário em caso de falta de energia, por exemplo.

Imaginando ainda um cenário de falta de energia, ou desligamento inesperado, é verificado também se existem medições salvas na memória flash, armazenadas durante o modo *offline* desenvolvido, explicado mais adiante. Caso existam, essas medições são enviadas imediatamente ao *broker* MQTT, sendo excluídas após o envio. A partir daí o código permanece em *loop*, verificando o horário atual e se alguma ordem de acionamento de cargas foi enviada pelo *broker*.

Foi previsto o envio de uma única medição a cada 15 minutos, nos minutos 00, 15, 30 e 45 de cada hora. Durante cada intervalo, são realizadas medições recorrentes (a cada 5 segundos) de potência, tensão, corrente, fator de potência e frequência. O objetivo é ao final do período obter o maior valor de demanda, a corrente média do período e as maiores variações de tensão, frequência e fator de potência em relação ao valor ideal. Dessa forma, são criadas variáveis que armazenam os valores medidos, sendo atualizadas a cada iteração caso as condições apresentadas acima sejam satisfeitas. Já a medição de consumo é obtida ao final do período de 15 minutos, através da diferença entre o valor atual do contador de energia e o valor no início do período. Isso evita que o *reset* no contador do PZEM-004T altere a informação, uma vez que o

valor enviado é apenas o valor consumido no período e não o do contador em si. Após receber cada uma das medições do PZEM-004T, o ESP32 faz ainda o cálculo do CRC16, evitando assim informações que por ventura venham corrompidas.

Para o envio ao broker MQTT os dados são compilados em uma única *string*, com cada valor separado por vírgula, além de serem incluídos a data e hora da mensagem, assim como o identificador único do dispositivo. A estrutura do *payload* enviado ao *broker* é apresentado na Figura 7.

Figura 7 – Estrutura de dados da string enviada ao broker MQTT

```
String: 215.6, ← Tensão
        0.05, ← Corrente
        4.8, ← Demanda
        0.000, ← Consumo
        60.0, ← Frequência
        0.30, ← Fator de Potência
        1, ← ID do dispositivo
        2020-11-09 15:30:06 ← Data e hora de envio
```

Fonte: O próprio autor

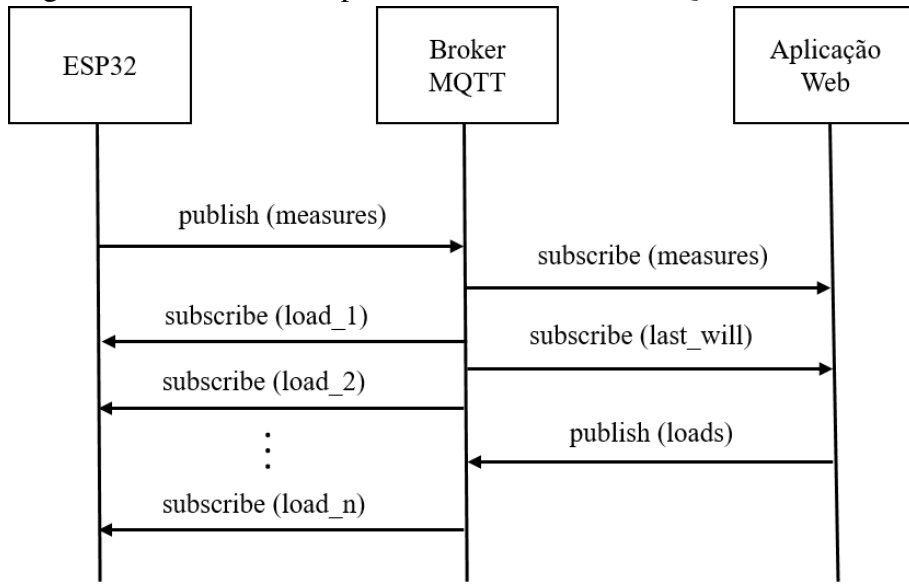
Todos os dispositivos, independente de serem de geração ou consumo utilizam o mesmo tópico chamado genericamente de "*measures*" para envio das medições. Na aplicação web, como será visto, cada medição é associada a um dispositivo através do identificador e salvo no banco de dados corretamente.

Para o acionamento remoto, foi previsto a utilização de um tópico específico para cada carga, chamado genericamente de "*load_1*", "*load_2*".. "*load_n*". Existindo uma rotina que identifica e realiza o acionamento de qualquer uma das cargas configuradas, caso uma mensagem de acionamento seja recebida.

A estrutura de tópicos criados no *broker*, apresentados nessa seção, bem como o fluxo da informação são mostrados na Figura 8.

De modo a evitar a perda de informação, uma rotina foi desenvolvida para verificar a conexão WiFi e a conexão com o *broker* MQTT. Caso seja detectada alguma desconexão o ESP32 inicia o modo *offline*, armazenando todas as medições na memória flash do dispositivo. Quando a conexão for reestabelecida o dispositivo envia as medições salvas e exclui o arquivo gerado. Foi separado 1.3 MB da memória flash para esse propósito o que garante uma autonomia de 234 dias sem conexão WiFi.

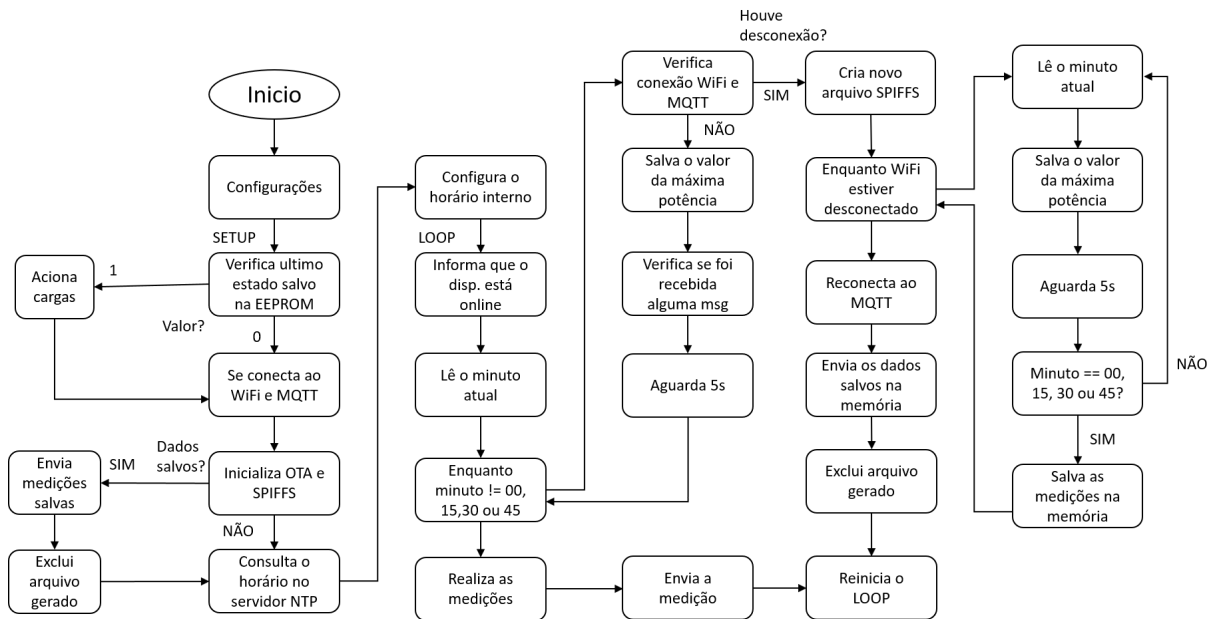
Figura 8 – Estrutura de tópicos criados no broker MQTT



Fonte: O próprio autor

Afim de elucidar o funcionamento do *firmware*, a Figura 9 sintetiza a lógica desenvolvida através de um fluxograma.

Figura 9 – Fluxograma do firmware desenvolvido



Fonte: O próprio autor

3.2 Aplicação web desenvolvida

A aplicação web proposta foi desenvolvida em Ruby, utilizando o *Ruby on Rails* (RoR). O RoR é um framework livre e de desenvolvimento ágil, que segue dois conceitos principais:

DRY (Don't Repeat Yourself): é o conceito em que o máximo de código em uma aplicação pode ser reaproveitado sem a necessidade de novas declarações, ou de código redundante.

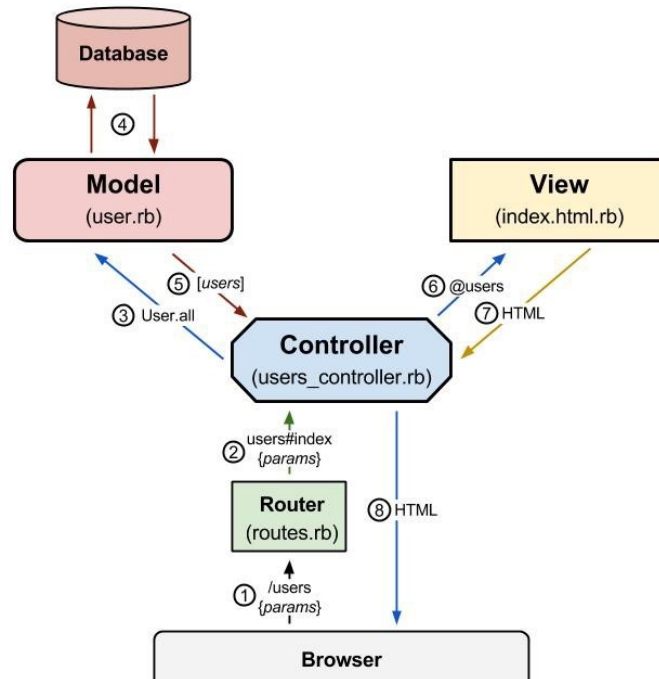
Convention over Configuration: é o conceito em que configurações comuns realizadas durante o desenvolvimento se transformam em convenções no framework, podendo ser utilizadas diretamente, sem a necessidade de retrabalho. Um exemplo disso é a criação de um CRUD (*Create-Read-Update-Delete*) em RoR, que pode ser feito utilizando apenas um comando. A partir daí, o framework se encarrega de criar todos os arquivos e configurações padrões pra começar o desenvolvimento.

O framework se baseia na arquitetura MVC (*Model-View-Controller*), onde a requisição do usuário feita na *View* passa pelo *Controller*, onde é tratada e enviada a um *Model* específico, que realiza a parte lógica e o acesso ao banco de dados. Como exemplo da arquitetura em RoR, pode-se citar o acesso a página de usuários de uma determinada aplicação. A partir do momento que o cliente acessa a página “/users”, uma rota aciona a ação no *Controller* específico. O *Controller* trata e envia a requisição ao *Model*, que acessa o banco de dados e retorna uma lista com todos os usuários cadastrados. Essa informação é passada pra *View* e exibido ao usuário, em formato HTML, JSON, ou XML, na maioria dos casos. O diagrama da situação exemplo pode ser visto na Figura 10.

Neste trabalho, o RoR foi utilizado para desenvolver uma aplicação web, capaz de receber os dados enviados no *broker* MQTT e apresentá-los de forma amigável ao usuário.

Inicialmente, a subscrição no tópico de medidas foi implementada na pasta *initializer* da aplicação, onde os arquivos são inicializados junto com as configurações do sistema. Dessa forma, no momento em que a aplicação inicia no servidor, uma nova instância é criada e a subscrição no tópico “*measures*” é realizada. Essa instância passa a funcionar de forma paralela as demais funcionalidades da aplicação, sendo responsável de maneira contínua, por receber e salvar no banco de dados as medições vindas pelo *broker* MQTT. Um outro *initializer* com uma nova instância foi criada para receber as mensagens de *keep alive* dos dispositivos conectados.

Figura 10 – Ciclo da resposta a requisição na arquitetura MVC



Fonte: Main (2015)

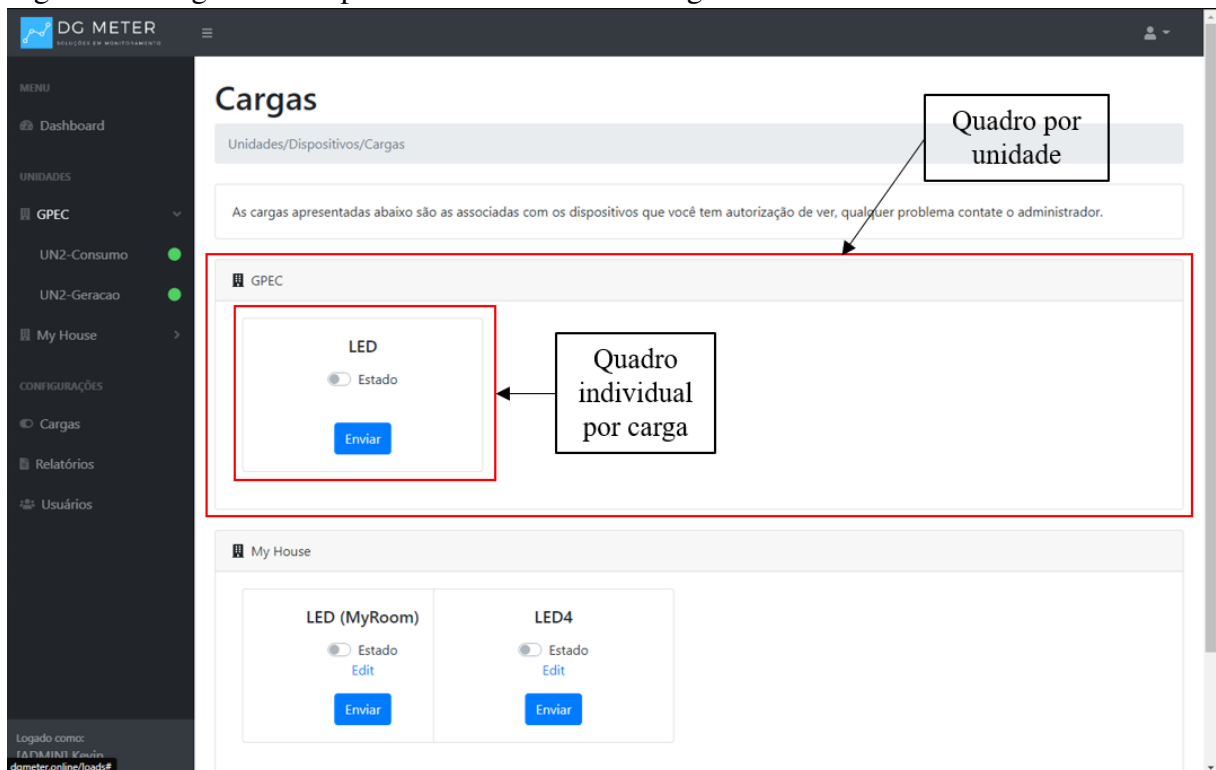
O estado de conexão recebido é exibido ao usuário e atualizado em tempo real, sem a necessidade de atualização da página. Para isso foi utilizado o *Action Cable*, recurso nativo do RoR que mantém uma conexão aberta entre o *front-end* em Javascript e o *back-end* em Ruby. Assim, no momento em que um dispositivo é desconectado, a aplicação responde alterando o estado da conexão, enviando um e-mail de alerta ao usuário e o impedindo de realizar alterações no estado das cargas conectadas daquele dispositivo.

No banco de dados, cada id de dispositivo é associado como sendo de geração ou consumo, dessa forma as medições são organizadas e exibidas ao usuário por dispositivo. Ao acessar cada dispositivo o usuário tem acesso aos gráficos de consumo e demanda, no caso do dispositivo de consumo, e geração e potência máxima do sistema no caso do dispositivo de geração. Cada gráfico é atualizado também em tempo real pela aplicação, e os dados podem ser visualizados por dia, mês, ano ou através de um intervalo personalizado de datas. Além da estrutura de gráficos descrita acima, são apresentados a frequência da rede e a tensão de forma instantânea ao usuário, sendo atualizados também pelo *Action Cable* do RoR.

O acionamento imediato das cargas é realizado pelo próprio usuário, que altera o estado de um *switch* e pressiona o botão pra enviar a informação. Ao fazê-lo o sistema atualiza o estado da carga no banco de dados e aciona o envio do novo estado da carga ao *broker* MQTT de

forma paralela, sem travar a aplicação. Por fim, o ESP32 que está subscrito no tópico recebe a mensagem e aciona ou não a carga selecionada. Dentro da aplicação o estado do *switch* é associados a "0" ou "1", onde mover o *switch* para a posição ligado, significa salvar no banco de dados a *string* "1" e envia-la ao dispositivo associado. Igualmente, no ESP32, ao receber a *string* "1" o microcontrolador aciona a carga e ao receber "0" desliga-a. A Figura 11 apresenta a interface de acionamento das cargas. Como pode-se perceber, as cargas são associadas as unidades que pertencem, permitindo uma melhor organização ao usuário.

Figura 11 – Página criada para o acionamento das cargas

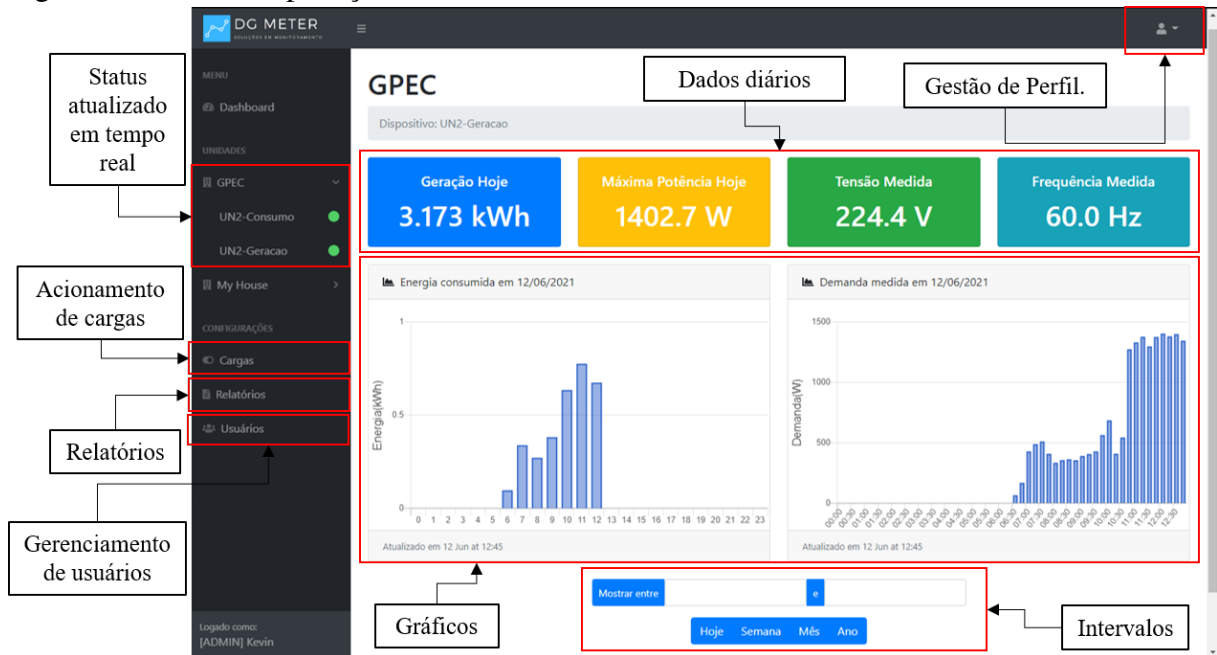


Fonte: O próprio autor

De modo a limitar o acesso a dados restritos, um sistema de usuários foi implementado, dessa forma cada usuário tem acesso apenas aos dispositivos da sua unidade, ou de outras unidades permitidas pelo administrador. Igualmente, o acionamento de cargas também só é permitido para usuários que possuam acesso aos dispositivos daquela unidade.

A aplicação web disponibiliza ainda uma aba para configurações, onde é possível alterar as configurações do usuário como nome, e-mail e senha. A Figura 12 apresenta a interface da aplicação web desenvolvida.

Figura 12 – Vista da aplicação web desenvolvida



Fonte: O próprio autor

3.3 Conclusão

Neste capítulo foi apresentado de uma forma macro e micro a estrutura do trabalho desenvolvido. Na análise do medidor inteligente proposto, foi explanado a cerca do hardware utilizado, sendo apresentada a precisão do módulo PZEM-004T e os recursos utilizados do ESP32 que justificam sua escolha. Do mesmo modo, o funcionamento do *firmware* desenvolvido foi apresentado, merecendo destaque a rotina para envio das medições, o armazenamento das medições em caso de desconexão e o acionamento remoto das cargas. Pôde-se ter também uma visão de como o medidor se comunica utilizando o protocolo MQTT e qual a estrutura de tópicos criada dentro do *broker*. Por fim, apresentou-se possibilidades de utilização do sistema que vão além da medição macro de energia.

No momento seguinte, foram apresentados os recursos utilizados para o desenvolvimento da aplicação Web, utilizada para apresentação dos dados ao usuário e armazenamento das medições através do banco de dados. Foi visto como instâncias da aplicação principal foram criadas, para realizar a armazenagem das medições e perceber desconexões dos dispositivos no *broker*, sem prejudicar a instância principal. Foi apresentado como os gráficos e quadros são atualizados em tempo real e como o acionamento das cargas é realizado dentro da aplicação.

Procurou-se, dentro do escopo do projeto, desenvolver um sistema de medição e

monitoramento que fosse o mais amigável possível ao usuário e que ao mesmo tempo fosse funcional e de baixo custo. Se poderá dizer que o objetivo foi concluído a partir dos resultados observados no próximo capítulo, onde um sistema piloto foi instalado em um laboratório em Fortaleza-CE.

4 RESULTADOS

Neste capítulo são apresentados os resultados do sistema instalado em um laboratório na cidade de Fortaleza-CE. Serão avaliados a confiabilidade dos resultados obtidos e como estes podem proporcionar uma economia de energia para a unidade. Por fim, com base nas medições obtidas, é apresentada a curva de carga dessa unidade.

4.1 Teste de precisão do medidor proposto

Inicialmente, o sistema de medição proposto foi avaliado em termos de precisão. Para realização do teste foram utilizadas cargas de diferentes valores, e os resultados obtidos foram comparados com os do multímetro *True RMS* Fluke 177. Esse equipamento possui uma precisão de 1% mais três dígitos na faixa de tensão do teste (até 600VAC) e 1,5% mais três dígitos na faixa de corrente do teste (até 10A) (FLUKE, 2015). Os resultados obtidos são apresentados na Tabela 2.

Tabela 2 – Comparação das medições obtidas pelo sistema de medição proposto e o multímetro Fluke 177

PZEM-004T			Fluke 177			Δ Potência
Tensão (V)	Corrente (A)	Potência (W)	Tensão (V)	Corrente (A)	Potência (W)	%
221.2	0.278	61.49	221.7	0.278	61.63	0.226
221.2	0.723	159.93	221.7	0.722	160.07	0.087
221.2	1.376	304.37	222.8	1.371	305.46	0.357
221.1	2.074	458.56	221.6	2.066	457.83	0.160
220.2	2.738	602.91	220.8	2.727	602.12	0.130
220.8	3.650	805.92	221.1	3.636	803.92	0.248
219.5	4.539	996.31	219.9	4.511	991.97	0.436
219.1	6.742	1477.17	219.2	6.720	1473.02	0.281
218.8	9.033	1976.42	218.7	9.010	1970.49	0.300

Fonte: O próprio autor.

Conforme os resultados observados na Tabela 2 pode-se observar que a precisão obtida entre a faixa de potência aproximada de 60W a 2000W está dentro dos 0,5% indicado pelo fabricante do módulo PZEM-004T (PEACEFAIR, 2020). No contexto da NBR 14520:2011, os medidores Cronos 6021L e o Cronos 7023L (ELETRA, 2021), instalados pela concessionária ENEL CE apresentam classe B de precisão, o que corresponde a um erro percentual de até 1%. Dessa forma, mesmo não tendo passado por um processo de certificação, é possível afirmar que o sistema proposto é capaz de fornecer medições precisas, sendo viável a comparação com os

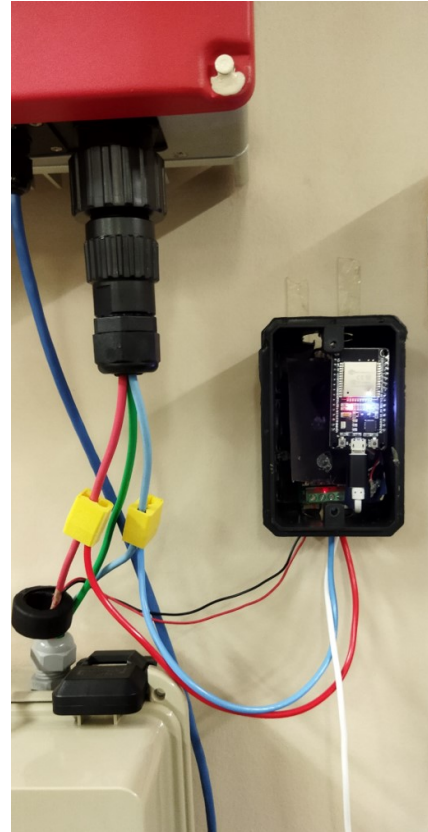
dados da concessionária de energia.

4.2 Resultados experimentais

A fim de validar o sistema de medição e a operabilidade da aplicação web foi realizada a instalação de protótipos em no laboratório do Grupo de Processamento de Energia e Controle (GPEC), em Fortaleza-CE. O laboratório possui hoje um fornecimento trifásico em 380V, ao qual é dividido em bancadas de testes, computadores, ar condicionados e outros equipamentos. O mesmo conta ainda com um sistema de geração solar instalado no telhado. O sistema tem capacidade de 1,675 kWp, distribuídos através de 5 placas de 335 Wp cada e é conectado a rede de distribuição através de um inversor de 1,5 kW monofásico.

De modo a monitorar o consumo e a geração dois protótipos foram instalados, o primeiro na saída do quadro de distribuição geral, que monitora apenas a energia consumida no laboratório e realiza o acionamento remoto das luzes da sala de estudos, enquanto o segundo foi instalado na saída do inversor e monitora apenas a energia gerada pelo sistema fotovoltaico. De modo a monitorar o consumo trifásico da unidade foram utilizados três módulos PZEM-004T conectados em um mesmo microcontrolador ESP32. Cada um dos módulos foi conectado em um mesmo barramento de transmissão e envio de dados, e controlados pelo ESP32 de forma a ser recebido a medição de apenas um módulo por requisição. A geração, por utilizar um inversor monofásico não apresentou o mesmo grau de dificuldade. Essa diferença entre geração e consumo foi útil para validar a utilização do sistema em rede monofásicas e trifásicas. A Figura 13 apresenta os protótipos instalados para medição de consumo e geração, respectivamente.

Figura 13 – Protótipos instalados para teste experimental



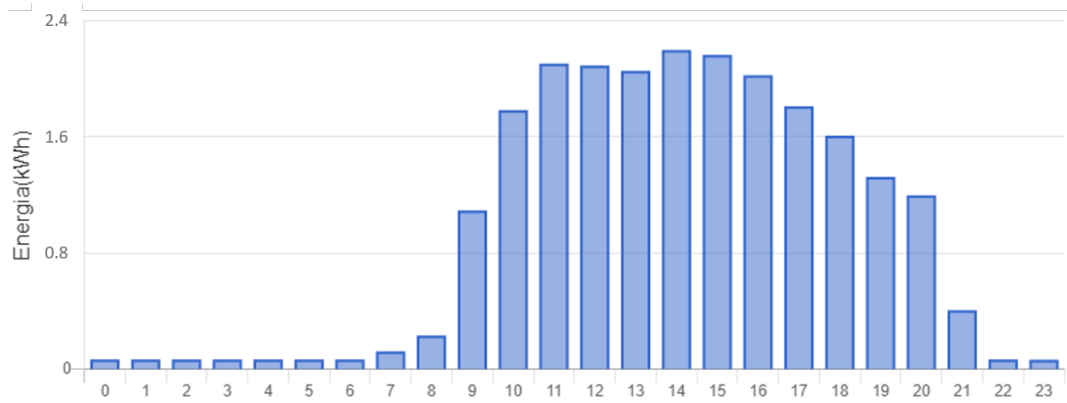
(a) Protótipo instalado para medição de consumo (b) Protótipo instalado para medição da geração

Fonte: O próprio autor

A unidade foi monitorada durante o intervalo de 50 dias, desde o dia 01 de Abril a 20 de Maio. Apesar da pandemia do novo Coronavírus, alguns alunos ainda continuaram indo ao laboratório em horário habitual, de modo que durante esse período foi possível a aquisição da curva de consumo da unidade. O perfil de consumo da unidade para os dias da semana e para os fins de semana é apresentado nas Figuras 14 e 15, respectivamente. Os dados apresentados foram obtidos através da média horária de consumo durante o período do experimento.

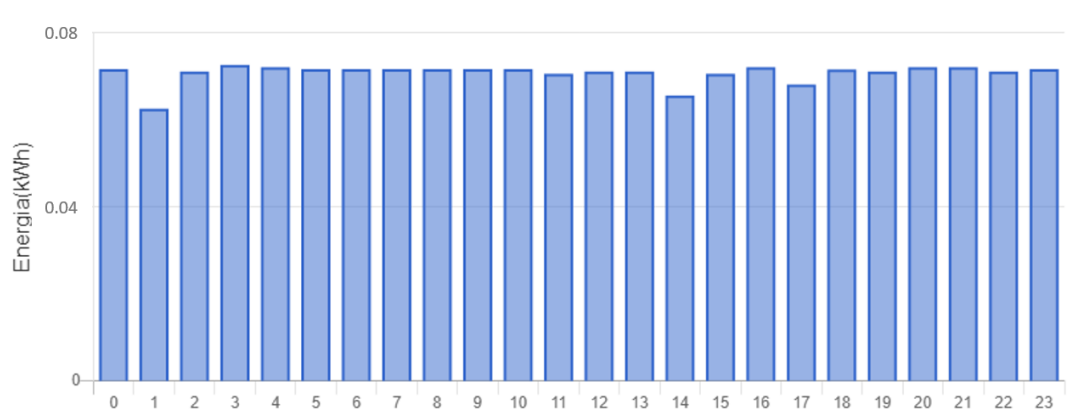
A partir da análise da Figura 14 é percebido um aumento no consumo de energia no período que vai de 9h a 20h, quando os pesquisadores chegam na unidade e acionam a maioria das cargas como ar condicionados, luzes e equipamentos. Contudo, pode-se observar também um consumo residual no momento em não há ninguém na instalação. Esse consumo apesar de não representar necessariamente desperdício de energia, pode ser avaliado pelo usuário a fim de verificar a possibilidade de economia de energia. Através da Figura 15 percebe-se também que a unidade é utilizada essencialmente durante a semana, permanecendo aos fins de semana com o mesmo consumo residual indicado anteriormente.

Figura 14 – Curva de consumo da unidade em dias úteis



Fonte: O próprio autor

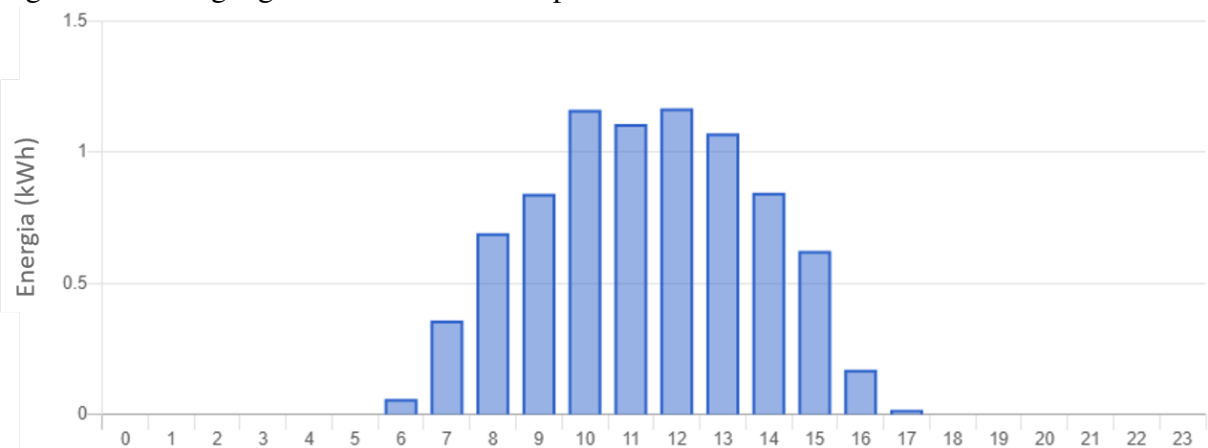
Figura 15 – Curva de consumo da unidade durante os fins de semana



Fonte: O próprio autor

Ao que diz respeito a energia gerada, a Figura 16 apresenta a geração no sexto dia do período de teste, em que as condições climáticas estavam favoráveis.

Figura 16 – Energia gerada no sexto dia do período de teste



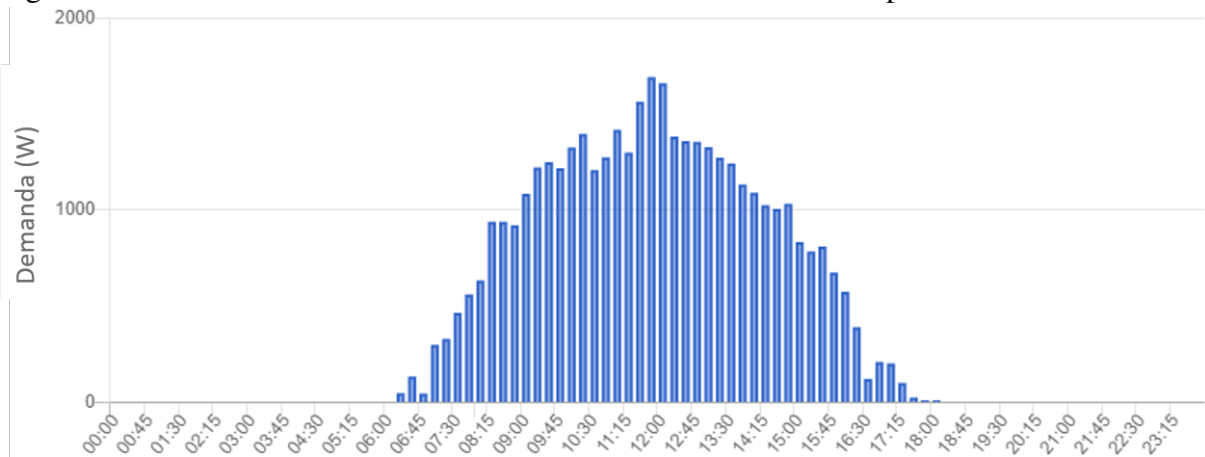
Fonte: O próprio autor

Pode-se observar uma conformidade do gráfico apresentado com o de um sistema

de geração solar típico. A continuidade apresentada indica que o sistema de geração está funcionando normalmente, sem problemas de desconexão ou falta de rede. Além disso, a quantidade de energia gerada por hora está de acordo com a irradiação solar em cada momento, o que indica que o sistema está fornecendo as informações corretas para cada horário.

De forma complementar, a máxima potência do sistema de geração também no sexto dia do período de teste é apresentado na Figura 17.

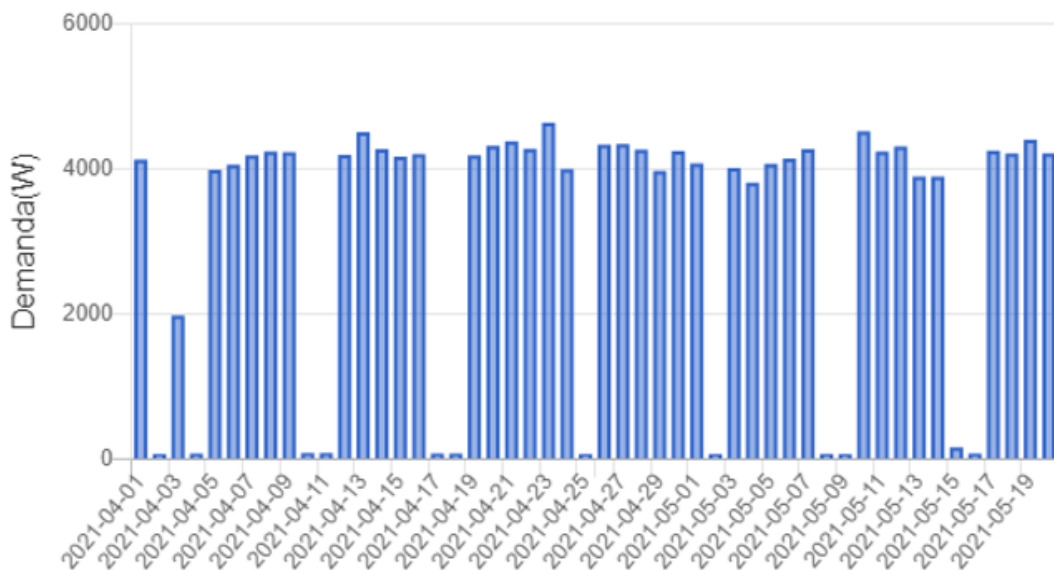
Figura 17 – Potência máxima do sistema fotovoltaico no sexto dia do período de teste



Fonte: O próprio autor

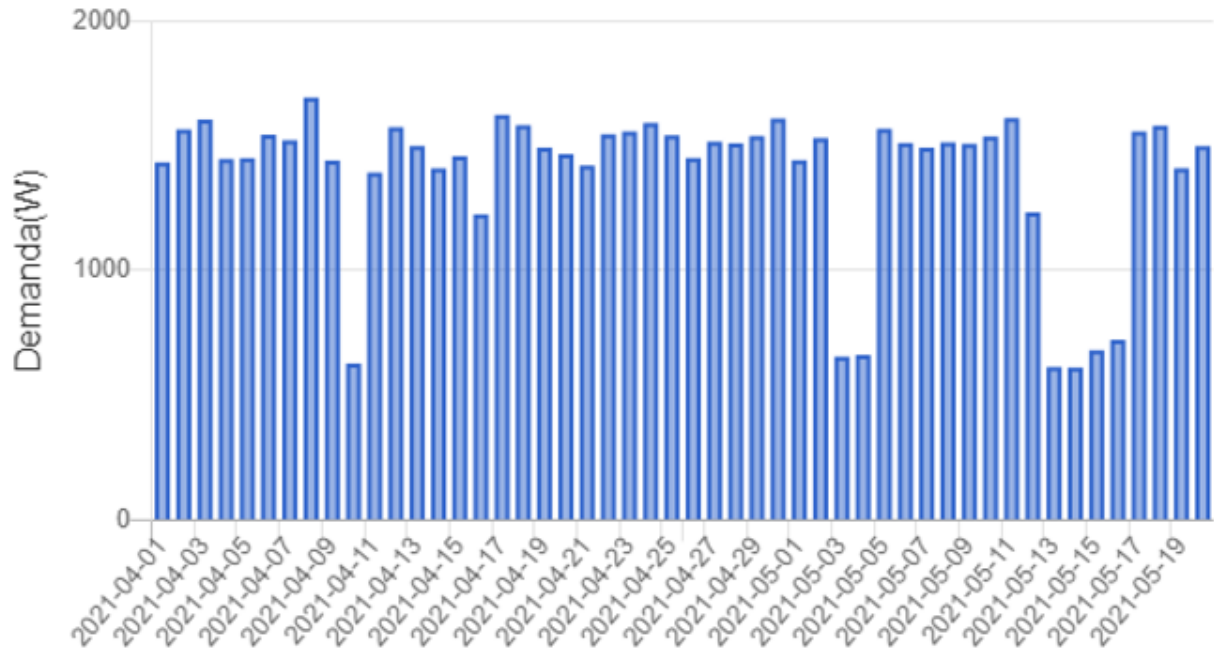
Já o gráfico de demanda da unidade e da potência máxima do sistema de geração, durante o período de teste, são apresentados nas Figuras 18 e 19, respectivamente.

Figura 18 – Demanda registrada durante o período de teste



Fonte: O próprio autor

Figura 19 – Potência máxima do sistema fotovoltaico durante o período de teste

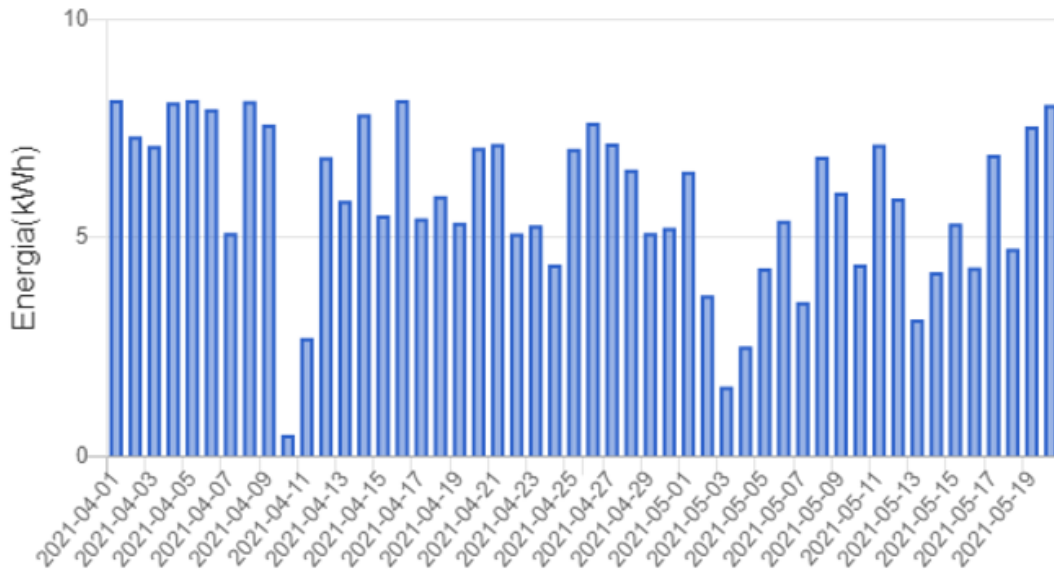


Fonte: O próprio autor

A partir da análise da Figura 18, percebe-se uma demanda aproximadamente constante durante os dias das semana. Foi registrado uma demanda máxima de pouco mais de 4,6 kW, que permanece próxima das demais medições obtidas durante o intervalo. Dessa forma, é possível uma previsibilidade da demanda contratada em torno desse valor. Já a potência máxima do sistema fotovoltaico, Figura 19, apresentou variações em alguns dias, decorrentes de períodos chuvosos, mas permaneceu na maior parte dos dias constante e próxima da potência nominal do inversor. Isso indica um bom dimensionamento dos painéis para o inversor instalado, assim como a elevada presença de recurso solar no período de testes.

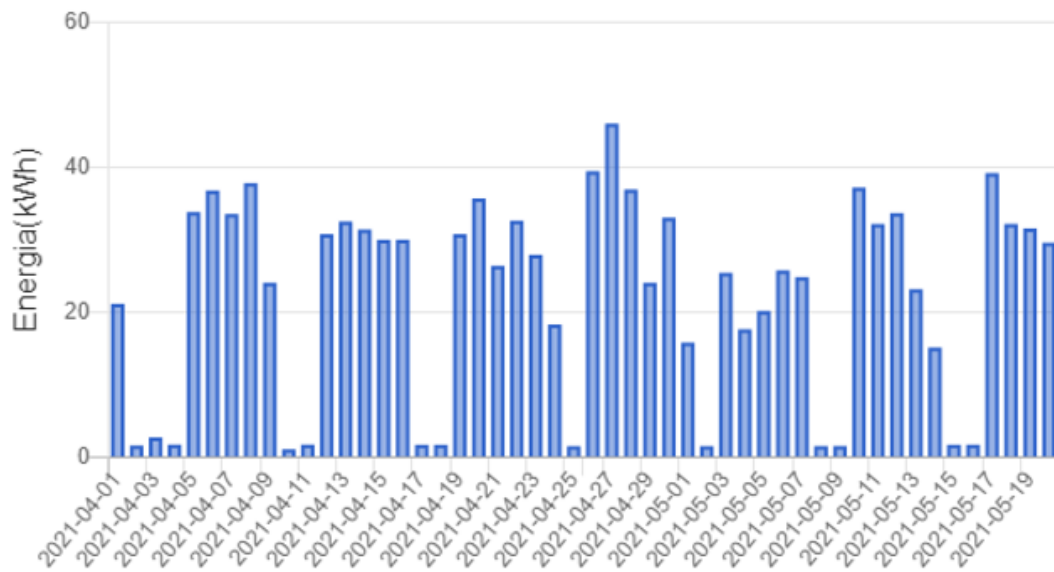
De modo geral, os gráficos da geração e do consumo obtidos durante o período de teste são apresentados nas Figuras 20 e 21.

Figura 20 – Geração registrada durante todo o período de teste



Fonte: O próprio autor

Figura 21 – Consumo registrado durante todo o período de teste



Fonte: O próprio autor

Durante o período de testes, foi visto uma variação significativa na geração do sistema, principalmente no período que compreende a primeira quinzena de Maio. Essa variação apesar de representar uma redução na geração do sistema, já é prevista por conta das condições climáticas, segundo aponta o calendário de chuvas do estado do Ceará (FUNCEME, 2021). É importante notar que os demais dias ficaram dentro do valor médio esperado de 7,85 kWh/dia previsto para um sistema de geração solar de 1,675 kWp, sendo apresentados em alguns dias valores até maiores que a média. A Equação 4.1, dada por Pinho e Galdino (2010), apresenta o

valor calculado com base em um rendimento de 80% e um HSP médio de 5,86 kWh/m² ao dia, dado pelo CRESESB (2021) para o município de Fortaleza-CE.

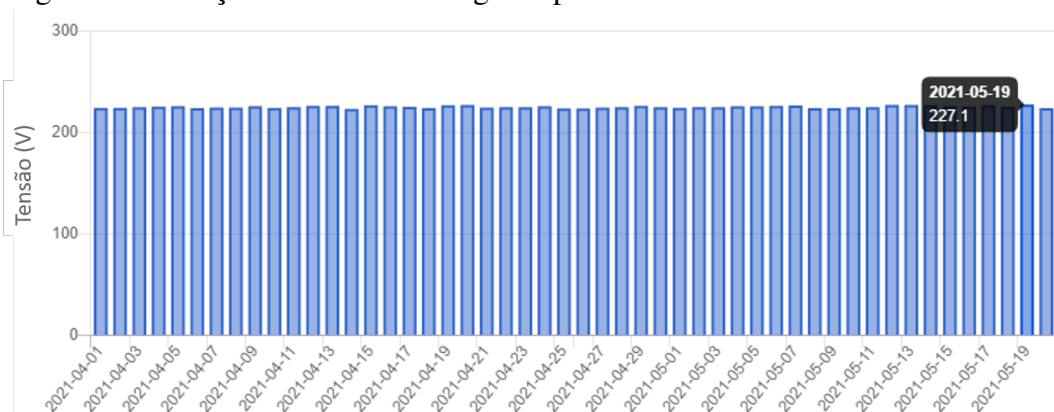
$$E_G(\text{kWh}/\text{dia}) = P_{FV} * HSP * TD \quad (4.1)$$

$$E_G(\text{kWh}/\text{dia}) = 1,675 * 5,86 * 0,8 = 7,85\text{kWh}/\text{dia} \quad (4.2)$$

O consumo, durante o período de testes, seguiu o comportamento esperado pelo gráfico de demanda da Figura 18, com valor de consumo bastante próximo durante os dias da semana, reduzindo consideravelmente aos fins de semana. Comparando a Figura 18 com a Figura 21, pode-se perceber ainda que o consumo registrado no segundo dia de medição (domingo), e de forma geral durante os fins de semana, foi proveniente apenas de equipamentos que permaneceram ligados enquanto a unidade estava fechada. Isso representa um potencial de redução de aproximadamente 1,72 kWh por dia, ou aproximadamente 51,6 kWh/mês, caso esses equipamentos permaneçam sempre ligados sem utilização (cargas não essenciais).

Adicionalmente foi avaliado a variação de tensão na unidade em relação ao valor nominal. A Figura 22 apresenta os valores com maior variação, obtidos durante o período de teste.

Figura 22 – Variação de tensão ao longo do período de teste



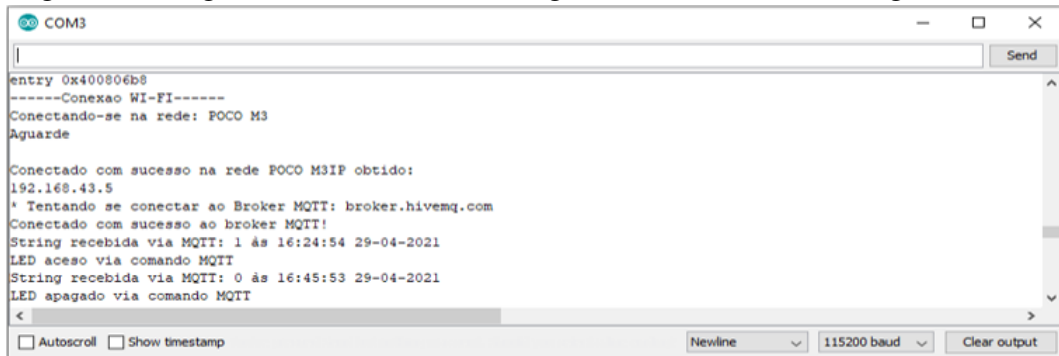
Fonte: O próprio autor

De modo geral, o gráfico apresenta variações máximas apenas acima do valor ideal, sendo registrado o valor de 227,1V no penúltimo dia de teste. Como estabelecido pelo módulo 8 do PRODIST, que permite uma variação máxima de tensão de 5% em relação ao ideal, esse valor está dentro do aceitável por norma, não oferecendo riscos aos equipamentos da unidade.

4.3 Validação do sistema de acionamento remoto

Dentro da perspectiva do trabalho proposto, o acionamento remoto de cargas permite maior conforto ao usuário, além de proporcionar economia de energia através do desligamento de cargas que foram esquecidas ligadas. Para validar o funcionamento desse sistema, foi realizado o acionamento de uma lâmpada de LED, em horário oportuno pelo usuário, via aplicação web. O comando foi realizado através da mudança de estado do *switch* correspondente. A Figura 23 apresenta, através do monitor serial, o recebimento do *payload* pelo microcontrolador e a lógica de acionamento da carga.

Figura 23 – Lógica de acionamento da carga, com horário das mensagens recebidas



```

entry 0x400806b8
-----Conexao WI-FI-----
Conectando-se na rede: POCO M3
Aguarde

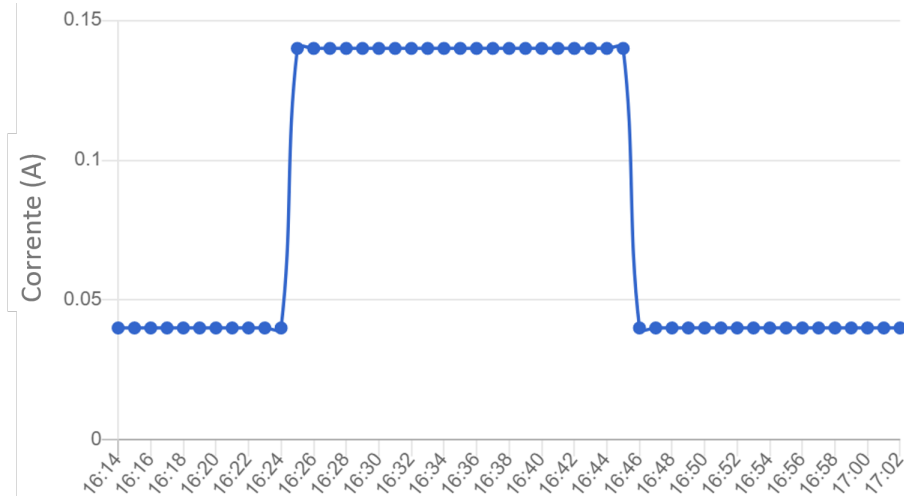
Conectado com sucesso na rede POCO M3IP obtido:
192.168.43.5
* Tentando se conectar ao Broker MQTT: broker.hivemq.com
Conectado com sucesso ao broker MQTT!
String recebida via MQTT: 1 às 16:24:54 29-04-2021
LED aceso via comando MQTT
String recebida via MQTT: 0 às 16:45:53 29-04-2021
LED apagado via comando MQTT

```

Fonte: O próprio autor

A fim de comprovar o acionamento da carga, a Figura 24 apresenta o gráfico da corrente medida no ponto de acionamento da lâmpada. Inicialmente apenas o próprio hardware de monitoramento estava ligado, sendo responsável pela corrente vista antes do acionamento. Ao receber o comando, pode-se observar um crescimento do gráfico, que indica que o acionamento da carga foi realizado de forma praticamente instantânea.

Figura 24 – Gráfico de corrente representando o acionamento da carga



Fonte: O próprio autor

4.4 Custos do Sistema Desenvolvido

Dentro da proposta apresentada, o hardware desenvolvido se mostrou como sendo composto por apenas quatro componentes: um módulo de medição PZEM-004T, um microcontrolador ESP32, uma placa de acionamento com relé de estado sólido, e uma fonte AC/DC de 220VAC para 5VDC. No caso de unidades com padrão de fornecimento trifásico podem ser incluídos mais módulos PZEM-004T para um mesmo ESP32, contudo a composição macro permanece a mesma. Dessa forma, o sistema proposto apresenta uma simplicidade de implementação, decorrente principalmente da modularidade dos componentes. Facilmente, um parte defeituosa pode ser substituída, permitindo o retorno do funcionamento em pouco tempo e com o mínimo de esforço humano.

O custo associado a cada um dos componentes que compõem o hardware do sistema é apresentado na Tabela 3. Os valores apresentados em reais são os menores obtidos em cotação com fornecedores internacionais.

Tabela 3 – Custo associado a cada um dos componentes do medidor inteligente proposto.

Dispositivo (Un.)	Valor (R\$)
PZEM-004T	23,97
ESP32	20,34
Módulo Relé	3,99
Fonte 220VAC/5VDC	5,20

Fonte: O próprio autor

Dessa forma, o medidor monofásico, já com o hardware de acionamento para uma carga, possui preço médio de pouco menos de R\$ 54. Enquanto, o medidor trifásico, também com o hardware para acionamento de uma carga, fica em torno de R\$ 102. Como são necessários dois medidores, um para consumo e outro para geração tem-se a previsão de aproximadamente R\$ 108 para unidades monofásicas e R\$ 204 para unidades trifásicas.

Para a produção em larga escala, a partir de 10 mil unidades, é possível ainda uma redução no valor dos equipamentos. Nessa condição, o medidor monofásico fica com preço médio em torno R\$ 46 enquanto o medidor trifásico em torno de R\$ 81.

Além dos custo com hardware, foi necessário a contratação de um serviço de VPS, com as configurações mínimas disponíveis (1 GB de RAM e 25 GB de ROM), no valor de 5 USD mensais. De acordo com a armazenagem contratada, estima-se que seja possível manter as medições de até 40 dispositivos, enviando dados a cada 15 minutos, pelo período de até 5 anos, sem a necessidade de contratar recursos adicionais.

É importante destacar que os custos apresentados acima dizem respeito apenas ao hardware necessário e a hospedagem da aplicação. Para o desenvolvimento de um produto final são necessários outros custos que não foram incluídos no total apresentado.

4.5 Conclusão

Neste capítulo foi apresentado os resultados experimentais do sistema proposto. A fim de validar sua operabilidade, dois medidores sendo um monofásico para medição da energia gerada, e um trifásico para medição de consumo foram instalados no laboratório do GPEC, em Fortaleza-CE. Através dos dados coletados durante os 50 dias de teste pode-se perceber que a unidade é frequentada majoritariamente durante a semana. Durante os dias de uso, a curva de carga cresce normalmente de 9h as 20h quando os pesquisadores chegam na unidade. Nos demais horários foi percebido consumo de energia residual, o que indica que equipamentos permanecem ligados mesmo quando não há ninguém na unidade. Considerando que esses equipamentos permaneçam ligados o dia todo e que se tratem de cargas não essenciais, o sistema foi capaz de indicar um potencial de redução de pouco mais de 50 kWh por mês. No que diz respeito a geração, foi observado, nos dias de teste uma geração do sistema condizente com o média esperada, acontecendo em alguns dias até mesmo a ultrapassagem desse valor.

A precisão do sistema foi validada através da medição de cargas com valores entre 60W e 2000W, sendo apresentado uma boa precisão se comparado as medições do multímetro

Fluke 177. O desvio apresentado, abaixo de 0,5%, ficou dentro do previsto no datasheet do módulo PZEM-004T. Assim como a precisão, o acionamento remoto das cargas também apresentou resultado satisfatório, ligando e desligando uma lâmpada LED, utilizada para teste, quase que instantaneamente.

Por fim, o custo do sistema desenvolvido foi analisado. Foi visto que o valor pode mudar de acordo com o padrão de fornecimento da unidade. De qualquer forma, os valores obtidos foram suficientes para que o sistema fosse considerado de baixo custo.

5 CONCLUSÕES E TRABALHOS FUTUROS

Neste trabalho foi proposto um sistema para monitoramento e gerenciamento de unidades com geração distribuída utilizando o protocolo de comunicação leve MQTT. Dentro dos objetivos desse trabalho, foi proposto o auxílio ao prosumidor no acompanhamento do consumo e geração do sistema, além do acionamento remoto de cargas dentro da unidade.

Durante o trabalho foi visto como a geração distribuída tem sido determinante para a mudança de uma matriz energética predominante poluente, em uma onde as fontes renováveis tem tido cada vez mais crescimento. Foi visto que a maioria dos países internacionais estudados favorecem a implantação da geração distribuída através de políticas públicas. Assim também fez o Brasil, que com a implantação da RN nº 482/2012 possibilitou ao consumidores que produzissem sua própria energia, e injetassem o excedente na rede, rendendo créditos a serem compensados no valor consumido ou em outras unidades consumidoras de mesmo dono e mesma concessionária. Foi visto que, com o medidor bidirecional, no entanto, o consumidor perde a informação do valor real consumido e gerado. Dificultando assim o gerenciamento energético da unidade.

Como apresentado, a medição inteligente tem papel importante na implantação da geração distribuída, fornecendo uma maior ciência ao consumidor sobre o que é consumido e permitindo a concessionárias e agências governamentais o maior controle do recurso energético. Esse recurso informativo, se mostrou ser capaz de reduzir o consumo a nível residencial, entregar ao consumidor uma melhor qualidade de energia, e aumentar a eficiência do sistema de transmissão e distribuição como um todo.

Desse modo, a estrutura geral de um medidor inteligente foi estudado em detalhes, merecendo destaque os protocolos de comunicação. Dentre os apresentados, o MQTT foi escolhido para este trabalho por ser um protocolo leve, propiciando um melhor funcionamento da aplicação mesmo em cenários onde a conexão não é tão estável.

Dentro do escopo proposto, foi apresentada em detalhes as partes que compõem este trabalho, sendo divididos de forma macro em um hardware de medição, um módulo de acionamento por relé, o protocolo de comunicação MQTT, uma aplicação Web, um banco de dados PostgreSQL e usuários conectados.

Pode-se perceber com a instalação do sistema no laboratório do GPEC que existe um potencial de economia de aproximadamente 50 kWh ao mês, decorrente de equipamentos que permanecem ligados continuamente na unidade. Esse valor apesar de não significar necessaria-

mente desperdício de energia, pode ser avaliado pelo usuário de forma reduzir o consumo mensal. Foi possível perceber também uma boa geração do sistema fotovoltaico instalado, apesar de haver a necessidade de um aumento na potência do sistema para suprir integralmente o consumo da unidade.

Durante o período de teste, o sistema de acionamento remoto se mostrou funcional, com resposta rápida aos comandos realizados pela aplicação web. A ferramenta pode ser uma boa opção para o desligamento de cargas que são esquecidas ligadas, ou simplesmente para um maior conforto do usuário.

Apesar de ser desenhado para a instalação na saída do quadro distribuição ou na conexão do inversor com a rede, o sistema desenvolvido pode ser utilizado também setorialmente dentro da unidade, ou até mesmo para medição e acionamento de um equipamento em isolado.

Entre as limitações do trabalho está a tensão de fornecimento e a máxima corrente de entrada do medidor utilizado, sendo necessário a instalação em baixa tensão e respeitando o limite de 100A, o que para algumas aplicações, principalmente industriais pode ser uma limitação. Além disso, o medidor utilizado não realiza a medição de energia reativa, sendo o fator de potência também limitado em valores positivos. Desse modo, o sistema proposto não se torna tão adequado para unidades que necessitem de uma maior resolubilidade nesse sentido .

De modo geral, o sistema atende a dor do usuário, apresentando uma forma de reduzir o consumo de energia através do *feedback* em tempo real, sendo ainda um sistema de baixo custo.

5.1 Trabalhos Futuros

Espera-se em trabalhos futuros implementar um mecanismo para acionamento automático de cargas, a depender da ultrapassagem de demanda, de consumo, ou pelo agendamento do próprio usuário. A fim de melhorar a integração do usuário com a aplicação, pretende-se realizar o desenvolvimento de um calculador de tarifas que apresente de forma simples ao usuários a fatura de energia mensal utilizando os valores obtidos pela aplicação web. O intuito é a comparação dos valores obtidos com os da concessionária de energia.

A fim de tornar o projeto proposta uma solução comercial existe a necessidade de realizar a encriptação dos dados entre o broker MQTT e os usuários conectados, proporcionando uma segurança da informação e proteção contra acionamentos indevidos das cargas.

Por fim, planeja-se a instalação de mais protótipos em residências e pequenas

indústrias, a fim de avaliar o comportamento do usuário em relação ao sistema desenvolvido, e como esse fator pode impactar na redução do consumo e utilização da energia gerada.

REFERÊNCIAS

- AL-ALI, A.; ZUALKERNAN, I. A.; RASHID, M.; GUPTA, R.; ALIKARAR, M. A smart home energy management system using iot and big data analytics approach. **IEEE Transactions on Consumer Electronics**, v. 63, n. 4, p. 426–434, 2017.
- AL-FUQAHA, A.; GUIZANI, M.; MOHAMMADI, M.; ALEDHARI, M.; AYYASH, M. Internet of things: A survey on enabling technologies, protocols, and applications. **IEEE Communications Surveys Tutorials**, v. 17, n. 4, p. 2347–2376, 2015.
- ANEEL. **Resolução Normativa nº 414/2010**. 2010. Disponível em: <https://aneel.gov.br/documents/656877/14486448/bren2010414.pdf>. Acesso em: 25 Abr. 2021.
- ANEEL. **Resolução Normativa Nº 482**. 2012. Disponível em: <http://www2.aneel.gov.br/cedoc/ren2012482.pdf>. Acesso em: 04 fev. 2021.
- ANEEL. **Resolução Normativa Nº 687**. 2015. Disponível em: <http://www2.aneel.gov.br/cedoc/ren2015687.pdf>. Acesso em: 04 fev. 2021.
- ANEEL. **Tarifa branca**. 2015. Disponível em: <https://www.aneel.gov.br/tarifa-branca>. Acesso em: 07 jun. 2021.
- ANEEL. **Módulo 3 – Acesso ao Sistema de Distribuição**. 2017. Disponível em: <https://bit.ly/3e9pvfT>. Acesso em: 04 mar. 2021.
- ANEEL. **Nota Técnica 0056/2017-SRD/ANEEL**. 2017. Disponível em: <https://bit.ly/3p8Lu8F>. Acesso em: 04 fev. 2021.
- ANEEL. **Revisão das regras de geração distribuída entra em consulta pública**. 2019. Disponível em: <http://bit.ly/3bffP1v>. Acesso em: 04 mar. 2021.
- ANEEL. **Geração Distribuída**. 2020. Disponível em: <http://bit.ly/power-bi-aneel>. Acesso em: 28 Jan. 2021.
- ANEEL. **Tarifa branca é opção para consumidores em 2020**. 2020. Disponível em: https://www.aneel.gov.br/sala-de-imprensa-exibicao/-/asset_publisher/XGPXSqdmFHRrE/content/tarifa-branca-e-opcao-para-consumidores-em-2020/656877. Acesso em: 05 mai. 2021.
- ANEEL. **Geração Distribuída - Busca por data de conexão**. 2021. Disponível em: <http://www2.aneel.gov.br/scg/gd/VerGD.asp>. Acesso em: 04 mar. 2021.
- AQUINO, A. T.; ALMEIDA, B. R.; GUIMARÃES, J. S.; PACHECO, J. d. O. Desenvolvimento de um sistema de baixo custo para monitoramento remoto do consumo de energia elétrica. **Congresso Brasileiro de Automática**, 09 2018.
- BARAI, G. R.; KRISHNAN, S.; VENKATESH, B. Smart metering and functionalities of smart meters in smart grid - a review. In: **2015 IEEE Electrical Power and Energy Conference (EPEC)**. [S. l.: s. n.], 2015. p. 138–145.
- BRASIL, G. do. **Governo zera imposto de importação de equipamentos de energia solar**. 2020. Disponível em: <http://bit.ly/2PAAt98z>. Acesso em: 04 mar. 2021.

CANO, C.; PITTOLO, A.; MALONE, D.; LAMPE, L.; TONELLO, A. M.; DABAK, A. G. State of the art in power line communications: From the applications to the medium. **IEEE Journal on Selected Areas in Communications**, v. 34, n. 7, p. 1935–1952, 2016.

COMISSION, E. **The roadmap for transforming the EU into a competitive, low-carbon economy by 2050**. 2021. Disponível em: https://ec.europa.eu/clima/sites/clima/files/2050_roadmap_en.pdf. Acesso em: 04 mar. 2021.

CORREIA, A. N. F. **Aplicações de Sistema de Monitoramento de Consumo de Energia Elétrica para Unidades Conectadas em Média Tensão**. Dissertação (Mestrado) – Universidade Federal do Ceará, 2019.

CRESESB. **Sundata v3.0**. 2021. Disponível em: <http://www.cresesb.cepel.br/index.php?section=sundata&>. Acesso em: 06 Mai. 2021.

ELETRA. **Catálogo Linha L BR**. 2021. Disponível em: <http://www.eletraenergy.com/images/catalogos/Catlogo-Linha-L.pdf>. Acesso em: 12 jun. 2021.

FAISAL, M.; KARIM, T. F.; PAVEL, A. R.; HOSSEN, M. S.; LIPU, M. H. Development of smart energy meter for energy cost analysis of conventional grid and solar energy. In: **2019 International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST)**. [S. l.: s. n.], 2019. p. 91–95.

FLUKE. **175, 177, 179 - True-rms Multimeters**. 2015. Disponível em: https://dam-assets.fluke.com/s3fs-public/175_____umpor0200.pdf. Acesso em: 07 jun. 2021.

FUNCEME. **Calendário de chuvas do estado do Ceará**. 2021. Disponível em: <http://www.funceme.br/app-calendario/diario/municipios/maxima/2021/6>. Acesso em: 06 Mai. 2021.

GARCIA, F. D.; MARAFÃO, F. P.; SOUZA, W. A. de; SILVA, L. C. Pereira da. Power metering: History and future trends. In: **2017 Ninth Annual IEEE Green Technologies Conference (GreenTech)**. [S. l.: s. n.], 2017. p. 26–33.

HUNKELER, U.; TRUONG, H. L.; STANFORD-CLARK, A. Mqtt-s — a publish/subscribe protocol for wireless sensor networks. In: **2008 3rd International Conference on Communication Systems Software and Middleware and Workshops (COMSWARE '08)**. [S. l.: s. n.], 2008. p. 791–798.

IRMAK, E.; KöSE, A.; GöçMEN, G. Simulation and zigbee based wireless monitoring of the amount of consumed energy at smart homes. In: **2016 IEEE International Conference on Renewable Energy Research and Applications (ICRERA)**. [S. l.: s. n.], 2016. p. 1019–1023.

ISE, F. **Net electricity generation in Germany in 2020**. 2021. Disponível em: <https://energy-charts.info/?l=de&c=DE>. Acesso em: 03 mar. 2021.

JONES, A.; IRWIN, M.; IZADIAN, A. Incentives for microgeneration development in the u.s. and europe. In: . [S. l.: s. n.], 2010. p. 3018 – 3021.

KODALI, R. K.; SORATKAL, S. Mqtt based home automation system using esp8266. In: **2016 IEEE Region 10 Humanitarian Technology Conference (R10-HTC)**. [S. l.: s. n.], 2016. p. 1–5.

KUROSAWA, S.; INOUE, T.; TSUKAMOTO, Y. Smart metering infrastructure in japan. In: . [S. l.: s. n.], 2015. p. 48 (5 .)–48 (5 .).

KURUPPU, K.; GUNATHILAKA, K.; WIJESINGHE, W.; THAYAPARAN, S. Power line communication for home automation in low income households. In: **2017 3rd International Conference on Power Generation Systems and Renewable Energy Technologies (PGSRET)**. [S. l.: s. n.], 2017. p. 28–31.

MAIN, M. **Rails Request-Response Cycle**. 2015. Disponível em: [https://https://medium.com/@matthewmain/rails-request-response-cycle-819e9cd8fa4e](https://medium.com/@matthewmain/rails-request-response-cycle-819e9cd8fa4e). Acesso em: 07 jun. 2021.

METERING, A. S.; VISALATCHI, S.; SANDEEP, K. K. Smart energy metering and power theft control using arduino gsm. In: **2017 2nd International Conference for Convergence in Technology (I2CT)**. [S. l.: s. n.], 2017. p. 858–961.

MORENO-MUNOZ, A.; ROSA, J. J. G. de la; LOPEZ, M.; CASTRO, A. R. Gil de. Grid interconnection of distributed generation: The spanish normative. In: **2009 International Conference on Clean Electrical Power**. [S. l.: s. n.], 2009. p. 466–470.

PEACEFAIR. **PZEM-004T V3.0 User Manual**. 2020. Disponível em: <https://bit.ly/3qc20G9>. Acesso em: 02 fev. 2021.

PINHO, J.; GALDINO, M. A. **Manual de Engenharia para Sistemas Fotovoltaicos**. [S. l.: CEP/EL/CRESESB, 2010. v. 2014.

SAMUHASILP, J.; PORA, W. Development of an automatic meter reading and outage management system using lorawan technology. In: **2018 IEEE 5th International Conference on Smart Instrumentation, Measurement and Application (ICSIMA)**. [S. l.: s. n.], 2018. p. 1–4.

SHIBU, N. B. S.; HANUMANTHIAH, A.; ROHITH, S. S.; YASWANTH, C.; KRISHNA, P. H.; PAVAN, J. V. S. Development of iot enabled smart energy meter with remote load management. In: **2018 IEEE International Conference on Computational Intelligence and Computing Research (ICIC)**. [S. l.: s. n.], 2018. p. 1–4.

SILVA, L. L. Internet of things : Pros and cons of coap protocol solution for small devices. In: . [S. l.: s. n.], 2016.

SIMOES, M.; ROCHE, R.; KYRIAKIDES, E.; MIRAOU, A.; BLUNIER, B.; MCBEE, K.; SURYANARAYANAN, S.; NGUYEN, P.; RIBEIRO, P. Smart-grid technologies and progress in europe and the usa. **Journal of Applied Probability - J APPL PROBAB**, 09 2011.

SREEDEVI, S. V.; PRASANNAN, P.; JIJU, K.; LEKSHMI, I. J. I. Development of indigenous smart energy meter adhering indian standards for smart grid. In: **2020 IEEE International Conference on Power Electronics, Smart Grid and Renewable Energy (PESGRE2020)**. [S. l.: s. n.], 2020. p. 1–5.

SURYANARAYANAN, S.; RIBEIRO, P. F.; SIMÕES, M. G. Grid modernization efforts in the usa and brazil - some common lessons based on the smart grid initiative. In: **IEEE PES General Meeting**. [S. l.: s. n.], 2010. p. 1–5.

UENO, T.; SANO, F.; SAEKI, O.; TSUJI, K. Effectiveness of an energy-consumption information system on energy savings in residential houses based on monitored data. **Applied Energy**, v. 83, n. 2, p. 166–183, 2006. ISSN 0306-2619. Disponível em: <https://www.sciencedirect.com/science/article/pii/S030626190500019X>.

UY, N. Q.; NAM, V. H. A comparison of amqp and mqtt protocols for internet of things. In: **2019 6th NAFOSTED Conference on Information and Computer Science (NICS)**. [S. l.: s. n.], 2019. p. 292–297.

VASSILEVA, I.; DAHLQUIST, E.; WALLIN, F.; CAMPILLO, J. Energy consumption feedback devices' impact evaluation on domestic energy use. **Applied Energy**, v. 106, p. 314–320, 2013. ISSN 0306-2619. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0306261913000688>.

WANG, F.; YANG, X.; ZHANG, L.; ZHANG, Y. Design of environmental monitoring system based on xmpp. In: **Proceedings of 2013 2nd International Conference on Measurement, Information and Control**. [S. l.: s. n.], 2013. v. 01, p. 509–512.

APÊNDICE A – CÓDIGOS-FONTES UTILIZADOS NO HARDWARE DE MEDIÇÃO

Código-fonte 1 – Firmware medidor monofásico

```
1 #include <PZEM004Tv30.h>
2 #include <WiFi.h>
3 #include <PubSubClient.h>
4 #include <ESPmDNS.h>
5 #include <WiFiUdp.h>
6 #include <ArduinoOTA.h>
7 #include <EEPROM.h>
8 #include "time.h"
9 #include "FS.h"
10 #include "SPIFFS.h"
11 #include <HardwareSerial.h>
12
13 #define PIN_LED      2 // porta I/O associada a carga
14 // topico da carga cadastrada
15 #define TOPICO_SUBSCRIBE_LED "testUN04C_topic_00"
16 // topico para envio das medicoes
17 #define TOPICO_PUBLISH_M   "test_le_medida"
18 // topico para envio de logs
19 #define TOPICO_LOGS "test_logs"
20 // ID unico dos dispositivo
21 #define ID_MQTT   "testUN04C"
22 // topico para envio das mensagens keep alive
23 #define WILLTOPIC  "test_keep_alive"
24 // condicao caso SPIFFS falhe
25 #define FORMAT_SPIFFS_IF_FAILED true
26 // Definido com base no id gerado ao criar o dispositivo
    no DB da App. Web
27 #define DV_ID "4"
```



```
28 // mensagem enviada ao broker para o dispositivo online
29 #define ID_ON "true,4"
30 // mensagem enviada ao broker desconexao
31 #define ID_OFF "false,4"
32 // tempo para se conectar ao wifi antes de iniciar outra
    tentativa
33 #define WIFI_TIMEOUT 15000
34
35 float previous_energy; // valor anterior de energia
36 float max_power; // maximo valor de potencia
37 int message_count; // contador de mensagens
38
39
40 const char* SSID = "GPEC-2.4"; // SSID / nome da rede WI-FI
    que deseja se conectar
41 const char* PASSWORD = "xxxxxxxx"; // Senha da rede WI-FI
    que deseja se conectar
42
43
44 const char* BROKER_MQTT = "xxxx"; //URL do broker MQTT que
    se deseja utilizar
45 int BROKER_PORT = 1883; // Porta do Broker MQTT
46
47 WiFiClient espClient; // Cria o objeto espClient
48 PubSubClient MQTT(espClient); // Instancia o Cliente MQTT
    passando o objeto espClient
49 // HardwareSerial MeterSerial(2);
50 PZEM004Tv30 pzem(&Serial2); //Conecta ao Serial2 do ESP32 (
    TX-17 e RX-16)
51
52
```

```
53 const char* ntpServer = "a.st1.ntp.br"; // servidor para
    obtencao do horario real
54 // UTC Time
55 const long  gmtoffset_sec = -10800; // fuso horario do
    Brasil
56 const int   daylightOffset_sec = 0; //
57
58 char* minutos; // variavel para armazenar o minuto atual
59
60 // Funcao: inicializa e conecta-se na rede WI-FI desejada
61 char* getMinute() {
62     struct tm timeinfo;
63     if(!getLocalTime(&timeinfo)){
64         Serial.println("Failed to obtain time");
65         return NULL;
66     }
67     char* timeMinute = (char*)malloc(3);
68     strftime(timeMinute,3, "%M", &timeinfo);
69     return timeMinute;
70 }
71 // Funcao: Cria um arquivo no diretorio SPIFFS
72 void writeFile(fs::FS &fs, const char * path, const char *
    message){
73     Serial.printf("Writing file: %s\r\n", path);
74
75     File file = fs.open(path, FILE_WRITE);
76     if(!file){
77         Serial.println("- failed to open file for writing")
            ;
78         return;
79     }
80     if(file.print(message)){
```

```
81         Serial.println("- file written");
82     } else {
83         Serial.println("- write failed");
84     }
85     file.close();
86 }
87 // Funcao: Acrescenta uma mensagem no arquivo selecionado
88 void appendFile(fs::FS &fs, const char * path, const char *
89     message){
90     Serial.printf("Appending to file: %s\r\n", path);
91     File file = fs.open(path, FILE_APPEND);
92     if(!file){
93         Serial.println("- failed to open file for appending
94             ");
95         return;
96     }
97     if(file.print(message)){
98         Serial.println("- message appended");
99     } else {
100         Serial.println("- append failed");
101     }
102     file.close();
103 }
104 // Funcao: Deleta um arquivo no diretcario SPIFFS
105 void deleteFile(fs::FS &fs, const char * path){
106     Serial.printf("Deleting file: %s\r\n", path);
107     if(fs.remove(path)){
108         Serial.println("- file deleted");
109     } else {
110         Serial.println("- delete failed");
111     }
112 }
```

```
111 }
112 //Funcao: inicia a conexao Wifi e aguarda 15 segundos pela
      conexao
113 void startWifi(void){
114
115     Wifi.disconnect(true);
116     Wifi.mode(WIFI_OFF);
117
118     delay(10);
119     Serial.println("-----Conexao WI-FI-----");
120     Serial.print("Conectando-se na rede: ");
121     Serial.println(SSID);
122     Serial.println("Aguarde");
123
124     Wifi.mode(WIFI_STA);
125     Wifi.begin(SSID, PASSWORD); // Conecta na rede WI-FI
126
127     unsigned long startAttemptTime = millis();
128
129     // Aguarda a conexao com o Wifi
130     while (Wifi.status() != WL_CONNECTED &&
131           millis() - startAttemptTime < WIFI_TIMEOUT){
132         delay(10);
133     }
134
135 }
136 // Funcao: inicializa e conecta-se na rede WI-FI desejada
137 void initWifi(void) {
138     // Inicia a conexao com a rede Wifi e aguarda 15
      segundos pela conexao
139     startWifi();
140
```

```
141 // caso algum problema ocorra um nova tentativa e
    realizada
142 while (WiFi.status() != WL_CONNECTED)
143 {
144     Serial.print(" Failed to Connect");
145     startWifi(); // Inicia novamente a conexao Wifi.
146 }
147
148 Serial.println();
149 Serial.print("Conectado com sucesso na rede ");
150 Serial.print(SSID);
151 Serial.println("IP obtido: ");
152 Serial.println(WiFi.localIP());
153 }
154
155 // Funcao: inicializa parametros de conexao MQTT
156 void initMQTT(void) {
157     MQTT.setServer(BROKER_MQTT, BROKER_PORT); //informa
        qual broker e porta deve ser conectado
158     MQTT.setCallback(mqtt_callback); //atribui
        funcao de callback (funcao chamada quando qualquer
        informacao de um dos topicos subscritos chega)
159     reconnectMQTT();
160 }
161
162 // Funcao: funcao de callback, chamada sempre que um
    informacao do topico chega
163 void mqtt_callback(char* topic, byte* payload, unsigned int
    length) {
164     String msg;
165
166     /* obtem a string do payload recebido */
```

```
167     for(int i = 0; i < length; i++)
168     {
169         char c = (char)payload[i];
170         msg += c;
171     }
172
173     Serial.print("Chegou a seguinte string via MQTT: ");
174     Serial.println(msg);
175
176     /* toma acao dependendo da string recebida */
177     if (msg.equals("1"))
178     {
179         digitalWrite(PIN_LED, HIGH);
180         Serial.print("LED aceso mediante comando MQTT");
181         // salva na EEPROM o estado recebido
182         EEPROM.write(0, HIGH);
183         EEPROM.commit();
184     }
185
186     if (msg.equals("0"))
187     {
188         digitalWrite(PIN_LED, LOW);
189         Serial.print("LED apagado mediante comando MQTT");
190         EEPROM.write(0, LOW);
191         EEPROM.commit();
192     }
193 }
194
195 // Funcao: reconecta-se ao broker MQTT
196 void reconnectMQTT(void) {
197     while (!MQTT.connected())
198     {
```

```
199     Serial.print("* Tentando se conectar ao Broker MQTT
      : ");
200     Serial.println(BROKER_MQTT);
201     if (MQTT.connect(ID_MQTT, NULL, NULL, WILLTOPIC, 0,
      false, ID_OFF))
202     {
203         Serial.println("Conectado com sucesso ao broker
      MQTT!");
204         MQTT.subscribe(TOPICO_SUBSCRIBE_LED);
205     }
206     else
207     {
208         Serial.println("Falha ao reconectar no broker."
      );
209         Serial.println("Havera nova tentatica de
      conexao em 2s");
210         delay(2000);
211     }
212 }
213 }
214 // Funcao: armazena a medicao atual no arquivo datalogger.
      txt
215 void sendSPIFFSMeasures (void) {
216     // Verifica se o arquivo foi criado
217     if (!SPIFFS.exists("/datalogger.txt")) return;
218
219     File file = SPIFFS.open("/datalogger.txt");
220     if(!file || file.isDirectory()){
221         Serial.println("- failed to open file for reading")
      ;
222         return;
223     }
```

```
224
225     //verifica se foi preenchido algum valor do arquivo e
        pula o titulo
226     if (!file.seek(14, SeekSet)) return;
227
228     char buffer[64];
229     while (file.available()) {
230         int l = file.readBytesUntil('\r', buffer, sizeof(
                buffer)-1);
231         buffer[l] = 0;
232         MQTT.publish (TOPICO_PUBLISH_M, buffer); // e se o
                dispositivo se desconectar durante esse processo
                ?
233         Serial.println(buffer);
234         delay(1000);
235     }
236     deleteFile(SPIFFS, "/datalogger.txt");
237 }
238
239 // Funcao: verifica o estado das conexoes WiFi e do broker
        MQTT
240 // E tenta a reconexao, caso nao consiga inicia a salvar as
        mensagens
241 // no diretoria SPIFFIS
242 void VerificaConexoesWiFiEMQTT(void){
243     if (!MQTT.connected()){
244         reconnectWiFi(); //se nao ha conexao com o WiFi, a
                conexao e refeita
245         reconnectMQTT(); //se nao ha conexao com o Broker, a
                conexao e refeita
246         sendSPIFFSMeasures();
247         MQTT.publish (WILLTOPIC, ID_ON);
```



```
248
249     }
250 }
251 // Recebe as ultimas medicoes e formata para o envio ao
    broker
252 char* takeMeasures(void) {
253
254     char voltage_str[10] = {0};
255     char current_str[10] = {0};
256     char power_str[10] = {0};
257     char energy_str[10] = {0};
258     char frequency_str[10] = {0};
259     char pf_str[10] = {0};
260     char MBuffer [58] = {0};
261     char *medida_str = (char*)malloc(58);
262
263     float voltage = pzem.voltage();
264     if(!isnan(voltage)){
265         sprintf(voltage_str, "%.1f", voltage);
266     }
267
268     float current = pzem.current();
269     if(!isnan(current)){
270         sprintf(current_str, "%.2f", current);
271     }
272
273     // pega o valor maximo medido e formata como char
274     sprintf(power_str, "%.1f", max_power);
275     max_power = 0;
276
277     float now_energy = pzem.energy();
278     if(!isnan(now_energy)){
```

```
279     float delta_energy = now_energy - previous_energy;
280     sprintf(energy_str, "%.3f", delta_energy);
281     // reseta o contador de energia quando estiver
        acima de 9000 kWh evitando erros de overflow
282     if (now_energy > 9000) {
283         pzem.resetEnergy();
284         previous_energy = 0;
285     } else {
286         previous_energy = now_energy;
287     }
288 }
289
290 float frequency = pzem.frequency();
291 if(!isnan(frequency)){
292     sprintf(frequency_str, "%.1f", frequency);
293 }
294
295 float pf = pzem.pf();
296 if(!isnan(pf)){
297     sprintf(pf_str, "%.2f", pf);
298 }
299
300     strcat(MBuffer, voltage_str);
301     strcat(MBuffer, ",");
302     strcat(MBuffer, current_str);
303     strcat(MBuffer, ",");
304     strcat(MBuffer, power_str);
305     strcat(MBuffer, ",");
306     strcat(MBuffer, energy_str);
307     strcat(MBuffer, ",");
308     strcat(MBuffer, frequency_str);
309     strcat(MBuffer, ",");
```

```
310     strcat(MBuffer, pf_str);
311     strcat(MBuffer, ",");
312     // incluir aqui o id do dispositivo
313     strcat(MBuffer, DV_ID);
314     strcat(MBuffer, ",");
315     char* fullDate = getTime();
316     strcat(MBuffer, fullDate);
317     strcpy(medida_str, MBuffer);
318     return medida_str;
319
320 }
321 // Funcao: Tenta se reconectar enquanto armazena as
    medicoes no diretorio SPIFFS
322 void reconnectWiFi(void) {
323
324     WiFi.disconnect(true);
325     WiFi.mode(WIFI_OFF);
326
327     WiFi.mode(WIFI_STA);
328     WiFi.begin(SSID, PASSWORD); // Conecta na rede WI-FI
329     delay(500);
330
331     writeFile(SPIFFS, "/datalogger.txt", "MODO OFFLINE");
332     Serial.println("MODO OFFLINE");
333
334     while (WiFi.status() != WL_CONNECTED)
335     {
336
337         minutos = getMinute();
338
339         float power = pzem.power();
340         if (power > max_power && !isnan(power)){
```

```
341     max_power = power;
342 }
343
344 ArduinoOTA.handle();
345
346 if((strcmp(minutos,"00") == 0 || strcmp(minutos,"15")
    == 0 || strcmp(minutos,"30") == 0 || strcmp(
    minutos,"45") == 0) && message_count == 0 ){
347     char *current_measure = takeMeasures();
348     char spiffs_line [64] = "\r\n";
349     strcat(spiffs_line, current_measure);
350     appendFile(SPIFFS, "/datalogger.txt", spiffs_line);
351     free(current_measure);
352     message_count = 1;
353 }
354
355 if (strcmp(minutos,"00") != 0 && strcmp(minutos,"15")
    != 0 && strcmp(minutos,"30") != 0 && strcmp(
    minutos,"45") != 0 && message_count == 1) {
356     message_count = 0;
357     delay(500);
358     // desconecta e conecta novamente afim de evitar
        erros
359     WiFi.disconnect(true);
360     WiFi.mode(WIFI_OFF);
361
362     WiFi.mode(WIFI_STA);
363     WiFi.begin(SSID, PASSWORD); // Conecta na rede WI-
        FI
364 }
365     delay(5000);
366     free(minutos);
```

```
367     }
368
369 }
370
371 void setup() {
372
373     Serial.begin(115200);
374     // Inicia o pino do LED como OUTPUT
375     pinMode(PIN_LED, OUTPUT);
376
377     /* Inicializacao da EEPROM */
378     EEPROM.begin (4);
379     /* Inicializa o LED com o estado salvo na EEPROM */
380     digitalWrite(PIN_LED, EEPROM.read(0));
381
382     /* Inicializa a conexao wi-fi */
383     initWiFi();
384
385     /* Inicializa a conexao ao broker MQTT */
386     initMQTT();
387
388     /* OTA Configuration */
389     ArduinoOTA
390     .onStart([]() {
391         String type;
392         if (ArduinoOTA.getCommand() == U_FLASH)
393             type = "sketch";
394         else // U_SPIFFS
395             type = "filesystem";
396
397         // NOTE: if updating SPIFFS this would be the place
398             to unmount SPIFFS using SPIFFS.end()
```

```
398     Serial.println("Start updating " + type);
399 })
400 .onEnd([]() {
401     Serial.println("\nEnd");
402 })
403 .onProgress([](unsigned int progress, unsigned int
404     total) {
405     Serial.printf("Progress: %u%%\r", (progress / (total
406         / 100)));
407 })
408 .onError([](ota_error_t error) {
409     Serial.printf("Error [%u]: ", error);
410     if (error == OTA_AUTH_ERROR) Serial.println("Auth
411         Failed");
412     else if (error == OTA_BEGIN_ERROR) Serial.println("
413         Begin Failed");
414     else if (error == OTA_CONNECT_ERROR) Serial.println("
415         Connect Failed");
416     else if (error == OTA_RECEIVE_ERROR) Serial.println("
417         Receive Failed");
418     else if (error == OTA_END_ERROR) Serial.println("End
419         Failed");
420 });
421
422 ArduinoOTA.begin();
423
424 // Inicia SPIFFS
425 if(!SPIFFS.begin(FORMAT_SPIFFS_IF_FAILED)){
426     Serial.println("SPIFFS Mount Failed");
427     return; // return pra onde ?
428     //ESP.restart();
429 }
```

```
423 Serial.println("Chegou aqui");
424 Serial.println(SPIFFS.totalBytes());
425
426 // Verificar se existe algum arquivo SPIFFS e enviar
    conteudo ao BROKER
427 sendSPIFFSMeasures();
428
429 // Configura o TIMER interno de acordo com o horario
    adquirido via NTP
430 configTime(gmtOffset_sec, daylightOffset_sec, ntpServer
    );
431 // seta o contador de mensagem para envio
432 message_count = 0;
433 /* Inicializa com o valor de energia salvo na memoria
    do PZEM*/
434 // Inicializa com a potencia atual
435 previous_energy = pzem.energy();
436 max_power = pzem.power();
437 delay(1000);
438 }
439
440 void loop() {
441
442     // Informa que o dispositivo esta ONLINE
443     MQTT.publish (WILLTOPIC, ID_ON);
444
445     // recebe o minuto atual
446     minutos = getMinute();
447
448     // verifica o momento exato para enviar as medicoes ou
        se ja foi enviado a medicao do periodo
```

```
449 while(strcmp(minutos,"00") != 0 && strcmp(minutos,"15")
      != 0 && strcmp(minutos,"30") != 0 && strcmp(minutos
      ,"45") != 0 || message_count == 1 ){
450     VerificaConexoesWiFIEMQTT();
451     MQTT.loop();
452     ArduinoOTA.handle();
453     float power = pzem.power();
454     if (power > max_power && !isnan(power)){
455         max_power = power;
456     }
457     delay(5000);
458     minutos = getMinute();
459     // reseta o contador apos o periodo de envio.
460     if(strcmp(minutos,"00") != 0 && strcmp(minutos,"15")
        != 0 && strcmp(minutos,"30") != 0 && strcmp(
        minutos,"45") != 0){
461         message_count = 0;
462     }
463 }
464
465 // forca a medicao ser enviada alguns segundos apos o
      horario
466 delay(3000);
467
468 // recebe um char com as medicoes realizadas
469 char *current_measure = takeMeasures();
470
471 // Envia as medicoes ao BROKER MQTT
472 MQTT.publish (TOPICO_PUBLISH_M, current_measure);
473 // Libera a memoria alocada pela variaveis...
474 free(current_measure);
475 free(minutos);
```



```
476 // seta o contador impedindo que mais de uma mensagem
      seja enviada
477 message_count = 1;
478 }
```