



DESENVOLVIMENTO DE UM SISTEMA DE MONITORAMENTO E ANÁLISE DE DADOS PARA EFICIÊNCIA ENERGÉTICA EM INDÚSTRIAS 4.0

Guilherme Balduino Lopes*¹, Renato Ferreira Fernandes Junior¹

¹FEELT - Universidade Federal de Uberlândia

Resumo - A constante busca pela otimização de processos tem sido o principal agente do desenvolvimento de novas tecnologias. Em meio a isso, tem se falado muito da otimização de utilidades voltadas ao desempenho das linhas de produção nas indústrias. Desta forma, este projeto propõe o desenvolvimento de um sistema que será responsável por monitorar toda a rede elétrica de uma planta industrial, enviando seus dados para a nuvem, onde poderão ser monitorados e analisados pelo setor de gestão da indústria através de dashboards interativos, de uma forma simples e eficaz, em tempo real, e de qualquer lugar que tenha acesso à internet.

Palavras-Chave – Eficiência Energética, Computação na Nuvem, Internet das Coisas Industrial, Modbus e MQTT.

DEVELOPMENT OF A MONITORING AND DATA ANALYSIS SYSTEM FOR ENERGY EFFICIENCY IN 4.0 INDUSTRIES

Abstract - The constant search for process optimization has been the main agent in the development of new technologies. Amidst this, much has been said about the optimization of utilities aimed at the performance of production lines in industries. Therefore, this project proposes the development of a system that will be responsible for monitoring the entire electrical network of an industrial plant, sending their data to the cloud, where it can be monitored and analyzed by the industry's management sector through interactive dashboards, in a simple and effective way, in real time, and from any place that has access to the internet.

Keywords – Energy Efficiency, Cloud Computing, Industrial Internet of Things, Modbus and MQTT.

I. INTRODUÇÃO

No setor industrial, responsável por cerca de 34% do consumo de energia elétrica do Brasil em 2020, o impacto da adoção de medidas para redução do consumo é enorme. Isso porque as ações de Eficiência Energética devem reduzir a quantidade de recursos naturais utilizados, enquanto mantêm ou aumentam a produtividade e a qualidade de vida [1].

A Indústria 4.0 é um movimento universal que está proporcionando uma profunda transformação das empresas por meio de tecnologias como IoT (*Internet of Things*, ou Internet das Coisas, em português), *Big Data* e Inteligência Artificial [2]. Ela se caracteriza por utilizar essas tecnologias digitais para otimizar os processos, melhorar a qualidade de produtos e serviços, reduzir o consumo de energia e contribuir para uma gestão mais assertiva. Por isso, a Eficiência Energética é um dos principais benefícios e um dos pilares da indústria 4.0 [14], [15].

Conceitualmente falando, Eficiência Energética é realizar o mesmo ou mais, consumindo menos recursos naturais e de maneira mais inteligente, sempre com foco na segurança, no conforto operacional e na qualidade do produto. E no conceito da Indústria 4.0, busca além de aprimorar os processos de produção, trazer soluções para problemas ambientais, qualidade do ambiente de trabalho e, diminuir o consumo de recursos, mantendo altos rendimentos [14].

A análise desse consumo de energia é feita, na maioria das vezes, através de medições das variáveis elétricas, que se tornam essenciais para a obtenção de indicadores chave de desempenho (KPIs) responsáveis por ajudar a medir a performance ou eficácia de alguma atividade realizada em uma empresa [1]. Nas indústrias, onde predomina o uso de motores elétricos, estes indicadores de desempenho ajudam a prever e identificar falhas, planejar manutenções preventivas, identificar queda de rendimento ou desgaste excessivo em equipamentos elétricos. Porém, a maneira tradicional para realizar estas medições requer o deslocamento de técnicos até cada ponto de consumo para realizar cada medição. Este método implica em um custo elevado, difícil acesso à informação e possibilidade de erro humano ao realizar cada medição.

Por outro lado, o ambiente industrial tem adotado cada vez mais soluções baseadas em redes de protocolos padrões, nos seus diferentes níveis de automação, integrando o nível de chão de fábrica com o nível corporativo, e também, em redes IoT [2], por permitirem uma interconexão direta destas redes padrões com o setor de gestão da indústria, através da internet.

Com seu objetivo voltado à Eficiência Energética em Indústrias 4.0, este trabalho propõe um sistema integrado de monitoramento de KPIs em plantas industriais, usando tecnologias de IoT, juntamente com um dos protocolos de rede industrial mais populares, apesar de antigo, o protocolo Modbus. O sistema será responsável por fazer a aquisição dos dados espalhados em uma rede Modbus e armazenar os

*guilhermealopes@ufu.br

mesmos na nuvem através do protocolo MQTT. Desta forma, será possível unificar todas as informações em um só local, não através de estimativas e cálculos manuais, mas sim de dados obtidos automaticamente, direto das fontes, e assim, auxiliar em tomadas de decisões que cooperem com a sustentabilidade ambiental e, conseqüentemente, também melhorem o resultado final da produtividade e lucratividade na produção industrial.

A seção II deste artigo, abordará alguns conceitos e ferramentas importantes de IoT e de Indústria 4.0. Na seção III, serão abordadas as metodologias de desenvolvimento do sistema proposto. Já na seção IV serão mostrados os resultados a partir dos testes preliminares, e por fim, a conclusão do trabalho na seção V.

II. INTEGRAÇÃO DE TECNOLOGIAS DE TI E TO NA INDÚSTRIA 4.0

A convergência entre plantas industriais e a área de Tecnologia da Informação (TI) tem sido uma das principais mudanças desta chamada Indústria 4.0. Isso resulta em uma convergência de múltiplas tecnologias díspares, algo que traz desafios de gerenciamento e segurança nunca vistos antes. Pode-se chamar essa nova realidade de TI Industrial, um ambiente onde três tipos de tecnologia convergem: Tecnologia da Informação, Tecnologia Operacional (TO), e Internet Industrial das Coisas, (*Industrial Internet of Things*, IIoT) [5].

A TO refere-se a redes industriais que são normalmente usadas nas estratégias de controle industrial. Essas redes industriais têm se consolidado a décadas e são compostas de diferentes tecnologias, dependendo do tipo de aplicação e processo industrial. Destacam-se as redes Profibus DP, Devicenet e Profinet nas redes de controle de manufatura e as redes FF, HSE e Profibus PA, no controle de processos contínuos. No contexto de redes de sistemas de energia, destacam-se as redes Modbus, DNP3 e IEC61850 [6].

Porém, apesar de resolverem uma grande gama de aplicações, estas redes de campo, como o Modbus, são bastante limitadas aos conceitos atuais de Indústria 4.0, uma vez que não possuem suporte para várias aplicações específicas, como por exemplo: aplicações onde os dispositivos fiquem trocando de rede ou que tenham endereço desconhecido; aplicações onde seja necessária uma camada de segurança dos dados ou controle de acesso; e principalmente, aplicações que envolvam o envio de dados para serviços em nuvem [7], [4].

Para que seja possível resolver problemas como listados acima, se torna necessária a implementação de novas tecnologias responsáveis por integrar estas redes de campo (TO) ao setor de TI. Para isso, existem os chamados *gateways*, dispositivos de borda dispostos a “intermediarem” essas duas pontas, comunicando os dados extraídos da TO, aos servidores nos *data centers* locais ou na nuvem. Estes, devem poder conectar-se diretamente ao equipamento de campo (sensores, atuadores, etc.) ou por meio de controladores lógicos programáveis (CLPs), dispositivos eletrônicos inteligentes (IED), sistema de controle distribuído (DCS), sistema de controle industrial / controle supervisão e aquisição de dados (ICS / SCADA), etc. que agregam dados de campo. Devendo assim, suportar uma ampla variedade de interfaces, incluindo

conexões com fio, sem fio e até serial (por exemplo, RS-232 ou RS-485) [8].

Dentre as principais tecnologias de *gateway* de IoT existentes na Indústria 4.0 que integram o chão de fábrica à *web*, podem ser citados MQTT, OPC DA/UA, CoAP, entre outras. Sendo o protocolo de comunicação MQTT um dos que mais se destacam pela sua facilidade de implementação, por ser amplamente utilizado, e principalmente, por ser uma tecnologia aberta [3], [8].

Neste contexto, a seguir são apresentados alguns conceitos importantes para o desenvolvimento desta pesquisa.

A. Intelligent Electronic Device

IED (*Intelligent Electronic Device*, ou Dispositivo Eletrônico Inteligente, em português), um termo que, em ambiente industrial, se refere a um equipamento com multifunções para a proteção, controle, monitoramento e medição, com inteligência interna e capacidade de se comunicar com sistemas de supervisão e aquisição de dados [17]. Por contarem com vários protocolos de comunicação integrados, estes IEDs podem receber dados de sensores e equipamentos elétricos e envia-los para um nível superior com bastante facilidade. Na Figura 1 é possível ver um exemplo de IED, a SEL-710.

Figura 1: SEL-710. Relé de Proteção do Motor



Fonte: Schweitzer Engineering Laboratories, Inc (2021).

B. Protocolo Modbus

Modbus é um protocolo aberto de comunicação desenvolvido pela Modicon para comunicação entre equipamentos industriais, como por exemplo, CLPs e IEDs [9]. É frequentemente usado para se estabelecer uma conexão serial (RS232/RS422/RS485), sendo este o Modbus RTU, ou uma conexão Ethernet (TCP/IP ou UDP/IP), chamado de Modbus TCP. Ambas as versões do protocolo, utilizam o modelo Cliente-Servidor (ou Mestre/Escravo), onde o cliente, ou seja, o dispositivo que solicita a informação, desempenha um papel de “Mestre” (*Modbus Master*), e o dispositivo servidor, que fornece a informação requerida, é chamado de “Escravo” (*Modbus Slave*). Porém, diferentemente do Modbus RTU, o Modbus TCP permite que um Escravo/Servidor se comunique com vários Mestres/Clientes [9], [4].

Em uma rede padrão Modbus TCP, podemos ter um mestre e até 247 escravos, sendo que cada um recebe um endereço fixo de 1 a 247. Vale ressaltar que, por este endereçamento, o Mestre Modbus pode tanto ler, quanto escrever informações nos escravos [4].

Quanto à segurança, o protocolo Modbus não possui nenhum mecanismo previsto: não possui senha, autorização,

certificados ou qualquer outro mecanismo de segurança existente [4].

C. Protocolo MQTT

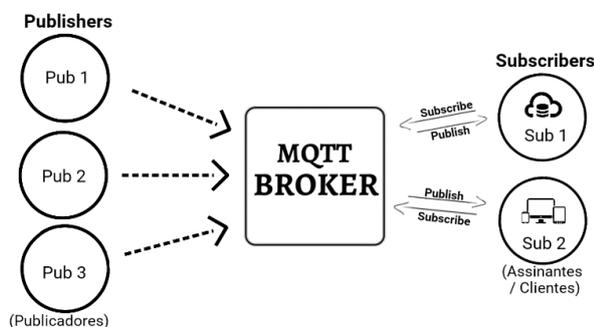
Uma das principais tecnologias de IoT utilizadas na comunicação entre a rede e a nuvem, é o protocolo MQTT (*Message Queuing Telemetry Transport*, ou Transporte Telemétrico de Enfileiramento de Mensagens, em português) um protocolo de rede do tipo *Publish/Subscribe* (Publica/Assina) que, por ser de fácil implementação e possuir uso moderado de banda, é normalmente destinado a sensores e pequenos dispositivos móveis [3], [2]. Essa tecnologia foi desenvolvida pela IBM no final dos anos 90, e sua finalidade original era conectar sensores de satélites ou pipelines de petróleo. Apesar de ter sido criado há um tempo, sua aplicabilidade ainda é excepcionalmente útil na atualidade, inclusive em diversos ramos empresariais [4]. Seu principal uso é fazer as máquinas trocarem informações, modalidade de comunicação conhecida como M2M (*Machine-to-Machine*, ou Máquina para Máquina, em português).

O protocolo MQTT tem um maior nível de segurança se comparado a outros protocolos de comunicação entre máquinas. Este suporta vários tipos de autenticações e mecanismos de segurança de dados, sendo possível também, fazer a criptografia ponta a ponta com segurança SSL para proteger os conteúdos das mensagens no protocolo [4].

O padrão *Publish/Subscribe* de troca de mensagens usado no MQTT, funciona da seguinte forma: elementos que desejam publicar informações, o fazem através do *Broker*, o intermediário que enfileira estas publicações recebidas e as envia aos elementos da rede que se inscreveram/assinaram para receber esta determinada informação [3]. A figura 1 mostra um exemplo da arquitetura do MQTT. Nela, os Publicadores (Pub1, Pub2 e Pub3) poderiam ser sensores em uma rede de automação, IHMs, CLPs, IEDs, ou um sistema SCADA, enquanto os Assinantes (Sub1 e Sub2) poderiam ser bancos de dados locais ou na nuvem, ou ferramentas de análises de dados, por exemplo.

Na arquitetura do protocolo MQTT a identificação das mensagens se dá por meio de tópicos (*topics*). Este tópico, lembra o conceito de *Uniform Resource Locator* (URL), onde os níveis são separados por barras (“/”) [3], [2].

Figura 1: Estrutura simplificada protocolo MQTT.



Fonte: Autores (2021)

Apesar do *Broker* representar um elo de fragilidade na rede ao centralizar as comunicações [3], ele permite um desacoplamento entre as partes comunicantes, algo não possível em modelos de comunicação do tipo cliente/servidor.

D. Armazenamento de dados em Nuvem

Um conceito existente na Indústria 4.0 é o armazenamento de dados em nuvem. O armazenamento em nuvem possui várias características que a torna superior ao uso do hardware dedicado, como serviço por self-service, rápida elasticidade, amplo acesso à rede, alocação de recurso de acordo com a demanda, e pagamento pelo uso, entre outros. O uso da nuvem descentraliza a informação para um armazenamento onde é possível acessar os dados de qualquer lugar, possibilitando acesso colaborativo e simultâneo a infraestrutura, softwares e informações [2].

Com relação ao funcionamento, a nuvem pode ser: pública, privada ou híbrida. A nuvem pública é administrada por grandes empresas, onde o usuário instala o software e as administradoras cuidam da manutenção e segurança das informações. Exemplos deste tipo de armazenamento são AWS (Amazon Web Service), Google Cloud Platform, Azure da Microsoft, Oracle Cloud, entre outras. Na nuvem privada, o cliente é responsável pela administração e segurança dos dados. O acesso é feito por rede virtual privada (VPN) e é utilizada por empresas que não querem compartilhar espaço com outros usuários. Na nuvem híbrida envolve o uso tanto da rede pública quanto privada [20].

Com relação ao banco de dados na nuvem, eles podem ser banco de dados SQL (*Structured Query Language*) que se tratam de bancos de dados relacionais, e os bancos NoSQL (*Not Only SQL*), termo utilizado para banco de dados não relacionais de alto desempenho, onde geralmente não é utilizado o SQL como linguagem de consulta. Esses bancos de dados NoSQL foram criados para ter uma performance melhor e uma escalabilidade mais horizontal para suprir necessidades onde os bancos relacionais não são eficazes. O NoSQL também proporciona uma performance melhor para o gerenciamento de dados das organizações, pois não há necessidade de agrupar os dados em um esquema de tabelas para usar as informações [11].

E. Business Intelligence

BI (*Business Intelligence*, ou Inteligência de Negócios, em português), é um processo orientado pela tecnologia, que combina análise empresarial, mineração de dados, visualização de dados, ferramentas/infraestrutura de dados, e práticas recomendadas para ajudar as organizações a tomar decisões impulsionadas por dados. Sendo assim uma forma inteligente de otimizar falhas e manter o negócio em destaque. Basicamente, o BI é um conjunto de teorias, metodologias, processos, tecnologias e estruturas, que transformam grandes quantidades de dados que, sozinhos, não significam muito, em informações essenciais para uma boa gestão [12].

Dentre as ferramentas *Business Intelligence*, se destaca o Power BI da Microsoft, uma plataforma unificada e escalonável para BI corporativo e por self-service [16]. Com ela é possível consolidar e tornar coerentes e visuais informações de várias fontes, desde um simples arquivo do Microsoft Excel ou um banco de dados local, a arquivos robustos localizados na nuvem.

O Power BI consiste em vários elementos que trabalham juntos, começando com estes três elementos básicos [16]:

- Um aplicativo de desktop do Windows, chamado Power BI Desktop.

- Um serviço SaaS (*software* como serviço) online, chamado de Serviço do Power BI.
- Aplicativos móveis do Power BI para dispositivos Windows, iOS e Android.

F. Interface para usuário final

No contexto da interface, o ecossistema do Python é repleto de ótimos *frameworks*, dentre eles se destacam o Django e o Flask, dois frameworks para desenvolvimento web bastante utilizados [13].

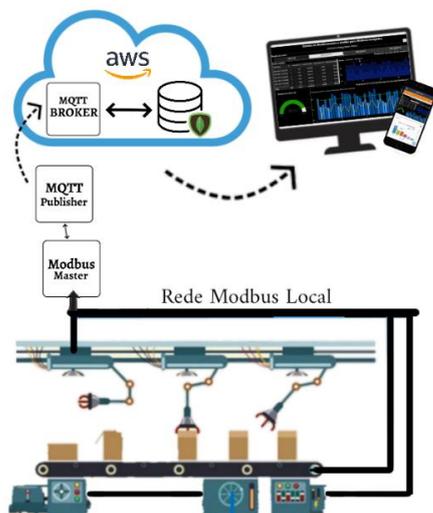
Django é um *framework* para desenvolvimento rápido para Web, escrito em Python, que utiliza o padrão MTV (*Model-Template-View*), que separa as responsabilidades da aplicação em camadas, aumentando a legibilidade do código e organização da aplicação. A principal premissa do Django é seu princípio DRY (*Don't Repeat Yourself*, ou Não Se Repita, em português), fazendo com que o desenvolvedor aproveite o máximo de código já criado em outros módulos da aplicação, evitando ao máximo a repetição e reescrita de código [13].

O Flask é, assim como o Django, um *framework* web de código aberto. Porém, o Flask possui apenas duas bibliotecas básicas, permitindo que o desenvolvedor instale apenas o que o projeto irá necessitar, o que torna a aplicação muito mais leve e rápida de ser executada. Não possui camada de acesso ao banco de dados, validação de formulários ou qualquer outro componente. Para isso, é necessário instalar bibliotecas de terceiros e, assim, prover determina da funcionalidade.

III. METODOLOGIA DE DESENVOLVIMENTO

Este projeto aborda, de forma prática, uma maneira de se conectar os dispositivos de um ambiente industrial, com foco principalmente nas variáveis elétricas, e com o objetivo de unificar todas as informações na nuvem para que esta informação possa ser analisada em tempo real, ou posteriormente pelo setor estratégico da empresa, estando este dentro da indústria, ou não, de forma fácil, precisa e segura. A Figura 2 mostra um esquemático da estrutura proposta por este trabalho.

Figura 2: Estrutura simplificada do projeto proposto.



Fonte: Autores (2021)

O projeto pode contar também, com equipamentos já dispostos pelo Laboratório de Automação, Sistemas Eletrônicos e Controle (LASEC) da Universidade Federal de Uberlândia (UFU), como Controladores e IEDs que se comunicam através de diversos protocolos, como Modbus e DNP3 por exemplo.

Visando atingir os objetivos do projeto, o mesmo foi dividido em cinco módulos.

A. Módulo Mestre Modbus

Por ser o de implementação mais fácil e estar presente na maioria dos ambientes industriais, o projeto em questão utilizará o modelo Modbus TCP/IP.

Este primeiro módulo consiste do desenvolvimento de um cliente Modbus responsável por monitorar todos os equipamentos referentes à rede elétrica de uma planta, como controladores e IEDs, e concentrar os dados de forma que a informação fique disponível para ser repassada para as camadas superiores.

B. Módulo Publicador MQTT

O Módulo Publicador MQTT é responsável por receber os dados lidos do Módulo Mestre Modbus e transmiti-los para a nuvem. A proposta deste módulo é o desenvolvimento de um *gateway*, o qual vai atuar como um *middleware* (termo referente a uma aplicação ou *software* capaz de intermediar, de forma transparente, dois ou mais serviços distintos [2]), para uma integração ou “conversão”, entre a rede de campo Modbus, e o protocolo de Internet das Coisas MQTT.

Portanto, este sistema fará tanto o papel de um Mestre Modbus, quanto de um *Publisher* MQTT, publicando dados em tempo real do Escravo Modbus, em um *Broker* MQTT localizado na nuvem da AWS.

A escolha pela plataforma AWS foi por sua facilidade, por ser gratuita para pequenos projetos e possível integração com MQTT nativo. Porém, existem outras plataformas de nuvem que poderiam ser utilizadas neste caso.

C. Módulo Armazenamento de Dados

Após os dados serem disponibilizados para o *Broker*, estes serão salvos em um banco de dados não relacional (NoSQL) MongoDB, também com comunicação na nuvem através da plataforma AWS. O MongoDB é um banco de dados de código aberto com alta performance que é aceito em diferentes sistemas operacionais, e tem como característica, ser orientado a documentos [11].

D. Módulo Monitoramento e Análise de Dados

Neste Módulo será desenvolvida a parte da aplicação voltada ao monitoramento e à análise dos dados. Após terem sido armazenados na nuvem, estes dados serão acessados remotamente pela ferramenta de análise Power BI. Através desta, serão criados dashboards interativos, possibilitando assim uma melhor visualização das informações, através de relatórios visuais contando com: curvas com faixas de operação aceitáveis; análises percentuais em relação à essas faixas de operação; máximos e mínimos dos parâmetros medidos; entre outros recursos disponíveis na ferramenta.

E. Módulo Interface Web

Aqui será criada uma aplicação web para o usuário final. Por se tratar de uma aplicação ainda simples, mas com potencial para se tornar uma aplicação robusta, foi escolhido o *framework* Django. Esta será uma interface para o sistema geral, onde será possível realizar cadastros de usuários, configurações de leitura, inclusões e exclusões de Slaves (dispositivos a serem monitorados), visualizar o banco de dados e também, o usuário poderá ser redirecionado para a página de análises e dashboards criados através do Power BI.

IV. RESULTADOS

Nesta seção serão mostrados os resultados obtidos do projeto até o presente momento. Estes experimentos e as análises são preliminares, visto que o trabalho está ainda em fase inicial de desenvolvimento.

Um primeiro levantamento realizado buscou testar a comunicação da IED SEL710 da LASEC com o computador pessoal, através do protocolo de rede Modbus TCP. Escrito em Python e utilizando uma biblioteca chamada *PyModbusTCP*, foi possível desenvolver o primeiro módulo deste projeto, o qual consistiu no desenvolvimento de um cliente Mestre Modbus, contando com um menu inicial para o usuário, dando-lhe a opção de realizar serviços de leitura, escrita e configuração, como mostrado na Figura 3.

Figura 3: Aplicação Cliente Modbus rodando.

```
--> Cliente Modbus TCP conectado..
--> Cliente MQTT conectado ao Broker..

-----
                    Cliente Modbus TCP
-----

Qual serviço?
1- Leitura
2- Escrita
3- Configuração
4- Sair
Nº Serviço: 1

Qual tipo de dado deseja ler?
1- Coil Status
2- Input Status
3- Holding Register
4- Input Register
Type: 1

Address: 100
Length: 5
Quantidade de leituras: 1

Começando leitura..

Leitura 1:[True, False, True, False, True]

Leituras Modbus foram publicadas no broker através do tópico /test/status/
E inseridas no Banco de Dados local com sucesso!!
```

Com a aplicação do Cliente Modbus desenvolvida, foi possível então, iniciar os testes de aquisição de dados. Para se ter uma melhor visualização dessas leituras, elas foram armazenadas e organizadas em uma base de dados local, através da biblioteca do SQLite3 no Python, por permitir a disponibilização de um pequeno banco de dados na própria aplicação, sem a necessidade de acesso a um SGBD (Sistema de Gerenciamento de Banco de Dados) separado [10]. Na Tabela I é possível ver alguns dos dados salvos através deste sistema.

Tabela I: Amostras de leituras através do Cliente Modbus

ID	Addr	Type	Display	Value	TimeStamp
1	00100	F01-CoilStatus	Booleano	1	2021-08-15 03:12:14
2	10100	F02-InputStatus	Booleano	0	2021-08-15 03:12:22
3	40100	F03- HoldingRegister	Floating Point	127.51	2021-08-15 03:12:32
4	30100	F04- InputRegister	Float (Swapped)	380.25	2021-08-15 03:12:45

Desta forma, foi possível ver os diferentes tipos (chamados de funções pelo protocolo Modbus) de leituras obtidos através da comunicação, seus endereços, tipos de *display* e seus respectivos *Time Stamps* (ou Amostras de Tempo, em português).

Após isso, teve início o desenvolvimento do segundo módulo deste projeto, o qual trabalhou a integração do protocolo Modbus com o protocolo MQTT. Foi implementada então, uma função capaz de publicar, em tempo real, através da biblioteca Eclipse paho-mqtt, os valores obtidos pelo Cliente Modbus, em um *Broker* MQTT localizado em um servidor na nuvem da AWS. Para isso, foi criada uma instância no serviço EC2 (Amazon Elastic Compute Cloud), uma parte central da plataforma de *Cloud Computing* da Amazon que permite que usuários utilizem o Console de Gerenciamento AWS para configurar máquinas virtuais para rodar aplicações pessoais [18].

Por se tratar de um sistema operacional bastante otimizado e gratuito na AWS, nesta instância, localizada em Ohio (EUA), foi criada uma máquina virtual com sistema operacional Ubuntu Server. Nesta, foi instalado e inicializado um dos *Brokers* Open Source mais conhecidos e utilizados, o Eclipse Mosquitto [19].

Abaixo, na Figura 4, é possível ver uma aplicação local no Prompt de Comando do Windows, rodando como um *Subscriber* (assinante) do tópico “*test/status*” do Cliente Modbus, através do *Broker* hospedado no servidor da AWS. Neste tópico, foram publicadas uma leitura de cada tipo de função, contando com 5 registros cada, a critério de exemplificação.

Figura 4: Assinante do tópico “*test/status*” do *Broker* na AWS.

```
Select Administrator Command Prompt - mosquitto_sub - h 3.134.40.193 - t "test/status"
C:\Program Files\Mosquitto>mosquitto_sub -h 3.134.40.193 -t "test/status"
Leitura 1 F01-CoilStatus: [True, False, True, False, True]
Leitura 1 F02-InputStatus: [False, True, True, False, True]
Leitura 1 F03-HoldingRegister: [127.51, 0.0, 23.786, 45.0, 0.0]
Leitura 1 F04-InputRegister: [380.25, 360.348, 10.12, 0.0, 24.7]
```

Com estes resultados, foi possível verificar e analisar as aquisições de dados da IED SEL710 através do protocolo Modbus TCP e também, a publicação dos mesmos, através do protocolo MQTT, em um servidor na nuvem, este podendo ser acessado de qualquer lugar, através da internet.

Após estas etapas terem sido concluídas e mostrarem resultados satisfatórios, iniciou-se então o desenvolvimento do terceiro módulo deste projeto. Este, juntamente com os demais módulos, estão ainda em fase de desenvolvimento.

V. CONCLUSÕES

Este trabalho teve como meta o estudo da integração de diferentes ferramentas voltadas à essa nova realidade trazida pela quarta revolução industrial, o mesmo tem uma importância significativa, vista a relevância da pesquisa sob a integração destas diferentes tecnologias na Indústria 4.0, não só com o intuito de melhorar a margem de lucro das empresas, mas também, cooperar com a sustentabilidade ambiental.

Até o presente momento, foi concluído o módulo 1 do projeto, onde é possível estabelecer uma comunicação Modbus com os equipamentos de energia, e o módulo 2, o qual é responsável pela publicação dos mesmos em um servidor na nuvem.

Com relação à rede industrial utilizada, este projeto implementou especificamente um cliente Modbus TCP. A utilização de outros protocolos de rede como cliente DNP3 poderia ser implementada no futuro. Desta forma, o projeto seria capaz de englobar diferentes equipamentos de rede de energia existentes no mercado.

Os próximos passos para a conclusão desta proposta envolvem o estudo da integração de todos os módulos e a otimização das técnicas utilizadas para o desenvolvimento das aplicações, juntamente com o estudo das análises dos dados para uma melhor eficiência energética na linha de produção.

Como projetos futuros, após a conclusão deste trabalho, ainda será estudada a implementação de técnicas de Aprendizado de Máquinas (*Machine Learning*) com o intuito de melhorar o sistema de análises, oferecendo previsibilidade e ajudando em tomadas de decisão, com o mínimo de intervenção humana. Também serão estudados aspectos de segurança da rede em seus diferentes níveis.

REFERÊNCIAS

- [1] Way 2 Blog. “Eficiência Energética no setor industrial”. Disponível em: <https://www.way2.com.br/blog/eficiencia-energetica-no-setor-industrial/>. Acesso:02/07/2021.
- [2] Cunha, M. J. et al. “Proposal for an iot architecture in industrial processes”. *IEEE. Industry Applications (INDUSCON), 2016 12th IEEE International Conference on*. [S.l.], 2016.
- [3] Barros, M. “MQTT - Protocolos para IoT” (2015). EMBARCADOS. Disponível em: <https://www.embarcados.com.br/mqtt-protocolos-para-iot/>. Acesso:01/06/2021.
- [4] Silva, C. de. S. F. “Modelos Cliente-Servidor do Modbus TCP e Publish-Subscribe do MQTT” (2019). Disponível em: <https://www.novus.com.br/>. Acesso: 10 /07/2021.
- [5] Data Center Dynamics. “Indústria 4.0 depende da convergência entre OT e TI para avançar no Brasil” (2020). Disponível em: <https://www.datacenterdynamics.com/>. Acesso: 01/06/2021.
- [6] Reynders, D. Mackay. S. Wright E. “Practical Industrial Data Communications” *Best Practice Techniques*, Newnes, 2005, ISBN 0 7506 6395 2.
- [7] Mazur, D.C. and Kay, J.A. “Benefits of IEC 61850 standard for power monitoring and management systems in forest products industries, in proceedings ”. *Conference Record of 2013 Annual IEEE Pulp and Paper Industry Technical Conference*, 2013.
- [8] Electronics, M. O papel crítico dos Gateways IoT (2020). EMBARCADOS. <https://www.embarcados.com.br/gateways-iot/>. Acesso: 01/06/21.
- [9] Silveira, C. B. “Saiba tudo sobre o protocolo Modbus” (2016). CITISYSTEMS. Disponível em: <https://www.citisystems.com.br/modbus/>. Acesso: 01/07/2021.
- [10] GSTI. “O que é SQLite?” (2021). *Portal GSTI - Um mundo de informações para TI*. Disponível em: <https://www.portalgsti.com.br/sqlite/sobre/>. Acesso: 05/07/2021.
- [11]Saphir. “Conheça os principais bancos de dados NoSQL (não-relacionais)” (2020). *Saphir Blog – Inteligência em Cloud Computing*. Disponível em: <https://blog.saphir.com.br/conheca-os-principais-bancos-de-dados-nosql-nao-relacionais/>. Acesso: 05/07/21.
- [12] Siteware. “O que é BI Business Intelligence” (2018). *Gestão Estratégica*. Disponível em: <https://www.siteware.com.br/blog/gestao-estrategica/o-que-e-bi-business-intelligence/>. Acesso: 01/07/21.
- [13] Sundfeld, R. “Django ou Flask? Eis a questão.” (2021). *Digital Innovation One*. Disponível em: <https://digitalinnovation.one/artigos/django-ou-flask-eis-a-questao>. Acesso: 10/07/21.
- [14] Zagonel. “Qual é a relação entre a ‘Indústria 4.0’ e a ‘eficiência energética na indústria?’” (2019). Disponível em: www.zagonel.com.br/iluminacao/blog/show/qual-e-a-relacao-entre-a-industria-40-e-a-eficiencia-energetica-na-industria-14/. Acesso: 20/07/2021.
- [15] Venturelli, M. “Eficiência Energética 4.0” (2020). *Marcio Venturelli – Automação Industrial, Digitalização e Indústria 4.0*. Disponível em: www.marcioventurelli.com/2020/06/26/eficiencia-energetica-4-0/. Acesso: 15/07/2021.
- [16] Microsoft. “O que é Power BI?”, Disponível em: <https://docs.microsoft.com/pt-br/power-bi/fundamentals/power-bi-overview>. Acesso: 15/06/21.
- [17] Pereira, A. Cascaes; et al. “Automação de Subestações e Usinas”. *Estado da arte e tendências utilizando a Norma IEC 61850, Anais do VII SIMPASE - Sétimo Simpósio de Automação de Sistemas Elétricos – Salvador -BA, 2007*.
- [18] Vincenzi, F. “Amazon Web Services – Máquina EC2”. *Sírius Guru –* Disponível em: <https://sirius.guru/aws/index.html>. Acesso: 15/08/21.
- [19] Mosquito. “Eclipse Mosquito – Um Broker MQTT de Código Aberto”. Disponível em: <https://mosquito.org/>. Acesso: 15/08/21.
- [20] Andrade, G. “Armazenamento em nuvem: como funciona e principais serviços” (2020). Disponível em: <https://digilandia.io/transformacao-digital/armazenamento-em-nuvem/>. Acesso: 15/08/21.