

UNIVERSIDADE DE SÃO PAULO
ESCOLA DE ENGENHARIA DE SÃO CARLOS
DEPARTAMENTO DE ENGENHARIA ELÉTRICA E DE COMPUTAÇÃO

MARCELA MARZOCHI BUENO QUIRINO

**NOVO MODELO DE SERVIÇO VOLTADO PARA AUTOMAÇÃO
INDUSTRIAL COM BASE EM IOT**

São Carlos
2017

Escola de Engenharia de São Carlos
Departamento de Engenharia Elétrica

MARCELA MARZOCHI BUENO QUIRINO

NOVO MODELO DE SERVIÇO VOLTADO PARA AUTOMAÇÃO INDUSTRIAL COM BASE EM IOT

Trabalho de Conclusão de Curso
apresentado à Escola de Engenharia de
São Carlos, da Universidade de São Paulo

Curso de Engenharia Elétrica com ênfase
em Sistemas de Energia e Automação

ORIENTADOR: Prof. Dr. Dennis Brandão

São Carlos
2017

AUTORIZO A REPRODUÇÃO TOTAL OU PARCIAL DESTE TRABALHO,
POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS
DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

Q8n

Quirino, Marcela Marzochi Bueno
NOVO MODELO DE SERVIÇO VOLTADO PARA AUTOMAÇÃO
INDUSTRIAL COM BASE EM IOT / Marcela Marzochi Bueno
Quirino; orientador Dennis Brandão. São Carlos, 2017.

Monografia (Graduação em Engenharia Elétrica com
ênfase em Sistemas de Energia e Automação) -- Escola de
Engenharia de São Carlos da Universidade de São Paulo,
2017.

1. Automação Industrial. 2. IoT. 3. SaaS. 4. Design
Science Research. 5. Cloud Computing. I. Título.

FOLHA DE APROVAÇÃO

Nome: Marcela Marzochi Bueno Quirino

Título: "Estudo de um novo modelo de serviço voltado para automação industrial com base em IOT"

Trabalho de Conclusão de Curso defendido e aprovado
em 29 / 11 / 17,

com NOTA 8,0 (oito , zero), pela Comissão Julgadora:

Prof. Associado Dennis Brandão - Orientador - SEL/EESC/USP

Mestre Bruno Henrique Oliveira - ICMC/USP

Mestre Guilherme Serpa Sestito - Doutorando - SEL/EESC/USP

Coordenador da CoC-Engenharia Elétrica - EESC/USP:
Prof. Associado Rogério Andrade Flauzino

DEDICATÓRIA

*À minha mãe Maria Angelina,
minha irmã Mayara, meu padrasto Ricardo.
Aos meus tios Lúcia e Dal.*

AGRADECIMENTOS

Agradeço primeiramente ao avô Allan que foi quem, antes de mim, me disse engenheira. Estendo os agradecimentos à toda minha família, tios, tias e primos. Saibam que só tenho coragem de enfrentar o mundo pois sei que são minha base e posso contar com isso. Agradeço *in memoriam* ao avô Luiz que, de sapato em sapato, nunca deixou de lado uma boa bossa, e também *in memoriam* à avó Iva que foi e sempre será um exemplo de resiliência. Tenho certeza que, mesmo com toda simplicidade, teriam se orgulhado em ver formar, por uma universidade pública, a primeira neta engenheira. Que dos seus netos em diante não seja um privilégio o acesso à educação de qualidade mas sim um direito.

Agradeço aos amigos de Ribeirão Preto com quem compartilhei maravilhosos anos da minha vida. Vocês sempre serão referência de amizade, parceria e união. Desde as intermináveis reuniões do Centro Estudantil e assembleias da Filô, os dias de completa doação no Circo da Física, até as noites mais bonitas onde contemplávamos o voo das corujas. Agradeço às amigas da República Gina, da qual sou feliz fundadora, por terem me ensinado na plenitude e com abundância de exemplos o significado de sororidade. Vocês foram e são essenciais na minha formação como mulher. Sou inteiramente grata por ter vocês comigo em minha caminhada. A minha casa sempre estará onde está meu coração, ele muda, a minha casa não.

À GAPeria, ao CAASO, à Tia Celly e a todos amigos que fiz nesta jornada.

Agradecer minha mãe Maria Angelina Marzochi e irmã Mayara seria pouco perto de todo suporte e amor que me deram nestes longos anos de faculdade. Cada conquista minha nestes últimos anos é nada mais que uma conquista nossa. Aprendemos juntas e com a primavera a se deixar cortar e voltar cada vez mais inteiras. E que cada dia mais o sol entre em nossas vidas por portas e janelas que insistimos em abrir e explorar. Pois então não só lhes agradeço como lhes dedico este trabalho. Agradeço também à meu padrasto Ricardo Goldemberg por me ensinar que amor tem várias formas e famílias se reinventam.

Por último e não menos importante, agradeço ao meu orientador Dennis Brandão que confiou em meu potencial e me deu oportunidade de trabalhar com ele. Agradeço também ao colega Murilo que sempre esteve disposto a me ajudar e é fonte imensa de conhecimento.

[...] Os Métodos de Pesquisa moldam a Linguagem que utilizamos para descrever o mundo, e a Linguagem molda como nós pensamos sobre o mundo [...]

Bensat & Weber (1996) [1]

RESUMO

QUIRINO, M. M. B. **Estudo de um novo modelo de serviço voltado para automação industrial com base em IoT**. 2017. Monografia (Trabalho de Conclusão de Curso) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2017.

O propósito deste trabalho é descrever as principais características e funcionalidades de um novo serviço voltado à automação industrial, assim como analisar as ferramentas disponíveis no mercado que oferecem processamento e armazenamento de dados em nuvem e que possibilitem a execução deste. A metodologia de pesquisa utilizada se baseia na Ciência do Artificial, conhecida como *Design Science Research*, cuja proposta é descrever um problema e propor artefatos para solucioná-lo. Foram analisadas alternativas de processamento e armazenamento em nuvem disponíveis no mercado das desenvolvedoras IBM, Microsoft, Amazon, Google e Siemens. Como alternativa para *gateway*, analisou-se o SIMATIC IOT2000 da Siemens. Foram descritos os protocolos de comunicação MQTT e HTTPs como alternativas, e os modelos de arquitetura de software Sistema Operacional Monolítico e modelo de Microserviços. Por fim, concluiu-se que os modelos de plataforma em nuvem mais flexíveis como o da Google e Amazon solucionam o problema de forma satisfatória. O gateway estudado cabe perfeitamente à solução. Para simplificação do desenvolvimento, optou-se pelo modelo monolítico de arquitetura, estabelecendo a comunicação entre elementos da rede e gateway com o protocolo MQTT.

Palavras-chave: Automação Industrial; IoT; SaaS; Design Science Research.

ABSTRACT

QUIRINO, M. M. B. **Study of a new service model dedicated to IoT based industrial automation**. 2017. Monograph (Course Completion Work) - School of Engineering of São Carlos, University of São Paulo, São Carlos, 2017.

The purpose of this work is to describe the main features and functionalities of a new service for industrial automation, as well as to analyze the tools available in the market that offer cloud data processing and storage, and also enable the execution of this Project. The research methodology used is based on Artificial Science, also known as Design Science Research, which purpose is to describe a problem and propose artifacts to solve it. Cloud processing and storage alternatives available from IBM, Microsoft, Amazon, Google and Siemens developers were analyzed. As an alternative to the gateway, Siemens SIMATIC IOT2000 was analyzed. The communication protocols MQTT and HTTPs were described as alternatives for the solution and the software architecture models Monolithic Operating System and Microservices model. Finally, it was concluded that the more flexible cloud platform models like Google and Amazon solve the problem in a satisfactory way. The gateway studied fits the solution perfectly. For simplification of development, the monolithic architecture model was chosen, establishing the communication between elements of the network and gateway with the MQTT protocol.

Keywords: Industrial Automation; IoT; SaaS; Design Science Research.

LISTA DE ILUSTRAÇÕES

Figura 1 - Etapas do Design Science Research definidas por Takaeda et al.....	30
Figura 2 - Esboço do funcionamento da comunicação entre as partes deste projeto.	33
Figura 3 - Esboço dos principais componentes e comunicações nas nuvens de processamento e armazenamento.....	33
Figura 4 - MindSphere da Siemes.....	38
Figura 5 - Google Cloud Platform for IoT [25].....	39
Figura 6 - Amazon AWS IoT [25].....	41
Figura 7 - Microsoft Azure IoT Suite [25].....	42
Figura 8 - IBM Bluemix [25].....	43
Figura 9 - Gráfico de Quadrantes comparando a flexibilidade em relação à facilidade de implementação de acordo com cada desenvolvedora.	45
Figura 10 - Gráfico de Quadrantes comparando ferramentas disponíveis para teste assim como tempo de teste por cada desenvolvedora	47
Figura 11 - Gateway SIOMATIC IOT2000 daSiemens. À esquerda, exemplo de componente montado no trilho de um painel industrial. À direita, componente sem a tampa, exibindo seu hardware, entradas, saídas e conexão com shield.	49
Figura 12 - Shield que pode ser adicionado ao SIOMATIC IOT2000. No canto superior esquerdo, foto do Shield apenas. No canto superior direito, pode-se observar a montagem do Shield na placa do gateway. Já a imagem inferior mostra o gateway fechado, com o Shield.....	50
Figura 13 - Replicação do Sistema Operacional Monolítico em novas máquinas da nuvem para múltiplos usuários.....	53
Figura 14 - Replicação da Arquitetura de Microserviços em novas máquinas da nuvem para múltiplos usuários.....	54

LISTA DE ABREVIATURAS E SIGLAS

API	Application-Programming Interface, do inglês, Interface de Aplicação de Programação
CLP	Computador Lógico Programável
DSR	<i>Design Science Research</i> , do inglês; Pesquisa em Ciência do Artificial ou Pesquisa em Ciência do Projeto
HTTP	Hypertext Transfer Protocol
HTTPS	Secure Hypertext Transfer Protocol, (
IaaS	<i>Infrastructure as a Service</i> , do inglês; Infraestrutura como um Serviço
IoT	<i>Internet of Things</i> , do inglês; Internet das Coisas
M2M	<i>Machine to Machine</i> , do inglês, de Máquina à Máquina
MQTT	<i>Message Queuing Telemetry Transport</i>
PaaS	<i>Platform as a Service</i> , do inglês; Plataforma como um Serviço
SaaS	Software as a Service, do inglês; Programa como um Serviço
SBC	<i>Single Board Computer</i> , do inglês; Computador de Placa Única

SUMÁRIO

1	Introdução.....	23
1.1	Objetivo	24
1.2	Estrutura do Trabalho.....	24
2	Revisão Bibliográfica.....	25
2.1	Industrial 4.0.....	25
2.2	Microprocessadores e Single Board Computers	26
2.3	Cloud Computing e Novos Modelos de Serviços	27
3	Metodologia - Design Science Research	29
4	Conscientização do Problema	31
5	Sugestão	35
5.1	Processamento e Armazenamento em Nuvem	35
5.2	Gateway - Dispositivos de Comunicação	36
5.3	Protocolos de Comunicação	36
5.4	Arquitetura de Processamento.....	37
6	Desenvolvimento e Avaliação	38
6.1	Cloud Computing	38
6.1.1	Apresentação das Principais Desenvolvedoras.....	38
6.1.1.1	Siemens (MindSphere)	38
6.1.1.2	Google (Google Cloud Platform for IoT).....	39
6.1.1.3	Amazon (AWS IoT)	41
6.1.1.4	Microsoft (Azure IoT Suite).....	42
6.1.1.5	IBM (Bluemix).....	43
6.1.2	Análise Comparativa das Desenvolvedoras	44
6.1.2.1	Flexibilidade x Facilidade	45
6.1.2.2	Ferramentas Disponíveis x Tempo de Teste.....	46
6.2	Gateway - Dispositivos de Comunicação	48
6.3	Protocolos de Comunicação	51
6.3.1	MQTT	51
6.3.2	HTTP/HTTPS.....	51
6.4	Arquitetura do Processamento.....	52
7	Conclusão.....	55
8	Futuras Aplicações	57
9	Referências¹.....	58

Anexo A – Detalhes da precificação da Google Cloud – IoT Core	60
Anexo B – Detalhes das Limitações de Teste da Amazon AWS IoT	61
Anexo C – Detalhes da Conta Gratuita e Precificação da Microsoft Azure e Hube IoT	63
Anexo D – Detalhes do Teste da IBM Bluemix	64
Anexo E – Detalhes do Datasheet do SIMATIC IOT2040	65

1 INTRODUÇÃO

A Primeira e a Segunda Revoluções Industriais aconteceram de forma lenta e gradual ao longo de aproximadamente dois séculos. Seus resultados, porém, impactaram permanente e positivamente os meios de produção, assim como a economia e os modelos de mercado e negócio [2]

Com a Terceira Revolução Industrial, computadores, microprocessadores, dentre outros eletrônicos, passaram a fazer parte das linhas de produção. Estes microprocessadores, inseridos desde sensores de segurança ao controle de produção e estoque, permitiram a automação das linhas de fábrica, aumentando ainda mais sua produtividade [2]. Até então, os equipamentos de automação industrial utilizavam o sistema analógico conhecido como 4-20mA, o qual envia informações através da modulação da corrente.

Com a necessidade de meios de comunicação mais eficientes, robustos, rápidos e com mais funções, desenvolveu-se o conceito de uma rede de comunicação voltada para o chão de fábrica, sendo esta nomeada genericamente de *fieldbus*. Diversos protocolos de comunicação foram desenvolvidos tanto por multinacionais como por iniciativas em plantas locais. Com o crescimento destas redes, surgiu a necessidade de padronização tanto do meio físico quanto digital desta comunicação, assim como sua arquitetura, dando origem a protocolos utilizados internacionalmente, como Foundation Fieldbus, Modbus, Profibus, entre outros [3].

Com a expansão dos novos maquinários microprocessados, surgiu também a necessidade destes equipamentos estarem conectados em rede para que pudessem se comunicar ou enviar dados à uma central de controle de uma forma rápida e eficiente.

O desenvolvimento da tecnologia nas últimas décadas permitiu grandes avanços na área de automação industrial e da computação em nuvem. Unindo estas duas áreas, surgiu o conceito de Internet das Coisas, conhecido por sua sigla em inglês, IoT (*Internet of Things*). Conectando os componentes da fábrica à internet e levando o processamento e armazenamento destes dados para uma infraestrutura em nuvem, criaram-se novas possibilidades de controle da rede fabril, dando início à quarta revolução industrial, chamada de Indústria 4.0.

Atualmente várias desenvolvedoras têm focado seus trabalhos para oferecer uma infraestrutura em nuvem de qualidade e criar plataformas para integração com os componentes industriais, trazendo soluções antes complexas ou inexistentes para a área. Algumas soluções já estão disponíveis no mercado, mas há muito o que se pesquisar e desenvolver.

1.1 OBJETIVO

O objetivo deste trabalho é analisar as soluções de plataforma e infraestrutura já disponíveis no mercado para desenvolver um novo modelo de serviço baseado na Indústria 4.0. Estudando o que já existe e seus potenciais de evolução, torna-se possível iniciar demais projetos de pesquisa na área, trazendo para o meio acadêmico o que há de mais recente em tecnologia.

1.2 ESTRUTURA DO TRABALHO

Seguido deste capítulo introdutório, tem-se a Revisão Bibliográfica (Capítulo 2) explanando os principais tópicos abordados no projeto. Em sequência, tem-se a explicação da Metodologia (Capítulo 3) utilizada para o desenvolvimento deste. Dando continuidade ao modelo seguido pela metodologia, faz-se o reconhecimento do problema, para o qual as soluções possíveis serão estudadas, no Capítulo 4 – Conscientização do Problema.

No Capítulo 5 – Sugestão, discorre-se sobre os principais artefatos que solucionariam o problema proposto. Seguido do Capítulo 6 – Desenvolvimento e Avaliação, no qual as principais soluções existentes no mercado são explicadas detalhadamente e suas características são avaliadas de acordo com as necessidades da solução proposta.

Encerra-se, então, com o Capítulo 7, trazendo as conclusões acerca das análises realizadas e as estimativas para aplicação em futuros projetos.

2 REVISÃO BIBLIOGRÁFICA

2.1 INDUSTRIAL 4.0

O termo Indústria 4.0, se referindo à uma quarta revolução industrial, surgiu em 2011 na Alemanha, em um artigo publicado pelo governo, como uma estratégia de alta tecnologia para 2020, visando contornar a crise financeira mundial. Após algumas mais aparições, o termo se tornou globalmente discutido como um dos maiores tópicos quando se trata de tecnologia e inteligência para a produção industrial. O conceito principal envolvendo a Indústria 4.0 é unir a tecnologia da informação e da comunicação à indústria e ao fornecimento de serviços, a fim de se promover um meio de produção digital, guiado pela informação, flexível, personalizável e até mesmo mais econômico e ambientalmente consciente [4].

Com o surgimento das redes de comunicação industrial, cada vez mais os dispositivos se encontram interconectados, gerando uma quantidade crescente de dados. Com esta quantidade de dados, surge também a necessidade de se processar, compreender, armazenar e utilizar estas informações a favor do crescimento e melhoria da produção. Para se gerir este volume de dados, o melhor recurso a ser utilizado é o processamento e armazenamento em nuvem, tornando necessário que, além de interconectados, estes dispositivos também se conectem à internet. Esta é a premissa base da Indústria 4.0 [5].

Além do processamento e armazenamento em nuvem dos dados coletados da rede, há a criação de sistemas inteligentes, tornando possível controlar a produção e estoque em tempo real, a degradação dos componentes da fábrica, assim como o agendamento autônomo de manutenções, a qualidade e eficiência da linha de produção, entre outros serviços. Quanto mais componentes e sistemas estiverem conectados à internet, mais simples é a comunicação entre eles [5].

Juntamente ao conceito da Indústria 4.0, vêm a especulação de infinitas possibilidades a serem estudadas e implementadas. Uma grande promessa desta revolução é a possibilidade de coletar dados suficientes de funcionamento de linhas ou equipamentos a fim de se alimentar uma rede de inteligência artificial. Com esta inteligência, seria possível acompanhar a saúde dos componentes e a qualidade do que se é produzido, assim como definir comportamentos em casos de falhas e anomalias [5].

Com uso de sensores apropriados, ainda mais informações podem ser coletadas e armazenadas de forma histórica. A agregação de todos estes dados de forma organizada é atualmente chamado de *Big Data*, e a extração da informação relevante destes dados,

correlacionando variáveis e compreendendo séries temporais e afins, é o que se dá o nome de *Data Mining*, do inglês, Mineração de Dados. O processamento, armazenamento, visualização e utilização destes dados de forma adequada é a chave do sucesso para o crescimento da Indústria 4.0 e estas ferramentas devem ser escolhidas de forma cautelosa [6].

Com estes dados, também seria possível criar um sistema de auto gestão da fabricação de modo que os ajustes fossem feitos de forma autônoma e de acordo com parâmetros como, por exemplo, a quantidade de pedidos dos clientes. Interconectando a fabricação, ao estoque, à loja online, criam-se até mesmo novas possibilidades de mercado baseado no *e-commerce* [4].

O ponto de partida, portanto, da transformação para a Indústria 4.0 é conseguir conectar os sensores da fábrica à internet. Em um estudo de 2011 feito pela Cisco (uma das gigantes mundiais em redes e comunicação, que oferece soluções desde uso doméstico à empresarial) a estimativa é que em 2015 existiriam 25 bilhões de dispositivos conectados à internet e este número dobraria em 2020, com uma média de aproximadamente 6,5 dispositivos conectados por habitante no planeta [7].

2.2 MICROPROCESSADORES E SINGLE BOARD COMPUTERES

Paralelo ao desenvolvimento industrial, ocorreu o desenvolvimento dos microprocessadores que se tornaram cada vez menores, com mais funcionalidades e maior capacidade de processamento graças aos avanços na fabricação de semicondutores [8].

Com a diminuição destes componentes, foi possível a criação de Computadores de Placa Única, em inglês, *Single Board Computer* (SBC). O SBC é uma plataforma com um microprocessador capaz de executar um sistema operacional completo e que possui várias funções embutidas (entradas e saídas digitais, porta USB, serial, ethernet, PWM, etc) assim como a possibilidade de adição de novos módulos conforme necessidade (conexão sem fio, display de 7 segmentos e afins) [9].

No intuito de se desenvolver um microprocessador que fosse fácil de programar, adicionar módulos de memória, entradas, saídas e afins, e ainda se disponibilizasse total acesso ao projeto do *hardware* e do *software*, foi criada a placa Arduino. Inicialmente projetado para auxiliar designers em projetos que necessitassem de uma eletrônica específica, o Arduino possui linguagem própria, porém bastante simples e de alto nível possibilitando que até mesmo pessoas com pouco conhecimento em programação o utilizem [10].

Atualmente, o uso de Arduino para desenvolver projetos de IoT tem crescido consideravelmente devido ao fato de ser *open-source* e financeiramente viável. Por sua

flexibilidade de funções, é possível encontrar descritos na literatura exemplos de diferentes usos dentro do ambiente de automação, desde controles simples de sensores à *gateways* desenvolvidos na plataforma [11].

2.3 CLOUD COMPUTING E NOVOS MODELOS DE SERVIÇOS

O termo *Cloud Computing* pode ser utilizado tanto para definir as aplicações, serviços e ferramentas que existentes no mercado atual e que são oferecidas via internet, por exemplo, serviço de armazenamento de documentos, *streaming* de filmes e seriados e redes sociais, como para definir o oferecimento de estruturas de hardware e de software através da internet, por exemplo, a locação de clusters ou servidores para executarem uma ferramenta que foi desenvolvida. [12].

Em outros termos, pode-se locar a infraestrutura de uma terceira e acessá-la via internet, de forma que não mais seja necessário investir em servidores locais que são de grande custo e, por vezes, volume. Isto possibilita grandes economias uma vez que o custo inicial de se construir um servidor é alto e, locando de outra empresa, pode-se utilizar apenas o necessário sem que haja imprevisto caso a demanda por mais máquinas aumente. Da mesma forma, pode-se contratar um serviço que não fique instalado e seja executado localmente, na infraestrutura física, mas sim é executado em servidores à distância e acessados via internet.

Além da vantagem de se contratar e financiar apenas a infraestrutura necessária para a sua necessidade, gerando economia, a *Cloud Computing* permite grande flexibilidade tanto na aplicação como no uso. Ou seja, uma vez contratado o serviço, pode-se alterar de forma simples o que exatamente se deseja que seja processado, assim como é simples alterar a quantidade de máquinas sendo executadas para sanar a demanda. Isso dá segurança ao desenvolvedor para criar novas ferramentas e serviços sem que se haja certeza do volume de uso. Caso o projeto tenha sucesso, é simples escalar – sendo o contrário tão viável quanto [13].

Estas novas possibilidades não só mudaram a forma de se pensar em computação e programação, mas a transformaram em um bem de consumo. Esta evolução da computação resultou também em novos modelos de mercado e serviços prestados, criando-se os conceitos de IaaS, PaaS e SaaS, explicados a seguir.

Infrastructure as a Service (IaaS), do inglês Infraestrutura como um Serviço, é o nome que se dá quando uma empresa disponibiliza servidores seus para serem locados e utilizados e cobra então por este serviço. Já *Platform as a Service* (PaaS), do inglês, Plataforma como um Serviço, muitas vezes se mistura com IaaS uma vez que as empresas que oferecem a infraestrutura também oferecem ferramentas para gerenciá-la, auxiliar na

comunicação e troca de dados e afins, criando-se então o acesso à uma plataforma de gerenciamento da infraestrutura locada. Quando se oferece um serviço hospedado na nuvem e disponibilizado através do acesso à internet, se dá o nome de *Software as a Service (SaaS)*, do inglês, Programa como um Serviço [12].

Com estas novas tecnologias e necessidades de mercado, surgem também novas possibilidades de desenvolvimento de produtos e oferecimentos de serviços nunca antes imaginados. Este trabalho se utilizará do método de pesquisa *Design Science Research (DSR)* para responder perguntas iniciais sobre as opções de processamento e armazenamento em nuvem existentes atualmente no mercado. Estas ferramentas possibilitam a criação de novos produtos e serviços de automação industrial via Internet das Coisas (IoT, do inglês, *Internet of Things*), estruturadas totalmente em nuvem.

3 METODOLOGIA - DESIGN SCIENCE RESEARCH

É comum no meio científico que se realizem pesquisas a fim de compreender, metrificar e caracterizar fenômenos naturais e, para este fim, existem metodologias muito bem estabelecidas e conceituadas. Estes métodos científicos são nada mais que um conjunto de regras e procedimentos a serem seguidos de forma a se obter conhecimento científico [14].

Estes processos de pesquisa, no entanto, nem sempre trazem resultados esperados ou relevantes quando utilizados em engenharia, uma vez que enfocam mais na teoria do que na aplicação prática. É relevante que as instituições validem e solidifiquem novos métodos de pesquisa para esta grande área [15].

Neste ponto, é ideal que se faça uma distinção entre conhecimento científico e conhecimento em engenharia, de forma que este último se utilize do primeiro com o propósito de projetar e construir artefatos para a solução de problemas [16] e que estes tenham as propriedades desejadas e objetivos bem definidos [17].

Sem que se crie uma oposição, é válido diferenciar ciência natural de ciência artificial, sendo o artificial algo produzido e inventado pelo homem - como o conceito de economia, aspectos da sociedade, organizações, maquinários, entre outros. Como grande parte do mundo em que vivemos atualmente se constitui do artificial, faz-se necessário um método de estudo baseado neste meio, e que este método consolide uma forma cientificamente aceita de se desenvolver mais artefatos para o meio artificial [17].

Um método de pesquisa científica já consolidado na engenharia de computação para desenvolvimento de softwares e sistemas inteligentes, e que tem sido amplamente aplicado à outras engenharias, é o *Design Science Research*, traduzido para o português como “Pesquisa em Ciência do Artificial” ou “Pesquisa em Ciência do Projeto” [18].

A ideia de *design* envolve indagar sobre novos sistemas que ainda não existem – tanto sistemas completamente novos como novos estados de sistemas já existentes. O que se questiona essencialmente é o funcionamento ou não daquele sistema imaginado e não sua viabilidade ou veracidade, como seria no estudo de um fenômeno natural. Uma característica importante do design é o uso de sistemas ideais como alvo ao se definir a situação inicial [19].

Uma vez definido o sistema ideal, deve-se definir sua classe de problema e os artefatos que geram soluções satisfatórias [18]. O conhecimento obtido pela Ciência do Projeto é generalizável dentro da classe de problema, já os artefatos são singulares para o caso real. O que não elimina o fato de que o conhecimento obtido através das soluções satisfatórias dos artefatos sejam aplicadas a problemas de mesma classe de forma a habilitar o crescimento do conhecimento na área [17].

Embora haja coerência entre autores ao se descrever a metodologia de pesquisa em *Design Science*, os processos em si podem ser descritos de forma diferente, seguindo o mesmo padrão de raciocínio, porém. Para a condução deste projeto, optou-se por seguir a literatura tal qual Takeda *et al*/ descreveram, sendo os passos do processo:

- (1) Conscientização – onde se ganha consciência do problema a ser solucionado;
- (2) Sugestão – são sugeridos artefatos que solucionam o problema;
- (3) Desenvolvimento – passo no qual os artefatos levantados são testados para a solução da problemática;
- (4) Avaliação – os resultados obtidos são avaliados;
- (5) Conclusão – disserta-se sobre os resultados obtidos e se validam os artefatos sugeridos no passo 2.

Tendo definido os passos que regem este estudo, dá-se seguimento à definição e conscientização sobre o problema.

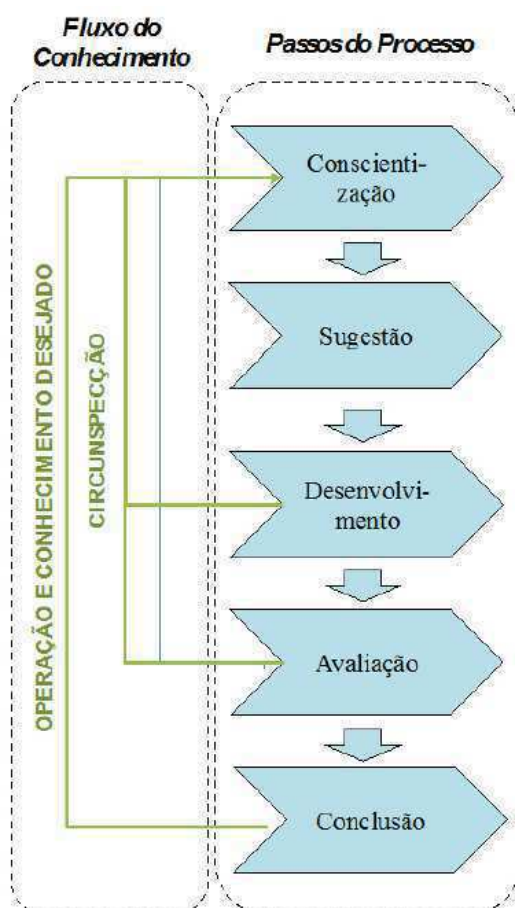


Figura 1 - Etapas do Design Science Research definidas por Takaeda et al.

4 CONSCIENTIZAÇÃO DO PROBLEMA

O propósito deste trabalho é descrever as principais características e funcionalidades de um novo serviço voltado à automação industrial, assim como analisar as ferramentas disponíveis no mercado que oferecem processamento e armazenamento de dados em nuvem e que possibilitem a execução deste. A partir destas ideias iniciais, o objetivo é que se torne possível o desenvolvimento de estudos mais aprofundados dos aspectos aqui levantados a fim de se iniciar a prototipagem deste novo serviço e então sua concretização.

Parte-se então de um sistema ideal que será caracterizado por uma linha de produção totalmente interconectada, na qual todos os sensores, atuadores, CLPs (Computador Lógico Programável) e demais elementos estejam discriminados e conectados em rede à internet – ou que esta conexão seja possível. Tem-se descrito um sistema fabril passível de se adaptar totalmente à Industrial 4.0.

O que se deseja desenvolver é um serviço (SaaS) prestado à este sistema idealizado que possa ser utilizado para diferentes fins. Os pontos principais a serem considerados e almejados são:

- **Acessibilidade:** o serviço deverá estar disponível online, acessível de qualquer terminal com internet, porém de forma segura, com controle de usuários e restrição de funcionalidades por usuário. Não é necessário que seja instalado em máquinas locais nem mesmo requer um processamento mínimo das máquinas que acessarão.
- **Infraestrutura:** toda a infraestrutura de processamento e armazenamento será alocada em nuvem de terceiros, não sendo necessária a criação e manutenção de servidores em local físico. É necessária apenas a manutenção da rede e sua conexão à internet.
- **Flexibilidade:** cria-se um serviço primário como base, cuja aplicação seja plausível em qualquer meio industrial, e então criam-se diversas funcionalidades que possam se adaptar às necessidades particulares de um determinado sistema. Com a facilidade de atualização que as plataformas online proporcionam, pode-se criar um serviço totalmente personalizável e maleável às mais diversas necessidades do meio industrial.
- **Supervisão:** possibilidade de criação de um sistema supervisor, também acessível de qualquer terminal com internet e que seja passível de entradas de comandos, personalização da exibição dos dados, alarmes,

linhas, etc, assim como exportação destas informações em formatos compatíveis com demais softwares (por exemplo “.csv”)

- Base de dados: uma rede de comunicação industrial transmite um volume considerável de dados, parte destes dados contém informações relevantes sobre a saúde da rede e de seus componentes; conhecendo-se os padrões de comunicação, é possível criar bancos de dados ricos em informações que possam ser utilizadas das mais diversas maneiras, como por exemplo, para alimentar uma função de inteligência artificial capaz de diagnosticar a rede apenas coletando dados durante sua operação.
- Inteligência Artificial: como citado anteriormente, é possível ter funcionalidades baseadas em inteligência artificial e aprendizado de máquinas, tornando o sistema ainda mais robusto e passível de utilizações em campo para consultoria e diagnóstico; extraindo e compreendendo as informações corretamente, também é possível configurar a linha de fábrica para seu melhor desempenho, trazendo benefícios inclusive econômicos para a empresa.
- Segurança: quando se fala em IoT, a primeira preocupação é com a segurança da informação que está sendo transferida, portanto é necessário que os protocolos de comunicação sejam extremamente seguros, assim como o acesso; é necessário que se preze pela qualidade do banco em que se está inserindo a informação uma vez que, em se tratando de armazenamento de dados para alimentação de uma rede de inteligência artificial, é necessário que não se perca ou se corrompam informações, permitindo seu uso posterior e garantindo a relevância do dado.
- Interoperabilidade: com a diversidade de protocolos de comunicação existentes assim como diferentes marcas e modelos de sensores, atuadores, CLPs e afins, é necessário que a plataforma dê suporte a pelo menos os itens mais comuns no mercado.
- Escalabilidade: para que se possa dar suporte de pequenas a grandes empresas, é necessário que a solução encontrada seja escalável e aplicável à diferentes plantas.

Em termos gerais, tem-se os componentes de uma fábrica conectados por meio de um dispositivo à internet. Os dados desta fábrica são levados por esta conexão à nuvem, onde ocorre o processamento e armazenamento destes. A interface de comunicação com esta nuvem por ser, por exemplo, um domínio na internet, de forma que o usuário tenha

acesso à estes dados e as ferramentas implementadas de qualquer computador com acesso à internet. O esboço desta ideia está representado na Figura 2.

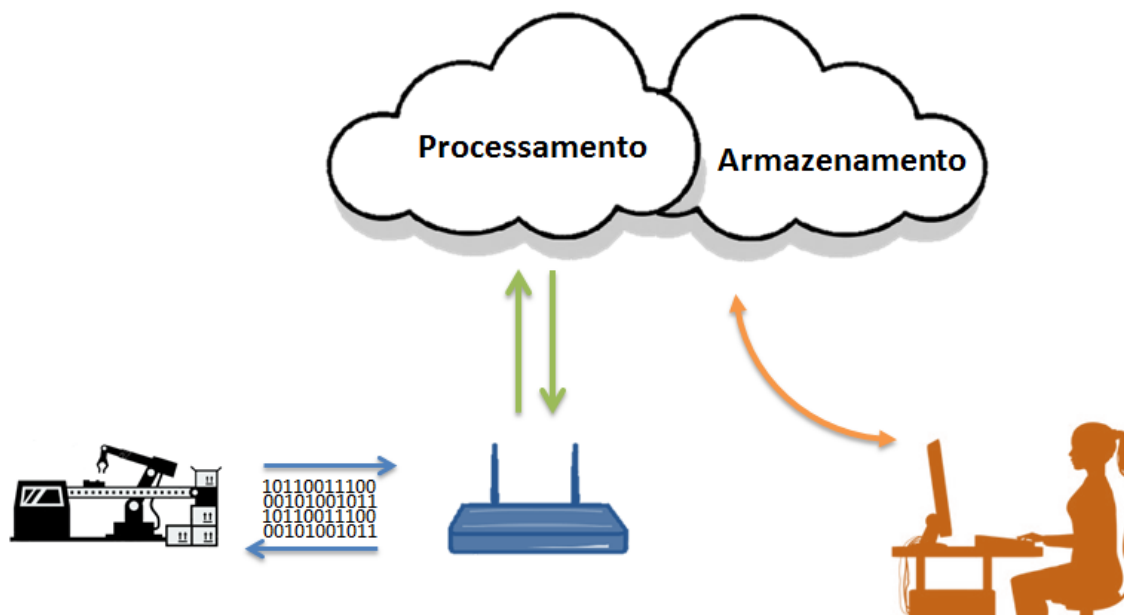


Figura 2 - Esboço do funcionamento da comunicação entre as partes deste projeto. À esquerda, tem-se a linha de fábrica conectada via *gateway* à nuvem, na qual ocorre o processamento e armazenamento dos dados, e pode ser acessada de qualquer terminal com internet.

Analisando mais a fundo os pontos levantados na listagem anterior, é possível fazer um esboço também das principais funções e comunicações dentro das nuvens de armazenamento e processamento, como exibido na Figura 3.

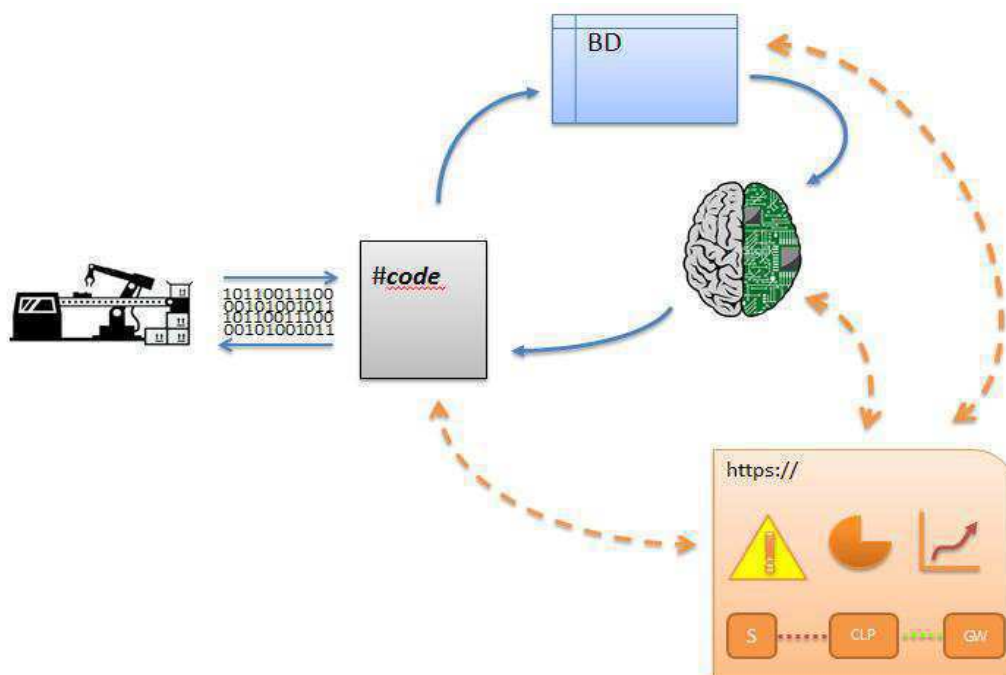


Figura 3 - Esboço dos principais componentes e comunicações da nuvem. Começando pela esquerda, tem-se a linha de fábrica transmitindo seus dados para o software em nuvem. Este se comunica com o banco de dados e ambos se comunicam com o sistema inteligente. Tanto o software, como o banco de dados, como o sistema inteligente, alimentam uma plataforma online.

No esboço, tem-se o programa processando os dados obtidos dos sensores, atuadores, CLPs e demais componentes da fábrica. Este programa pode exibir em tempo real os dados de saúde da rede para uma interface em um *website*. Também se comunica com a plataforma de armazenamento que, por sua vez, alimenta a um sistema de inteligência artificial. Este sistema pode tanto ser configurado pelo usuário através da interface *online* como pode retornar dados para o programa tomar as ações necessárias, criando-se assim um sistema supervisório. O banco de dados também é acessível ao usuário que pode utilizar ferramentas para visualizá-los e interpretá-los.

Uma vez definidos tanto o sistema ideal quanto o objetivo em que se quer chegar, é necessário definir os artefatos que solucionarão este problema. Sendo o passo seguinte a este, a definição de como eles serão analisados de maneira que os resultados tenham forma similar e sejam passíveis de comparação.

5 SUGESTÃO

Ao longo deste capítulo, serão descritos os artefatos que podem solucionar o problema levantado, assim como será definida a análise a ser feita acerca de cada classe de artefato.

5.1 PROCESSAMENTO E ARMAZENAMENTO EM NÚVEM

É primordial para o desenvolvimento deste projeto que se escolha uma plataforma ou serviço de processamento e uma de armazenamento de dados em nuvem. Alguns projetos optam por desenvolver sua própria nuvem e mantê-la acessível localmente, porém os requisitos para se desenvolver tal ferramenta estão aquém do âmbito deste trabalho. Com isso, objetiva-se analisar quais nuvens estão disponíveis no mercado atualmente que satisfaçam os objetivos propostos.

É possível diferenciar as nuvens em dois tipos distintos de serviços oferecidos: IaaS e PaaS. Quando se opta pela infraestrutura de um terceiro (IaaS), está se optando por locar os *clusters* de terceiros e neles armazenar ou a aplicação desenvolvida (*software*) ou o banco de dados obtido pela aplicação, ou ambos. Há de se atentar para questões gerenciais de dados e de clusters uma vez que o desenvolvimento destas é de certa complexidade.

Alguns IaaS oferecem recursos para gerenciar automaticamente o cluster locado, adicionando ou removendo mais máquinas conforme a demanda momentânea, assim como diversas outras ferramentas de busca rápida em banco de dados, leitura e escrita e inserção de novas versões da aplicação. Quando se contratam estas funcionalidades, pode-se caracterizar como um PaaS. Tudo o que não for oferecido pela terceira, deve ser desenvolvido no projeto e requer conhecimento específico na área.

Já as terceiras que oferecem uma plataforma como serviço (PaaS) têm, embutidas em seu leque de ferramentas, alternativas para lidar com os dados, ferramentas de busca e inserção no banco, ferramentas de visualização e análise dos dados do banco, gerenciamento automático dos clusters, aceleração de processamento, ferramentas para executar as atualizações das versões mais recentes do seu programa, ferramentas de diagnóstico de falhas e até mesmo ferramentas já desenvolvidas especificamente para a automação industrial, com módulos que podem ser inseridos à plataforma conforme interesse.

Com relação às alternativas presentes no mercado, este trabalho analisará as principais características oferecidas pela Siemens (*MindSphere*), pela Google (*Google Cloud Platform for IoT*), pela Amazon (AWS IoT), pela Microsoft (*Azure IoT Suite*) e pela IBM (Bluemix).

5.2 GATEWAY - DISPOSITIVOS DE COMUNICAÇÃO

A definição de *gateway*, como a tradução do próprio nome sugere, é um dispositivo dedicado a ser o “portal” entre um sistema físico local e um serviço localizado em servidores da internet. O exemplo mais clássico de um *gateway* de uso doméstico é o roteador de internet – ele que conecta a rede residencial à prestadora de internet.

O *gateway*, dentro da automação, é quem faz a comunicação dos dispositivos de uma rede industrial com a internet. Para isso, é necessário que ele seja um dispositivo inteligente, que consiga se comunicar em diferentes protocolos *fieldbus* e consiga absorver uma quantidade alta de dados assim como interpretá-los na mesma velocidade de forma a não prejudicar o funcionamento da indústria.

Idealizando um modelo simples de *gateway*, suas funções básicas seriam estas descritas: flexibilidade de comunicação, escalabilidade para muitos componentes, alto processamento de dados, comunicação eficiente com a internet.

Várias fabricantes de componentes para automação industrial produzem *gateways* para comunicação das plantas. A maioria delas possuem soluções que se comunicam via ethernet, porém a única no mercado que apresenta um modelo exclusivamente desenvolvido para IoT é a Siemens. Mais detalhes sobre este componente estarão no Capítulo 10.

5.3 PROTOCOLOS DE COMUNICAÇÃO

Neste projeto, tem-se vários dispositivos e plataforma se comunicando constantemente. Para que esta comunicação ocorra de forma segura e eficiente, é necessário que se escolham protocolos de comunicação adequados e suportados pelos meios que se conectarão.

Iniciando-se a análise pela linha de fábrica, existem diversos protocolos adotados e nem sempre tem-se a presença de apenas um em uma planta. Os protocolos de comunicação *fieldbus* mais utilizados são o Foundation *Fieldbus*, *Modbus* e *Profibus*. É necessário, portanto, que o *gateway* saiba interpretar estes dados e comunica-los à nuvem.

A comunicação com a nuvem, pelo alto volume de dados, deve acontecer de forma ágil. O tamanho das mensagens deve ser reduzido sem que se percam informações importantes. A simplificação desta comunicação não pode ser em contra partida à segurança, de maneira que este canal seja tanto estável como inacessível à terceiros.

Dentre as ferramentas de processamento e armazenamento em nuvem, os protocolos de comunicação devem estar estabelecidos pela própria prestadora de serviço, ou seja, a desenvolvedora da IaaS ou PaaS, e normalmente se dá através de API

(Application-Programming Interface, do inglês, Interface de Aplicação de Programação).

Já a comunicação destas ferramentas com uma interface acessível por um *website* deve também ser ágil e segura, assim como o acesso ao *website* e às informações contidas nele deve ser restrita. No capítulo seguinte, serão discutidos os detalhes sobre os protocolos MQTT e HTTP/HTTPS.

5.4 ARQUITETURA DE PROCESSAMENTO

Como todo o sistema operacional deste projeto será armazenado em nuvem, é necessário que se decida a melhor arquitetura a ser utilizada, tanto para garantir melhor funcionamento como para minimizar os custos de operação (dados transmitidos, uso de máquinas, etc) para que seja economicamente viável.

Atualmente existem diversos modelos de arquitetura a serem utilizados, o mais comum, porém, é a Monolítica. Um outro modelo que têm crescido muito, principalmente para a aplicação em nuvem e IoT, é a estrutura de Microserviços. Ambos modelos serão discutidos no capítulo a seguir.

O modelo mais comum de arquitetura é a monolítica, conhecida como Sistema Operacional Monolítico (SOM) e, em linhas gerais, é assim definida pois seu código todo é como uma coleção de procedimentos, dispostos de forma contínua e interconectados. Podem haver funções que chamam outros procedimentos e assim por diante, porém o código é único.

6 DESENVOLVIMENTO E AVALIAÇÃO

O foco deste capítulo é trazer informações específicas sobre os artefatos levantados no capítulo anterior e então compará-los para que, futuramente, se possam extrair conclusões acerca deles e diminuir as opções que solucionam o problema levantado.

6.1 CLOUD COMPUTING

Primeiramente, serão analisadas as plataformas mais populares disponíveis no mercado voltadas à área de IoT com ferramentas específicas para automação industrial. Todas as informações descritas foram obtidas nos websites dos fornecedores: MindSphere da Siemens [20]; Google Cloud Platform for IoT [21]; AWS IoT da Amazon [22]; Azure IoT Suit da Microsoft [23]; Bluemix da IBM [24].

Para se analisar a viabilidade de utilização destas em um projeto, é necessário se responder algumas perguntas acerca de seus funcionamentos, possibilidades de implementação, versatilidade, comunicação, ferramentas disponíveis, limitações, custos e afins.

Foi estudada então a definição inicial de cada uma, de acordo com o desenvolvedor, as possibilidades de comunicação com dispositivos industriais, a existência tanto de um local de armazenamento dos dados na nuvem como seu processamento, também a viabilidade de se realizar testes antes de adquirir o produto.

6.1.1 Apresentação das Principais Desenvolvedoras

6.1.1.1 Siemens (MindSphere)



Figura 4 - MindSphere da Siemes [25]

Definição:

MindSphere é um sistema operacional hospedado em nuvem da Siemens que permite integração com milhares de dispositivos assim como outras ferramentas da fabricante – desde aplicações para desenho de projetos de plantas industriais à análise de *Big Data*.

Comunicação com dispositivos:

Permite integração com os dispositivos da Siemens e outros parceiros de forma simplificada estilo *plug-and-play*. Possui ferramentas que podem ser utilizadas para a tradução e adaptação à protocolos industriais e outras linguagens de máquina.

Solução para armazenamento dos dados:

A Siemens oferece três modelos de nuvem: pública, privada ou local. Mais detalhes sobre capacidade de armazenamento, ferramentas de busca, gerenciamento e visualização da base de dados não ficam claras na descrição da plataforma.

Solução para processamento dos dados:

Todo o processamento de dados é feito pelo sistema operacional. Por ter parte do seu sistema aberto, permite certa personalização e criação de novas ferramentas. A maior parte das funções, porém, são aplicações fechadas já desenvolvidas pela Siemens.

Opção de teste:

Informação não disponível na plataforma online do desenvolvedor.

6.1.1.2 Google (Google Cloud Platform for IoT)

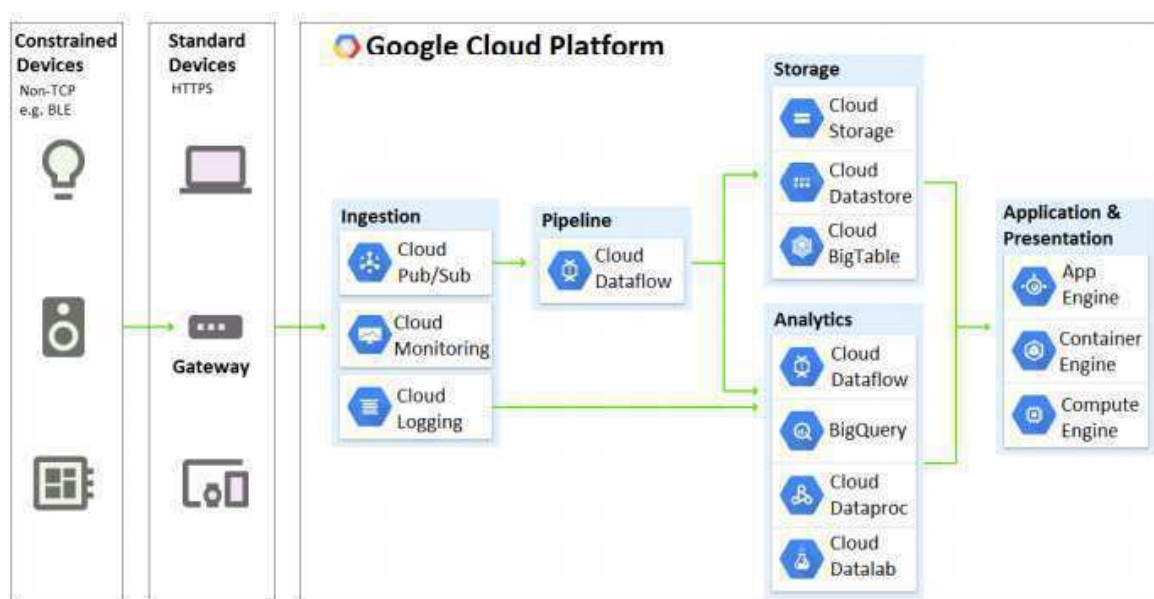


Figura 5 - Google Cloud Platform for IoT [25].

Definição:

Google Cloud Platform for IoT é uma PaaS com serviço completo de ponta-a-ponta que oferece diversas soluções para a comunicação e tratamento dos dados obtidos da rede.

Comunicação com Dispositivos:

Esta plataforma tem 3 tipos de entrada e tratamento de dados:

- Cloud Pub/Sub: é um serviço de mensagem em tempo real que permite comunicar com diversas aplicações e dispositivos ao mesmo tempo, sendo totalmente escalável por suportar até 10 mil mensagens por segundo. Comunica com os dispositivos da rede industrial via gateway.

- Cloud Monitoring: coleta dados da plataforma para exibi-los em gráficos, painéis e alertas.

- Cloud Logging: função que coleta os dados de *login* na plataforma de forma que estes possam ser visualizados, filtrados e exportados.

Solução para armazenamento dos dados:

Depois de coletados os dados, o Google oferece um serviço de *pipeline* chamado Cloud Dataflow que pode ser utilizado para pré-processar o alto volume de dados antes que estes sejam armazenados.

Para o armazenamento em si, oferecem 3 ferramentas distintas: Cloud Storage, Cloud Datastore e Cloud BigTable. Ainda oferece serviços de visualização e análise desta base de dados, como o BigQuery, o Cloud Dataproc e o Cloud Datalab. O BigQuery, por exemplo, é uma ferramenta que permite a visualização e análise de dados armazenados nas BigTables, permitindo a construção de *dashboards* totalmente personalizados, manipulados por uma interface dinâmica com comandos em blocos. O BigTable, permite a extração de relatórios e é acessível de forma simples com comandos de banco de dados em SQL.

Solução para processamento dos dados:

A Google traz algumas soluções para processamento de dados voltado para IoT, dentre elas a Cloud IoT Core. Esta é uma ferramenta totalmente gerenciável que permite o processamento de todos dados obtidos assim como a criação de funções que gerenciam e controlam remotamente os dispositivos, podendo ser inseridos inclusive no processo de aprendizagem de máquina.

Opção de teste:

A Google disponibiliza US\$300 para serem utilizados com qualquer ferramenta pela duração de 12 meses, sem restrição de funcionalidades. Seu método de cálculo de custo é de acordo com o uso (dados processados, dados armazenados, etc). Traz uma diversidade de opções para que o usuário possa escolher a melhor relação custo benefício e economizar.

6.1.1.3 Amazon (AWS IoT)

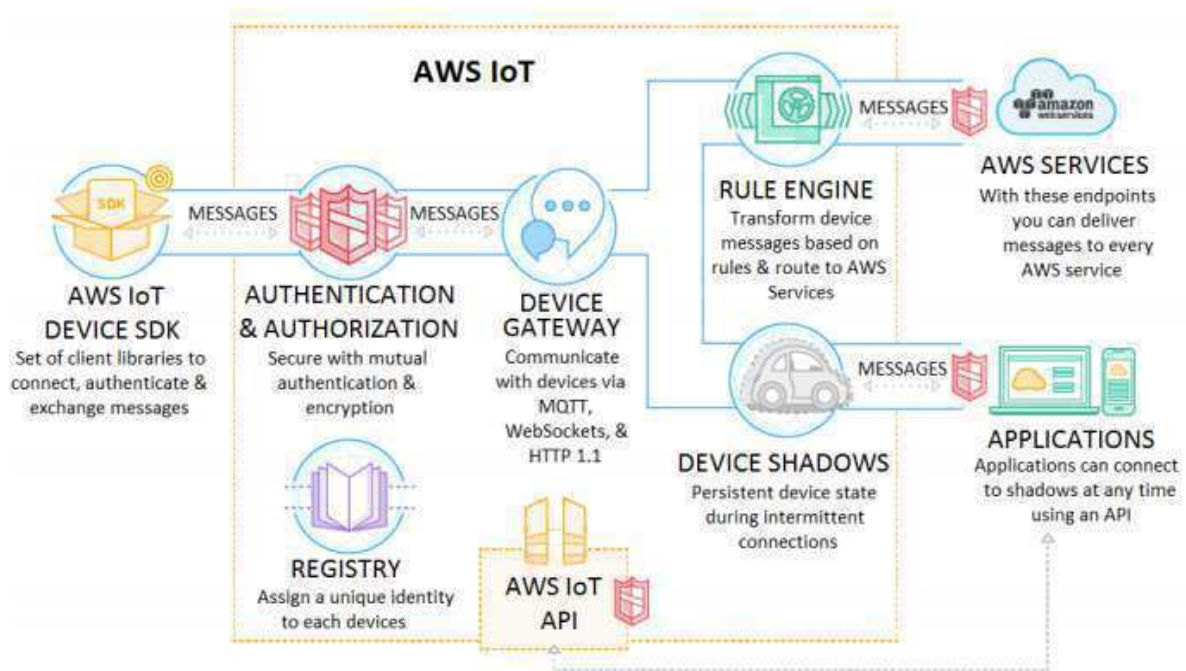


Figura 6 - Amazon AWS IoT [25].

Definição:

A AWS da Amazon é uma plataforma desenvolvida para IoT que permite integração com dispositivos, processamento e armazenamento dos dados, assim como oferece diversas ferramentas para gerenciá-los em tempo real, obtendo os estados que se encontram e enviando comandos.

Comunicação com dispositivos:

A comunicação com os dispositivos é feita através do AWS IoT SDK Device que se utiliza do protocolo MQTT e tem ferramentas de autenticação para garantir a segurança da conexão. A AWS suporta bilhões de aplicativos interconectados e trilhões de mensagens trocadas. Para mecanismos intermitentes, é possível criar uma sombra destes e continuar a gerenciá-los à distância, lendo os estados e enviando comandos para quando estiverem operantes.

Solução para armazenamento dos dados:

Depois de captados os dados dos dispositivos autenticados, é possível armazená-los em algum serviço de banco de dados da Amazon como, por exemplo, o S3. A Amazon oferece diversas mais ferramentas de análise destes dados e uma ferramenta exclusiva de *Machine Learning*.

Solução para processamento dos dados:

A AWS IoT é uma ferramenta que permite a criação de regras aplicadas aos dados de um ou mais dispositivos. Estes dados são processados conforme as regras

criadas e direcionadas para a aplicação desejada. A plataforma também permite a criação de diversas aplicações em uma infinidade de ferramentas disponibilizadas pela Amazon, como AWS Lambda, Amazon Kinesis, Amazon S3, Amazon Elasticsearch.

Opção de teste:

A Amazon oferece a possibilidade de utilizar o AWS IoT de forma a trocar 250.000 mensagens (publicadas ou entregues) por mês, por até 12 meses, de forma gratuita. Para todas as outras ferramentas, também há a possibilidade de teste gratuito, sendo necessário consultar o desenvolvedor para mais detalhes (algumas ofertas, dentro de parâmetros específicos, nunca expiram, já outras estão totalmente gratuitas por um tempo determinado).

6.1.1.4 Microsoft (Azure IoT Suite)

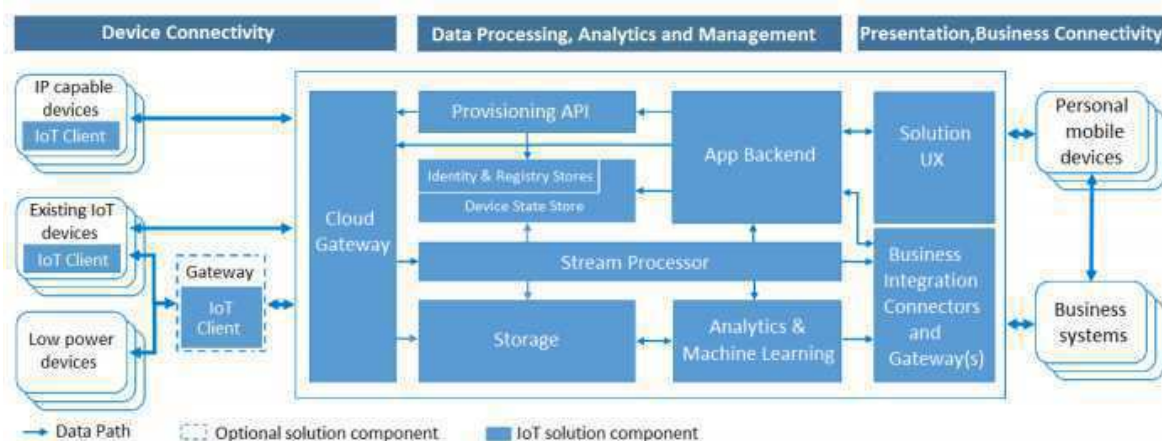


Figura 7 - Microsoft Azure IoT Suite [25].

Definição:

O Azure IoT Suite é um SaaS que permite integração com os componentes da sua de forma rápida, escalável e sem interromper a linha de produção. O software funciona basicamente como um sistema supervisorio que identifica com facilidade e precisão quais componentes apresentam falha. Interface gráfica bastante intuitiva, de fácil navegação e que permite integração com outras ferramentas da Microsoft como, por exemplo, um sistema de agendamento de manutenção preventiva. As informações de cada componente são dispostas na interface em tempo quase real, sendo os alertas personalizáveis.

Comunicação com dispositivos:

É necessário contratar o serviço Azure IoT Hub para conectar com dispositivos através de *gateways*. Os protocolos aceitos de linguagem são HTTP, MQTT e AMQP.

Solução para armazenamento dos dados:

A Microsoft traz diversas soluções de armazenamento em nuvem que podem ser contratadas. Não sendo nenhuma específica para este tipo de armazenamento, é necessário

que se faça uma análise da necessidade do projeto e então se opte pela melhor opção – algumas das nuvens que permitem movimentação rápida de dados, de forma segura, eficiente e escalável. Algumas das alternativas são para tipos específicos de arquivos ou apenas *backup* de dados ou sites. Uma das nuvens especificamente foi dimensionada ser um repositório de dados para *Big Data*.

Solução para processamento dos dados:

A Microsoft têm diversas ferramentas para processamento de dados, seja para gerenciar o fluxo e controlar as máquinas dos clusters, fazer análises de *Big Data* ou visualizar os dados em tempo real, telemetria dos dispositivos conectados, *Machine Learning*, etc.

Opção de teste:

A Microsoft oferece US\$200 para ser utilizado com qualquer produto Azure. Após iniciado o teste, oferecem as ferramentas mais populares para serem testadas por até 12 meses, cada uma com uma limitação específica a seu uso (quantidade de mensagens trocas, quantidade de usuários ou componentes conectado, quantidade de blocos de lógica de funcionamento, hora de uso, etc). Oferece também, com restrições, uso contínuo e gratuito de 25 de suas ferramentas. Nem todas são de interesse deste projeto, porém algumas permitem *load balance* dos *clusters* utilizados, monitoramento de quantidade de mensagens trocadas, etc.

6.1.1.5 IBM (Bluemix)

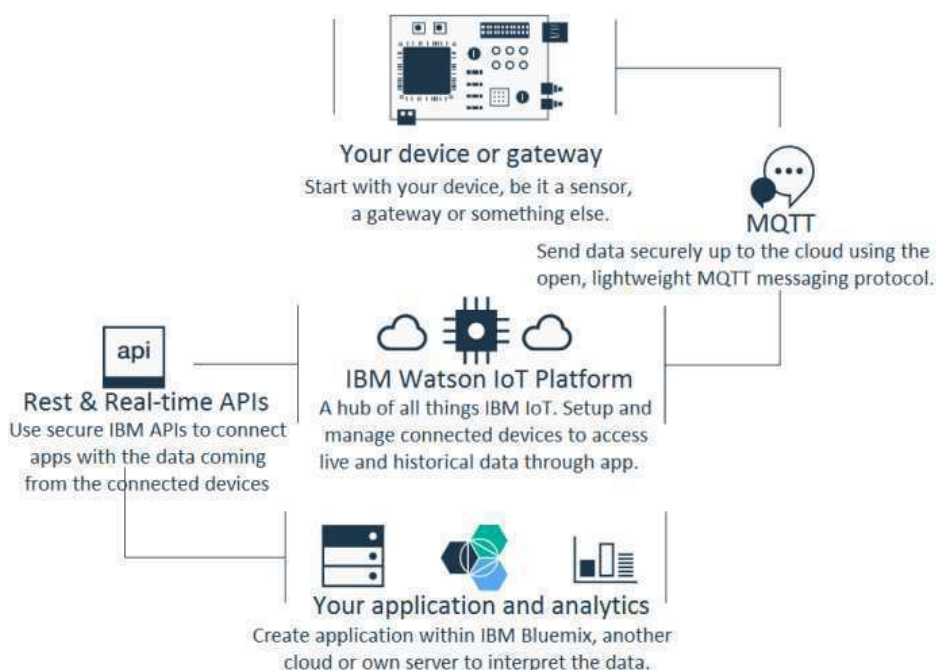


Figura 8 - IBM Bluemix [25].

Definição:

Bluemix da IBM é uma PaaS, definida como uma plataforma de desenvolvimento open cloud, e está hospedada no serviço de infraestrutura (IaaS) da IBM, chamado Softlayer. Para utilizar o Bluemix em uma rede industrial, é necessário que esta esteja conectada com o Watson IoT for Bluemix, que faz a comunicação entre rede e plataforma.

Comunicação com dispositivos:

Tem comunicação com os dispositivos da rede industrial e se utiliza do protocolo de linguagem MQTT ou HTTP. Para conectar um dispositivo ao Watson, é necessário primeiro registrá-lo no Bluemix. Utiliza comunicação via API para levar os dados dos dispositivos para os aplicativos de gerenciamento.

Solução para armazenamento dos dados:

A IBM traz diversas opções de armazenamento em nuvem para fins específicos, sendo 2 deles aplicáveis aos fins deste projeto. Alguns são escaláveis com relação ao número de máquinas operando e podem possuir ferramentas de busca rápida para o usuário final.

Solução para processamento dos dados:

A plataforma permite que se criem aplicativos de gerenciamento dos dados obtidos ou que estes dados sejam conectados à outra plataforma, seja em nuvem ou local.

Opção de teste:

A IBM oferece 2GB de memória de tempo de execução e de armazenamento por 30 dias, assim como acesso a 10 ferramentas do desenvolvedor. Mais detalhes sobre uso, tráfego e ferramentas não estão disponíveis na plataforma online da IBM.

6.1.2 Análise Comparativa das Desenvolvedoras

Para se fazer uma análise de forma visual, os dados apresentados na seção 6.1.1 serão dispostos em tabelas e gráficos. O objetivo é que a escolha da plataforma se dê de forma intuitiva de acordo com a necessidade do projeto, para isso, serão comparados os seguintes critérios:

- Flexibilidade x Facilidade:

Flexibilidade diz respeito à quanto o desenvolvedor dá liberdade para que o contratante desenvolva novas ferramentas e utilize as existentes de forma personalizada, se adaptando perfeitamente ao uso e sem gerar gastos adicionais.

Facilidade dirá respeito à quanto é simples construir novas funcionalidades e integrá-las às outras existentes, ou até mesmo integrar as soluções existentes.

- Ferramentas disponíveis no teste x Tempo de teste

Ferramentas disponíveis no teste, como o próprio nome sugere, é a quantidade

de ferramentas que o desenvolvedor disponibiliza durante o período de teste. Há de se levar em conta, também, as limitações destas ferramentas uma vez que alguns desenvolvedores as disponibilizam, porém limitam a quantidade de horas a serem utilizadas ou a quantidade de mensagens trocadas.

Tempo de teste se refere à quantos meses a plataforma ou o serviço estará disponível para ser avaliada. Este dado é de extrema importância uma vez que, dentro do ambiente acadêmico, os projetos podem ser desenvolvidos ao longo de alguns anos e, algumas vezes, sem financiamento. Ter disponibilidade de teste tanto para analisar a viabilidade de uso da ferramenta como também justificar seu uso para a agência financiadora é crucial ao sucesso do projeto.

6.1.2.1 Flexibilidade x Facilidade

Analisando-se as opções de integração entre plataformas e ferramentas, os protocolos de comunicação disponíveis, as linguagens de programação suportadas, a possibilidade de criação de qualquer software dentro da plataforma assim como novas ferramentas e funcionalidades, pontuou-se cada desenvolvedora. O resultado da análise se encontra na Figura 9.

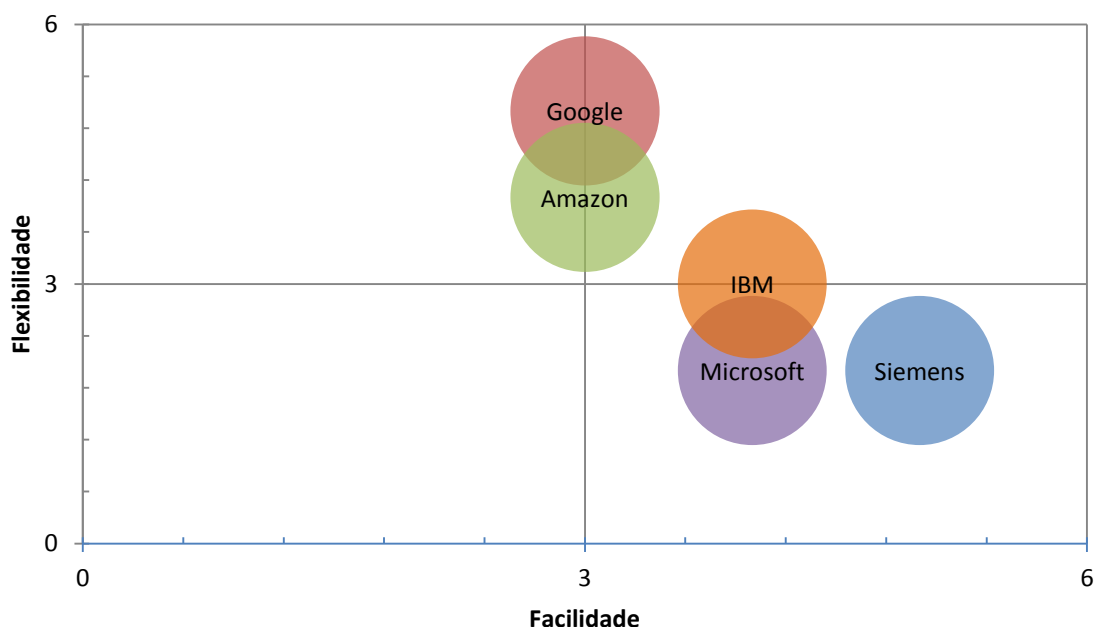


Figura 9 - Gráfico de Quadrantes comparando a flexibilidade em relação à facilidade de implementação de acordo com cada desenvolvedora.

Todas as plataformas estudadas dão a opção de desenvolvimento, mas há de se destacar que algumas, como Google e Amazon, permitem que o usuário faça qualquer uso desejado do serviço, enquanto outras, como Microsoft e Siemens, disponibilizam

ferramentas para criação de novas funcionalidades dentro apenas do uso para IoT.

Todas elas integram com outros serviços ou softwares de suas desenvolvedoras ou parceiras, porém algumas integram apenas com softwares que requerem licença extra e têm funcionalidade definida, sem a possibilidade de adaptação ao uso. Por exemplo, a Siemens possibilita que se comunique com seu software (que pode ser tanto instalável como versão para nuvem) TIA Portal que auxilia na construção das redes de comunicação industriais, mas o uso ou aplicação do software é o padrão já comercializado.

Enquanto isso, a maioria das ferramentas disponíveis pela Google e pela Amazon são métodos para gerenciar seus dados, para personalizar seu processamento e os *clusters* utilizados, para gerir os modelos de busca, de armazenamento, de visualização dos dados, controle de gastos e afins. Dentro de suas plataformas é possível que se instale qualquer software desejado em uma infinidade de linguagens de programação.

Por outro lado, flexibilidade parece estar relacionada inversamente à facilidade de uso destes serviços. Mesmo tendo ferramentas bastante intuitivas, a Google e a Amazon requerem que o software que definirá a função básica do serviço criado seja inteiramente programado pelo usuário. Com isso, estas desenvolvedoras foram classificadas com menor facilidade de implementação.

Por outro lado, a Siemens classifica sua plataforma como *plug-and-play*, expressão utilizada no meio digital para indicar que basta conectar os dispositivos à ela para que sejam reconhecidos, a comunicação seja estabelecida e as funcionalidades desejadas comecem a operar.

Já as demais plataformas trazem soluções também mais simples de implementação, porém com funcionalidades pré-estabelecidas e voltadas para a área de IoT. Todas as plataformas são escaláveis e, de certa forma, têm possibilidade de criação de novas funções e personalização do uso. Nenhuma delas é apresenta extrema dificuldade no uso ou questões que possam vir a serem críticas, inviabilizando o projeto. Mesmo as que facilitam e aconselham o uso com dispositivos da mesma marca ou parceiros, não inviabilizam o uso para demais fabricantes – como é o caso da Siemens, grande produtora de componentes para automação industrial.

6.1.2.2 Ferramentas Disponíveis x Tempo de Teste

As desenvolvedoras foram posicionadas nos quadrantes de acordo com o tempo de teste em meses disponíveis e a quantidade de ferramentas que podem ser utilizadas dentro deste período. O resultado desta análise se encontra na Figura 10.

Como não há uma normalização ou padronização da métrica de precificação do uso, foi necessário realizar uma terceira análise. Esta análise se constituiu em avaliar as

limitações impostas para a utilização das ferramentas. Por exemplo, a Google disponibiliza US\$300 para serem utilizados por até 12 meses. Já a Amazon disponibiliza 250.000 mensagens por mês.

Para que seja possível comparar as duas plataformas, é necessário compreender a precificação da Google por cada mensagem para tentar quantificar de acordo com a Amazon. Na análise feita, 250.000 mensagens por mês seria mais vantajoso que o valor livre da Google somente se o tamanho das mensagens, no total mensal, não excederem 250MB – abaixo deste valor, nada é cobrado pela Google (Anexo A – Detalhes da precificação da Google Cloud – IoT Core).

Há também um fator a ser levado em conta com relação a quantidade de ferramentas disponíveis que se refere às limitações impostas sobre o uso de cada ferramenta no período de teste. Novamente, se utilizando da comparação entre Google e Amazon, a primeira não restringe o uso de nenhuma ferramenta. Já a Amazon oferece limitações em todas as ferramentas, porém 12 delas, dentro de certos parâmetros de uso, jamais serão cobradas (Anexo B – Detalhes das Limitações de Teste da Amazon AWS IoT).

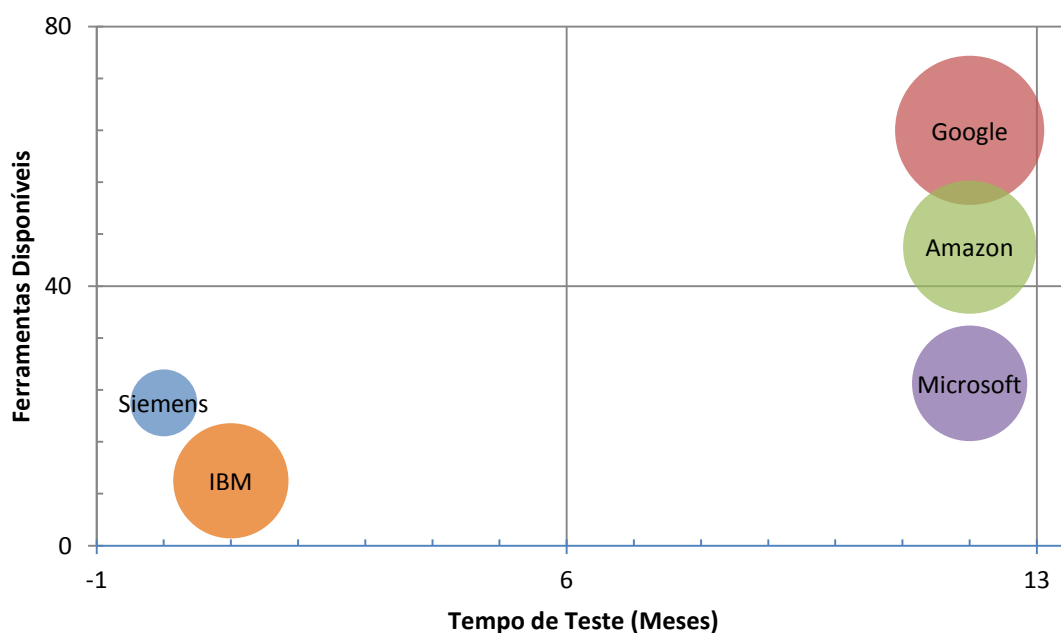


Figura 10 - Gráfico de Quadrantes comparando ferramentas disponíveis para teste assim como tempo de teste por cada desenvolvedora

Considerando estes fatores subjetivos, foi dimensionada a área da esfera que representa cada desenvolvedor no gráfico de quadrantes. As dimensões foram dadas de forma comparativa, sendo baseadas sempre nas informações providas nos *websites* dos mesmos.

Três das plataformas disponibilizam 12 meses de teste gratuito. Comparando entre elas, a Google é quem oferece o maior leque de ferramentas que podem ser utilizadas em conjunto com a solução IoT, sendo seguida pela Amazon e pela Microsoft, que não

oferece todas para teste (Anexo C – Detalhes da Conta Gratuita e Precificação da Microsoft Azure e Hube IoT).

A Amazon possui restrições por quantidade de mensagens trocadas ou armazenamento para a maioria de suas ferramentas neste período, porém oferece 12 delas de forma gratuita sempre, desde que se trabalhe nos limites estabelecidos.

A Google, dentre as três, recebeu maior pontuação em relação ao caráter subjetivo uma vez que o valor para teste pode ser utilizado por qualquer ferramenta, já a Amazon limita o uso de todas ferramentas e isso pode gerar uma incongruência nas necessidades – por exemplo, pode-se transferir uma alta quantidade de dados, porém o armazenamento está restrito a 1GB. Novamente, a Microsoft não disponibiliza teste de todas ferramentas.

Dentre as restantes, não há nenhuma informação sobre disponibilidade de período de teste do MindSphere pela Siemens e, ao que parece, as demais ferramentas da desenvolvedora não estão disponíveis para compra no Brasil. Já a IBM oferece 1 mês para testar a plataforma BlueMix com as funcionalidades embutidas à ela (Anexo D – Detalhes do Teste da IBM Bluemix).

6.2 GATEWAY - DISPOSITIVOS DE COMUNICAÇÃO

A Siemens elevou este conceito idealizado e desenvolveu, juntamente à RS Components e à Arduino, um dispositivo inteligente, capaz de processar os dados em alta velocidade e tomar decisões independentes, controlando a rede industrial. O objetivo da desenvolvedora era criar um elemento que fosse crucial e dedicado à integração da rede à Internet das Coisas. As características descritas a seguir foram obtidas no website da Siemens [26] e do *datasheet* do componente [27].

O gateway em questão foi apresentado sob o nome de SIMATIC IOT2000 (tem também as versões 2020 e 2040, ainda em uso experimental e para fins educacionais). Trata-se de uma plataforma *open source*, o que significa que seu código é livre para alterações e personalizações, porém seu *hardware* não é livre e a Siemens não disponibilizou detalhes sobre sua composição. Seguem detalhes técnicos disponibilizados abaixo:

- Processador Intel Quark x1020 (+Secure Boot) com 1 núcleo, velocidade 400MHz e 16KB de Cache;
- 1GB de memória RAM;
- 2 portas Ethernet;
- 2 interfaces RS232/485;

- Clock RTC com bateria própria;
- Suporte para Linux Yocto.

Como qualquer outro componente de automação da Siemens, possui carcaça para uso em ambiente industrial, podendo inclusive ser anexada aos trilhos dos painéis de controle. A desenvolvedora também garante robustez de hardware e funcionamento constante, podendo operar em modo contínuo, sem risco à produção.



Figura 11 - Gateway SIOMATIC IOT2000 da Siemens. À esquerda, exemplo de componente montado no trilho de um painel industrial. À direita, componente sem a tampa, exibindo seu hardware, entradas, saídas e conexão com shield.

O que é inédito neste componente, além de ter sido desenvolvido especialmente para a Indústria 4.0, é o fato de ser composto por um processador Intel Quark, similar ao utilizado em outras SBCs, e dar suporte ao Linux Yocto. Isto significa que é por si só um microcomputador e que permite a instalação de uma infinidade de softwares, programados em diferentes linguagens e com diferentes níveis.

Além desta flexibilidade, conta ainda, de forma inédita em ambiente industrial, com uma interface de conexão com Arduino UNO ou qualquer um de seus *shields*, possibilitando que se utilize as facilidades da plataforma Arduino *open source* e *open hardware*, de simples programação, diretamente no *gateway*. Como a memória do Arduino não é grande o suficiente para programas extensos, o *gateway* tem uma entrada para cartão de memória, possibilitando a execução de programas mais extensos e complexos.

Apesar de existirem diversos trabalhos envolvendo a plataforma Arduino na Indústria 4.0, sua composição tanto física (em termos de encapsulamento) como de *hardware* não têm robustez suficiente para aplicação industrial. A interação desta com os demais componentes também não se dá de forma simples e escalável devido às limitações de entradas e saídas.

Unindo uma plataforma com processamento robusto e fácil integração à rede industrial à outra de extrema flexibilidade e fácil programação, as possibilidades de soluções a serem desenvolvidas são muitas. O *Shield* que possibilita a integração com o Arduino também traz outras opções de portas, permitindo fácil conexão com padrões industriais e expandido a capacidade do *gateway*. É possível, através destas portas chamadas de GPIOs

(*General Purpose Input/Output*, do inglês, Entradas e Saídas de Uso Genérico), selecionar arbitrariamente a linguagem de programação ou ambiente de desenvolvimento que se deseja utilizar. Os detalhes técnicos das interfaces do *Shield* estão dispostos a seguir.

- Arduino: GPIO;
- 5 Entradas Digitais (DI) de 24V, 20Hz e atraso de 1.5ms, com isolamento elétrica;
- 2 Saídas Digitais (DO) com isolamento elétrica, velocidade 10Hz;
- 2 Entradas Analógicas (AI) de 9 bits, 0-10V e 0-20mA selecionáveis.



Figura 12 - Shield que pode ser adicionado ao SIOMATIC IOT2000. No canto superior esquerdo, foto do Shield apenas. No canto superior direito, pode-se observar a montagem do Shield na placa do gateway. Já a imagem inferior mostra o gateway fechado, com o Shield.

O SIMATIC IOT2000 possui ainda um PCI (Peripheral Component Interconnect, do inglês, Conector de Componente Periférico) da Intel para conexão de uma placa para *Bluetooth* e *Wifi*. O software está disponível no site da Intel e é de fácil instalação no sistema operacional Linux. Possui também uma entrada USB e uma mini USB, compatíveis com outros receptores e SBCs, como, por exemplo, o comunicador wifi da Raspberry Pi.

Com duas portas para ethernet, o SIMATIC IOT2000 tem a capacidade de coletar uma grande quantidade de dados dos componentes da rede, compacta-los, traduzi-los para diferentes protocolos de comunicação e enviá-los à nuvem para seu processamento e armazenamento. Com seu processador e suas saídas, é possível enviar comandos da

nuvem para a rede industrial, elevando o potencial de controle de toda a planta. Pode-se pensar em um cenário no qual diversos setores da planta ou mesmo diferentes plantas estejam interconectadas pela nuvem e, com o processamento instantâneo dos dados, é possível ter mapeado todo o processo de produção de forma que sua eficiência seja máxima.

6.3 PROTOCOLOS DE COMUNICAÇÃO

6.3.1 MQTT

Message Queuing Telemetry Transport (MQTT) foi desenvolvido por Andy Stanford-Clark (IBM) e Arlen Nipper (Eurotech) em 1999 para monitoramento de uma linha de óleo através do deserto [28]. Este protocolo aberto é leve, é baseado no modelo *publisher/subscriber*, de fácil implementação e estabelece uma comunicação assíncrona entre as partes. Por suas características, pode ser um bom recurso para situações em que a banda de comunicação é restrita ou inconsistente. Também pode ser um ótimo recurso a ser utilizado em dispositivos com processamento limitado e com pouca memória [29].

Como o protocolo utiliza o padrão *publisher/subscriber*, permite que muitas mensagens sejam transmitidas, desassociando qual mensagem é de qual componente [29]. Em linhas gerais, este modelo de comunicação faz com que as mensagens pertençam à “tópicos” e os receptores que têm interesse nestes tópicos “se inscrevem” para receber estas mensagens apenas.

Com relação à segurança, MQTT não se utiliza de criptografia para proteger a comunicação, porém é possível utilizar a versão “Secure-MQTT” que criptografa as mensagens de ponta a ponta utilizando as camadas TLS (*Transport Layer Security*) e SSL (*Secure Socket Layer*) [28].

6.3.2 HTTP/HTTPS

O Hypertext Transfer Protocol, (HTTP) foi definido pela World Wide Web Consortium (W3C) que é a principal organização responsável pela padronização da World Wide Web (WWW). O HTTP é baseado no modelo cliente/servidor, no qual o cliente faz uma requisição e o servidor responde a requisição com a informação desejada, sendo, portanto, uma conexão intermitente [25].

Para grandes fluxos de dados, esta intermitência não é favorável pois não garante a conexão entre as pontas – a conexão é estabelecida para se fazer a requisição e é encerrada depois de respondida. Porém é o protocolo mundialmente utilizado para

comunicação de dados via internet [25].

Como o nome diz, o protocolo baseia-se na comunicação de hipertextos, que são textos estruturados (cabeçalho, corpo, requisição, etc) e, como no MQTT, possui uma versão segura, o HTTPS, que também se utiliza das camadas TLS e SSL para criptografia da mensagem [25].

6.4 ARQUITETURA DO PROCESSAMENTO

Nesta etapa, se discutirá sobre dois modelos de sistema operacional que podem ser utilizados na programação do processamento dos dados em nuvem. Esta análise visa levantar questões relativas às vantagens e desvantagens de cada modelo mediante a diferentes propósitos:

- uma arquitetura bem planejada traz vantagens na velocidade de processamento;
- pensando o processamento e a troca de mensagens de forma eficiente, há ganho também em custos de hospedagem na nuvem;
- há de se considerar a dificuldade de implementação e de atualização de novas versões do sistema operacional de acordo com a limitação de mão de obra do projeto.

O modelo mais comum de arquitetura é a monolítica, conhecida como Sistema Operacional Monolítico (SOM) e, em linhas gerais, é assim definida pois seu código todo é como uma coleção de procedimentos, dispostos de forma contínua e interconectados. Pode haver funções que chamam outros procedimentos e assim por diante, porém o código é único.

Este modelo tradicional é dividido entre interface do usuário, chamada de *front-end*, componentes de lógica de serviço e banco de dados. O componente do *front-end* roda diretamente no dispositivo do usuário, por exemplo, na página da internet pelo navegador. O componente responsável por armazenar o banco de dados é o chamado *back-end*. Já os componentes de lógica de serviço que serão hospedados na nuvem e nela processados e são estes componentes que conectam as requisições do usuário feitas no *front-end* aos dados disponíveis no *back-end* [30].

A vantagem desta arquitetura é que para projetos menores, com uma quantidade reduzida de procedimentos, funções e também mão de obra, é mais fácil de se implementar. Como o código é único, ele pode ser pensado de maneira mais consistente e desenvolvido por menos pessoas. Dentro do código é definido tudo que é necessário e são descritas todas suas funções. Ele não é executado de forma linear, podendo uma função chamar outra, mas é, em linhas gerais, programado e arquitetado linearmente [30].

Este ponto, que traz vantagens, traz também desvantagens. Para que novas

versões sejam lançadas, as correções são pouco mais trabalhosas já que haverá de se cuidar para que todas as partes envolvidas na alteração sejam corrigidas (chamada de funções, envio de valores de variáveis, etc). Caso haja qualquer falha, o sistema todo falha [30].

Outro ponto crítico do monólito é sua escalabilidade. Caso a aplicação seja para multiusuários e o fluxo de acesso seja alto, pode ser que o sistema operacional não comporte o uso e seja necessário replicá-lo. Mesmo que os usuários estejam divididos em diferentes perfis, utilizando diferentes ferramentas da aplicação, caso qualquer uma delas se sature, é preciso replicar todo o sistema operacional para uma nova máquina da infraestrutura contratada, gerando gastos desnecessários [31].

A ideia da replicação do monólito para novas máquinas da infraestrutura está exemplificada na Figura 13 - Replicação do Sistema Operacional Monolítico em novas máquinas da nuvem para múltiplos usuários.

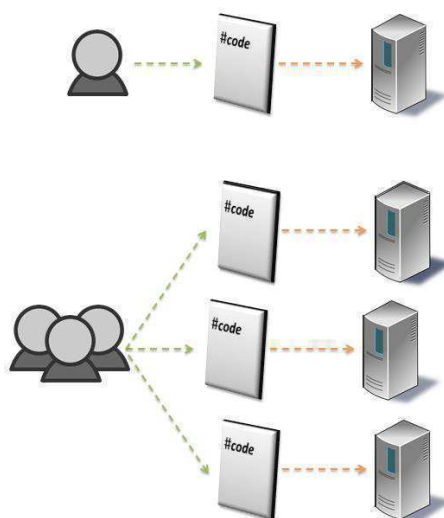


Figura 13 - Replicação do Sistema Operacional Monolítico em novas máquinas da nuvem para múltiplos usuários.

A tendência atual tem sido a migração deste sistema operacional para o modelo de Microserviços (MSA, *Microservices Architectures*) que foi pensado especialmente para o processamento em nuvem. Ao invés de um bloco único como o modelo anterior, os microserviços constituem blocos específicos que cumprem uma função apenas, de forma prática e eficiente. Os microserviços se conectam e se comunicam conforme necessário e podem ser desenvolvidos em linguagens distintas caso necessário – só é preciso garantir a comunicação [30].

Este modelo de blocos específicos que cumprem funções únicas permite que seu uso seja escalável, uma vez que, caso o fluxo de um serviço esteja saturado, é necessário a replicação apenas desta função para uma nova máquina da infraestrutura. É possível também que sejam feitas correções e atualizações em cada serviço

independentemente – os chamados *deploys* [30]. Neste caso, o *deploy* é feito em uma máquina apenas e as demais que estiverem em uso replicam a versão atualizada. Esta ideia está exemplificada na Figura 14 - Replicação da Arquitetura de Microserviços em novas máquinas da nuvem para múltiplos usuários..

Este formato de replicação pode gerar economia de processamento e, por conseguinte, de custo do serviço contratado para infraestrutura. Pensando ainda mais em aperfeiçoar este modelo de replicação de microserviços, a Amazon desenvolveu uma ferramenta chamada AWS Lambda. Segundo estudos, a redução de custo pode chegar em até 77% com esta ferramenta que automatiza os *deploys* [31].

Apesar de suas vantagens aparentes, é necessário se fazer ressalvas com relação à este modelo de arquitetura. Como os microserviços funcionam independentemente, nem sempre é possível se manter um registro atualizado do uso de cada componente. É difícil também acompanhar quais serviços estão operantes e quais apresentam falhas na execução. Como se trata de uma arquitetura mais complexa já que cada microserviço tem que ser programado de forma a ser totalmente autônomo e a comunicação entre todos deve funcionar plenamente e ser eficiente, este modelo exige mais dedicação na hora de ser implementado. Pode ser que para projetos menores, com menos mão de obra, não seja totalmente viável [32].

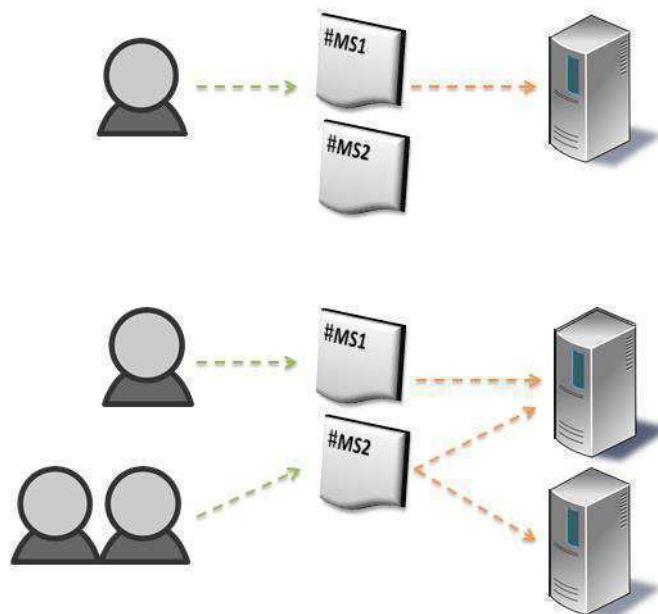


Figura 14 - Replicação da Arquitetura de Microserviços em novas máquinas da nuvem para múltiplos usuários.

A escolha do método correto pode ser complexa porém é crucial, principalmente a médio e longo prazo, uma vez que as consequências desta escolha se refletirão quando realmente houver a necessidade de escalabilidade e de melhorias para futuras versões.

7 CONCLUSÃO

Trazendo um método de pesquisa atual, este trabalho analisou as alternativas existentes no mercado para se desenvolver um serviço de automação industrial voltado à Indústria 4.0 e IoT. Nos capítulos iniciais, se definiram as características principais que este serviço precisa ter para que seja desenvolvido de forma satisfatória.

O propósito, então, é retomar à estas exigências e definir, com base em todos artefatos analisados anteriormente, quais as possíveis e, por que não, melhores alternativas disponíveis, dando continuidade ao *design* deste serviço.

Baseando-se na análise feita em termos de facilidade de uso e implementação versus sua flexibilidade, foi possível concluir que as plataformas da Google e da Amazon se mostraram as mais flexíveis tanto para processamento como armazenamento de dados em nuvem uma vez que é possível se instalar e executar nelas qualquer programa que se tenha desenvolvido.

Para compensar a dificuldade de implementação devido à liberdade oferecida, ambas trazem diversas ferramentas que auxiliam a gestão e comunicação destes dados e serviços disponíveis, tornando seu uso cada vez mais intuitivo. Ao se analisar a quantidade de ferramentas, aplicações e recursos disponibilizados por todas desenvolvedoras estudadas, fica evidente que Google e Amazon têm se dedicado à compreender as necessidades dos clientes e estão alinhadas com a evolução da indústria e economia mundiais.

Outra vantagem na adesão de seus serviços são os planos de tempo de teste das plataformas e ferramentas e, além deste período, possuem custo economicamente favorável tanto em método de precificação como em valor absoluto quando comparadas às outras alternativas apresentadas. Também não há um modelo de fidelidade contratual, permitindo que, caso necessário, se encerre a contratação do plano sem demais custos ou complicações.

O gateway disponível para o projeto, SIMATIC IOT 2000 da Siemens mostrou-se como a alternativa mais moderna atualmente para se fazer a integração da rede industrial com a nuvem. Por construção, o gateway possibilita, de forma confiável e robusta, uma infinidade de criações para diferentes usos e ainda há muito o que se estudar e desenvolver para atingir todo seu potencial.

Analisando as plataformas escolhidas e os protocolos de comunicação de máquinas mais utilizados, o protocolo MQTT é comportando por ambas e inclusive indicado por elas. Para se fazer a comunicação M2M (*Machine to Machine*, do inglês, de máquina à máquina) o protocolo MQTT é mesmo o mais indicado para estabelecer a comunicação da

rede da fábrica com a nuvem. A interpretação dos protocolos *fieldbus* e futura tradução para MQTT é feita pelo próprio *gateway*. Como estudado, o protocolo internacionalmente utilizado na internet é o HTTPs e deve ser utilizado para coletar dados de entrada e saída na interface web e comunica-lo ao *front end* da aplicação.

Pensando em arquitetura de *software*, há de se fazer um estudo mais detalhado das implicações da escolha entre monolítica e de microserviços uma vez que, depois de estabelecida, a migração entre elas pode se dar de forma complexa. Ambas trazem vantagens e desvantagens, portanto aconselha-se medir também os esforços necessários para implementação de cada uma e cruzar com os dados de mão de obra e horas de engenharia disponíveis para o projeto. Para equipes menores, sistemas mais simples e com pouco volume de uso, aconselha-se o uso da arquitetura monolítica.

Assim sendo, este estudo se concretiza como um primeiro passo no desenvolvimento de um novo serviço voltado para IoT, fornecendo as informações iniciais do que deve ser considerado e projetado para futuras aplicações.

8 FUTURAS APLICAÇÕES

O objetivo final que se prevê para este estudo é a criação de uma ferramenta completa de gerenciamento, supervisão, diagnóstico e controle de linhas de produção industriais. Há de se dar os primeiros passos nesta direção de forma cautelosa uma vez que decisões no momento presente terão grande influência na capacidade do projeto se desenvolver ou não.

Agregando este princípio às pesquisas em realização no laboratório, pode-se primeiramente aplicar este conhecimento ao estudo de elementos únicos da rede, para depois ampliá-lo. Uma vez atingindo o sucesso deste projeto, pode-se expandir a ferramenta para o diagnóstico de diferentes componentes de rede até que se tenha criado uma aplicação que consiga identificar os elementos, diferenciar a comunicação entre eles e obter diferentes resultados sobre eles, de acordo com as especificações de cada um.

Esta ferramenta pode ser utilizada tanto para se manter o controle sobre a saúde da rede como para o oferecimento de um serviço de diagnóstico da rede quando esta apresenta falhas, tornando possível apenas com a obtenção dos dados em operação, distinguir e caracterizar seus componentes e localizar os elementos que apresentam erro ou comportamento diferente do esperado.

Soluções mais simples podem também ser desenvolvidas pelos alunos das disciplinas do curso para serem aplicadas nos laboratórios de ensino, nas plantas já existentes para uso educacional, trazendo para o ambiente acadêmico o que se tem de mais moderno em desenvolvimento industrial na atualidade.

Independente dos caminhos que se decida seguir, este trabalho almeja se estabelecer como boa fundação para o crescimento de mais uma área de conhecimento dentro da universidade, trazendo para a pesquisa acadêmica as infinitas possibilidades que a IoT nos proporciona.

9 REFERÊNCIAS¹

- [1] I. Bensat e R. Weber, "Research Commentary: Rethinking Diversity' in Information Systems Research," *Information Systems Research*, vol. 7, n. 4, pp. 389 - 399, 1996.
- [2] R. Drath and A. Horch, "Industrie 4.0: Hit or Hype? [Industry Forum]," vol. June 2014, 2014.
- [3] J.-P. Thomesse, "Fieldbus Technology and Industrial Automation," in *Emerging Technologies and Factory Automation, IEEE 10th Conference on Emerging Technologies and Factory Automation*, Catania, Itália, 2005.
- [4] K. Zhou, T. Liu e L. Zhou, "Industry 4.0: Towards Future Industrial Opportunities and Challenges," em *12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, Zhangjiajie, 2015.
- [5] J. Lee, H.-A. Kao e S. Yang, "Service innovation and smart analytics for Industry 4.0 and big data environment," em *6th CIRP Conference on Industrial Product-Service Systems*, Karlsruhe, 2014.
- [6] X. Wu, X. Zhu, G.-Q. Wu e W. Ding, "Data mining with big data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, n. 1, pp. 97-107, 2014.
- [7] D. Evans, *The Internet of Things - How the Next Evolution of the Internet is Changing Everything*, Cisco IBSG, 2011.
- [8] D. Sima, "Decisive aspects in the evolution of microprocessors," *Proceedings of the IEEE*, pp. 1896 - 1926, Dezembro 2004.
- [9] S. J. e. a. Johnston, "Applicability of commodity, low cost, single board computers for Internet of Things devices," em *Internet of Things (WF-IoT), 2016 IEEE 3rd World Forum on*, Reston, 2016.
- [10] Arduino, "Arduino - Introduction," Arduino, 2017. [Online]. Available: <https://www.arduino.cc/en/Guide/Introduction#>. [Acesso em 15 09 2017].
- [11] A. Grygoruk e J. Legierski, "IoT gateway - implementation proposal based on Arduino board," em *2016 Federated Conference on Computer Science and Information Systems (FedCSIS)*, Gdansk, 2016.
- [12] M. Armbrust e e. al, "A View of Cloud Computing," *Communications of the ACM*, vol. 53, n. 4, pp. 50-58, 2010.
- [13] J. Pan e J. McElhannon, "Future Edge Cloud and Edge Computing for Internet of Things Applications," *IEEE Internet of Things Journal*, p. 1, 30 Outubro 2017.
- [14] M. A. P. A. e. a. Andery, *Para compreender a ciência: uma perspectiva histórica*, Rio de Janeiro: Garamond, 2004.
- [15] K. W. Platts, "A Process Approach to Researching Manufacturing Strategy," *International Journal of Operations & Production Management*, vol. 13, n. 8, pp. 4-17, 1993.
- [16] P. A. C. Miguel, *Metodologia de Pesquisa para Engenharia de Produção e Gestão de Operações*, Rio de Janeiro: Abepro, 2012.
- [17] H. A. Simon, *The Sciences of the Artificial*, London: MIT Press, 1996.
- [18] D. P. Lacerda, A. Dresch, A. Proença e J. A. V. A. Júnior, *Gestão & Produção - Design Science Research: método de pesquisa*, São Carlos: Epub, 2013.

- [19] A. G. L. Romme, "Making a Difference: Organization as Design," *Organization Science*, vol. 14, n. 5, pp. 558-573, 2003.
- [20] Siemens Ltda, "Mindsphere - Sistema Operacional IoT - Brasil," Siemens Ltda, 2017. [Online]. Available: <https://www.siemens.com/br/pt/home/produtos/software/mindsphere.html>. [Acesso em 18 10 2017].
- [21] Google, "Google Cloud Platform," Google Inc., [Online]. Available: <https://cloud.google.com/>. [Acesso em 18 10 2017].
- [22] Amazon, "AWS - Amazon Web Services," Amazon, [Online]. Available: <https://aws.amazon.com/pt/iot/>. [Acesso em 18 10 2017].
- [23] Microsoft, "Azure IoT Suite - Solução de Nuvem IoT," Microsoft, 2017. [Online]. Available: <https://www.microsoft.com/pt-br/internet-of-things/azure-iot-suite>. [Acesso em 18 10 2017].
- [24] IBM, "IBM Bluemix," IBM, [Online]. Available: <https://www.ibm.com/br-pt/marketplace/cloud-platform>. [Acesso em 10 18 2017].
- [25] P. Kumar, *Internet of Things (IoT) Development Platforms – A Case of IBM Bluemix*, Germany, 2016.
- [26] Siemens, "Industrial IoT - PC-based Automation - Siemens," Siemens AG 1996-2017, [Online]. Available: <http://w3.siemens.com/mcms/pc-based-automation/en/industrial-iot/Pages/Default.aspx>. [Acesso em 2017 10 27].
- [27] Siemens, *Operating Instructions - SIMATIC IOT/2020/2040*, 2016.
- [28] P. Thota e Y. Kim, "Implementation and Comparison of M2M Protocols for Internet of Things," em *2016 4th Intl Conf on Applied Computing and Information Technology/3rd Intl Conf on Computational Science/Intelligence and Applied Informatics/1st Intl Conf on Big Data, Cloud Computing, Data Science & Engineering*, Las Vegas, 2016.
- [29] IBM, *MQTT V3.1 Protocol Specification*, Eurotech, International Business Machines Corporation (IBM), 2010.
- [30] C.-Y. Fan e S.-P. Ma, "Migrating Monolithic Mobile Application to Microservice Architecture: An Experiment Report," em *2017 IEEE 6th International Conference on AI & Mobile Services*, 2017.
- [31] M. Villamizar e e. al, "Infrastructure Cost Comparison of Running Web Applications in the Cloud using AWS Lambda and Monolithic and Microservice Architectures," em *2016 16th IEEE/ACM International Symposium on Cluster, Cloud, and Grid Computing*, Bogotá, 2016.
- [32] J. Lewis e M. Fowler, *Microservices - a definition of this new architectural term*, 2014.

¹De acordo com o estilo bibliográfico IEEE 2006.

ANEXOS

ANEXO A – DETALHES DA PRECIFICAÇÃO DA GOOGLE CLOUD – IOT CORE



Google Cloud Internet of Things Core



ENVIAR COMENTÁRIOS

Pricing



Beta

This is a beta release of Cloud IoT Core. This product might be changed in backward-incompatible ways and is not subject to any SLA or deprecation policy.

Cloud IoT Core is priced according to the data volume used in a calendar month:

Monthly data volume	Price per MB	Registered devices	Minimum message size
Up to 250 MB	\$0.00	Unlimited, within QPS maximums	1024 bytes
250 MB to 250 GB	\$0.0045	Unlimited, within QPS maximums	1024 bytes
250 GB to 5 TB	\$0.0020	Unlimited, within QPS maximums	1024 bytes
5 TB and above	\$0.00045	Unlimited, within QPS maximums	1024 bytes

Data volume is based on data exchanged by devices that are connected to Cloud IoT Core. There is no charge for create, read, update, and delete operations through the [device manager](#).

Billable messages

MQTT

- CONNECT
- PUBLISH (both cloud- and device-bound)
- PUBACK (ack of device configuration, cloud-bound)
- SUBSCRIBE
- PINGREQ

HTTP

- Requests: total bytes in the body
- Responses: total bytes in the body

Nonbillable messages

MQTT

- CONNACK
- PUBACK (publish telemetry or state, from cloud to device)
- SUBACK
- PINGRESP
- DISCONNECT
- UNSUBSCRIBE
- UNSUBACK

ANEXO B – DETALHES DAS LIMITAÇÕES DE TESTE DA AMAZON AWS IOT

Detalhes do nível gratuito da AWS

★ EM DESTAQUE 12 MESES GRATUITOS SEMPRE GRATUITO CATEGORIAS DE PRODUTO TODAS

<p>SERVIÇOS DE APLICAÇÕES</p> <p>Amazon API Gateway</p> <p>1 milhão</p> <p>de chamadas de API recebidas por mês</p> <p>Publicação, manutenção, monitoramento e segurança de APIs em qualquer escala</p> <p>Saiba mais sobre o Amazon API Gateway »</p>	<p>IDENTIDADE, SEGURANÇA E CONFORMIDADE</p> <p>Amazon Cloud Directory</p> <p>1 GB</p> <p>de armazenamento por mês</p> <p>Serviço de criação de diretórios totalmente gerenciado e nativo na nuvem para dados com diversas hierarquias</p> <p>Saiba mais sobre o Amazon Cloud Directory »</p>	<p>ARMAZENAMENTO E ENTREGA DE CONTEÚDO</p> <p>Amazon CloudFront</p> <p>50 GB</p> <p>de armazenamento</p> <p>Serviço da web para distribuir conteúdo para usuários finais com baixa latência e altas velocidades de transferência de dados</p> <p>Saiba mais sobre o Amazon CloudFront »</p>
<p>CENTRAL DE CONTATOS</p> <p>Amazon Connect</p> <p>90 minutos</p> <p>por mês de uso do Amazon Connect</p> <p>O Amazon Connect é uma central de contatos fácil de usar e baseada na nuvem com escalabilidade para apoiar empresas de todos os portes</p> <p>Saiba mais sobre o Amazon Connect »</p>	<p>COMPUTAÇÃO</p> <p>Amazon EC2</p> <p>750 horas</p> <p>por mês</p> <p>Capacidade computacional redimensionável na nuvem</p> <p>Saiba mais sobre o Amazon EC2 »</p>	<p>COMPUTAÇÃO</p> <p>Amazon EC2 Container Registry</p> <p>500 MB</p> <p>por mês de armazenamento</p> <p>Armazene e recupere imagens do Docker</p> <p>Saiba mais sobre o Amazon EC2 Container Registry »</p>
<p>ARMAZENAMENTO E ENTREGA DE CONTEÚDO</p> <p>Amazon EFS</p> <p>5 GB</p> <p>de armazenamento</p> <p>Serviço de armazenamento de arquivos simples, escalável e compartilhado para instâncias do Amazon EC2</p> <p>Saiba mais sobre o Amazon EFS »</p>	<p>ARMAZENAMENTO E ENTREGA DE CONTEÚDO</p> <p>Amazon Elastic Block Storage</p> <p>30 GB</p> <p>qualquer combinação de propósito geral (SSD) ou magnético</p> <p>Volumes de armazenamento de bloco persistentes, resilientes e com baixa latência para instâncias do EC2</p> <p>Saiba mais sobre o Amazon Elastic Block Storage »</p>	<p>SERVIÇOS DE APLICAÇÕES</p> <p>Amazon Elastic Transcoder</p> <p>20 minutos</p> <p>de transcodificação de áudio</p> <p>Serviço de transcodificação de mídia totalmente gerenciado</p> <p>Saiba mais sobre o Amazon Elastic Transcoder »</p>
<p>BANCO DE DADOS</p> <p>Amazon ElastiCache</p> <p>750 horas</p> <p>de uso do nó cache.t2micro</p> <p>Serviço da web que facilita a implantação, a operação e a escalabilidade de um cache de memória na nuvem</p> <p>Saiba mais sobre o Amazon ElastiCache »</p>	<p>ANÁLISE</p> <p>Amazon Elasticsearch Service</p> <p>750 horas</p> <p>por mês de uma instância t2.micro.elasticsearch ou t2.small.elasticsearch Single-AZ</p> <p>Serviço gerenciado que facilita a implantação, a operação e a escalabilidade de clusters do Elasticsearch na Nuvem AWS</p> <p>Saiba mais sobre o Amazon Elasticsearch Service »</p>	<p>SERVIDORES DE VIDEOGAMES</p> <p>Amazon GameLift</p> <p>125 horas</p> <p>por mês de uso da instância c3.large.gamelift do Amazon GameLift, mais 50 GB de armazenamento de propósito geral (SSD) do EBS</p> <p>Hospedagem simples, rápida e econômica de servidores dedicados de jogo.</p> <p>Saiba mais sobre o Amazon GameLift »</p>
<p>INTELIGÊNCIA ARTIFICIAL</p> <p>Amazon Lex</p> <p>10.000</p> <p>solicitações de texto por mês</p> <p>Crie chatbots de voz e texto de bate-papo</p> <p>Saiba mais sobre o Amazon Lex »</p>	<p>SERVIÇOS MÓVEIS</p> <p>Amazon Pinpoint</p> <p>5.000</p> <p>usuários direcionados gratuitos por mês</p> <p>Notificações por push direcionadas a aplicativos móveis</p> <p>Saiba mais sobre o Amazon Pinpoint »</p>	<p>INTELIGÊNCIA ARTIFICIAL</p> <p>Amazon Polly</p> <p>5 milhões</p> <p>de caracteres por mês</p> <p>Transforme texto em falas realistas</p> <p>Saiba mais sobre o Amazon Polly »</p>

Detalhes do nível gratuito da AWS

★ EM DESTAQUE

🕒 12 MESES GRATUITOS

🔄 SEMPRE GRATUITO

📂 CATEGORIAS DE PRODUTO

▼ TODAS

BANCO DE DADOS

Amazon RDS

750 horas

por mês de uso do banco de dados db.t2.micro (mecanismos de banco de dados aplicáveis)

Serviço de banco de dados relacional gerenciado para MySQL, PostgreSQL, MariaDB, Oracle BYOL ou SQL Server

[Saiba mais sobre o Amazon RDS »](#)

INTELIGÊNCIA ARTIFICIAL

Amazon Rekognition

5.000

imagens por mês

Serviço de reconhecimento de imagens com base em aprendizado profundo

[Saiba mais sobre o Amazon Rekognition »](#)

ARMAZENAMENTO E ENTREGA DE CONTEÚDO

Amazon S3

5 GB

de armazenamento padrão

Infraestrutura de armazenamento de objetos segura, durável e escalável

[Saiba mais sobre o Amazon S3 »](#)

ANÁLISE

AWS Data Pipeline

3

pré-condições de baixa frequência

Orquestração para fluxos de trabalho controlados por dados

[Saiba mais sobre o AWS Data Pipeline »](#)

INTERNET DAS COISAS

AWS Greengrass

3 dispositivos

gratuitos

Recursos locais de computação, sistema de mensagens, armazenamento de dados em cache e sincronização para dispositivos conectados

[Saiba mais sobre o AWS Greengrass »](#)

INTERNET DAS COISAS

AWS IoT

250 mil mensagens

(publicadas ou distribuídas) por mês

Conecte dispositivos à nuvem

[Saiba mais sobre o Amazon IoT »](#)

FERRAMENTAS DE GERENCIAMENTO

AWS OpsWorks for Chef Automate

10

nós por mês

Gerenciamento de configurações com o Chef Server e o Chef Automate

[Saiba mais sobre o AWS OpsWorks for Chef Automate »](#)

FERRAMENTAS DO DESENVOLVEDOR

AWS Trusted Advisor

4

verificações de desempenho e segurança

Especialista em otimização da Nuvem AWS

[Saiba mais sobre o AWS Trusted Advisor »](#)

Doze meses gratuitos e produtos sempre gratuitos

O nível gratuito da AWS inclui ofertas que expiram 12 meses após a inscrição e outras ofertas que nunca expiram.

[Saiba mais »](#)

COMPUTAÇÃO

Elastic Load Balancing

750 horas

por mês compartilhados entre load balancers Classic e Application

Distribuição automática de tráfego de entrada de aplicações em várias instâncias do Amazon EC2

[Saiba mais sobre o Elastic Load Balancing »](#)

ANEXO C – DETALHES DA CONTA GRATUITA E PRECIFICAÇÃO DA MICROSOFT AZURE E HUBE IOT

Microsoft Azure

Crie sua conta gratuita do Azure hoje mesmo


Comece com um crédito de R\$670
Comece gratuitamente com R\$670 em créditos para usar em qualquer produto Azure por 30 dias.

Continue usando os produtos gratuitos
Crie sua próxima grande ideia com acesso gratuito aos nossos produtos mais populares por 12 meses e a mais de 25 produtos sempre gratuitos.

Não pague nada até escolher
Usamos os dados do seu cartão de crédito para verificação de identidade, mas você só receberá cobrança se optar por fazer a atualização.

– Hub IoT		Conecte, monitore e controle os ativos de IoT em execução em um amplo conjunto de sistemas operacionais e protocolos para iniciar rapidamente seu projeto de Internet das Coisas.	
TIPO DE EDIÇÃO	PREÇO POR UNIDADE (POR MÊS)	NÚMERO TOTAL DE MENSAGENS/DIA POR UNIDADE	TAMANHO DO MEDIDOR DE MENSAGEM
Gratuito	Gratuito	8.000	0,5 KB
S1	R\$332	400.000	4 KB
S2	R\$3.320	6.000.000	4 KB
S3	R\$33.200	300.000.000	4 KB

ANEXO D – DETALHES DO TESTE DA IBM BLUEMIX



Inscreva-se para obter um IBMid e criar sua conta do Bluemix

Experimente o Bluemix grátis por 30 dias

Inicie a construção imediatamente.

Sua avaliação não requer cartão de crédito. Você só precisa inscrever-se e iniciar a construção.

App de produção? Sem problema.

Nós fornecemos a você 2 GB de memória de tempo de execução e de contêiner grátis por 30 dias, além de acesso para provisão de até 10 serviços.

Estamos aqui para ajudar.

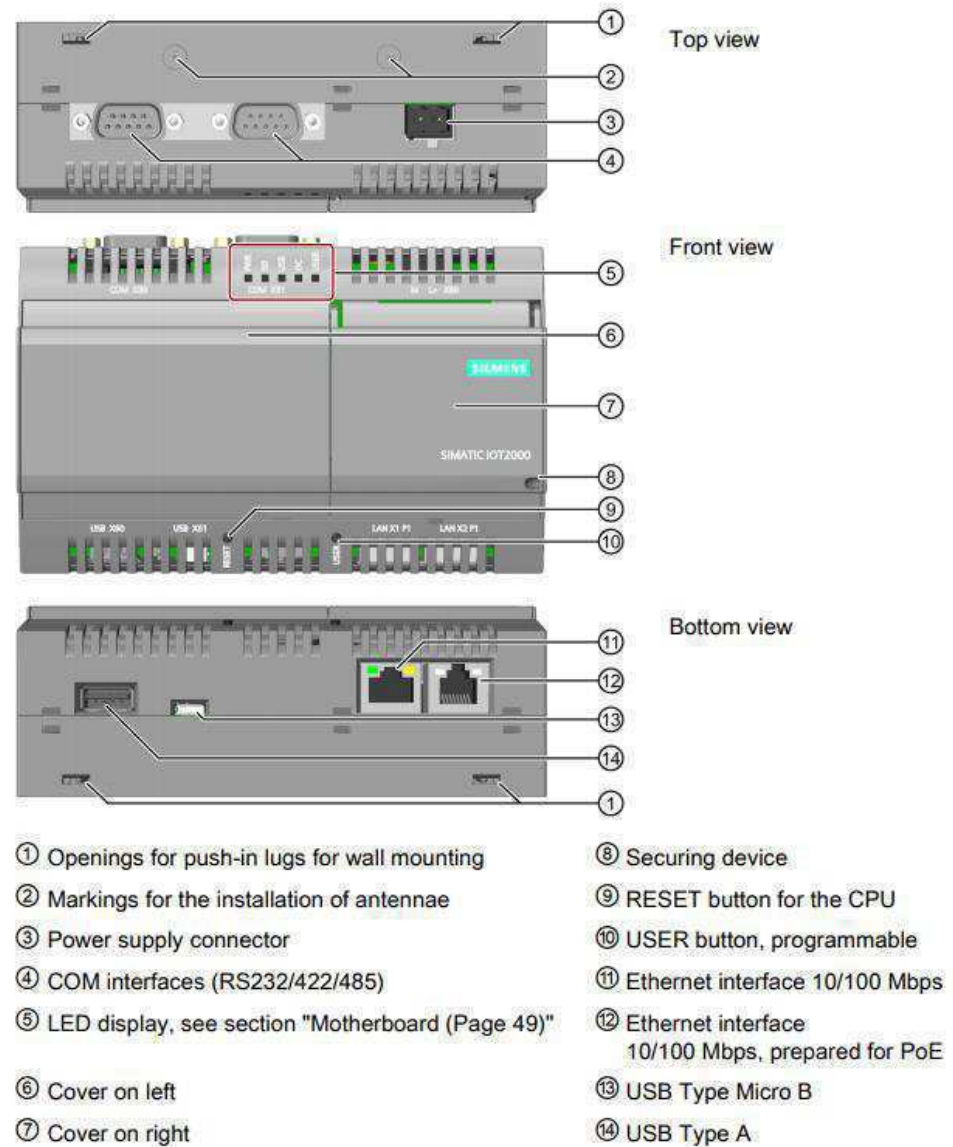
Sua avaliação é fornecida com suporte de help desk grátis. Pergunte o que quiser ao longo do processo.

ANEXO E – DETALHES DO DATASHEET DO SIMATIC IOT2040

Imagem obtida do *datasheet* fornecido pela Siemens em seu website [27].

1.2.2 SIMATIC IOT2040

The following figures show the configuration and interfaces of the SIMATIC IOT2040.



5.1 Insert Micro SD card

Requirement

- The device is disconnected from the power supply.
- Micro SD card that is suitable for industrial use.

Procedure

Installation

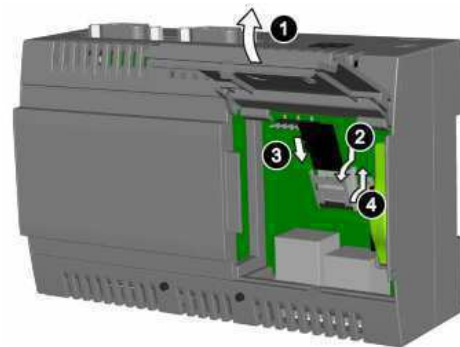
NOTICE

Inserting a memory card

If you are using the Micro SD card in a device installed in a system, you must observe the safety regulations for work on electrical systems.

Carefully insert the Micro SD card into the Micro SD holder without applying excess force.

1. Open the cover on the right.
2. Carefully press the Micro SD holder down and lift the holder forwards.
3. Push the Micro SD card correctly aligned into the supporting frame. The contacts of the Micro SD card must point in the direction of the motherboard.
4. Push the supporting frame back and carefully push the supporting frame upwards until it engages.



5.2 Install Arduino shield

Requirement

- The device is disconnected from the power supply.
- An Arduino shield

Procedure

NOTICE

Install Arduino shield

Do not under any circumstances insert the Arduino shield incorrectly. Ensure that the contact pins of the Arduino shield connect correctly with the terminal strips of the motherboard.

Arduino shield with operator control or display elements

Some Arduino shields have operator control and display elements. The device loses its approval certificates if you drill or mill openings in the cover in order to make the operator control or display elements of the Arduino shield accessible or visible from outside. In this case, the customer is responsible for the re-approval of the device.

Note

Power consumption

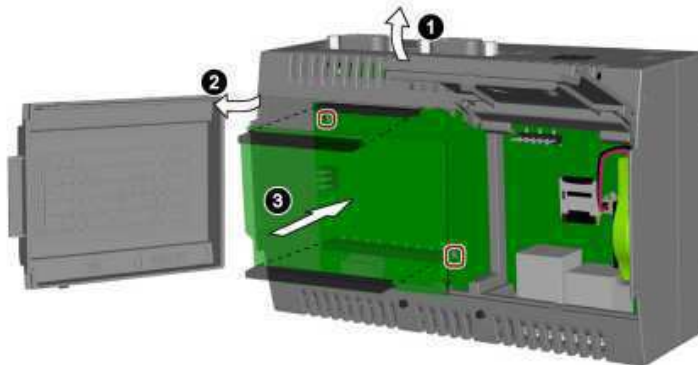
If the power consumption of the Arduino shield is too high, the device will be damaged.

Note the information in section "Technical data (Page 45)".

Ambient temperature

The temperature in the housing of the device can be up to 30 °C above the maximum permissible ambient temperature of the device. Make sure that the maximum permissible ambient temperature of the Arduino shield is specified accordingly.

1. Open cover on the right and lift up the cover.



2. Open the cover on the left by slightly raising the cover and lifting it to the left.
3. Insert the Arduino shield into the motherboard. Ensure that the contact pins of the Arduino shield fit perfectly on the contact strips of the motherboard and that the components of the Arduino shield do not touch the components of the motherboard.

5.3

Install Mini PCIe card

You can install a Mini PCIe card in a device of the type IOT2000.

Procedure

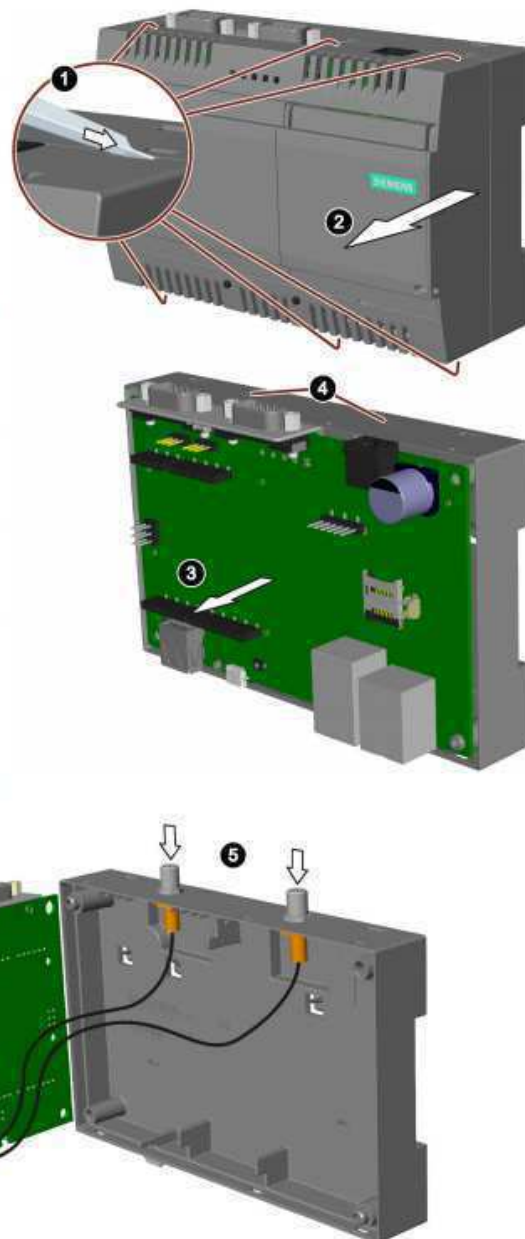
The following example describes the installation of a Mini PCIe WLAN card, including mounting of the antenna jacks. If you install a different Mini PCIe card, the work steps 4, 5 and 7 are not required.

When the enclosure is secured with two screws at the rear panel, remove the two screws.

Remove the battery, see section "Replace the backup battery (Page 36)".

Then follow these steps:

1. Loosen the front panel of the housing from the rear panel of the housing. Carefully press with the blade of a flat-blade screwdriver in the marked recesses and carefully pull on the appropriate place on the front panel of the enclosure.
2. Remove the front panel of the enclosure.
3. Remove the motherboard.
4. Drill the bushings for the antenna sockets with a corresponding diameter at the markings shown.
5. Install the antenna sockets in the enclosure.
6. Insert the Mini PCIe card in the Mini PCIe interface on the motherboard from below as illustrated.
7. Connect the antenna cables to the Mini PCIe card.



Then install the motherboard again and close the housing.