



REVISTA BRASILEIRA DE MECATRÔNICA
FACULDADE SENAI DE TECNOLOGIA MECATRÔNICA

FERRAMENTA DE CONTROLE DE PRODUÇÃO E DEMANDA COM COMPUTAÇÃO EM NUVEM

PRODUCTION AND DEMAND CONTROL TOOL WITH CLOUD COMPUTING

Daniel Carvalho dos Santos^{1, i}

Crishna Irion^{2, ii}

Thiago Tadeu Amici^{3, iii}

Jorge Antonio Giles Ferrer^{4, iv}

Data de submissão: (28/02/2023) Data de aprovação: (10/04/2023)

RESUMO

A quarta revolução Industrial, denominada de Indústria 4.0, mostra que a aplicação das suas tecnologias habilitadoras na indústria traz resultados importantes em termos de produtividade, eficiência, controles de estoque e gestão estratégica. Um dos pontos principais da gestão estratégica é checar o equilíbrio entre produção inteligente e demanda. Este importante departamento é conhecido, em algumas indústrias, como Produção *versus* Demanda (PvD) e tem este nome justamente pelo natural conflito entre as duas disciplinas. O objetivo deste artigo é apresentar uma ferramenta de controle, que utiliza as tecnologias de Computação em Nuvem e Integração de Sistemas para minimizar este conflito e facilitar tomadas de decisões estratégicas de gestão da produção.

Palavras-chave: Computação em Nuvem; Integração de Sistemas; Indústria 4.0; Produção *versus* Demanda; Produção inteligente.

ABSTRACT

The fourth Industrial Revolution, called Industry 4.0, shows that the application of its enabling technologies in industry brings important results in terms of productivity, efficiency, inventory controls and strategic management. One of the main points of strategic management is to check the balance between intelligent production and demand. This important department is known in some industries as Production *versus* Demand (PvD), and it has this name precisely because of the natural conflict between the two disciplines. The objective of this article is to

¹ Engenheiro Mecatrônico e Pós-graduando em Indústria 4.0 EAD na Faculdade SENAI de Tecnologia Mecatrônica. E-mail: dansoucar@gmail.com

² Mestre em Ciência da Linguagem pela UNIVAS. E-mail: crishna.irion@senai.br

³ Mestre em Automação e Controle de Processos da Faculdade SENAI de Tecnologia Mecatrônica. E-mail: thiago.amici@sp.senai.br

⁴ Doutor em Engenharia Mecânica pela Universidade Estadual de Campinas. E-mail: jorge.ferrer@sp.senai.br

present a control tool, which uses Cloud Computing and Systems Integration technologies to minimize this conflict and facilitate strategic decision-making in production management.

Keywords: Cloud Computing; Systems Integration; Industry 4.0; Production versus Demand; Smart Production.

1 INTRODUÇÃO

No mundo globalizado e cada dia mais competitivo, os desperdícios de tempo, matéria-prima ou mesmo a ocupação do time operacional em tarefas não prioritárias ou não tão lucrativas, pode ser fatal para definir o futuro de qualquer tipo de manufatura. O equilíbrio entre qual produto precisa realmente ser fabricado, ou o que deve ser priorizado, cria um conflito natural entre produção e demanda, responsabilidade do departamento de PvD. Diversos setores da manufatura têm demonstrado preocupação com este equilíbrio, sobretudo, o mercado de alimentos e perecíveis. Produzir acima da demanda e ainda prover desperdício gera prejuízo em dobro.

Outro desafio das manufaturas modernas é a integração de setores e departamentos sem renunciar aos resultados individuais. Nesta linha, mesmo atualmente, não é difícil encontrar manufaturas ainda trocando relatórios de demanda e produção feitos manualmente, reuniões de apontamento de produção e alinhamento da demanda ocorrendo uma vez por semana, ou seja, desta forma, os setores produtivos podem não ter acesso às necessidades da demanda em tempo real.

O objetivo deste artigo é propor uma ferramenta que auxilia nas tomadas de decisões estratégicas da manufatura, apresentar em tempo real as demandas ao departamento de produção e disponibilizar ao cliente final um portal em página *web* para cadastro dos pedidos. A ferramenta proposta conta com aplicação das tecnologias habilitadoras da Indústria 4.0 de Computação em Nuvem e Integração de Sistemas.

2 REVISÃO DE LITERATURA

2.1 Gestão da Demanda

A gestão da demanda tem papel chave na manufatura moderna, e este tema é sempre abordado em literaturas que tratam de eficiência produtiva. De acordo com Favaretto (2001, p.58) “A gestão da carteira de pedidos e da previsão de vendas, em conjunto, é chamada de gestão da demanda. Estas atividades são ligação da empresa com o mercado consumidor”. O termo demanda, de acordo com a plataforma Significados (2022), é definido como: “Demanda significa a quantidade de um bem ou serviço que os consumidores desejam adquirir por um preço definido em um mercado “. Juntando gestão e demanda, a interpretação simplificada, fica compreendida como a ciência que sistematiza as práticas de administração da quantidade de produtos ou serviços que o mercado necessita.

De acordo com Arnold, Chapman e Clive (2008), a responsabilidade desta administração vai um pouco mais além e mensura as capacidades e limitações produtivas:

O marketing se concentra em atender às necessidades do cliente, mas as operações, por meio do gerenciamento de materiais, devem fornecer os recursos. A coordenação dos planos por essas duas partes é o gerenciamento da demanda. (Arnold, Chapman e Clive, 2008, p.217, tradução nossa).

Alguns autores, dessa linha de literatura, destacam que existem macroatividades dentro desta disciplina que são fundamentais para um trabalho eficiente na manufatura, como é o caso de Pround (2007, p.484, tradução nossa) que cita: “A gestão da demanda tem quatro requisitos fundamentais: 1. Previsão [...] 2. Comunicações [...] 3. Influência [...] 4. Priorização e Alocação”. A ferramenta de controle proposta neste artigo, ajuda as manufaturas no desenvolvimento de algumas macroatividades, como a ‘Previsão da Demanda’ que utiliza dados históricos e análises dos cenários de consumo para fazer previsões de vendas e suas quantidades.

Neste caso, a ferramenta de controle já conta com um banco de dados contendo o histórico de demanda e produção armazenados na nuvem, podendo ser acessados para análise em tempo real. O mesmo ocorre na macroatividade de ‘Priorizar e Alocar’ que consiste em corrigir o processo produtivo quantas vezes forem necessárias para melhor servir a demanda. É nesta macroatividade que a ferramenta proposta tem o seu principal objetivo, que é proporcionar a visão das alterações de demanda em tempo real e, conseqüentemente, propiciar tomadas de decisões precisas para reprogramar o processo produtivo também em tempo real. Outra atividade que é beneficiada pela ferramenta é a de ‘Controlar os Indicadores de Performance do Processo’, que podem ser calculados e indicados em tempo real, dentre eles, o *Overall Equipment Effectiveness* (OEE - Eficiência Global do Equipamento).

A importância em ter os dados de produção, eficiência e perdas em tempo real foi identificada em um estudo de caso no qual foram analisadas duas empresas de laticínios com falhas na Gestão da Demanda. Nesse estudo, foi identificado por Melo e Alcântara (2012, p. 15): “as previsões de vendas não são compartilhadas com os produtores rurais ou com as lojas varejistas. Desta forma, percebe-se por parte da organização um alinhamento interno, mas não há integração entre empresas”. Ainda neste estudo, as duas empresas observadas utilizam planilhas de previsão de vendas que são replicadas ao time estratégico em períodos pré-determinados, uma delas com intervalos mensais, ou seja, informações não atuais. As duas empresas observadas também possuíam problemas em comum. De acordo com Melo e Alcântara (2012, p. 15): “Nas duas empresas as metas de vendas são acompanhadas, porém não há um controle das distorções entre planejado *versus* realizado, além de não haver análise de erros de previsão”.

A eficiência na Gestão da Demanda na manufatura foi destacada por Corrêa e Gianesi (1992, p. 20): “A tecnologia de manufatura - sua escolha e gestão - tem sido considerada uma das principais áreas de decisão dentro da função de administração de produção”. De forma geral, o conflito gerado entre ‘Produção *versus* Demanda’ é bem natural que ocorra em muitas empresas. O que não deveria ocorrer é que estes conflitos se tornem grandes problemas, como a subprodução com alta demanda ou superprodução sem demanda, ou pior, os dois cenários simultâneos dentro da mesma organização sob pontos de vista de produtos ou setores diferentes.

Auxiliar o departamento de PvD, prevenir desperdícios e aumentar a eficiência produtiva é a motivação pelo desenvolvimento de uma ferramenta de controle da produção e demanda. Para que esta ferramenta seja eficiente, algumas características são desejáveis, como o armazenamento em banco de dados, atualização das informações em tempo real, mobilidade para ser acessível em qualquer lugar e possuir interface amigável para a utilização.

O armazenamento em banco de dados facilita a organização, pois o controle sai das planilhas individuais e passam para um único centro de informações. A atualização das informações em tempo real é possível mediante consultas e gravações no banco de dados

através das interfaces de usuário. Para que a ferramenta seja acessível em qualquer lugar, precisa estar disponível na Internet. Até é possível utilizar um computador físico com acesso à Internet para reunir todas estas características, mas eleva bastante a complexibilidade da implantação, pois é necessário garantir fontes seguras de energia elétrica para este computador, e links redundantes de Internet, além de mais equipes para executar manutenções e rotinas de revisão.

Para atender todas as características desejáveis para o bom desempenho da ferramenta e diminuir as complexibilidades de implantação, infraestrutura e manutenção, a utilização dos recursos da Computação em Nuvem se encaixa neste contexto, pois o computador passa a ser virtual, o banco de dados pode ser contratado e expandido conforme a necessidade, a infraestrutura, a internet e a manutenção são responsabilidade da provedora da nuvem. Logo, praticamente toda a infraestrutura é obtida com a facilidade de contratação de um serviço. O ponto que deve ser observado é que existem muitos serviços disponíveis e é necessário escolher quais se encaixam nas necessidades da ferramenta.

2.2 Computação em Nuvem

O termo Computação em Nuvem é descrito por Buyya, Broberg e Goscinski (2011, p. 03, tradução nossa) como: “A computação em nuvem foi cunhada como um termo guarda-chuva para descrever uma categoria de serviços sofisticados de computação sob demanda inicialmente oferecidos por provedores comerciais, como Amazon, Google e Microsoft”. De forma simplificada, Computação em Nuvem pode ser entendida como o fornecimento de serviços virtualizados de computação e acessíveis via Internet. Neste modelo de computação, o cliente final não necessita prover a infraestrutura com altos investimentos para desenvolver os projetos. Em contrapartida, no geral, os serviços são fornecidos mediante a manutenção de contratos de prestação de serviços.

A possibilidade de utilizar recursos sofisticados de qualquer lugar do mundo e sem necessidade de implantação de hardwares próprios, faz com que a infraestrutura do provedor seja chamada de “nuvem”. Segundo Buyya, Broberg e Goscinski (2011, p. 03, tradução nossa): “O principal princípio por trás desse modelo é fornecer computação, armazenamento e software “como um serviço””. Por mais que o termo Computação em Nuvem dá um enfoque futurista ao tema, os serviços são bem ‘tradicionais’ no mundo da computação. Segundo Kim (2009, p. 66, tradução nossa): “As ideias por trás dessas formas de serviço não são novas. O fato de os usuários poderem acessar esses serviços a partir de navegadores da Web via Internet os torna serviços de 'nuvem””.

A ferramenta de controle da produção e demanda necessita de dois serviços de nuvem para atender as necessidades propostas. O primeiro serviço é um computador virtual também conhecido como *Virtual Machine* (VM), ou ainda Instância. Nesta instância fica alocado um software capaz de interagir com o banco de dados e que possua propriedade de interface para os usuários através da página *web*. O segundo serviço é um banco de dados para armazenar as informações dos pedidos e o controle da produção e entregas.

2.2.1 Serviços da Nuvem

Em muitas literaturas sobre o tema, os serviços podem ser divididos em 3 grandes blocos, sendo eles: *Infrastructure as a Service* (IaaS), *Platform as a Service* (PaaS) e *Software as a Service* (SaaS). De acordo com Veras (2012, p. 143): “*Infrastructure as a Service* (IaaS) é a

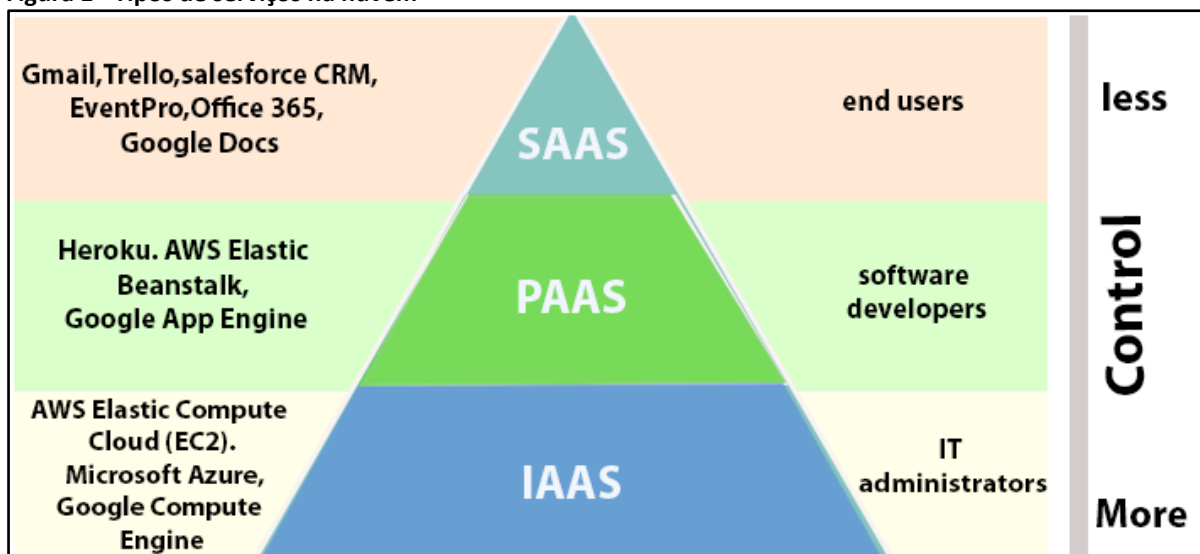
capacidade que o provedor tem de oferecer uma infraestrutura de processamento e armazenamento de forma transparente para o cliente”. Neste modelo de serviço, a instância (ou *VM*) fica totalmente no controle do contratante, podendo ser utilizada para a instalação de softwares, armazenar informações etc.

A Plataforma como Serviço (PaaS) da nuvem, segundo Veras (2012, p. 169) “O conceito de PaaS, menos conhecido, está vinculado ao uso de ferramentas de desenvolvimento de software oferecidas por provedores de serviços”. Neste serviço, os clientes desenvolvedores não necessitam instalar o *software* em seus computadores. O desenvolvimento ocorre diretamente na plataforma do provedor. Este modelo de contratação do serviço substitui os tradicionais custos de licenças e os provedores são monetizados através do acesso/utilização.

O *Software* como Serviço (SaaS) da nuvem é exemplificado por Veras (2012, p. 192) como: “Os aplicativos são oferecidos como serviços por provedores e acessados pelos clientes por aplicações como o browser”. Neste modelo de serviço, o usuário utiliza o aplicativo ou função que também é utilizado por vários outros usuários em uma mesma plataforma mantida pelo provedor.

A Figura 1 mostra uma síntese dos tipos de serviços de nuvem. O lado esquerdo da pirâmide mostra alguns exemplos de serviços em cada camada, e o lado direito, os típicos usuários de cada serviço. Nesta mesma figura, do lado direito da pirâmide é mostrado o grau de controle que os contratantes da nuvem possuem, da camada com menos acesso (*less*), até à camada com mais acesso (*more*) ao gerenciamento.

Figura 1 - Tipos de serviços na nuvem



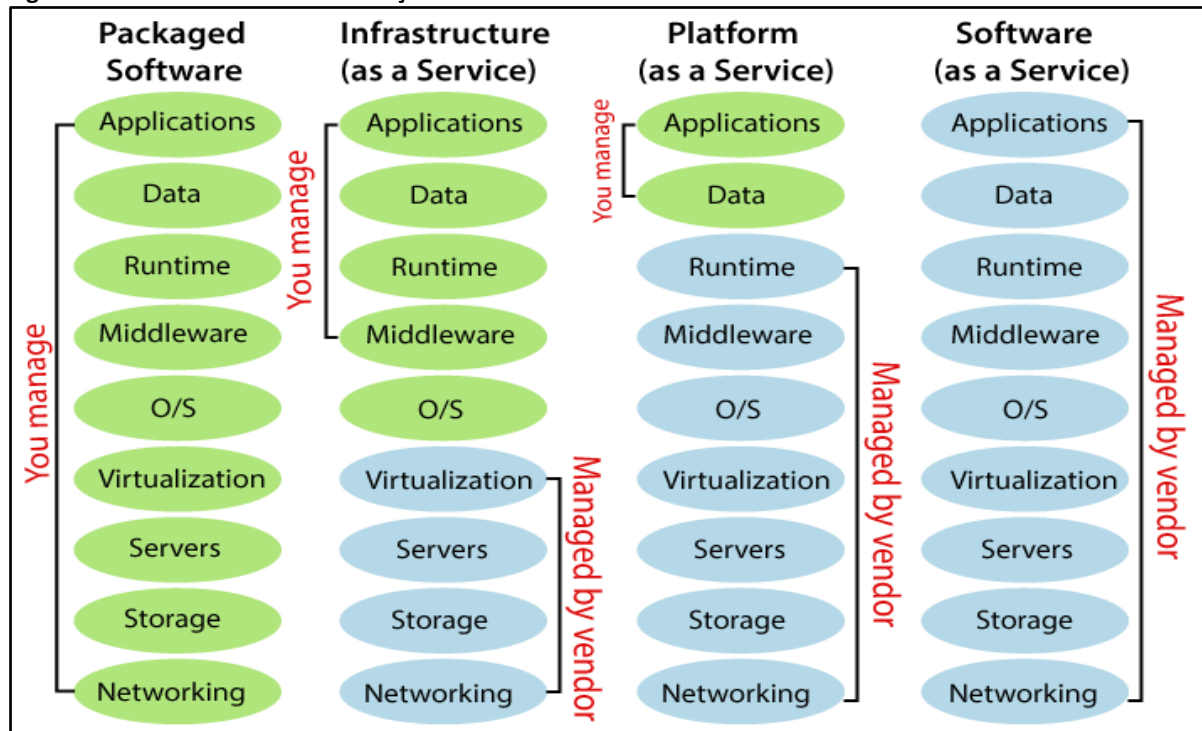
Fonte: javaTpoit (2022a).

A Figura 2 ilustra o sistema 100% gerenciado e mantido (inclusive a infraestrutura) pela organização (*Packaged Software*) e sistemas que utilizam serviços em nuvem. Ainda analisando a figura, o nível de gerenciamento pelo usuário (*you manage*), fica bem definido em cada tipo de serviço. É importante conhecer o nível de gerenciamento para a escolha mais adequada de serviço. A instância, que é um IaaS, necessita de instalação de software e configurações, já o serviço de banco de dados, por ser SaaS, já está pronto, sendo necessário apenas configurar as tabelas.

O provedor escolhido para o desenvolvimento da ferramenta é o *Amazon Web Services* (AWS) por possuir um sistema robusto e expansível. A instância na nuvem da AWS é chamada

de *Amazon Elastic Computing Cloud* (EC2) e o serviço de banco de dados é chamado de *Relational Database Service* da Amazon (RDS). Estes dois serviços formam a infraestrutura para o funcionamento da ferramenta.

Figura 2 – Gerenciamento dos serviços da nuvem



Fonte: javaTpoit (2022b).

2.2.2 Amazon EC2

A instância EC2, segundo a Amazon (2022a) é: “O *Amazon Elastic Compute Cloud* (Amazon EC2) é um serviço web que disponibiliza capacidade computacional segura e redimensionável na nuvem”. O EC2 se encaixa no grupo de serviços do tipo IaaS e foi inclusive utilizado como exemplo por Veras (2012, p. 143): “Um exemplo de IaaS é o Amazon EC2”.

Para a utilização do serviço EC2 é necessário configurar as características e recursos de uma máquina virtual completa:

- Tipo de drive de armazenamento: *Solid State Drive* ou *Hard Disc Drive* (HDD);
- Tamanho do armazenamento em *Gigabytes* (GB);
- Quantidade de memória *Random Access Memory* (RAM);
- Quantidade de núcleos e processadores virtuais;
- Sistema operacional;

O serviço EC2 possibilita a criação, com poucos cliques, de uma máquina virtual já pronta para a instalação dos serviços e aplicações necessárias para o funcionamento dos projetos. Para a proteção cibernética, o EC2 conta com a possibilidade de criação de políticas para acesso. Por conta de todas estas versatilidades, o EC2 foi escolhido para ser alocada a principal funcionalidade da ferramenta aqui proposta.

O objetivo da utilização do serviço EC2 na ferramenta de controle da produção, é hospedar um software que tenha os recursos necessários para interagir com o banco de dados e possuir interface com os usuários em navegador web. Existem muitos softwares que

possuem os recursos desejados, e neste caso, o Node-RED da IBM foi escolhido por não possuir custos de licenciamento e por possuir linguagem mais simples para as integrações com o banco de dados e com as interfaces de usuário.

2.2.3 Amazon RDS

O serviço de banco de dados da AWS, segundo a Amazon (2022b) é: “O *Amazon Relational Database Service* (Amazon RDS) é uma coleção de serviços gerenciados que facilita a configuração, operação e escalabilidade de bancos de dados na nuvem”. O RDS é classificado como um SaaS, e possibilita a criação de instâncias de bancos de dados SQL. Neste formato de serviço, a cobrança ocorre de acordo com o volume de dados armazenados. No banco de dados relacional somente é possível a gravação de dados, ou seja, números, letras, códigos, nomes etc. O armazenamento de documentos, imagens, gravações, ou outros formatos de arquivos não são possíveis neste tipo de banco de dados.

O serviço Amazon RDS permite a utilização de algumas plataformas de banco de dados. Para esta aplicação, o MySQL (banco de dados relacional da Oracle) foi escolhido por ter configuração e integração sem grandes complexibilidades, sem renunciar a boa capacidade de armazenamento. A vantagem da utilização de um banco de dados na nuvem, é que as informações atuais estão disponíveis para serem acessadas em tempo real e os dados históricos estão disponíveis para serem analisados, apontando padrões e tendências de demanda, atendendo desta forma, a parte da gestão da manufatura.

Por outro lado, a interface com o cliente é um portal de *electronic commerce* (*e-commerce*) onde os pedidos são armazenados no banco de dados RDS exposto. Neste ponto, ocorre a integração entre cliente e manufatura.

2.3 Node-RED

A IBM desenvolveu o Node-RED para ser uma ferramenta de programação visual, de código aberto, baseado em linguagem Java. A descrição que consta na página oficial (<https://nodered.org/>) da ferramenta, já explica muito o objetivo deste software inovador:

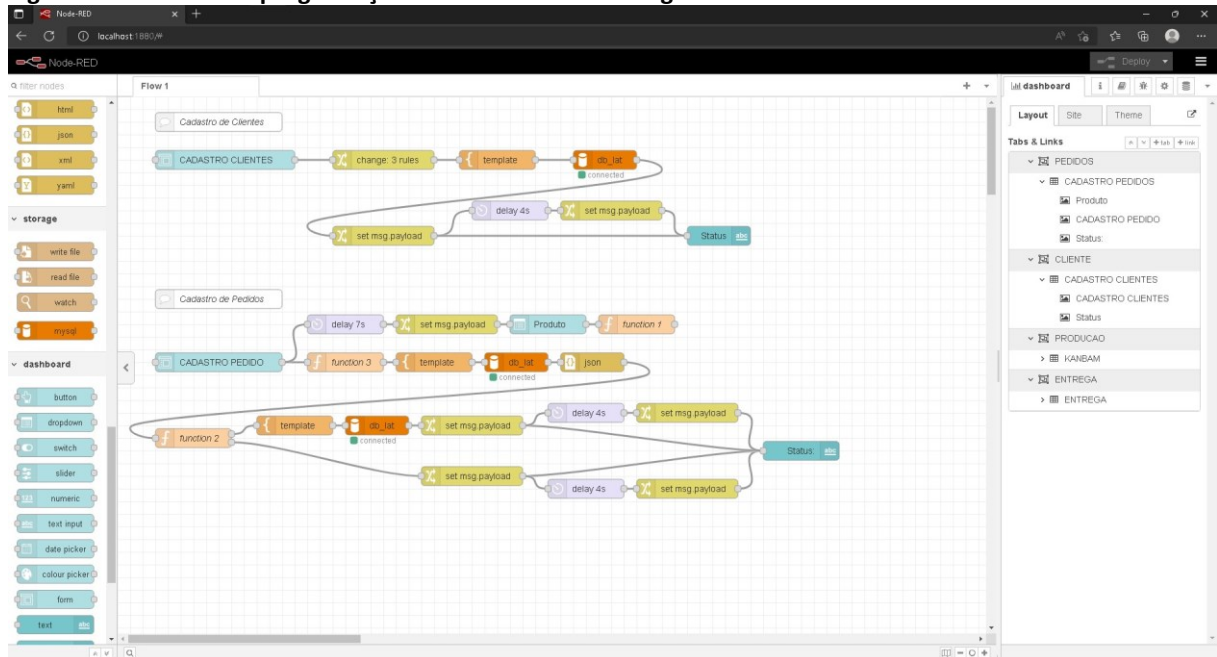
Node-RED é uma ferramenta de programação para conectar dispositivos de hardware, APIs e serviços online de maneiras novas e interessantes. Ele fornece um editor baseado em navegador que facilita a conexão de fluxos usando a ampla variedade de nós na paleta que podem ser implantados em seu tempo de execução com um único clique. (NODERED, 2022).

Uma característica que torna o Node-RED muito interessante é o fato de ser uma ferramenta de código aberto e colaborativo, ou seja, *Application Programming Interfaces* (APIs) desenvolvidas por diversos fabricantes e desenvolvedores para poderem ser integrados, compartilhados e melhorados. Outra característica interessante é o inovador visual de programação em navegador com blocos, ou nós, que possibilita a interligação entre diferentes funções e fluxos. Por fim, o Node-RED possui a funcionalidade de criação de *dashboards*, que são interface gráfica de usuário.

A Figura 3 mostra a estrutura de programação do Node-RED. A biblioteca de blocos fica do lado esquerdo da página e as informações e instruções do bloco selecionado, bem como a depuração do código, ficam do lado direito. A estrutura de blocos de programação do

Node-RED, pode ter *msg.topic* (ponto de ligação do lado esquerdo do bloco), *msg.payload* (ponto de ligação do lado direito do bloco), ou ambos de acordo com o tipo de função. No Node-RED, o *msg.topic* contém uma função ou o resultado de um código em Java. No geral, os blocos executam sua função com o código que chega no *msg.topic* e o resultado é enviado no *msg.payload*.

Figura 3 - Estrutura de programação do Node-RED em navegador



Fonte: Elaborada pelo autor

Na figura acima também é possível observar diferentes tipos de blocos dos quais, listados abaixo, estão os principais:

- Bloco de eventos cíclicos (*inject*): pode carregar funções para que sejam executadas ciclicamente a cada tempo pré-configurado;
- Bloco de eventos de mudança (*change*): realiza alterações/funções nos códigos 'msg.' como o *msg.topic*, *msg.payload* ou outros tipos de variáveis;
- Bloco com eventos de chaveamento (*switch*): realiza escolha de rota, neste caso, a rota vai ser escolhida de acordo com o resultado do algoritmo obtido no *topic*. Neste tipo de bloco, os pontos de ligação de *payload* possui mais de uma opção (caminho);
- Bloco de retardo (*delay*): realiza retardos na execução da função;
- Bloco de função (*function*): realiza funções completas como montar uma estrutura de dados para interação com banco de dados;
- Bloco de depuração (*debug*): possui somente o ponto de ligação *topic* e possibilita a visualização dos códigos que estão neste ponto da programação;

Além das funções básicas do Node-RED, para o desenvolvimento da ferramenta de controle da produção e demanda, é necessário instalar a biblioteca de *dashboard* e a API do MySQL (banco de dados relacional da Oracle). Ao instalar as APIs no Node-RED, novos blocos destas funções são agregados à biblioteca na lateral direita. A partir deste ponto, a ferramenta de controle ganha forma, pois já conta com software para a integração com o banco de dados e a função *dashboard* no formato de portal *web* para cadastro de clientes, pedidos e apresentação em tempo real da demanda para o departamento de PvD.

2.4 Portal *e-commerce*

A melhor definição para portal *e-commerce* é uma porta para o comércio eletrônico. Essa definição tem como alicerce muitas literaturas sobre o assunto. De acordo com o SEBRAE (2017): “De maneira bem simples, o termo *e-commerce* é a abreviação em inglês de *electronic commerce*, cuja tradução em bom português significa comércio eletrônico”. Ao associar o termo Portal à abreviação *e-commerce*, é transmitido o conceito de porta, ou seja, é o meio de acesso eletrônico do cliente ao produto/serviço.

A solução do portal *e-commerce* atende duas demandas. Primeiro a demanda do cliente, que cadastra o pedido a qualquer momento, com a vantagem de o portal, sendo hospedado na nuvem que é a proposta deste estudo, operar ‘24 X 7’ todos os dias do ano. Segundo, atende a demanda de suprir com dados o departamento de PvD, para que as ações estratégicas sejam tomadas. Neste momento, o intuito deste estudo é informar as demandas, sem integração com a automação industrial que comanda a planta, já que esta última parte do trabalho pode também ser incorporada no futuro.

O portal *e-commerce* inteligente, não só identifica o cliente, como também identifica as quantidades e prazos. O painel de visualização da demanda, que será utilizado pelo time operacional e pelo setor de PvD, também deve ser inteligente para que as prioridades, quantidades e prazos sejam facilmente visualizados.

3 METODOLOGIA

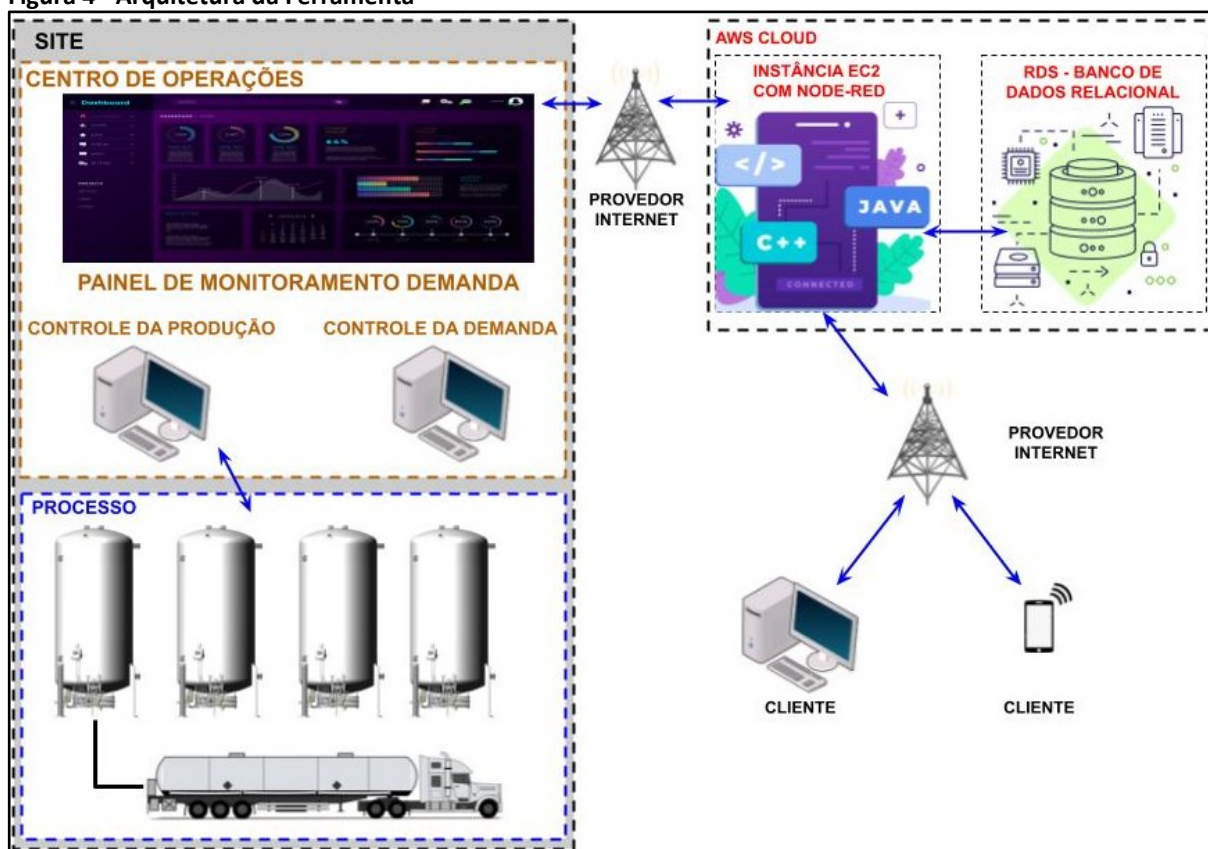
Para a implantação da ferramenta de controle de Produção e Demanda, com Computação em Nuvem, foi necessário contratar dois serviços da AWS: uma instância Amazon EC2 e um banco de dados Amazon RDS. O software Node-RED também tem papel central, pois fornece os *dashboards* para as interfaces de usuário e tem integração total com o banco de dados. A estratégia de utilizar serviços em nuvem atende a uma necessidade importante da indústria moderna que é a conectividade e rapidez na resposta. A unidade de produção precisa saber em tempo real sobre a demanda que deve ser priorizada. Para a aplicação da ferramenta, foi criada uma indústria fictícia de laticínios, contudo, pode ser amplamente aplicada em diversas manufaturas.

3.1 Arquitetura do sistema

A arquitetura da ferramenta de controle de Produção e Demanda com Computação em Nuvem está representada na Figura 4. No canto inferior direito dessa figura, estão os clientes que podem cadastrar seus pedidos no portal *e-commerce*. O servidor em nuvem da AWS, que abriga a página *web* em uma instância EC2 e o banco de dados em um RDS, estão representados pelo retângulo na parte superior direita. Já dentro da unidade produtiva, representado pelo retângulo na parte esquerda nomeado como *site*, estão os setores que vão utilizar as informações gravadas no banco de dados para executar o gerenciamento e priorização da produção.

A instância EC2 contém o sistema operacional Windows. Este sistema operacional hospeda o Node-RED que é responsável por três integrações: uma com o banco de dados, outra com a interface de clientes e a última com os departamentos produtivos.

Figura 4 - Arquitetura da Ferramenta



Fonte: Elaborada pelo autor

3.2 Configuração da Instância EC2

A instância EC2 foi configurada com o sistema operacional de 64 bits Windows 10 Profissional, em inglês, e utiliza os seguintes recursos de hardware:

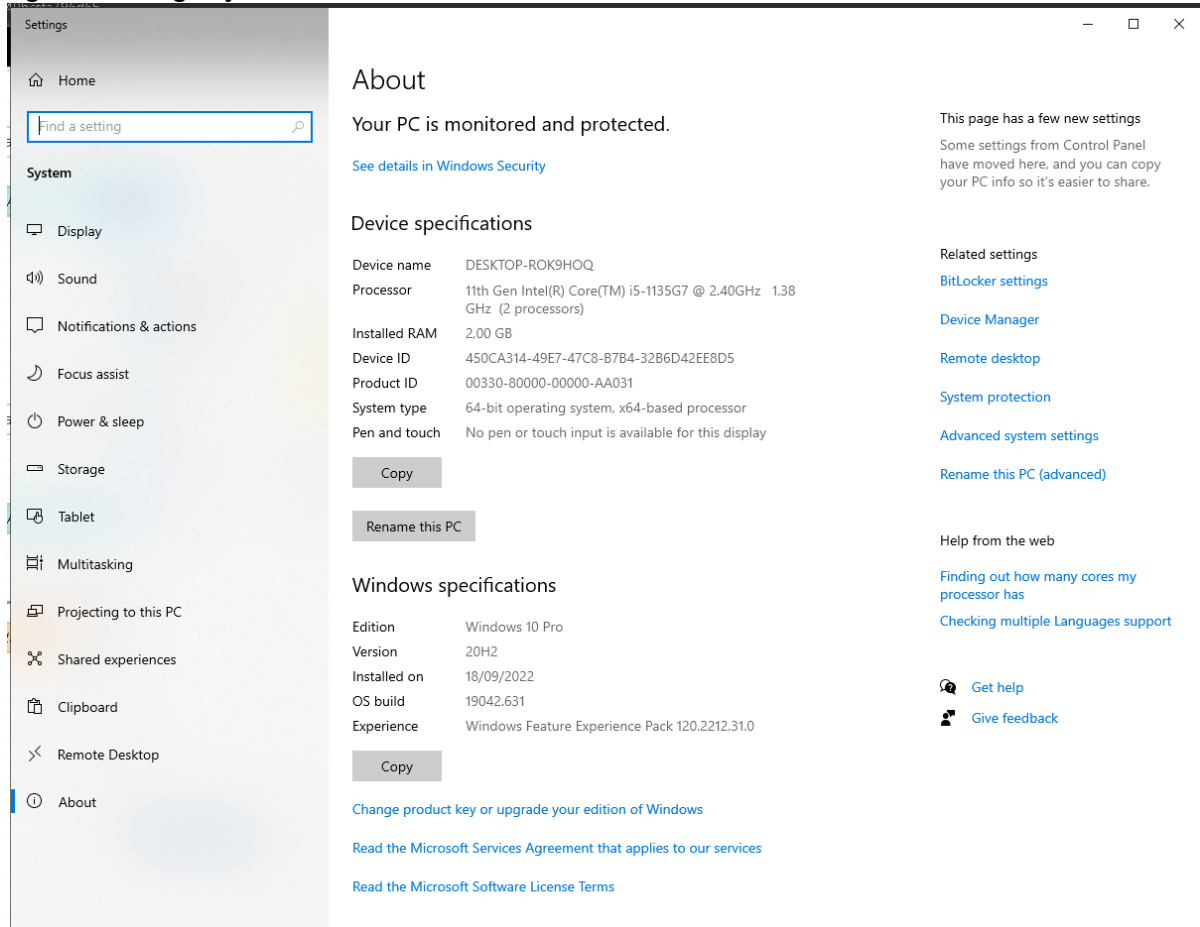
- 02 Processadores I5-1135G7 @ 2,40 GHz 1,38 GHz;
- 02 GB de RAM;
- 100 GB SSD;

A página de propriedades da máquina virtual está representada na Figura 5. É importante ressaltar que, por se tratar de máquina virtual, com o aumento da demanda de clientes simultâneos conectados, a configuração pode ser facilmente expandida para evitar lentidão ou inaccessibilidade das páginas/serviços.

A figura mostra também, uma informação muito importante que é a versão do sistema operacional (20H2). De acordo com a versão do Windows instalada, as bibliotecas do Node-RED, ou outros serviços de *web service*, podem não funcionar corretamente. A recomendação da Microsoft e dos fornecedores de software é de sempre utilizar a versão mais atualizada disponível.

Uma opção para que a configuração receba mais recursos somente quando necessário, é a contratação dos serviços de elasticidade da AWS, ou seja, mais recursos de hardware são colocados e retirados de acordo com a quantidade de conexões simultâneas.

Figura 5 - Configuração da Instância EC2



Fonte: Elaborada pelo autor

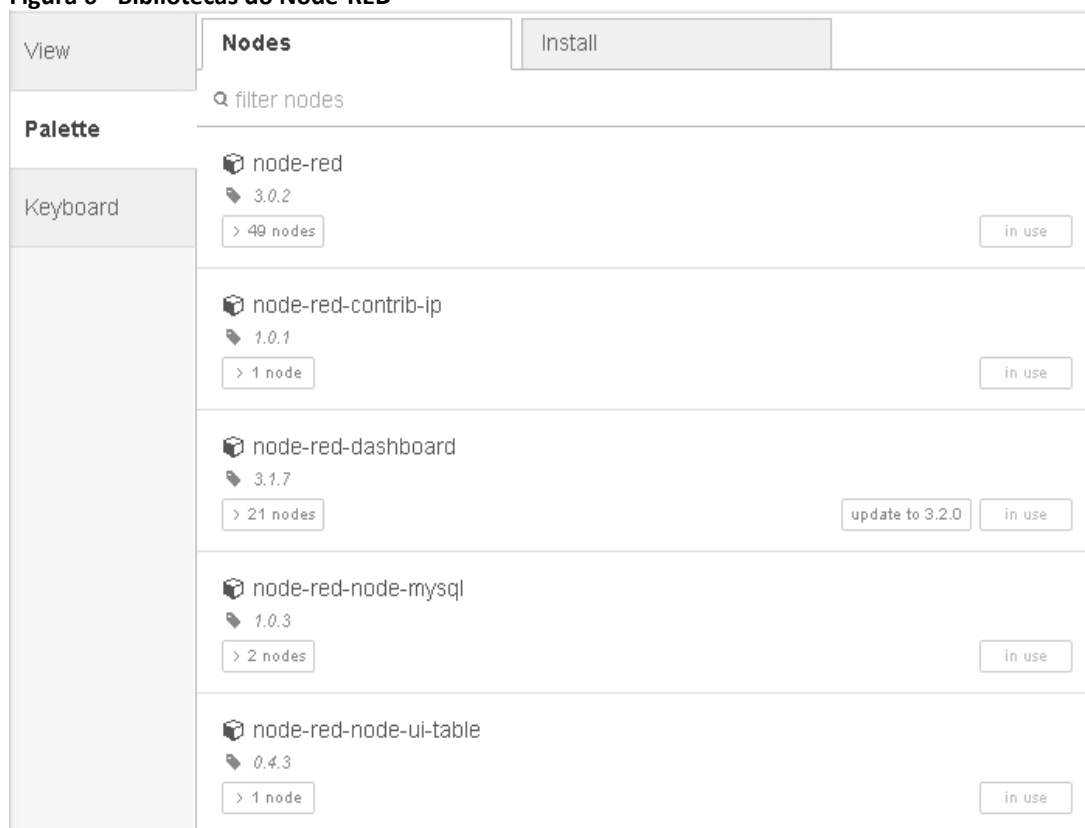
3.3 Instalação do Node-RED na Instância EC2

Para realizar a instalação do Node-RED, o método mais adequado é seguir as orientações da página oficial da ferramenta (<https://nodered.org/>). Após baixar o software, para seguir com a instalação do Node-RED na instância EC2, é necessário executar o seguinte comando no Prompt do Windows ou no Windows Powershell: *"npm install -g --unsafe-perm node-red"*.

Com o Node-RED devidamente instalado, é necessário executar o comando *"node-red"* no Prompt do Windows para que o software comece a executar as funções de programação. A programação do Node-RED é feita em página *web*, sendo assim, para abrir a página de programação é necessário abrir algum navegador *web* e digitar o caminho: *"http://localhost:1880"* (1880 é a porta padrão que funciona o Node-RED e pode ser alterada conforme a necessidade). O Node-RED funciona nos principais navegadores, como o Google Chrome, Microsoft EDGE, Firefox etc. Após a instalação do Node-RED, a etapa seguinte foi instalar as bibliotecas de API.

A Figura 6 apresenta o Node-RED e as bibliotecas instaladas.

Figura 6 - Bibliotecas do Node-RED



Fonte: Elaborada pelo autor

As bibliotecas e APIs podem ser instaladas acessando o menu *Manage Palette*. Para o funcionamento da ferramenta, foram instaladas algumas bibliotecas para funções específicas:

- node-red-contrib-ip*: Bloco utilizado para identificar o IP público da instância EC2;
- node-red-dashboard*: Conjunto de blocos com funções do dashboard, como botões, textos e campos de informações;
- node-red-node-mysql*: Conjunto de blocos com funções para integração com o Banco de Dados MySQL;
- node-red-node-ui-table*: Bloco para apresentação de tabela no dashboard;

3.4 Configuração das tabelas no Banco de dados RDS

A ferramenta de controle de Produção e Demanda com Computação em Nuvem necessita de um banco de dados (neste caso nomeado 'db_lat') e duas tabelas (nomeadas 'tab_cliente' e 'tab_pedidos'). O software escolhido para manipular o banco de dados e configurar as tabelas, foi o *MySQL Workbench*.

A tabela 'tab_cliente' possui 4 colunas. A coluna ID, que é o parâmetro do tipo 'numérico', possui a função auto-incremental, que a cada nova entrada no banco, um novo número subsequente é gerado. As colunas 'Cliente' e 'END', que identificam consequentemente, o nome do cliente e o seu endereço, são do tipo 'string', ou seja, é um campo a ser preenchido com caracteres alfanuméricos. E a coluna 'CNPJ', que também é do tipo 'numérico', para que seja inserido o CNPJ do cliente.

A figura 7 mostra em detalhe, a estrutura do banco de dados na lateral esquerda, na área *Schemas*. O banco de dados 'db_lat' está em destaque indicado pela letra 'A', as tabelas

estão em destaque indicado pela letra ‘B’ e na área central está o conteúdo da tabela de clientes, já com alguns cadastros, que está em destaque indicado pela letra ‘C’.

Figura 7 - Estrutura do Banco de Dados MySQL em detalhe

The screenshot shows the MySQL Workbench interface. On the left, the 'SCHEMAS' tree shows the 'db_lat' database selected, with a red arrow 'A' pointing to it. Under 'db_lat', the 'Tables' folder is expanded, and 'tab_cliente' is highlighted with a red arrow 'B'. The main window shows a SQL query editor with the following code:

```

1 • USE db_lat;
2 • SELECT * FROM db_lat.tab_pedidos;
3 • delete from tab_pedidos where ID = 10;
4 • select PRODUTO, PRAZO from tab_pedidos where PRODUTO =
5 • SELECT * FROM db_lat.tab_cliente;

```

Below the query editor, the 'Result Grid' is displayed, showing a list of clients. A red arrow 'C' points to the grid. The grid has the following data:

ID	CLIENTE	CNPJ	END
1	EXTRA	222333444	RUA 15 DE ABRIL SP
2	BIG	555333222	AV. SAO JOAO, 200 PR
3	ESSO	654987321	AV. HUM, 1200 RJ
4	BOMBRIL	264899999	AV. DO ESTADO, 5500 SP
11	DANCLOUD	999999999	AV. SANTOS, 2222 SP
13	CSN	112233	Av. Amaro, 5000
15	Octavia Alimentos	268268268	Av. Estado, 5000 SP
16	BILL FOOD	42908490328	Av. do Cascalho, 2222 MG
17	Amici Thiago Foods	123456789	Rua Santo Abdré, 100 SP
18	Crishna Supermercados	987654321	Rua Taboão, 22 PR

Fonte: Elaborada pelo autor

A tabela “tab_pedidos” possui 6 colunas. A coluna ‘ID’, novamente, é o parâmetro do tipo ‘numérico’ e ‘auto incremental’, ou seja, a cada novo pedido cadastrado no banco de dados, um novo número subsequente é gerado. A coluna ‘cliente’ e ‘produto’, indicam o nome do cliente e o produto que foi solicitado. Estes dois campos são do tipo ‘string’ e são aceitos caracteres alfanuméricos. A coluna ‘QDE’, também é do tipo ‘numérico’, para que seja inserida a quantidade do produto a ser produzida.

Por fim, as colunas ‘prazo’ e ‘entrega’ indicam o prazo desejado pelo cliente e data efetiva da entrega. Estas colunas são do tipo ‘data’, ou seja, são aceitos conjuntos de caracteres numéricos separados por hífen (-) que correspondem à data, com o seguinte formato: AAAA-MM-DD. Neste caso, o ano possui quatro algarismos e, o mês e o dia possuem dois algarismos cada.

As colunas do tipo ‘data’ são críticas para a integração com o Node-RED, pois os dados devem ser inseridos no banco de dados exatamente neste formato. Nesta integração, foi necessário transformar o formato da data contida no Node-RED para se adequar ao banco de dados. A figura 8 mostra a estrutura e conteúdo da tabela ‘tab_pedidos’. É possível perceber que já existem alguns pedidos cadastrados através do portal *e-commerce*.

Figura 8 - Estrutura da tabela 'tab_pedidos'

The screenshot shows MySQL Workbench with the following details:

- Navigator:** Shows the database structure for 'db_lat', including tables 'tab_cliente' and 'tab_pedidos'.
- SQL Editor:** Contains five queries:
 - USE db_lat;
 - SELECT * FROM db_lat.tab_pedidos;
 - delete from tab_pedidos where ID = 10;
 - select PRODUTO, PRAZO from tab_pedidos where PRODUTO = 'Leite' and ENTREGA is null;
 - SELECT * FROM db_lat.tab_cliente;
- Result Grid:** Displays 11 rows of data for 'tab_pedidos'. The columns are ID, CLIENTE, PRODUTO, PRAZO, QDE, and ENTREGA.

ID	CLIENTE	PRODUTO	PRAZO	QDE	ENTREGA
1	DANCLOUD	Leite	0000-00-00	2	2022-09-01
2	DANCLOUD	Leite	0000-00-00	3	2022-09-08
3	BOMBRIL	Queijo	2022-10-10	22	2022-09-08
4	BOMBRIL	Manteiga	2022-11-01	2	NULL
5	BILL FOOD	Leite	2022-09-16	22	NULL
6	Octavia Alimentos	Creme Leite	2022-09-29	10	2022-09-10
7	Octavia Alimentos	Leite Conde...	2022-11-30	7	NULL
8	BILL FOOD	Queijo	2022-11-23	4	NULL
9	BOMBRIL	Manteiga	2022-10-07	5	NULL
11	FOX	Manteiga	2022-10-24	3	NULL
- Action Output:** Shows the execution results of the queries:

#	Time	Action	Message	Duration / Fetch
1	22:02:33	USE db_lat	0 row(s) affected	0.000 sec
2	22:02:42	SELECT * FROM db_lat.tab_cliente LIMIT 0, 1000	13 row(s) returned	0.016 sec / 0.000 sec
3	22:04:04	SELECT * FROM db_lat.tab_pedidos LIMIT 0, 1000	27 row(s) returned	0.015 sec / 0.000 sec

Fonte: Elaborada pelo autor

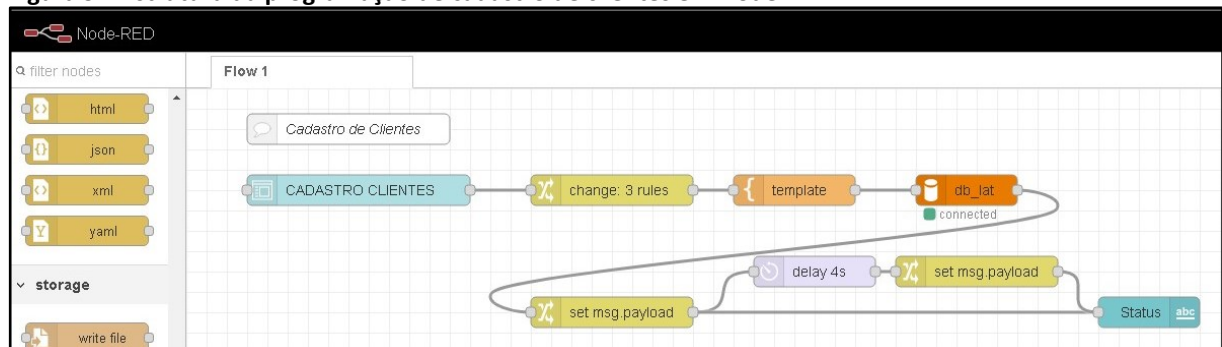
3.5 Programação Node-RED

A programação do Node-RED, consiste em quatro estruturas: a primeira faz a função de cadastro de clientes; a segunda faz a função de cadastro de pedidos; a terceira faz a função de apresentação da demanda em tempo real e a quarta faz a função de baixa dos pedidos entregues. Estas quatro estruturas compõem as bases das quatro telas do *dashboard*, ou seja, cada estrutura tem uma função no projeto e uma tela.

3.5.1 Cadastro de clientes

A primeira estrutura, que executa a função de cadastro de clientes, está representada na Figura 9. Os dois blocos da cor azul, são também elementos do *dashboard*, ou seja, o primeiro bloco é um formulário onde os clientes inserem as informações para cadastro, e o segundo é uma mensagem de confirmação que o cliente foi cadastrado corretamente.

Figura 9 – Estrutura da programação de cadastro de clientes em Node-RED



Fonte: Elaborada pelo autor

O bloco em laranja-escuro (db_lat) da figura acima é originado da API do MySQL. Com a utilização desta API, a integração do Node-RED com o banco de dados é facilitada, pois os comandos utilizados no Node-RED são os mesmos utilizados no MySQL. A Figura 10 apresenta o código utilizado no cadastro de um novo cliente no banco de dados.

Figura 10 - Código de cadastro de cliente do Node-RED

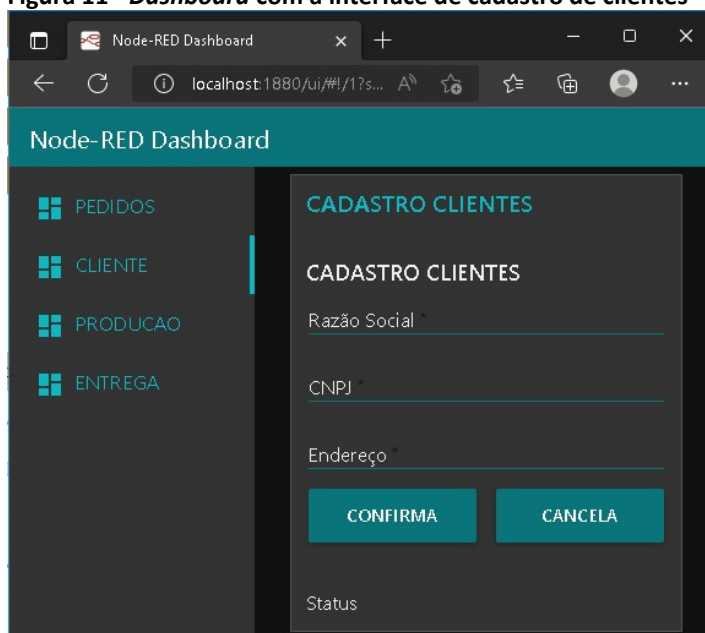


Fonte: Elaborada pelo autor

Para acessar a programação do Node-RED, é necessário utilizar um navegador *web* na instância EC2 e digitar o endereço: <http://localhost:1880>. Já para acessar a *dashboard*, utiliza-se o mesmo endereço só que com o complemento *"/ui"*, que significa *user interface*, resultando no endereço: <http://localhost:1880/ui>.

A Figura 11 apresenta o *dashboard* com a interface de usuário para cadastro de clientes. Também é possível observar na figura, que a navegação entre as funções é através do menu lateral esquerdo. Para efetuar um cadastro de cliente, basta digitar a Razão Social, o CNPJ e o Endereço nos campos correspondentes e confirmar no botão 'Confirma'. Uma mensagem de confirmação vai aparecer no campo 'Status', se o cadastro foi bem-sucedido ou se houve alguma falha, como a tentativa de cadastrar um cliente já cadastrado.

Figura 11 - Dashboard com a interface de cadastro de clientes

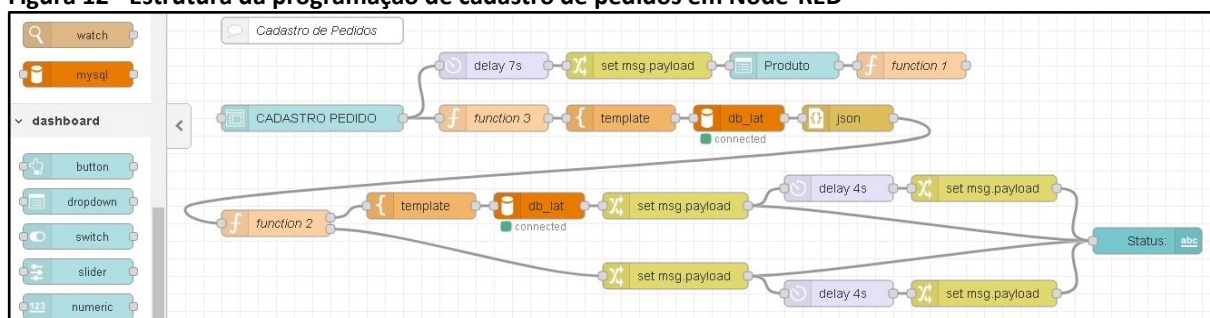


Fonte: Elaborada pelo autor

3.5.2 Cadastro de pedidos

A segunda estrutura contém as funções que executam a função de cadastro de pedidos e está representada na Figura 12. Novamente, os blocos na cor azul são elementos do *dashboard*. Os blocos da cor laranja-escuro, são blocos de integração com o banco de dados MySQL.

Figura 12 - Estrutura da programação de cadastro de pedidos em Node-RED



Fonte: Elaborada pelo autor

O bloco da cor laranja escuro na parte superior da figura, executa uma função de *select* (selecionar) no banco de dados para verificar se o cliente digitado está devidamente cadastrado e após a confirmação, uma mensagem vai ser apresentada informando que o pedido foi cadastrado corretamente, ou que houve algum erro. O código do comando *select* está representado na Figura 13.

Figura 13 - Código do comando *select* do Node-RED

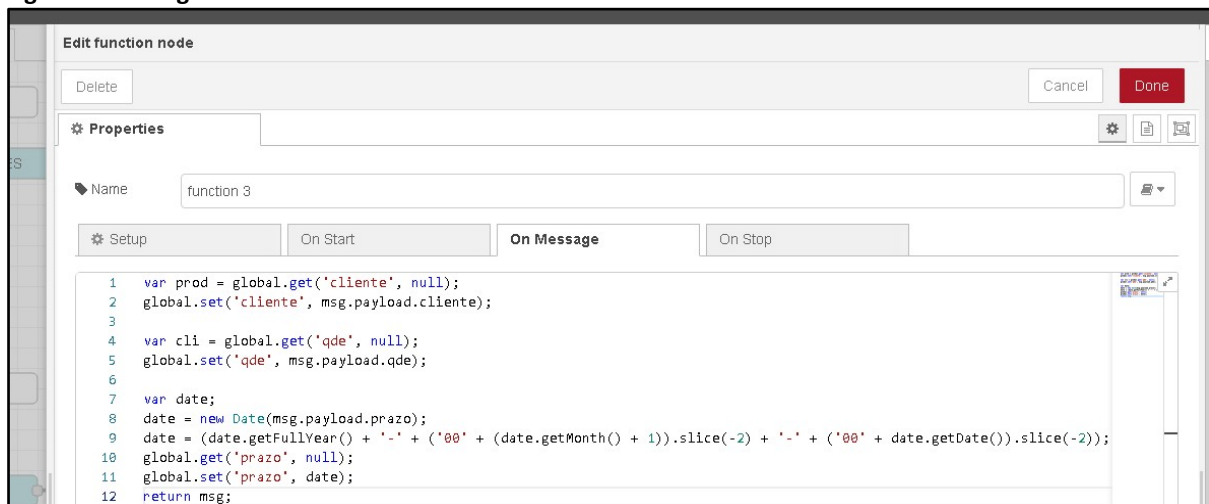


Fonte: Elaborada pelo autor

O bloco da cor laranja-escuro na parte inferior da Figura 12, possui um código para a inserção de um novo pedido no banco de dados, contendo entre outras informações, o prazo. Antes de aplicar a informação do prazo, é necessário ajustar o formato da data para que seja compatível com o formato do campo no MySQL.

A Figura 14 mostra o código para conversão do dado originado no Node-RED (data+hora) para o formato do MySQL, que é somente data no formato AAAA-MM-DD.

Figura 14 - Código de conversão do formato da data



```

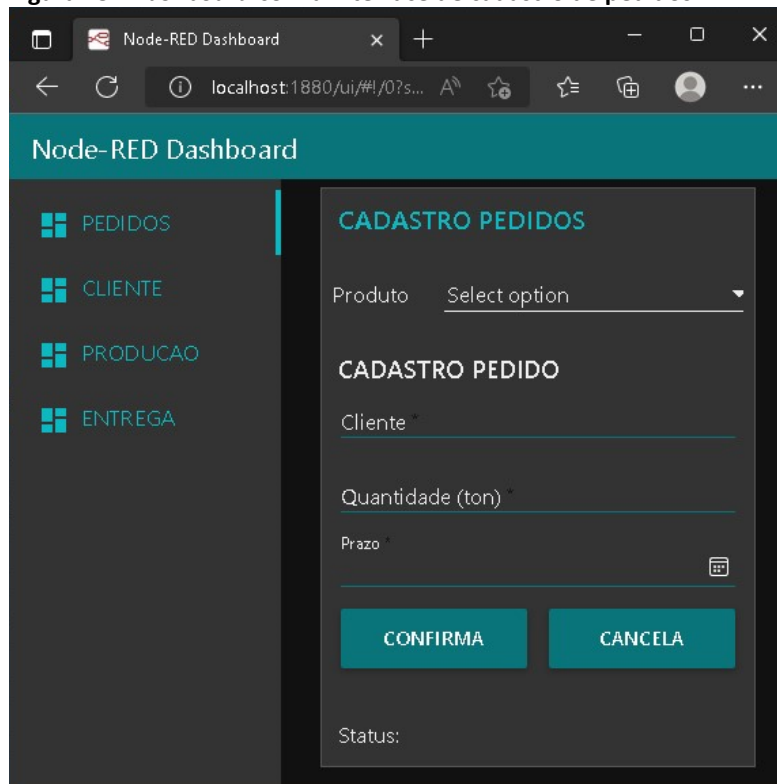
1  var prod = global.get('cliente', null);
2  global.set('cliente', msg.payload.cliente);
3
4  var cli = global.get('qde', null);
5  global.set('qde', msg.payload.qde);
6
7  var date;
8  date = new Date(msg.payload.prazo);
9  date = (date.getFullYear() + '-' + ('00' + (date.getMonth() + 1)).slice(-2) + '-' + ('00' + date.getDate()).slice(-2));
10 global.get('prazo', null);
11 global.set('prazo', date);
12 return msg;

```

Fonte: Elaborada pelo autor

Na Figura 15 é apresentado o *dashboard* com a interface de usuário para cadastro de pedidos. Para efetuar um cadastro de pedidos, basta selecionar o produto desejado na caixa de seleção, digitar o cliente, a quantidade e o prazo nos campos correspondentes e confirmar no botão 'Confirma'. Uma mensagem de confirmação vai aparecer no campo 'Status', se o cadastro foi bem-sucedido ou se houve alguma falha, como a tentativa de cadastrar um pedido para um cliente não cadastrado.

Figura 15 - *Dashboard* com a interface de cadastro de pedidos

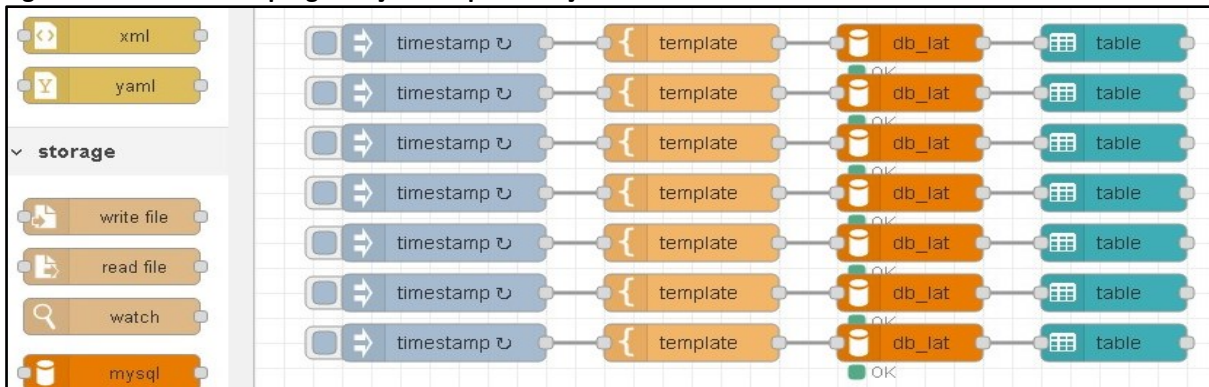


Fonte: Elaborada pelo autor

3.5.3 Apresentação da demanda em tempo real

A terceira estrutura executa a função de apresentação da demanda em tempo real e está representada na Figura 16. Este trecho de programação é executado a cada 10s, para que os novos pedidos cadastrados no banco de dados sejam apresentados na tela de visualização da demanda.

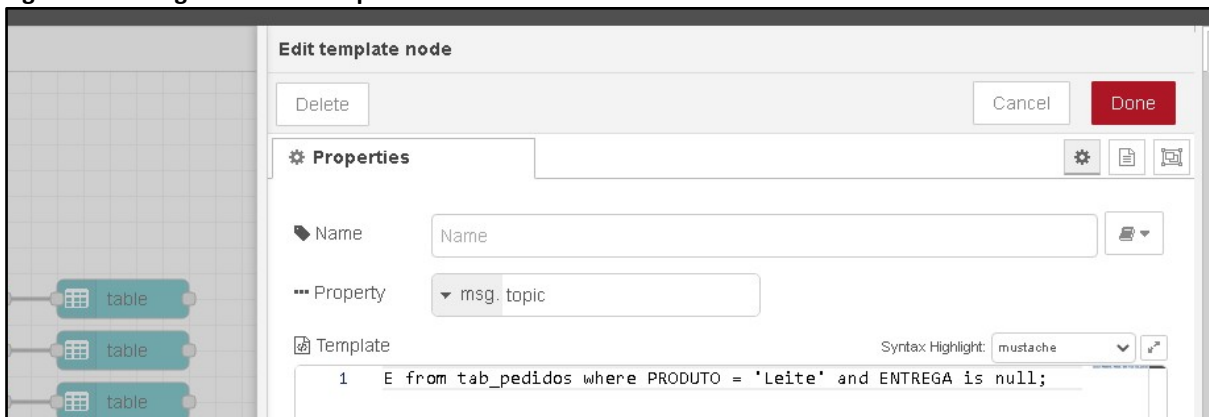
Figura 16 - Estrutura da programação de apresentação da demanda em Node-RED



Fonte: Elaborada pelo autor

O código para buscar os produtos ainda não entregues está representado na Figura 17. Basicamente, o código busca todos os pedidos daquele produto (classe) que não tenham dados no campo de entrega, e apresenta na tela em forma de tabela.

Figura 17 - Código de busca dos produtos em aberto



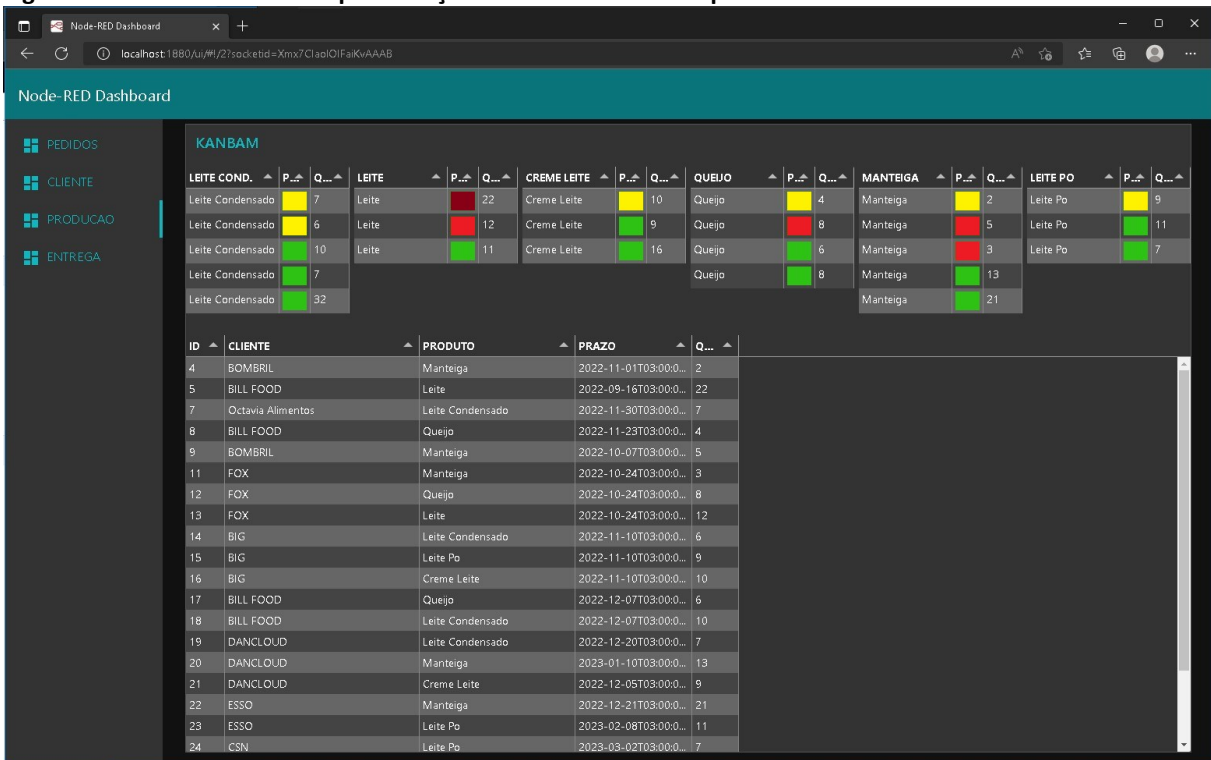
Fonte: Elaborada pelo autor

Na Figura 18 é apresentado o *dashboard* com a demanda de produção em tempo real, no formato de KANBAM. A escolha de uma fábrica de laticínios fictícia foi para exemplificar onde esta ferramenta pode ser útil, pois em indústrias de processos contínuos, a escolha com qual produto deve ser produzido a partir da mesma matéria-prima faz parte da rotina.

A ordem de prioridade dos produtos que devem ser produzidos segue a seguinte ordem para facilitar a visualização no KAMBAM:

- Vermelho-escuro: Produto em atraso;
- Vermelho: Produtos com prazo próximo da entrega;
- Amarelo: Produtos com prazo médio para a entrega;
- Verde: Produtos com prazo controlado para a entrega;

Figura 18 - *Dashboard* com a apresentação da demanda em tempo real

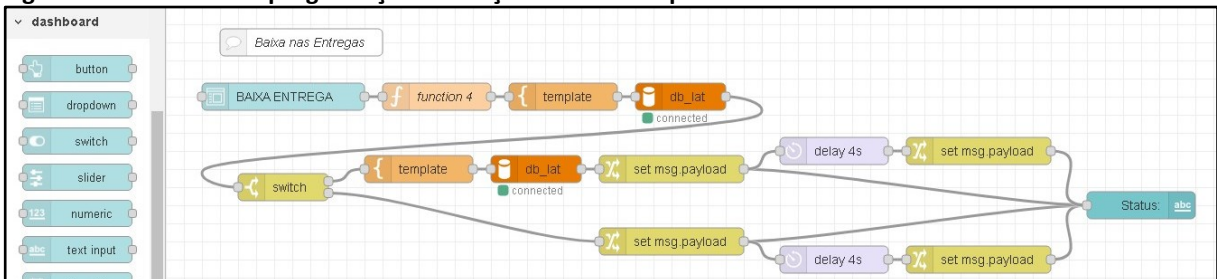


Fonte: Elaborada pelo autor

3.5.4 Baixa de pedidos entregues

A quarta estrutura executa a função de baixa dos pedidos entregues e está representada na Figura 19.

Figura 19 - Estrutura da programação da função de baixa de pedidos em Node-RED

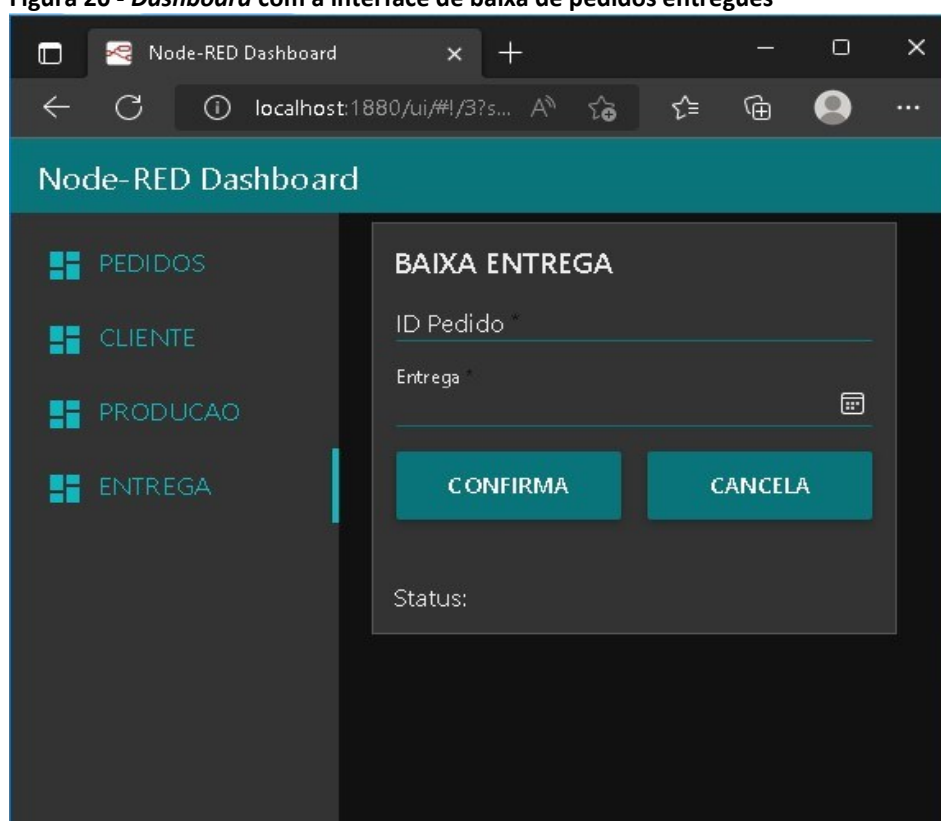


Fonte: Elaborada pelo autor

Na Figura 20 é apresentado o *dashboard* com a interface de usuário para baixa de pedidos entregues. Para efetuar uma baixa de produto entregue, basta digitar o ID do pedido e a data da entrega nos campos correspondentes e confirmar no botão 'Confirma'.

Uma mensagem de confirmação vai aparecer no campo 'Status', se a baixa foi bem-sucedida ou se houve alguma falha, como a tentativa de dar baixa em pedido já entregue.

Figura 20 - *Dashboard* com a interface de baixa de pedidos entregues



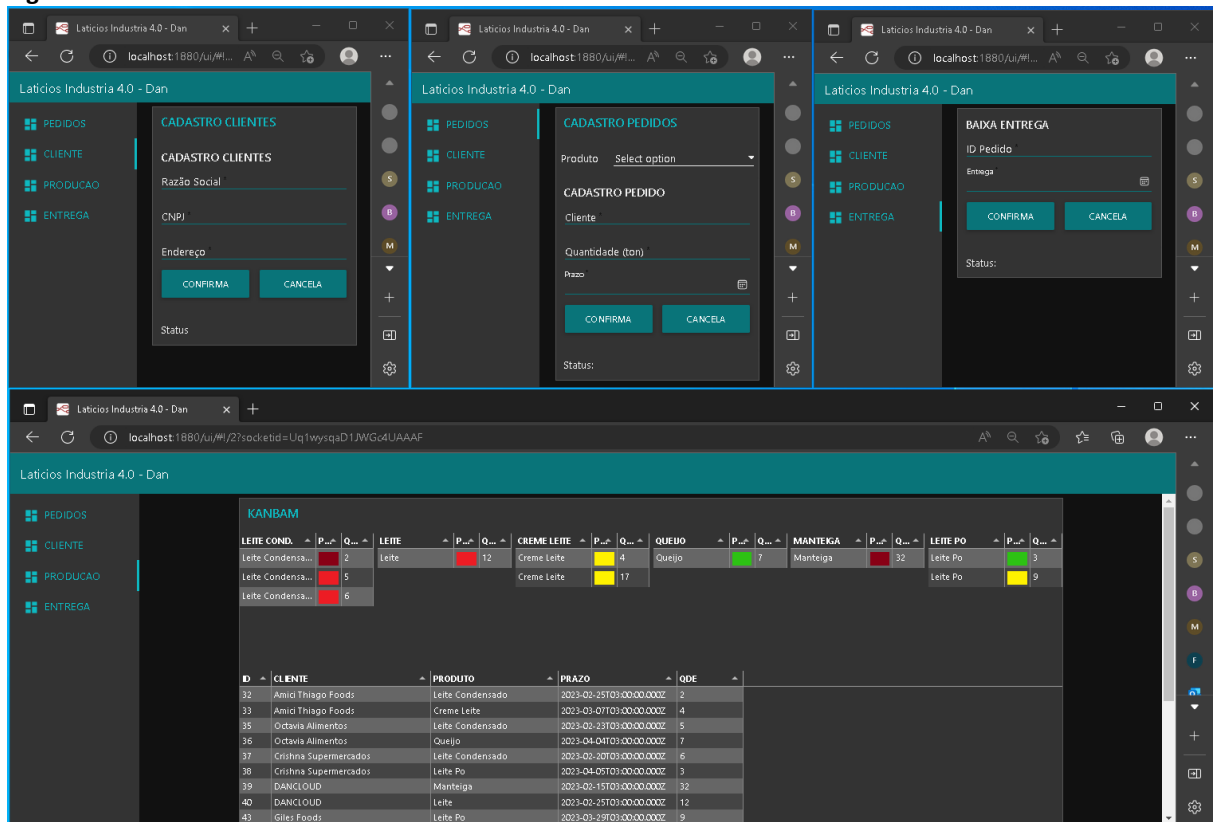
Fonte: Elaborada pelo autor

4 RESULTADOS E DISCUSSÕES

Os testes efetuados em cadastramentos de clientes fictícios, bem como pedidos também fictícios, foram satisfatórios. A dinamização das telas de interface ficou bem agradável e com interação intuitiva, sem necessidade de treinamentos complexos, o que viabiliza a utilização da ferramenta em cenários de indústrias com grandes equipes de operação ou com grande rotatividade de pessoas.

A apresentação da demanda também foi agradável, e a cada pedido novo cadastrado, surge um novo cenário. A equipe de PvD visualiza os dados em tempo real, conforme mostra a figura 21, ajudando e embasando as tomadas de decisão que é o objetivo central deste artigo.

Figura 21 – Telas de interface da ferramenta



Fonte: Elaborada pelo autor

Em relação aos custos, a Tabela 1 mostra os valores estimados para o desenvolvimento do projeto e a Tabela 2 mostra os valores estimados para a operação, ou seja, para a utilização dos serviços de nuvem.

Tabela 1 – Custo estimado de Implantação

Custo de Implantação			
Profissional	HH	HH/hora*	Sub-total
Arquiteto JAVA	40	R\$ 250,00	R\$ 10.000,00
Plataforma Catho (2023): "Arquiteto Java [...] Média Salarial: R\$ 9.611,09".			
Salário-hora: R\$ 9.611,09 / 200 horas = R\$ 48,05			
Custo Encargos: R\$ 48,05 * 2 = R\$ 96,11			
Taxas Adm, Impostos, Custo-fixos e lucros: R\$ 96,11 * 2,5 = R\$ 240,27			
* Estimativa razoável para valor de HH de Arquiteto Java: R\$ 250,00/hora.			

Fonte: Elaborada pelo autor

Tabela 2 – Custo estimado de Operação

Custo de Operação			
Serviço	Custo/Mês**	Câmbio (3/3/23)	Real
Amazon EC2	\$51,39	5,20	R\$ 267,23
Amazon RDS	\$15,13	5,20	R\$ 78,68
Total / Mês			R\$ 345,90
**AWS Pricing Calculator Amazon(2023c).			

Fonte: Elaborada pelo autor

O custo estimado do valor de Homem-Horas (HH) para o desenvolvimento foi obtido com base no salário médio que consta na plataforma Catho (2023). Os custos estimados de mensalidade dos serviços da nuvem foram obtidos utilizando a ferramenta de cálculo *AWS Pricing Calculator* da Amazon. A Figura 22 mostra as características e custos estimados dos serviços.

Figura 22 – AWS Pricing Calculator

Estimate summary				
Upfront cost	Monthly cost	Total 12 months cost		
0.00 USD	66.52 USD	798.26 USD		
		Includes upfront cost		

Detailed Estimate				
Name	Group	Region	Upfront cost	Monthly cost
Amazon EC2	No group applied	US East (Kansas City 2)	0.00 USD	51.39 USD
Description: Controle de Demanda Config summary: Tenancy (Shared Instances), Operating system (Windows Server), DT Inbound: Not selected (0 TB per month), DT Outbound: Not selected (0 TB per month), DT Intra-Region: (0 TB per month), Workload (Consistent, Number of instances: 1), Advance EC2 instance (t3.medium), Pricing strategy (On-Demand), Enter any placeholder cost such as Licensing to add to your estimate (), null (0)				
Amazon RDS for MySQL	No group applied	US East (Ohio)	0.00 USD	15.13 USD
Description: BD Controle Demanda Config summary: Storage for each RDS instance (General Purpose SSD (gp2)), Storage amount (30 GB), Quantity (1), Instance type (db.t4g.micro), Utilization (On-Demand only) (100 %Utilized/Month), Deployment option (Single-AZ), Pricing strategy (OnDemand)				

Fonte: Amazon (2023c).

5 CONCLUSÃO

A ferramenta de controle da produção e demanda desenvolvida utilizando a tecnologia habilitadora da Indústria 4.0 de Computação em Nuvem e a ferramenta gratuita do Node-RED, é eficiente no auxílio às equipes produtivas e seus controles, pois não exige grande complexibilidade de desenvolvimento, e ainda possui duas vantagens decisivas na escolha de sistemas que oferecem funções semelhantes, que são o tempo de implantação e o custo.

Em relação ao tempo de implantação, com base na experiência do desenvolvimento deste artigo, é possível estimar com segurança que com 40 HH de um Arquiteto Java, por exemplo, uma versão inicial já com todas as funcionalidades desta ferramenta descrita é implantada.

A utilização da tecnologia de Computação em Nuvem, como base da ferramenta de auxílio à produção e ao departamento de PvD, torna a implantação simples, pois não há necessidade de aquisição de materiais, equipamentos ou quaisquer *hardwares*.

A forma simples e intuitiva das telas atualizadas em tempo real, deixa claro quais produtos devem ser priorizados. Além disso, o time operacional tem a oportunidade de compreender algumas tomadas de decisão e propor alternativas que priorizem os prazos mais curtos sem onerar demais as horas produtivas em trocas de *setup* da manufatura.

A ferramenta de controle de produção e demanda, demonstrada neste artigo, apresenta grande potencial de aplicação para diversos setores industriais que serão

beneficiados com uma redução significativa de desperdícios de tempo graças à conectividade e disponibilidade de dados *online*, o que permitirá a tomada de decisões assertivas no momento certo.

REFERÊNCIAS

- AMAZON. **Elastic compute cloud**. 2022a. Disponível em: <https://aws.amazon.com/pt/ec2/getting-started/>. Acesso em: 22 out 2022.
- AMAZON. **Relational Database Service**. 2022b. Disponível em: <https://aws.amazon.com/pt/rds/>. Acesso em: 22 out 2022.
- AMAZON. **AWS Pricing Calculator**. 2023c. Disponível em: <https://calculator.aws/#/estimate?id=cc575acb5e393476a208918778129cb9aca263c7>. Acesso em: 04 mar 2023.
- ARNOLD, J. R. Tony; CHAPMAN, Stephen N.; CLIVE, Lloyd M. **Introduction to materials management**. 6. ed. New Jersey-US. Pearson Education, Inc. 2008.
- BUYA, R.; BROBERG, J.; GOSCINSKI, A. **Cloud computing: principles and paradigms**. New Jersey-US. John Wiley & Sons. 2011.
- CATHO. **Arquiteto Java: Média Salarial**. 2023. Disponível em: <https://www.catho.com.br/profissoes/arquiteto-java/>. Acesso em 03 mar 2023.
- CORRÊA, H. L.; GIANESI, I. G. N. **Just in Time, MRP II e OPT - um enfoque estratégico**. São Paulo: Atlas, 2. ed., 1992.
- FAVARETTO, F. **Uma contribuição ao processo de gestão da produção pelo uso da coleta automática de dados de chão de fábrica**. 2001. Tese (Doutorado em Engenharia Mecânica) – Escola de Engenharia, Universidade de São Paulo, São Carlos, 2001.
- JAVATPOINT. **Tipos de serviços na nuvem**. 2022a. Disponível em: <https://www.javatpoint.com/introduction-to-cloud-computing>. Acesso em 12 out. 2022.
- JAVATPOINT. **Gerenciamento dos serviços da nuvem**. 2022b. Disponível em: <https://www.javatpoint.com/introduction-to-cloud-computing>. Acesso em: 12 out. 2022.
- KIM, Won. **Cloud computing: today and tomorrow**. 2009. Disponível em: www.jot.fm/issues/issue_2009_01/column4/. Acesso em: 28 set. 2021
- MELO, D. de C.; ALCÂNTARA, R. L. C. A Gestão da Demanda na Cadeia de Produção da Indústria de Laticínios: uma análise dos problemas e abordagens para melhoria, **Revista Contabilidade, Gestão e Governança**, v. 15, n. 2, 2012. Disponível em: <https://revistacgg.org/index.php/contabil/article/view/337>. Acesso em: 20 out. 2022.

NODERED. **Node-RED: Low-code programming for event-driven applications**. 2022. Disponível em: <https://nodered.org>. Acesso em: 22 out 2022.

PROUND, John F. **Master scheduling: a practical guide to competitive**. 3. ed. New Jersey-US. John Wiley & Sons. 2007.

SEBRAE. **Informe de mercado: e-commerce**. 2017. Disponível em: <https://www.sebrae.com.br/Sebrae/Portal%20Sebrae/UFs/PA/Sebrae%20de%20A%20a%20Z/Ebook-Ecommerce.pdf>. Acesso em: 17 out. 2022.

SIGNIFICADOS. **Demanda (Economia)**. 2022. Disponível em: <https://www.significados.com.br/demanda/>. Acesso em 22 out 2022.

VERAS, Manoel. **Cloud computing: nova arquitetura da TI**. Rio de Janeiro: Brasport, 2012.

AGRADECIMENTOS

Agradeço imensamente à minha orientadora, aos mestres formadores da banca e à bibliotecária Jesuina pela incomensurável paciência e orientação prestada – sem sombra de dúvidas tornou esta obra muito mais rica e clara. Agradeço também à minha família que compreendeu perfeitamente os momentos de sacrifício do convívio familiar para dedicar a este artigo.

Sobre os autores:

ⁱ DANIEL CARVALHO DOS SANTOS

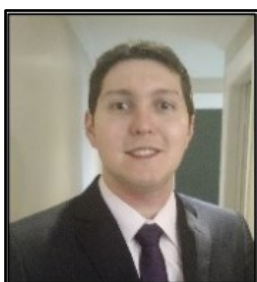


Possui graduação em Engenharia Mecatrônica pelas Faculdades Integradas de São Paulo - (FISP-SP - 2010). Também é Tecnólogo em Automação pelo Instituto Federal de Educação, Ciência e Tecnologia de SP (IFSP - 2005). cursando atualmente a Pós-graduação em Indústria 4.0 EAD pela Faculdade SENAI de Tecnologia Mecatrônica (2022). Tem experiência na área de Automação, Engenharias e Projetos de Smart OT. É Coordenador de Tecnologia Operacional pela Air Liquide Brasil Ltda.

ⁱⁱ CRISHNA IRION



Docente no Ensino Superior, desde 2005, Doutoranda em Ciência da Computação (UFU- início julho/2022), Mestre em Ciência da Linguagem (UNIVAS-2012), Créditos no Mestrado em Redes de Telecomunicações (INATEL-2006), Especialista em Informática Empresarial (UNIFEI -2003), Bacharel em Ciência da Computação pela Fundação de Ensino e Tecnologia de Alfenas (UNIFENAS-1999). Atualmente Professora Educação Profissional Tecnológica no SENAI/SCS Diretora (2021/22) da Maratona Mineira de Programação. Representante do Sudeste na Maratona de programação SBC. Entusiasta no ensino de programação em todos os níveis (desde fundamental à graduação e comunidade), sendo voluntária em vários projetos filantrópicos.

iii THIAGO TADEU AMICI (Banca)

Ministra aulas na pós-graduação de Indústria 4.0 nas modalidades presencial e EAD, no MBA de Gestão de Projetos aplicados a inovação em Indústria 4.0 e nas graduações em Engenharia de Controle e Automação, em Tecnologia em Mecatrônica e Tecnologia em Análise e Desenvolvimento de Sistemas na Faculdade SENAI de Tecnologia Mecatrônica. Assessora também o Instituto SENAI de Tecnologia Metalmeccânica em projetos industriais com foco na Indústria 4.0, onde desenvolveu inúmeros projetos como integrador de relevância nacional e internacional. Tem experiência na área de Engenharia Elétrica, Automação Industrial, Mecatrônica, Robótica e Indústria 4.0. Experiência internacional na aprovação de linha de produção (Cavemil) em Milão na Itália e sua instalação no Brasil.

Endereço para acessar este CV:

<http://lattes.cnpq.br/9165856219131658>

iv JORGE ANTONIO GILES FERRER (Banca)

Doutor em Engenharia Mecânica pela Universidade Estadual de Campinas – UNICAMP. Mestre em Engenharia Mecânica pela Escola Politécnica da Universidade de São Paulo - USP. Engenheiro Mecânico pela Pontificia Universidad Católica del Perú. Possui Licenciatura plena em pedagogia para educação profissional em ensino médio, pela Universidade Metodista de Piracicaba - UNIMEP. Atualmente é docente da Faculdade SENAI de Tecnologia Mecatrônica em São Caetano do Sul SP, onde ministra disciplinas no Curso Superior de Tecnologia em Mecatrônica, Curso de Engenharia de Automação e Controle e na pós-graduação na área de Gestão de Projetos e Produção (Lean Manufacturing e Virtualização de Sistemas Produtivos). É membro do Banco de Avaliadores do Sistema Nacional de Avaliação da Educação Superior do Ministério de Educação do Brasil - MEC e Avaliador do Projeto de Formação Profissional Dual em Mecatrônica da Câmara de Comércio e Indústria Brasil-Alemanha.