

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ  
DEPARTAMENTO ACADÊMICO DE ELETROTÉCNICA  
ENGENHARIA ELÉTRICA**

**GABRIEL FERREIRA DE BRITO**

**INDÚSTRIA 4.0: DESENVOLVIMENTO DE UMA FERRAMENTA PARA ANÁLISE  
DE EFICIÊNCIA DE UMA LINHA DE PRODUÇÃO EM UMA INDÚSTRIA DE  
PLÁSTICOS DA REGIÃO METROPOLITANA DE CURITIBA**

**TRABALHO DE CONCLUSÃO DE CURSO**

**CURITIBA  
2019**

**GABRIEL FERREIRA DE BRITO**

**INDÚSTRIA 4.0: DESENVOLVIMENTO DE UMA FERRAMENTA PARA ANÁLISE  
DE EFICIÊNCIA DE UMA LINHA DE PRODUÇÃO EM UMA INDÚSTRIA DE  
PLÁSTICOS DA REGIÃO METROPOLITANA DE CURITIBA**

Trabalho de Conclusão de Curso de Graduação em Engenharia Elétrica, apresentado à disciplina de TCC 2, do Departamento de Acadêmico de Eletrotécnica (DAELT) da Universidade Tecnológica Federal do Paraná (UTFPR), como requisito parcial para a obtenção do título de Engenheiro Eletricista.

Orientador: Prof. Dr. Jorge A. Leludak

**CURITIBA**

**2019**

Gabriel Ferreira de Brito

# INDÚSTRIA 4.0: DESENVOLVIMENTO DE UMA FERRAMENTA PARA ANÁLISE DE EFICIÊNCIA DE UMA LINHA DE PRODUÇÃO EM UMA INDÚSTRIA DE PLÁSTICOS DA REGIÃO METROPOLITANA DE CURITIBA

Este Trabalho de Conclusão de Curso de Graduação foi julgado e aprovado como requisito parcial para a obtenção do Título de Engenheiro Eletricista, do curso de Engenharia Elétrica do Departamento Acadêmico de Eletrotécnica (DAELT) da Universidade Tecnológica Federal do Paraná (UTFPR).

Curitiba, 05 de junho de 2019.

---

Prof. Antonio Carlos Pinho, Dr.  
Coordenador de Curso  
Engenharia Elétrica

---

Profa. Annemarlen Gehrke Castagna, Mestre  
Responsável pelos Trabalhos de Conclusão de Curso  
de Engenharia Elétrica do DAELT

## ORIENTAÇÃO

---

Jorge Assade Leludak, Dr.  
Universidade Tecnológica Federal do Paraná  
Orientador

## BANCA EXAMINADORA

---

Daniel Balieiro Silva, Mestre  
Universidade Tecnológica Federal do Paraná

---

Marco Antonio Buseti de Paula, Pós-doutorado  
Universidade Tecnológica Federal do Paraná

A folha de aprovação assinada encontra-se na Coordenação do Curso de Engenharia Elétrica

## **AGRADECIMENTOS**

Primeiramente a Deus, pela coragem, disciplina e sabedoria que me mantiveram no caminho e me ajudaram a chegar até aqui.

A minha esposa Luíza e meu filho Joaquim pelo amor, compreensão e carinho durante as horas ausentes dedicadas para conclusão deste trabalho.

Aos meus pais, por todo carinho e tempo investido, além dos sacrifícios que realizaram para que eu pudesse me dedicar aos estudos.

Aos meus avós, que através do trabalho árduo superaram condições de escassez para que seus filhos e netos chegassem muito mais longe.

Ao colega Andrei Kanenovski que contribuiu significativamente com seus conhecimentos permitindo assim o êxito deste estudo.

À empresa envolvida neste trabalho pela confiança e a disponibilização de tempo e recursos sem os quais este trabalho não seria possível.

Aos amigos, pela paciência, pelos conselhos e pelos bons momentos que tornam meu caminho mais fácil.

Aos professores, que fizeram da arte de ensinar suas profissões e tanto contribuíram para o meu desenvolvimento profissional.

Ao meu orientador, Jorge Assade Leludak, pelo tempo e atenção disponibilizados e por todo o conhecimento dividido.

A banca examinadora, pelas críticas construtivas essenciais para alcançar a excelência deste trabalho.

## RESUMO

BRITO, Gabriel F. **Indústria 4.0: Desenvolvimento de uma Ferramenta para Análise de Eficiência de uma Linha de Produção em uma Indústria de plásticos da Região Metropolitana de Curitiba.** 2019. 90 f. Trabalho de Conclusão de Curso (Graduação) – Curso de Engenharia Elétrica. Universidade Tecnológica Federal do Paraná, Curitiba, 2019.

A Indústria tem vivido mudanças significativas nos últimos anos. O aumento de competitividade, a evolução e surgimento de novas tecnologias, aliado a mão de obra qualificada tem permitido a transição da indústria para uma nova era centrada na informação e denominada de quarta revolução industrial ou indústria 4.0. No Brasil este movimento ainda encontra espaços para amadurecimento, principalmente nas pequenas e médias empresas. Dentre as principais barreiras estão o alto investimento e falta de informações sobre os retornos financeiros destas novas tecnologias, mão de obra qualificada, incorporação dos meios de produção antigos às novas plataformas de comunicação e a segurança da informação. Apesar destes obstáculos a indústria brasileira necessita encontrar meios para alcançar novos níveis de modernização. A construção dos processos dentro da cadeia de produção com foco na informação permite rastreabilidade dos produtos, controle detalhado dos estoques de matéria-prima e produto e acabado, monitoramento dos índices de satisfação do cliente, comunicação entre os elementos presentes no chão de fábrica e a execução de atividades autônomas através da inteligência artificial, resultando em um benefício principal: a melhoria da performance dos processos. A partir disto, este trabalho propõe o desenvolvimento de uma ferramenta para coleta, armazenamento e análise de dados de uma linha de produção em uma empresa de porte médio. Ao final foi possível determinar a eficiência produtiva da linha de produção, a quantidade material produzido e não utilizado (rejeitos) e o consumo de energia determinando o custo despendido por peça, reduzindo assim o tempo para processamento das informações e elaboração de ações corretivas.

Palavras-chave: Indústria 4.0, pequenas e médias empresas, eficiência, consumo de energia, linha de produção.

## ABSTRACT

BRITO, Gabriel F. **Industry 4.0: Production Line Efficiency Analysis Tool Development from a Plastic Industry of Curitiba Metroplitan Region.** 2019. 90 f. Trabalho de Conclusão de Curso (Graduação) – Curso de Engenharia Elétrica. Universidade Tecnológica Federal do Paraná, Curitiba, 2019.

Industry has come through significant changes in the last years. The increase of competitiveness, combined with growth of new technologies and qualified workforce has granted industry the transition to a new era centered in the information, defined as fourth revolution or industry 4.0. Brazil has a great potential and is able to harvest many benefits from this movement especially for small and medium sized companies. Among main obstacles are high investment and lack of information on financial return of these new technologies, skilled labor, old machinery update to current electronic communication protocols and information security. Despite these obstacles Brazilian industry needs to find the means to achieve such level of modernization. The construction of processes in the production chain focused on information allows product traceability, raw material stock and finished goods detailed control, customer satisfaction ratio monitoring, communication within the machines at shop floor area and execution of autonomous activities through artificial intelligence, enabling the overall process improvement. From that, this study proposes a tool development to collect, store and analyze data from a production line of a medium sized company. At the end it was possible to determine process efficiency, the quantity of material produced and not used and also the energy consumption giving out the cost per part produced, thus reducing significantly the amount of time to evaluate and define corrective actions upon the data generated by production line.

Key-words: Industry 4.0, small and medium sized companies, energy consumption, production line.

## LISTA DE FIGURAS

FIGURA 1 - UTILIZAÇÃO DO BIG DATA POR EMPRESAS ENTREVISTADAS PELA CONSULTORIA PWC EM 2016.....	23
FIGURA 2 - REDES DE COMUNICAÇÃO E INTERFACE COM USUÁRIO ATRAVÉS DA "INTERNET DAS COISAS".....	24
FIGURA 3 - ECOSSISTEMA IIOT.....	25
FIGURA 4 - MODELO DE IMPLEMENTAÇÃO GERENCIAMENTO DE ENERGIA COM BASE EM IOT.....	28
FIGURA 5 - CLP COM ESTRUTURA TIPO FIXA.....	32
FIGURA 6 - CLP COM ESTRUTURA MODULAR.....	33
FIGURA 7 - CLP WAGO 750-881.....	34
FIGURA 8 – EXTRUSORA.....	39
FIGURA 9 – LINHA DE EXTRUSÃO.....	40
FIGURA 10 – CLP E CARTÕES ADICIONAIS.....	43
FIGURA 11 – CIRCUITO DE MEDIÇÃO DE ENERGIA.....	44
FIGURA 12 – ENCODER.....	45
FIGURA 13 – PARÂMETROS DE COMUNICAÇÃO COM BANCO DE DADOS XAMPP.....	45
FIGURA 14 – LAYOUT BANCO DE DADOS.....	46
FIGURA 15 – APLICAÇÃO JAVA.....	47
FIGURA 16 – STATUS DA LINHA DE PRODUÇÃO.....	48
FIGURA 17 – BANCO DE DADOS PARA APLICAÇÃO JAVA.....	49
FIGURA 18 – IMPORTAÇÃO DE DADOS POWER BI.....	49
FIGURA 19 – EDIÇÃO POWER BI.....	50
FIGURA 20 – FLUXOGRAMA DAS ETAPAS DE DESENVOLVIMENTO.....	51
FIGURA 21 – MESCLAR CONSULTAS.....	52
FIGURA 22 – COLUNAS BANCO DE DADOS POWER BI.....	53
FIGURA 23 – CÁLCULO DE VELOCIDADE REAL.....	53
FIGURA 24 – CÁLCULO ALTERAÇÃO DE VELOCIDADE PERCENTUAL.....	53
FIGURA 25 – CRIAÇÃO INTERFACE COM USUÁRIO.....	54
FIGURA 26 – METROS PRODUZIDOS.....	54
FIGURA 27 – METROS SOLICITADOS.....	54
FIGURA 28 – EFICIÊNCIA GERAL.....	55
FIGURA 29 – EFICIÊNCIA GERAL E FLUTUAÇÃO DE VELOCIDADE.....	55

FIGURA 30 – TEMPO POR <i>STATUS</i> (%).....	56
FIGURA 31 – CÁLCULO DE REJEITOS (%).....	57
FIGURA 32 – REJEITOS (%).....	57
FIGURA 33 – EFICIÊNCIA POR ITEM (%).....	58
FIGURA 34 – EFICIÊNCIA POR ITEM (%) – 2184-035.....	58
FIGURA 35 – EFICIÊNCIA POR ITEM (%) – 3310-275.....	59
FIGURA 36 – CÁLCULO DE CONSUMO (KWH) POR METROS PRODUZIDOS....	60
FIGURA 37 – CONSUMO (KWH) A CADA 100 METROS PRODUZIDOS.....	61
FIGURA 38 – COMPARATIVO LINHAS EXTRUSÃO (KWH / 100 METROS).....	62
FIGURA 39 – POTÊNCIA POR STATUS.....	63
FIGURA 40 – COMPARATIVO LINHAS EXTRUSÃO (R\$ / 100 METROS).....	63
FIGURA 41 – RELATÓRIO APROVEITAMENTO ENERGIA.....	64
FIGURA 42 – INTERFACE <i>ONLINE POWER BI</i> .....	65
FIGURA 43 – INTERFACE PARA <i>SMARTPHONE</i> .....	65



## LISTA DE TABELAS

TABELA 1 – AVALIAÇÃO DOS POTENCIAIS BENEFÍCIOS.....	16
TABELA 2 – CUSTO UNITÁRIO (R\$/KWH) – 2018.....	61

## **LISTA DE ABREVIATURAS**

CNI – Conselho Nacional da Indústria

PME – Pequenas e Médias Empresas

M2M – Machine-to-Machine

CPS – Cyber-Physical System

ASQ – American Society for Quality

PAAS – Platform as a Service

SAAS – Software as a Service

IAAS – Infrastructure as a Service

NIST – National Institute of Standard and Technology

IHM – Interface Homem-Máquina

IOT – Internet of Things

IIOT – Industrial Internet of Things

CLP – Controlador Lógico Programável

SQL – Structured Query Language

## SUMÁRIO

<b>1.</b>	<b>INTRODUÇÃO .....</b>	<b>12</b>
1.1	TEMA.....	12
1.2	DELIMITAÇÃO DO TEMA.....	13
1.3	PROBLEMAS E PREMISSAS .....	14
1.4	OBJETIVOS .....	15
1.4.1	Objetivo Geral.....	15
1.4.2	Objetivos Específicos .....	15
1.5	JUSTIFICATIVA.....	15
1.6	PROCEDIMENTOS METODOLÓGICOS.....	17
1.7	ESTRUTURA DO TRABALHO .....	18
<b>2</b>	<b>REFERENCIAL TEÓRICO.....</b>	<b>18</b>
2.1	INDÚSTRIA 4.0 .....	18
2.1.1	Computação em Nuvem .....	19
2.1.2	<i>Big Data Analytics</i> .....	21
2.1.3	Internet das Coisas.....	23
2.1.4	<i>Smart Energy Metering</i> .....	26
2.2	<i>LEAN MANUFACTURING</i> .....	27
2.3	CONTROLADORES LÓGICOS PROGRAMÁVEIS.....	31
2.3.1	Controlador Lógico Programável Wago.....	33
2.4	BANCO DE DADOS .....	34
2.4.1	Banco de Dados XAMPP.....	35
2.4.2	Microsoft <i>Power BI</i> .....	36
2.4.3	Aplicação Java .....	36
<b>3</b>	<b>FERRAMENTA DE ANÁLISE DE EFICIÊNCIA.....</b>	<b>37</b>
3.1	AMBIENTE DE APLICAÇÃO: LINHA DE PRODUÇÃO.....	37
3.2	METODOLOGIA DE CÁLCULO DE EFICIÊNCIA .....	39

3.3	AUTOMAÇÃO E FLUXO DE DADOS.....	40
<b>4</b>	<b>ETAPAS DE DESENVOLVIMENTO .....</b>	<b>42</b>
4.1	AQUISIÇÃO DE DADOS ENERGIA .....	42
4.2	AQUISIÇÃO DE DADOS PRODUÇÃO .....	43
4.3	ARMAZENAMENTO DE DADOS .....	44
4.4	STATUS DA LINHA DE PRODUÇÃO - APLICAÇÃO JAVA.....	45
4.5	ANÁLISE DE DADOS.....	48
4.6	FLUXOGRAMA DAS ETAPAS DE DESENVOLVIMENTO .....	49
<b>5</b>	<b>ESTUDO DE CASO – APLICAÇÃO DA FERRAMENTA .....</b>	<b>51</b>
5.1	ANÁLISE DE EFICIÊNCIA, REJEITOS E VELOCIDADE DA LINHA DE PRODUÇÃO.....	51
5.1.1	CÁLCULO DE EFICIÊNCIA.....	53
5.1.2	CÁLCULO DE REJEITOS .....	55
5.1.3	SEGMENTAÇÃO DE DADOS .....	56
5.3	ANÁLISE DE CONSUMO DE ENERGIA.....	58
5.4	PUBLICAÇÃO DE ANÁLISES E RELATÓRIO PARA SMARTPHONE .....	63
<b>6</b>	<b>CONSIDERAÇÕES FINAIS .....</b>	<b>65</b>
6.1	DIFICULDADES E DESAFIOS FUTUROS.....	65
6.2	CONCLUSÕES.....	66
	<b>REFERÊNCIAS .....</b>	<b>69</b>
	<b>APÊNDICE A – CÓDIGO CLP WAGO .....</b>	<b>74</b>
	<b>APÊNDICE B – CÓDIGO JAVA.....</b>	<b>78</b>
	<b>APÊNDICE C – BANCO DE DADOS XAMPP .....</b>	<b>90</b>

## 1. INTRODUÇÃO

### 1.1 TEMA

A transformação digital ou digitalização tem atingido todas as áreas do nosso cotidiano, proporcionando soluções pertinentes para grandes desafios de mobilidade urbana (*smart cities*), eficiência energética (*smart grid*) e aumento da produtividade industrial e manufatura avançada através da indústria 4.0 (CNI; 2016a).

Indústria 4.0 é o termo designado para quarta revolução industrial e busca propor conexões através de Sistemas Ciberfísicos (CPS) aumentando a integração entre todos os participantes da cadeia de valor (DELOITTE; 2014). Esta integração por sua vez trará grandes impactos nos ganhos de produtividade no chão de fábrica, o encurtamento de prazos e aumento da eficiência no uso dos recursos como matéria-prima e energia. Outro benefício importante é a flexibilidade da linha de produção viabilizando a customização dos produtos através da comunicação instantânea entre os diferentes participantes da cadeia produtiva (CNI; 2016).

No início dos anos 90 ocorreram as primeiras tentativas de aproximação de automação e produção enxuta, sendo chamada de *Lean Automation*. Atualmente existem novas áreas para *Lean Automation* devido ao potencial das tecnologias da indústria 4.0 (KOLBERG & ZUHLKE, 2015). A produção enxuta ou *Lean Manufacturing* foi fundada por Ono na Toyota, descrevendo como a produção e processos devem ser organizados para alcançar menor tempo de espera, com mínimo custo e máxima qualidade (ONO, 1988). A fusão destes conceitos, produção enxuta e transformação digital, aprofundam ainda mais as relações entre produção e eficiência com foco na competitividade.

As tecnologias habilitadoras exercem função fundamental dentro da indústria do futuro. Muitas delas foram desenvolvidas no século XX, mas apenas agora, na quarta revolução industrial, tem desempenhado papéis relevantes, trabalhando como catalisadores que permitem soluções flexíveis e individualizadas. Dentre elas estão: internet das coisas, *big data*, robótica avançada, impressão 3D e inteligência artificial (DELOITTE, 2014).

De acordo com o relatório Competividade Brasil de 2018 da Confederação Nacional da Indústria (CNI) o Brasil ocupa a 17ª posição geral dentre 18 países

avaliados. Fator importante para este resultado está no fraco desempenho na variável compra governamental de produtos de tecnologia avançada, investimentos em Pesquisa e Desenvolvimento e a baixa capacidade de inovar nas empresas e por consequência baixa utilização das tecnologias habilitadoras.

Neste cenário o desenvolvimento da indústria 4.0 no Brasil envolve soluções complexas e por vezes de investimento elevado. Desta forma, a implementação dos sistemas ciberfísicos tendem a ocorrer primeiro nas grandes indústrias do que nas indústrias de pequeno e médio porte (PME). Schroder (2017) afirma que:

Apesar do grande potencial econômico da indústria 4.0 nas PMEs, estas ainda permanecem cautelosas. Por exemplo, 5% das PMEs estão completamente conectadas e um terço delas estão tomando os primeiros passos nesta direção ou pelos tem planos concretos para fazerem isto.

Estes fatos expõe a necessidade da criação de soluções acessíveis, como a ferramenta a ser desenvolvida durante este trabalho, que permitam a aquisição de dados e aumento da eficiência dos processos produtivos elevando o nível de maturidade das indústrias brasileiras de pequeno e médio porte em relação às novas tecnologias, assim como avanço do grau de competitividade e redução de custos.

## 1.2 DELIMITAÇÃO DO TEMA

Dentre as tecnologias habilitadoras presentes na indústria 4.0, uma das mais representativas é a Internet das Coisas *ou Internet of Things (IoT)*. A *IoT* realiza a conexão entre diferentes tipos de equipamentos e redes de comunicação para armazenamento e análise de dados. Estes equipamentos possuem diferentes tipos de sensores que coletam as informações e permitem o monitoramento em tempo real da situação do aparelho, como: temperatura, pressão, ruído, consumo de energia e índices de qualidade (AMAZONAS, 2010).

A princípio a Internet das Coisas foi desenvolvida para aplicações residenciais dentro do conceito de *smart home* conectando eletrodomésticos, permitindo assim o monitoramento e controle de funcionamento a distância, através de um *smartphone*, por exemplo. Atualmente a *IoT* possui também vasta aplicação nas indústrias, não somente para coleta de dados, mas também para comunicação máquina-máquina (M2M). Este tipo de interação otimiza os processos de produção já que a comunicação

e tomadas de decisão são realizadas de forma direta e sem interferência humana. Outras aplicações dentro da indústria e manufatura são economia de energia, controle de poluição, monitoramento de ciclo de vida dos produtos, rastreamentos dos produtos na cadeia produtiva e controle de processos de produção (CASAGRAS, 2008).

Neste estudo será desenvolvido uma solução, com base no conceito de Internet das Coisas e Computação em Nuvem, para coleta de dados de uma linha de produção em um processo de extrusão de uma indústria de plásticos da Região Metropolitana de Curitiba. Estes dados estarão relacionados a eficiência produtiva da linha de manufatura em estudo, através das informações de consumo de energia, tempo de *setup*, quantidades de metros produzidos, comparação entre velocidade real e velocidade ideal da máquina e quantidade de rejeitos durante tempo de *setup* e produção. O objetivo final é propor um ambiente para análise das informações e tomada de decisões com foco nas melhorias de processo e sustentabilidade do empreendimento.

### 1.3 PROBLEMAS E PREMISAS

Uma pesquisa realizada pela CNI (BRASIL, 2016b) considerando as 10 principais tecnologias digitais (impressão 3D, *IoT*, *big data*, *cloud computing*, e outras) revela que 42% das empresas entrevistadas não identificaram quais tecnologias tem maior potencial para desenvolver a competitividade da indústria. Dentre as PMEs o desconhecimento ainda é maior, sendo que 54% não souberam responder à questão.

No *ranking* dos setores que utilizam pelo menos uma das 10 tecnologias digitais listadas o de produtos e materiais plásticos conta com pouco menos da metade das empresas (49%). Se comparado com o setor de equipamentos de informática, produtos eletrônicos e ópticos o índice sobe para 61% (BRASIL, 2016b).

Em relação aos benefícios da adoção das tecnologias digitais, 54% dos empresários esperam redução de custos e 50% deles o aumento de produtividade. Entretanto, 66% dos entrevistados acreditam que o custo de implantação é maior barreira, em segundo lugar tem-se a falta de clareza sobre a definição de retorno do investimento e a estrutura e cultura da empresa, com 26% e 24% respectivamente (BRASIL, 2016b).

Diante deste cenário, é possível desenvolver uma ferramenta simples para aquisição de dados e a avaliação dos pontos de melhoria do processo com o foco nos custos e produtividade?

## 1.4 OBJETIVOS

### 1.4.1 Objetivo Geral

Desenvolver uma ferramenta de fácil implementação para aquisição de informações referente a eficiência produtiva de uma linha de produção em uma indústria de plásticos da região metropolitana de Curitiba, frente às soluções já existentes no mercado.

### 1.4.2 Objetivos Específicos

- a. Revisão da bibliografia;
- b. Aplicar a ferramenta a uma linha de produção;
- c. Determinar a eficiência produtiva, dentro de um intervalo de tempo, comparando a quantidade ideal e quantidade real de metros produzidos;
- d. Determinar a relação de consumo de energia e produção;
- e. Criar uma interface de comunicação para apresentação dos dados e auxílio na tomada de decisões.

## 1.5 JUSTIFICATIVA

De modo geral a implantação dos Sistemas Ciberfísicos (CPS) criam a expectativa do aumento de produtividade e desta forma ocorre a redução dos estoques, otimização da logística e redução dos custos de manutenção. Entre outros



benefícios ainda estão o aumento da qualidade do produto e a customização em massa (DELOITTE, 2014).

Segundo a pesquisa realizada pela *American Society for Quality (ASQ)*, em 2014, 82% das organizações que implementaram a manufatura inteligente experimentaram aumento na eficiência e 49% obtiveram menos produtos com defeitos. Outro aspecto importante são as potenciais reduções dos custos de produção, como demonstrado na TABELA 1:

TABELA 1 – AVALIAÇÃO DOS POTENCIAIS BENEFÍCIOS

<b>Tipo de Custo</b>	<b>Valor total</b>
Custo de inventário	-30% a -40%
Custos de produção	-10% a -20%
Custos logísticos	-10% a -20%
Custos complexos	-60% a -70%
Custos de qualidade	-20% a -30%

FONTE: SCHRODER (2017).

Embora a produção enxuta procure estabelecer um fluxo único e contínuo com máximo variedade de produtos o conceito não é adequado para produção de itens customizados. Desta forma, a integração entre CPS e estações de trabalho (*lean automation*) alteram os tempos de ciclo e encontram a relação ideal entre a utilização máxima de cada estação de trabalho e fluxo contínuo de produtos, criando uma produção dinâmica que automaticamente adota os programas de produção no momento (KOLBERG & ZUHLKE, 2015). A criação deste novo ambiente é essencial para criar um sistema de produção desacoplado do mercado, que possuem altas taxas de variação de demandas e tipos de produto. E este é um dos principais atributos da transformação digital: atender níveis de demanda com elevado índice de customização, mas mantendo os níveis de competitividade e sustentabilidade do negócio (ERLACH, 2013).

Neste contexto, a internet das coisas ou *IoT*, permite controle dos processos através de uma interface humano-máquina. Ao coletar dados sobre como os produtos se comportam as empresas podem prever futuros comportamentos e melhorar experiências do consumidor além de otimizar o funcionamento de máquinas e outros elementos presentes na linha de produção (UDEN, 2017).

Além disso a *IoT* é capaz de mapear todos os processos simultaneamente criando um espaço para as tomadas de decisões, a fim de atender as alterações de

demanda com máximo de eficiência, ou seja, menor consumo de insumos (matéria prima e energia) e em menor tempo. De modo complementar, a internet das coisas promove a transparência das ações correntes no chão de fábrica minimizando os desperdícios. Isto ocorre, pois, esta tecnologia encurta o tempo entre a ocorrência de problemas e a tomada de decisões eficientes, relacionadas, por exemplo, ao sequenciamento de produção e gerenciamento de manutenções (SHROUF, 2014).

Assim, a relevância deste trabalho está em aplicar estes conceitos ao monitoramento dos elementos que compõem os processos produtivos, habilitando os tomadores de decisão a melhorar a performance e a sustentabilidade da indústria em análise.

## 1.6 PROCEDIMENTOS METODOLÓGICOS

Este trabalho será dividido em quatro etapas: a primeira será uma revisão literária sobre os pontos de partida para elaboração da pesquisa (indústria 4.0), a segunda em um aprofundamento na pesquisa e desenvolvimento do hardware da ferramenta a ser aplicada, a terceira se refere à criação da interface de comunicação e a última na elaboração de relatório com base na análise dos dados colhidos na linha de produção.

A primeira parte trará os conceitos da indústria 4.0, baseando-se no relatório divulgado pela consultoria Deloitte (2014), que estabelece quais são seus principais pilares e o ímpeto do uso das tecnologias com crescimento exponencial, como computação em nuvem, Big Data e Internet das Coisas.

Em seguida será realizada um aprofundamento na pesquisa para desenvolvimento da ferramenta através da utilização de controladores lógicos programáveis e sua integração com banco de dados e sensores. A aplicação será em uma indústria de plásticos que produz peças para automóveis. Os dados serão fornecidos pelo painel de controle das máquinas envolvidas na linha de produção, como Extrusoras e máquinas de controle de velocidade.

A terceira parte será a criação da interface de comunicação e disposição dos dados criando um espaço para análise das informações e tomada de decisões. Os dados serão capturados dentro de um intervalo específico de produção.

E por último a elaboração de um relatório final mostrando a eficiência do processo em termos produtividade, material não aproveitado e consumo de energia,

durante o intervalo de aquisição de dados. A análise destas informações também permitirá a elaboração de possíveis ações corretivas, preventivas e preditivas.

## 1.7 ESTRUTURA DO TRABALHO

No capítulo 1, Introdução, o objetivo deste trabalho é discutido, pontuando os problemas e as premissas para o desenvolvimento do estudo, além das justificativas para realizar uma pesquisa sobre o tema e sobre os métodos para obtenção dos resultados esperados.

No capítulo 2 haverá uma revisão bibliográfica apresentando aspectos específicos sobre a Indústria 4.0, Produção Enxuta e Controladores Lógicos Programáveis.

O capítulo 3 consistirá na metodologia e etapas para implementação da ferramenta para aquisição dos dados de consumo de energia, eficiência e rejeitos e a elaboração da interface de comunicação com o usuário.

O capítulo 4 apresentará os resultados informando a eficiência, índice de rejeitos e consumo de energia da linha de produção com base nas informações obtidas.

No capítulo 5, a conclusão sobre os resultados e possíveis soluções para as situações identificadas.

E por fim a inserção de apêndices, anexos e referências bibliográficas.

## 2 REFERENCIAL TEÓRICO

### 2.1 INDÚSTRIA 4.0

A consultoria Deloitte (2014) define os quatro principais pilares da indústria 4.0 como sendo a comunicação vertical, a integração horizontal, o aprimoramento dos

processos e produtos através da engenharia e aceleração pelo crescimento exponencial de tecnologias disruptivas.

A comunicação vertical faz uso dos sistemas ciberfísicos para habilitar as indústrias a agirem rapidamente sobre flutuações de demanda, níveis de estoques e/ou perdas. Além disso, permite a localização do produto em qualquer etapa do processo e a identificação de desvios na qualidade reduzindo o desperdício. O uso destas tecnologias também possibilita o gerenciamento de manutenção evitando paradas de máquinas e robôs. A integração horizontal refere-se à geração de uma nova cadeia de valor envolvendo as áreas de desenvolvimento do produto, compras, produção, marketing, vendas e pós-venda, incluindo os fluxos logísticos de entrada e saída de materiais. A integração destes processos permite transparência nas ações a partir do fornecedor da matéria prima, durante o processamento do produto até entrega no cliente final. Todas estas etapas permitem a flexibilização dos processos e criação de uma espécie de memória do produto (DELOITTE, 2014).

O terceiro eixo, aprimoramento através da engenharia, atua na adaptação dos sistemas de produção de acordo com novos produtos e seus ciclos de vida. Este eixo é fundamental para indústria do futuro, já que uma das tendências é a manufatura de produtos específicos e individualizados para cada consumidor. Por último as tecnologias disruptivas ou habilitadoras. Estas tecnologias funcionam como catalisadores acelerando o processo de transformação digital, soluções individualizadas e flexibilidade. Dentre as principais tecnologias habilitadoras estão: internet das coisas, *big data*, robótica colaborativa, impressão 3D (manufatura aditiva) e inteligência artificial (DELOITTE, 2014).

### 2.1.1 Computação em Nuvem

A Computação em Nuvem ou *Cloud Computing* alterou profundamente os padrões de infraestrutura para armazenamento de informações, simplificando a fase de implementação de hardwares ou *data centers*. Além disso, a customização do ambiente em nuvem permite o aumento ou redução de espaço para armazenamento de dados de acordo com a demanda do usuário. Possíveis custos com a instalação de um *data center*, (sobredimensionamento) do equipamento ou a instalação de novos hardwares no futuro são eliminados (TAURION, 2009).

Segundo o *National Institute of Standard and Technology* dos Estados Unidos, os serviços em nuvem são divididos em três modelos: *Software as a Service (SaaS)*, *Platform as a Service (PaaS)* e *Infrastructure as a Service (IaaS)*. No SaaS é a camada mais alta, sendo responsável por disponibilizar aplicações completas ao usuário final, através de um *web browser* ou algum programa específico. A camada intermediária é o PaaS, sendo este o ambiente onde o desenvolvedor pode criar aplicações em nuvem para o usuário final. E a camada mais inferior é o IaaS, responsável por armazenar, processar e promover a comunicação entre os diferentes tipos de software, ou seja, controlar de forma geral os sistemas de operação, armazenamento e aplicações instaladas. O NIST ainda define que os ambientes em computação em nuvem podem ser de quatro tipos. Ambientes públicos, com gerenciamento de uma entidade privada, acadêmica ou governamental, mas com o acesso para o público em geral. Ambientes privados, com o acesso restrito das informações somente a usuários ligados a entidade. Ambientes comunitários, onde os usuários possuem interesse em comum em determinadas informações, mas pertencem a diferentes organizações. E os ambientes híbridos, onde é possível combinar os ambientes acima em um único sistema em nuvem (NIST, 2011).

Dentro da quarta revolução industrial, as vantagens da nuvem são ainda maiores. A adoção deste sistema realiza a integração horizontal de CPS, que produzem uma quantidade enorme de dados, estes precisam ser estocados e analisados. Estas análises podem ser acessadas em qualquer lugar em qualquer momento, já que as informações são processadas em tempo real, e os resultados obtidos através de um smartphone, por exemplo, realizando a descentralização da cadeia de criação de valor (CAPGMEMINI, 2014). A popularização dos recursos da computação em nuvem nos últimos anos permite que esta tecnologia seja acessível para todas as empresas, incluindo as PMEs, que possuem um potencial grande a ser explorado. A agilidade e customização entregues pela nuvem aumentam competitividade frente aos concorrentes e a migração para o *Cloud Computing* dentro das pequenas e medias empresas é rápida, pois estas não possuem instalações de TI robustas ou muito antigas como vistas em empresas de grande porte (BATEMAN, 2018).

De acordo com relatório técnico de Armbrust et al (2009) a Universidade da Califórnia em Berkeley ainda restam alguns obstáculos para adoção da computação em nuvem como a segurança das informações contra hackers e ataques cibernéticos,

disponibilidade de acesso e continuidade do sistema e a interoperabilidade entre os provedores de computação em nuvem para que seja possível o acesso das informações de diferentes plataformas pelo usuário. No Brasil, a nuvem se tornou uma alternativa para redução de custos e confiança nos serviços tem aumentado, principalmente devido a evolução dos sistemas de segurança dos provedores. Entretanto, a segurança dos dados também é de responsabilidade do usuário. A consultoria Gartner previu que em 2016 95% dos incidentes de segurança envolvendo a nuvem foram culpa do usuário e ou administrador, configurando um novo obstáculo: a contratação de profissionais especializados em segurança da informação para gestão da infraestrutura em sistemas em nuvem a fim de evitar a perda de dados (MOREIRA, 2016).

### 2.1.2 *Big Data Analytics*

*Big Data Analytics* compõe um espaço para armazenamento e análise de uma grande quantidade de informações, sejam elas de origem humana ou artificiais. Através do processamento dos dados provenientes destas fontes é possível traçar padrões de comportamento e descobrir relações ainda desconhecidas, como preferências dos clientes, tendências de mercado ou otimização dos processos de produção evitando problemas de qualidade ou mal desempenho de máquinas e robôs. Desta forma o *Big Data* auxilia os profissionais especialistas a analisar um grande volume de dados não estruturados ou semiestruturados, permitindo as organizações tomarem decisões estratégicas baseada em modelos preditivos, estatísticas e análises avançadas (ROUSE, 2017).

*Big Data* é caracterizado pela grande variedade e grande volume de dados, além da velocidade em que estes devem ser processados, caracterizando os três “Vs” do *Big Data*: volume, variedade e velocidade. Estas características demandam grandes espaços de armazenamento e um alto poder computacional para análises rápidas. Considerando este aspecto, as nuvens públicas se tornaram uma alternativa econômica viável para a implementação de grandes projetos de *Big Data*. Isto porque a nuvem possui elasticidade podendo aumentar a capacidade de servidores e permite que as organizações paguem pela quantidade exata de serviços, podendo desabilitá-los ao fim do projeto. Outro fator importante para o gerenciamento do *Big Data* é a mão de obra especializada. É esperado que estes analistas sejam capazes de

entender e formular requisições adequadas a fim de direcionar os projetos que atendam às necessidades das organizações coletando informações relevantes para tomada de decisões. Outra função que aparece neste contexto são os cientistas de dados, que utilizam as ferramentas disponibilizadas, para mineração dos dados e análise técnica dos resultados obtidos (ROUSE, 2017).

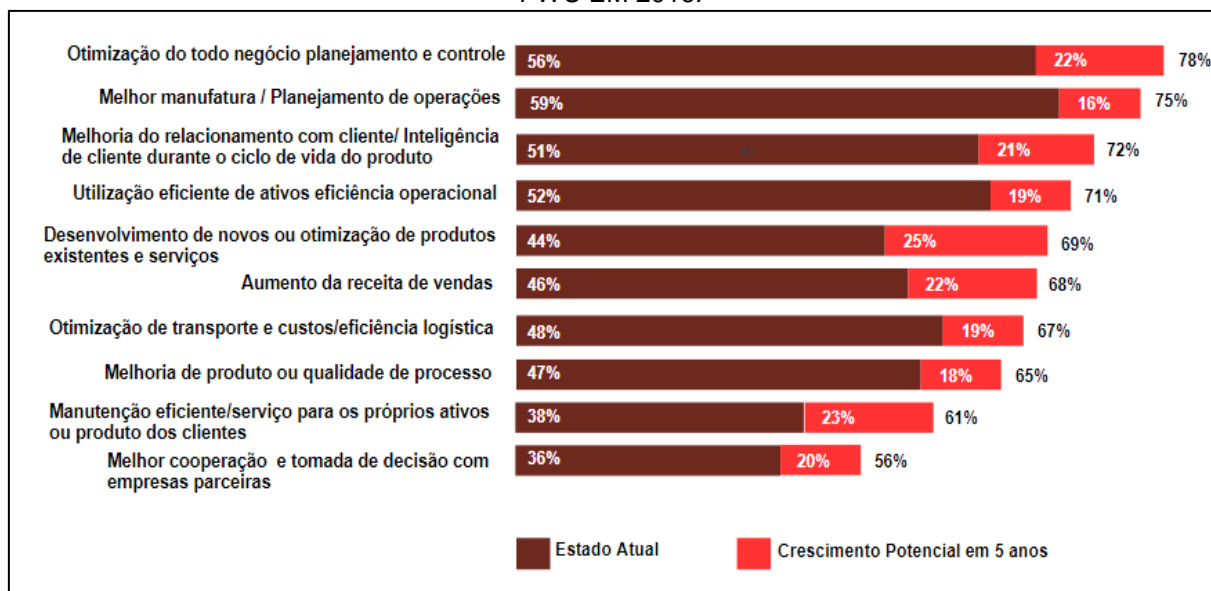
Na indústria a captura e análise de informações está no centro da transformação digital, assim a seleção das fontes de dados e o tratamento destas informações estão diretamente ligadas aos resultados e a competitividade das organizações. Dentro das indústrias haverá um fluxo contínuo de dados emitidos através de sensores e sistemas de monitoramento, em uma atmosfera conhecida como *Industrial Internet of Things* ou *IloT*. É neste campo que o *Big Data* tem uma vasta aplicação, pois os milhões de dados recebidos através dos pontos de monitoramento são rapidamente processados permitindo que as próprias máquinas tenham capacidade de tomar decisões, realizar transferência de informações com outras máquinas, em um processo chamado de comunicação Máquina a Máquina (*M2M*) reduzindo a interferência humana no chão de fábrica e nos custos, provocando o aumento da produtividade (BOYES, 2018).

Além de fontes internas de dados é possível explorar as informações capturadas a partir das tendências de consumo, criando produtos que atendam diretamente as expectativas do cliente. A indústria 4.0 não irá só modificar as relações dentro das indústrias, mas também irá criar um ambiente para interpretação dos dados enviados pelo consumidor, permitindo a customização em massa, o melhor gerenciamento do ciclo de vida dos produtos e os índices de satisfação com os serviços prestados (GEISSBAUER, 2016).

Segundo a consultoria PWC, para obterem sucesso as empresas devem utilizar os dados de maneira preditiva, observando o mercado, os clientes e o próprio negócio. O gráfico apresentado na FIGURA 1, originado pela pesquisa da consultoria em 2016, mostra o estado atual e o potencial de crescimento nos próximos cinco anos da utilização do *Big Data Analytics*. Nas primeiras posições estão a otimização de todo negócio (planejamento e controle), melhoria da manufatura e planejamento de operações e o desenvolvimento das relações com o cliente ao longo do ciclo de vida do produto. Destaca-se também o desenvolvimento de novos ou otimização de produtos, com o crescimento previsto de 25%, o aumento de receitas, com 22% e a

possível de cooperação com empresas parceiras, representando 20%, além dos avanços nas áreas de logística, qualidade e manutenção.

FIGURA 1 - UTILIZAÇÃO DO *BIG DATA* POR EMPRESAS ENTREVISTADAS PELA CONSULTORIA PWC EM 2016.



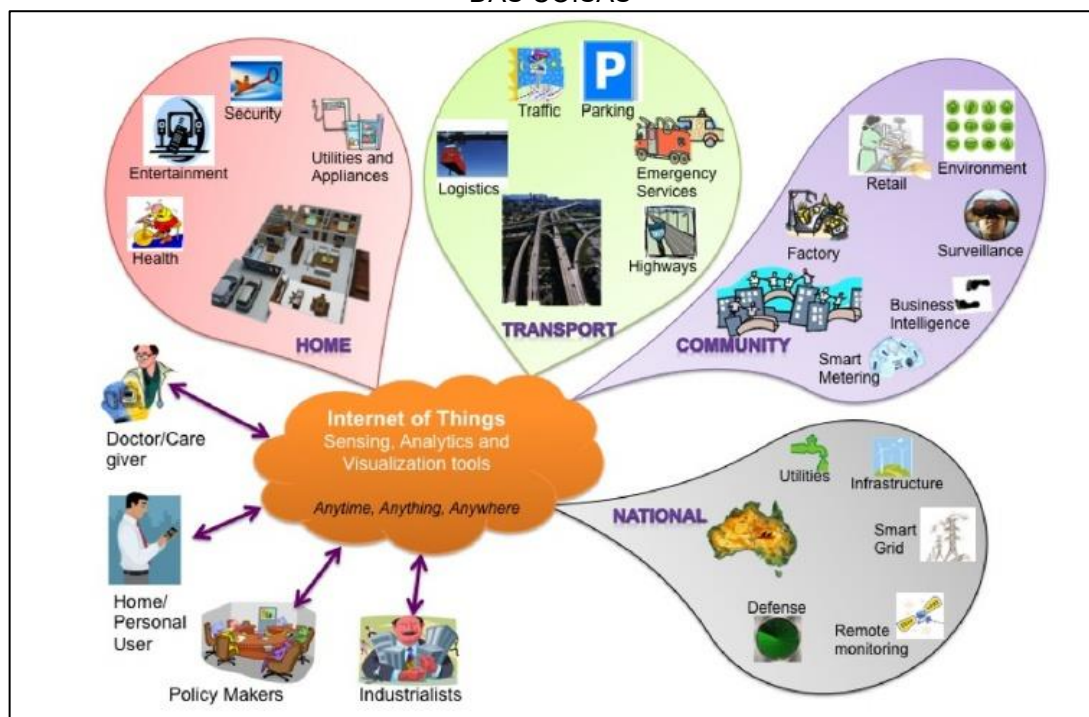
FONTE:GEISSBAUER (2016)

### 2.1.3 Internet das Coisas

A internet das coisas ou *internet of things* é conceito que define a conectividade dos objetos (coisas) ao nosso redor. Esta conectividade permite o controle destes objetos bem como o seu monitoramento, originando um fluxo intenso de informações entre os aparelhos e os usuários. As redes de comunicação podem ser domésticas, com a conectividade dos aparelhos eletrodomésticos, redes industriais e até mesmo nas cidades, com a implementação das redes de comunicação para veículos autônomos, gerenciamento de tráfego e de distribuição e consumo de energia, conforme apresentado na FIGURA 2. De acordo com Gubbi et al (2013) a internet das coisas é a interconexão de dispositivos, com a capacidade sensorial, de atuação e de compartilhar informações dentro de uma atmosfera comum. Isto será alcançado através de sensoriamento abrangente, análise de dados e a computação em nuvem como a criadora desta atmosfera, capaz de abrigar diferentes fontes de informação e manipulação dos dados.



FIGURA 2 - REDES DE COMUNICAÇÃO E INTERFACE COM USUÁRIO ATRAVÉS DA "INTERNET DAS COISAS"



FONTE:GUBBI (2013)

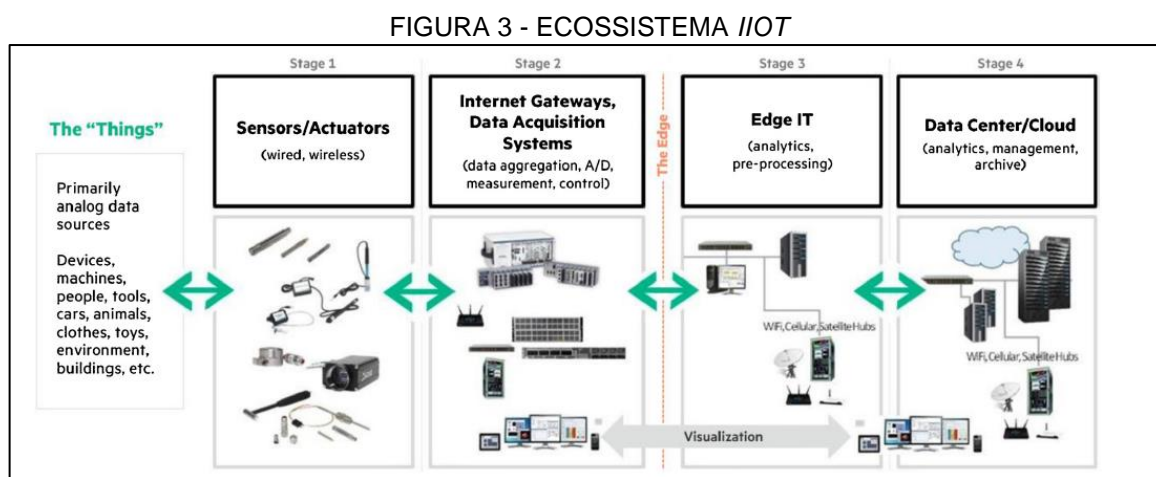
A computação em nuvem também encontra espaço nesta aplicação, pois oferece ótima infraestrutura para armazenamento dos dados e interface com usuário através de acesso remoto, além das vantagens econômicas (facilidade de implementação e baixo custo de manutenção), criando um ecossistema chamado de *Cloud of Things*. Os novos dispositivos que vão sendo adicionados a rede encontram facilidade de integração e suporte na nuvem para envio e recebimento de informações, devido a elasticidade de armazenamento. Além disso, a computação em nuvem oferece recursos para que profissionais desenvolvam aplicações e softwares com foco na análise e visualização dos dados, sendo este um dos pilares para o desenvolvimento do *machine learning* e a inteligência artificial. Entretanto desenvolver aplicações na nuvem em linguagens de baixo nível pode ser um pouco complexo, assim as aplicações em *IoT* devem ser capazes de expressar facilmente análises, de maneira transparente e lógica, receber fluxo de dados diretamente dos sensores ou de banco de dados e por fim em caso de eventos importantes divulgar estas informações através das plataformas de visualização (GUBBI, 2013).

Aplicações da internet das coisas a indústria originou um novo termo: *IIoT* ou *Industrial Internet of Things*. Estes dispositivos, como sensores e/ou atuadores, são capazes de transmitir informações automaticamente, comunicar seu *status* a outros

dispositivos ao seu redor, além de permitir seu gerenciamento e análise contínua do fluxo de dados, compondo um sistema ciberfísico. Desta forma a *IIoT* foi concebida para fins diferentes das aplicações convencionais, pretendendo conectar motores, máquinas, redes de abastecimento, sejam elas de energia, gás ou água, e sensores a nuvem através da internet. O objetivo é criar uma rede descentralizada, onde seja possível monitorar, coletar, trocar e analisar dados e tomar decisões sem a interferência direta humana, promovendo a qualidade dos serviços e produtos entregues (BOYES, 2018).

Um modelo de ecossistema de *IIoT* é exibido na

FIGURA 3. As “coisas” são representadas pelas fontes analógicas de informação, como pessoas, ferramentas, carros, máquinas e outros. Os sensores são responsáveis por captar estas informações e transmiti-las às unidades de aquisição de dados (controladores lógicos programáveis, barramentos, roteadores, etc.). E então há as fases de processamento, análise das informações e armazenamento em servidores ou em nuvem (BOYES, 2018).



FONTE:BOYES (2018)

A consultoria Brain e Company estima que a *IIoT* irá movimentar mais de 300 bilhões de dólares até 2020 (Rouse, 2018), mas ainda restam alguns desafios para implementação da Internet das Coisas na Indústria. Entre eles está o desafio da segurança das informações, tanto na transmissão, garantindo a confiabilidade das informações, quanto no armazenamento, impedindo que informações importantes sejam perdidas ou vazadas. É necessário também não subestimar a fase implementação da rede de *IoT*, contando com profissionais qualificados para mapear

e quantificar as informações relevantes para corporação, evitando o sobredimensionamento do sistema e ainda avaliar quais os riscos e a complexidade de integração com as tecnologias já existentes dentro da indústria. Outro fator importante a ser analisado é a possibilidade de integração com outras redes de *IoT* presentes na cadeia de valor, como clientes, fornecedores, sistemas de abastecimento de energia, e ambientes externos como construções, meios de transporte e até mesmo as cidades (BOYES, 2018).

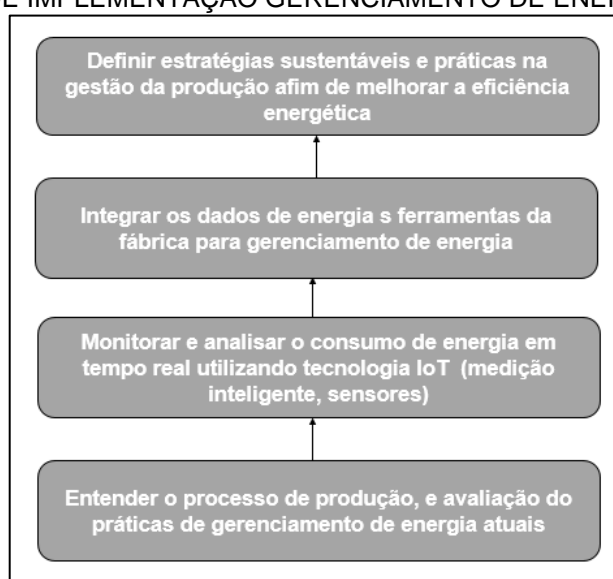
#### 2.1.4 *Smart Energy Metering*

A gestão de energia é um dos pilares da Indústria 4.0, pois a fábrica inteligente busca minimizar os desperdícios, aumentar a eficiência, além de reduzir impactos ambientais. Estes são fatores importantes na manutenção da competitividade das indústrias, onde utiliza-se o mínimo de recursos para trazer o máximo de valor aos produtos. Assim, a indústria 4.0 necessita do aprimoramento contínuo das práticas de gestão de energia e produção através da integração com os sistemas ciberfísicos. Segundo a ABESCO (Associação Brasileira das Empresas de Serviços de Conservação de Energia) a Indústria é o setor com maior capacidade de gerar riquezas, mas também com maior capacidade para desperdiçar energia através dos sistemas de ventilação, ar comprimido, ar condicionado e aquecimento (SANTOS, 2018). Atualmente, as indústrias representam 36% do consumo de energia no Brasil e há um grande potencial a ser explorado em busca da sustentabilidade, eficiência e diversificação da matriz energética e redução dos custos (EPE, 2017).

A eficiência energética, entretanto, requer atenção ao consumo das linhas de produção e máquinas e nesse ambiente os medidores inteligentes ou *smart meters* transmitem informações em tempo real que auxiliam comportamento sustentável e a redução de custos. Por exemplo, é possível coordenar quais os melhores horários para produção evitando os horários de pico, obter informações referente aos índices de produtividade versus consumo de energia, e identificar comportamento irregular de máquinas e planejar ações corretivas ou preventivas a fim de evitar a sobrecarga do sistema e perdas (SHROUF, 2014). Os medidores inteligentes serão parte da rede de comunicação no chão de fábrica, sendo capazes de se comunicar com outros dispositivos, enviando e recebendo comandos, reforçando assim a importância do *IoT* para o alcance do consumo inteligente.

É possível realizar o gerenciamento de energia baseado na plataforma e recursos de *IoT* em quatro etapas. A primeira fase é entender o processo de produção e quais os mecanismos já utilizados para controle de energia. A segunda etapa constitui no monitoramento e análise do consumo através da utilização dos medidores inteligentes. Nesta fase é necessário identificar quais máquinas e parâmetros (potência ativa, potência reativa, etc.) serão monitorados. A terceira fase é a integração dos dados a softwares de apoio e outras ferramentas que permitam as ações dos *decision makers*. E por fim, a definição de estratégias e rotinas para eficiência energética, que no futuro poderão ser incorporadas às atividades de gestão da produção (SHROUF, 2014). Este modelo foi desenvolvido pelo departamento Politécnico de Milão e a Universidade Politécnica de Madrid e é exibido na FIGURA 4. A primeira etapa é entender os processos gerenciais atuais e práticas voltadas consumo de energia. Em seguida, o acompanhamento em tempo real através de redes de *IoT* (sensores inteligentes) irá realizar um diagnóstico dos componentes presentes na fábrica e coletar dados para análise. Estes dados por sua vez devem ser integrados a ferramentas de gestão já utilizados e servir como base para gestão de produção com foco em eficiência energética. De maneira geral as ferramentas modernas de gerenciamento de energia devem ser capazes de processar um grande volume de dados, em diferentes pontos do processo de produção, garantindo a sustentabilidade e eficiência e a integração com diferentes fontes de energia.

FIGURA 4 - MODELO DE IMPLEMENTAÇÃO GERENCIAMENTO DE ENERGIA COM BASE EM IOT



FONTE: SHROUF (2014)

## 2.2 LEAN MANUFACTURING

O termo *Lean Manufacturing* ou produção enxuta busca a integração entre os processos e informações tendo como princípio a redução de desperdícios. Aliado a transformação digital a produção enxuta pode alcançar resultados mais expressivos, pois a transparência dos processos e a reatividade aos desvios e falhas são maiores e mais rápidos, aumentando a confiabilidade das operações. O contrário também é válido, em indústrias onde os conceitos de produção *lean* já são aplicados a adaptação as novas tecnologias é mais ágil e eficiente. Em ambientes não estruturados a implementação das metodologias de produção em Indústria 4.0 podem não atingir os resultados esperados e não agregar valor ao processo produtivo. É possível dizer que o *Lean Manufacturing* e a quarta revolução industrial andam juntos (SILVA, 2017).

Os conceitos da produção enxuta tiveram seu princípio no *Toyota Production System* (TPS) criado por Taiichi Ohno após a Segunda Guerra Mundial no Japão (Liker, 2004). Naquele momento a Toyota e o Japão passavam por uma grave crise, sofrendo as consequências da guerra. A filosofia criada por Ohno tinha como missão reestabelecer a prosperidade do negócio, através de métodos de redução de desperdício e melhoria contínua. De acordo com Cruz (2013), há cinco princípios chaves do *Lean Manufacturing*:

- **Identificar o Valor:** Este valor está referenciado na perspectiva do cliente e definido de acordo com sua necessidade. Para empresas é necessário descobrir esta necessidade e atendê-la, cobrando o valor que o cliente identifica como justo pagar.
- **Identificar a Cadeia de Valor:** A cadeia de valor é composta por três tipos de atividades. Atividades que acrescentam valor, atividades que não acrescentam valor, entretanto são importantes para produção, como manutenção e qualidade. E atividades que não possuem qualquer valor associado, e estas devem ser eliminadas pois estas geram desperdícios. Mapear a cadeia de valor, portanto, pode identificar atividades do terceiro tipo e aumentar a produtividade.
- **Estabelecer o Fluxo Contínuo:** Definir um fluxo sem paradas e inventários. Esta ação reduz o tempo de fabricação dos produtos, processamento dos pedidos e inventários, aumentando a competitividade da empresa e reduzindo de custos.

- **Produção *Pull*:** Produzir de acordo com a solicitação do cliente, e não “empurrar” os produtos para o cliente. Isto reduz possíveis custos com inventários e aumenta disponibilidade de mão de obra.
- **Obter a perfeição:** Focar na perfeição dos produtos e minimizar as perdas, através da melhoria contínua ou *Kaizen*.

Dentro destes conceitos existem diversas ferramentas e técnicas associadas como o próprio *Kaizen*, o *Single Minute Exchange o Die (SMED)*, *5s*, *Six Sigma*, *Value Stream Mapping* e outros, sempre com o foco de reduzir os desperdícios. Para Ohno (1988) os desperdícios são atividades que despendem recursos, mas não aumentam o valor do produto vendido ao cliente e podem ser divididos em sete tipos: desperdícios de excesso de produção, movimentação, transportes, esperas, reprocessamento de produtos, inventários e defeitos. Em 1996, Womack e Jones, identificaram um oitavo desperdício e que se refere a subutilização das pessoas, tanto de mão obra quanto de ideais e melhorias a serem aplicadas no processo de produção (CRUZ, 2013).

Devido a sua simplicidade e grandes benefícios a teoria *Lean* se espalhou rapidamente, inclusive para indústrias fora do ramo automotivo. Entretanto, as novas concepções de mercado limitaram sua utilização. Observa-se uma crescente tendência para flexibilização de demandas, devido às fortes oscilações no mercado, e a produção de itens individuais e específicos, onde não é possível incorporar os princípios de fluxo contínuo de produção com sequências e tempos de operação fixos. (KOLBERG & ZUHLKE, 2015).

Como descrito anteriormente, a tecnologia exerce papel importante neste novo cenário, fortalecendo os princípios da produção enxuta, minimizando os custos e desperdício, permitindo total controle, flexibilização e transparência dos processos originando novo conceito: o *Lean Automation* (KOLBERG & ZUHLKE, 2015).

O fluxo contínuo de peças, por exemplo, pode ser aprimorado através da utilização de realidade aumentada, onde o operador possui informações precisas sobre o ciclo de produção realizando a gestão visual *just-in-time*. O operador também pode ser notificado rapidamente sobre a ocorrência de falhas e atuar sobre o problema. Nestes casos, utilizando dispositivos inteligentes conectados a rede, como *smart watches*, o operador recebe notificações da localização e o momento da falha,

descentralizando as informações. Silva (2017) traça um paralelo entre as tecnologias habilitadoras e como sua empregabilidade pode fornecer suporte a produção enxuta:

- **Identificar o Valor:** Conectar os sistemas entre o cliente e a indústria obtendo um feedback imediato do produto, e através do *Big Data Analytics* processar e analisar estas informações, aprimorando o ciclo de vida dos produtos.
- **Identificar a Cadeia de Valor:** Utilização de tecnologias para monitoramento dos operadores para definir operações simplificadas e o trabalho padronizado, a presença de robôs colaborativos integrados a linha de produção para auxiliar nas tarefas, e a documentação digital dos processos (setup, materiais e máquinas) com visualização em tempo real, são um dos exemplos de como é possível mapear a cadeia de valor de forma mais rápida e eficiente.
- **Estabelecer o Fluxo Contínuo:** Realizar a gestão em tempo real da sequência e tamanhos dos lotes em função da eficiência dos processos de produção, data de entrega e custo de armazenamento e o aprimoramento do layout através de simulações 3D considerando melhor sequência e rotas de produção.
- **Produção Pull:** Integrar a comunicação entre cliente e fornecedor (externo ou interno) sobre a necessidade real, mapeando todas as etapas da cadeia de valor. Outro aspecto interessante é a aplicação em rotinas de abastecimento em linha, para áreas de *picking* e *kitting*, informando necessidade de reabastecimento, quantidade de material ou eventuais falhas e necessidades emergenciais.
- **Obter a perfeição:** Utilização de sistemas para detecção de paradas, perdas de rendimento e cálculo em tempo real de indicadores de desempenho. Estes sistemas permitem o tratamento dos dados e aprendizagem dinâmica das máquinas, para prever futuras falhas e manter a qualidade dos produtos.

Desenvolver a indústria em sua quarta fase de evolução irá depender das estratégias de cada empresa, o mercado consumidor e a quantidade de recursos disponíveis para investimentos. Porém, outro passo importante para implementação

destas tecnologias é a adoção da cultura *Lean* nas organizações. Apesar da distância temporal que separam os dois eventos, ambos focam na sustentabilidade e eficiência dos processos, e a sua integração aceleram os resultados e promovem a melhoria contínua (SILVA, 2017).

### 2.3 CONTROLADORES LÓGICOS PROGRAMÁVEIS

Os controladores lógicos programáveis ou CLPs são amplamente aplicados em automação e podem ser considerados computadores industriais, substituindo os circuitos mais tradicionais a relés. Com decorrer do tempo as funções do CLP aumentaram oferecendo recursos além de comutação de contatos (relés) mas também de temporização, contagem, comparação, controle e processamento de sinais. Todas estas funções são controladas por um programa criado de acordo com as necessidades de aplicação do projeto. Esta é uma grande vantagem, pois aumenta a confiabilidade da lógica criada quando ocorre a transferência do programa de um CLP para outro, além disso o programa é editável e permite maior flexibilidade para alterações. Estas habilidades reduziram significativamente a quantidade de fios em projetos de automação industrial e minimizaram a ocorrência de erros em casos onde era necessário alterar a lógica do projeto inicial (SILVEIRA, 2003).

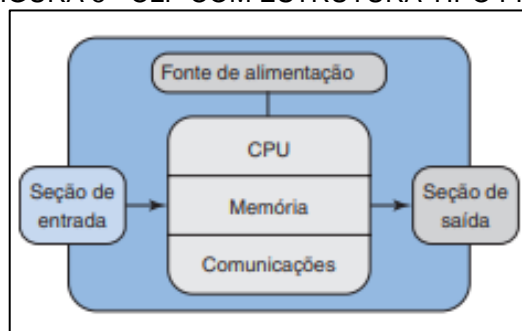
O primeiro CLP foi concebido em 1968 pela *Bed Ford Association* nos Estados Unidos da América. A empresa foi contratada para desenvolver um dispositivo que substituísse os relés e recebeu o nome de MODICON (*Modular Digital Controller*). Os CLPs são compostos por unidade central de processamento que recebe as informações através de uma entrada de dados, analógica ou digital. As informações são processadas por um programa armazenado em sua memória. O CLP devolve os resultados por uma unidade de saída. As unidades de entrada e saída podem estar conectadas a dispositivos de campo, como sensores ou relés, responsáveis por executar as ações. A linguagem de programação frequentemente utilizada é a Ladder, pois se aproxima de um esquema elétrico baseado em relés e armazena os dados na memória do CLP em forma de operações (SILVEIRA, 2003).

Em relação a forma de estruturar os módulos de entrada e saída há duas formas: fixas ou moduladas (PETRUZELLA, 2014). A estrutura fixa, apresentada na FIGURA 5, não possui unidade removível, assim o processador e os módulos são montados juntos e os terminais possuem quantidade limitada de entradas e saídas. A



maior vantagem desta configuração é o baixo custo, podendo ser acrescentado unidades fixas adicionais. Porém, há baixa flexibilidade e se alguma unidade apresentar defeito é necessário realizar a substituição de todo conjunto.

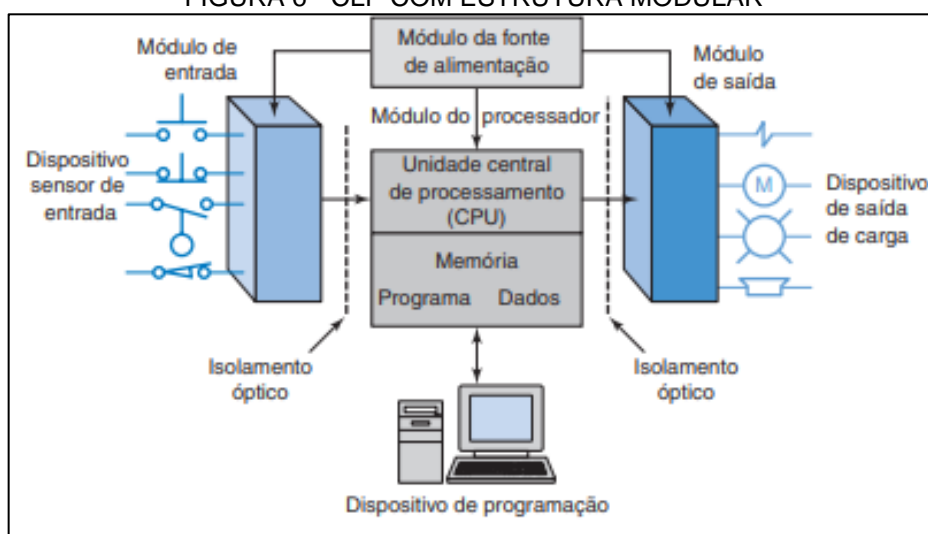
FIGURA 5 - CLP COM ESTRUTURA TIPO FIXA



FONTE: PETRUZELLA (2014)

Os CLPs modulados possuem unidades que podem ser conectadas ou desconectadas aumentando a flexibilidade, já que são possíveis misturar diferentes unidades de acordo com a disponibilidade do fabricante. Geralmente, os CLPs modulados são compostos pelo processador (CPU), módulos de entrada e saída, fonte de alimentação e a interface de para programação e monitoração, conforme apresentado na FIGURA 6. Os módulos de entrada e saída podem ser conectados direto ao CLP ou através de racks, que estabelecem uma conexão como uma série de contatos localizada na parte traseira. Neste caso, o CLP também é conectado ao rack e pode se comunicar com todos os módulos e a fonte (PETRUZELLA, 2014).

FIGURA 6 - CLP COM ESTRUTURA MODULAR



FONTE: PETRUZELLA (2014)

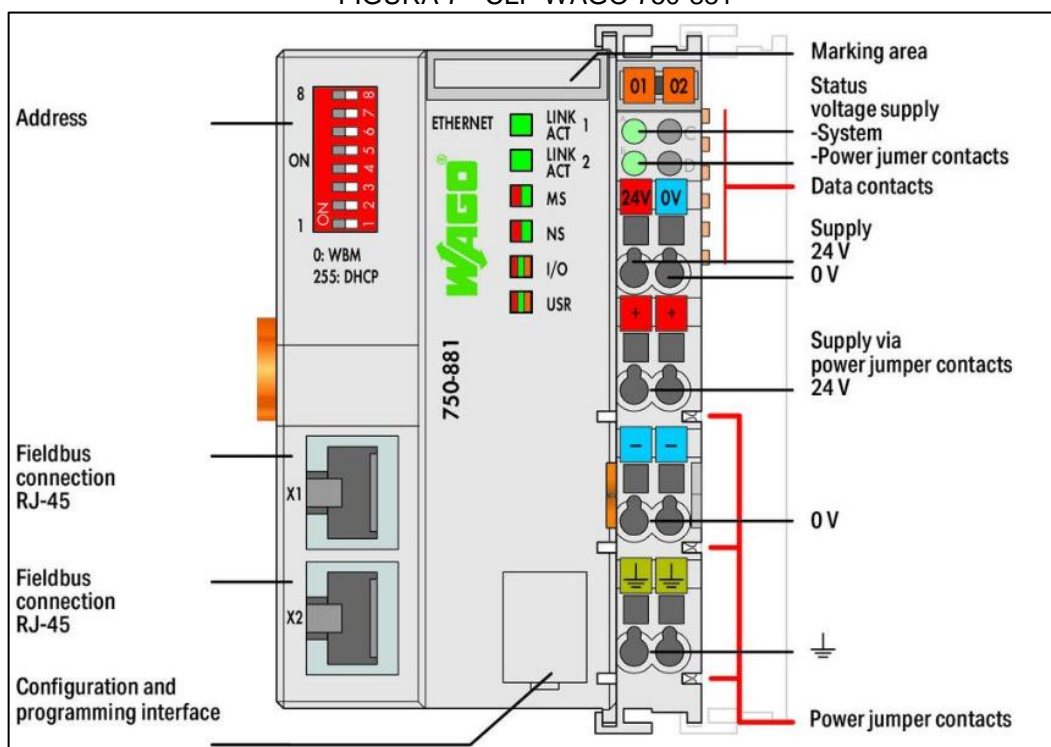
O dispositivo de programação tem a função desenvolver o programa em linguagem Ladder, criada para simplificar a programação utilizando elementos gráficos. Os dispositivos de programação mais comuns são computadores pessoais (PC) e se comunicam com o processador por links em série, paralelo ou Ethernet. O tamanho da memória da CPU, onde o dispositivo de programação irá inserir o programa, depende da aplicação e está relacionada a número de entradas e saídas, tamanho do programa, necessidade de programas de supervisão, coleta de dados e expansão futura. O programa por sua vez irá determinar os diferentes modos de operação do CLP. Entretanto, é possível definir um ciclo de funcionamento genérico e que se repete ao longo do funcionamento do CLP, que consiste em ler as entradas, executar o programa, diagnóstico e comunicação e por fim atualizar as saídas (PETRUZELLA, 2014).

Apesar de ter participado ativamente da terceira revolução industrial, onde ocorreu uma intensa automação dos processos e a aquisição dos robôs, os CLPs também encontram espaço na Indústria 4.0 pois são elementos essenciais nas redes industriais já existentes. Os controladores lógicos programáveis podem se comunicar com outros controladores, ou com qualquer outro equipamento e executar atividades de controle, coleta de dados e monitoramento de processos. Os CLPs mais modernos contam com conectividade 3G/4G e acesso wireless a internet. Esta modernização permite ao CLP incorporar as redes de *IoT* e enviar dados para armazenamento em nuvem (WAGO, 2017).

### 2.3.1 Controlador Lógico Programável Wago

A WAGO é uma multinacional alemã, fundada em 1951 na cidade de Minden. A empresa é especializada na produção de equipamentos eletrônicos e conectores elétricos. O CLP modular escolhido para desenvolvimento da aplicação foi o Wago 750-881 (FIGURA 7), com comunicação Ethernet. A programação pode ser feita através do software CODESYS (Wago, 2017). O software, bem como os módulos de entrada e saída e as unidades remotas serão descritos mais detalhadamente no capítulo 4, na etapa de desenvolvimento.

FIGURA 7 - CLP WAGO 750-881



FONTE: WAGO (2017)

## 2.4 BANCO DE DADOS

Banco de dados como um conjunto de dados que possuem alguma conexão entre si e quando organizados podem gerar informações específicas, como o cadastro de clientes de uma loja, os produtos produzidos por uma empresa ou os livros presentes em uma biblioteca. O sistema deve conter o mínimo de duplicação, permitindo que a informação desejada seja encontrada rapidamente, reduzindo o número de falhas durante a busca (OLIVEIRA, 1992).

Há muitas décadas atrás os arquivos eram organizados fisicamente, e o grande volume de informações impressas exigia um bom sistema de indexação para que se pudesse recuperar a informação quando se precisasse dela. Estes modelos antigos foram fundamentais para o desenvolvimento e experimentação dos primeiros bancos de dados nos anos de 1960. Os dois principais modelos deste período foram o CODASYL (*Comitte for Data Systems Language*), modelo em rede, e o modelo hierárquico IMS (*Information Management System*). Mais tarde, em 1970, Edgar Frank Codd propôs o modelo de dados relacional se tornando o padrão desde então. O modelo de Codd também deu origem a linguagem SQL (*Structured Query Language*), utilizada para o gerenciamento de banco de dados por grandes empresas, como a

IBM e Oracle, nos anos de 1980 em seus primeiros produtos comerciais (REZENDE, 2006).

Desde então a linguagem SQL se tornou padrão para criação de banco de dados. Dentre as principais vantagens estão a simplicidade de programação e a popularidade. Muitos profissionais, mesmo de áreas diferentes de Engenharia e Tecnologia da Informação, aprendem rapidamente programar em SQL e encontram vasto suporte na comunidade, como exemplos e tutoriais. Outra vantagem é disponibilidade de soluções *Open Source* e a interoperabilidade entre diferentes fornecedores. Apesar de pequenas variações na sintaxe dos programas, uma vez aprendido a teoria da linguagem é possível se adaptar rapidamente às diferentes plataformas (REZENDE, 2006).

O SQL e os sistemas de gerenciamento de banco de dados relacional já possuem uma idade avançada, entretanto ainda atendem as principais demandas atuais, pois apresentam interfaces amigáveis ao usuário, são capazes de armazenar uma grande quantidade de informações e grande parte das aplicações se expressam muito bem através do relacionamento entre os dados. Para sistemas onde as informações não são estruturadas e possuem uma grande variedade, não sendo possível prever todas as possíveis trocas e combinações de dados o SQL abre espaço para a utilização de novos bancos de dados, chamados de NoSQL (CÔRREA, 2017). Estas características, onde os dados não possuem estruturas rígidas, foram responsáveis por definir os novos modelos para o armazenamento de informações alavancando o termo *Big Data*. Contudo, apesar das diferenças iniciais, a cultura amplamente estabelecida, fez com que as organizações também desenvolvessem aplicações para consultas ao *Big Data* utilizando o SQL, contrariando as expectativas de que esta linguagem ficaria reservada apenas para situações convencionais. Se bem-sucedido, este movimento garantirá a utilização da linguagem SQL nos próximos anos, dando suporte ao *Big Data*, *Machine Learning* e as redes de *IoT* presentes na indústria (ORACLE, 2016).

#### 2.4.1 Banco de Dados XAMPP

XAMPP (Apache, MySQL, PHP, Perl) é uma plataforma *open source* criada pela Apache Friends que permite aos desenvolvedores criar servidor web local para fins de teste e projetos em linguagem MySQL. A Apache Friends, é um projeto sem

fins lucrativos, criados por Kai Seidler e Kay Vogelgesang em 2002 (Apache Friends, 2018). Esta plataforma será utilizada para criar o banco de dados que armazenará os dados coletados da linha de produção.

#### 2.4.2 Microsoft *Power BI*

O Power BI é uma ferramenta desenvolvida pela Microsoft para visualização de dados e criação de relatórios interativos compartilháveis e com várias opções de customização. O Power BI é capaz de obter dados de diversas fontes, incluindo bancos de dados SQL, páginas da internet e planilhas de dados, armazenados localmente ou em servidores na nuvem. Os relatórios gerados no Power BI podem ser compartilhados rapidamente através da versão paga Power BI Pro. (Microsoft, 2018). Entretanto, para fins de análise e criação de modelos para smartphones é possível acessar a versão gratuita, a mesma utilizada durante o desenvolvimento deste trabalho.

#### 2.4.3 Aplicação Java

A linguagem programação Java foi criada no início dos anos 90 por um grupo de Engenheiros chamados de *Green Team*, liderados por James Gosling. O objetivo inicial era criar uma interface que permitisse que os aparelhos eletrodomésticos pudessem se comunicar, criando uma rede interativa com usuário. Porém, difusão da linguagem Java cresceu a partir das aplicações para internet. Em 1995 o *Green Team* criou o navegador *Netscape Navigator* que incorporava tecnologia Java. Atualmente, essa linguagem e programação é amplamente empregada na internet e para desenvolvimento de sistemas e aplicativos para celulares (Oracle, 2018). As aplicações em Java também permitem a criação de interfaces gráficas e manipulação de bancos de dados. Estas características foram importantes na escolha da linguagem Java para a criação da interface de comunicação com usuário e armazenamento de informações da ferramenta de análise de eficiência desenvolvida.

### 3 FERRAMENTA DE ANÁLISE DE EFICIÊNCIA

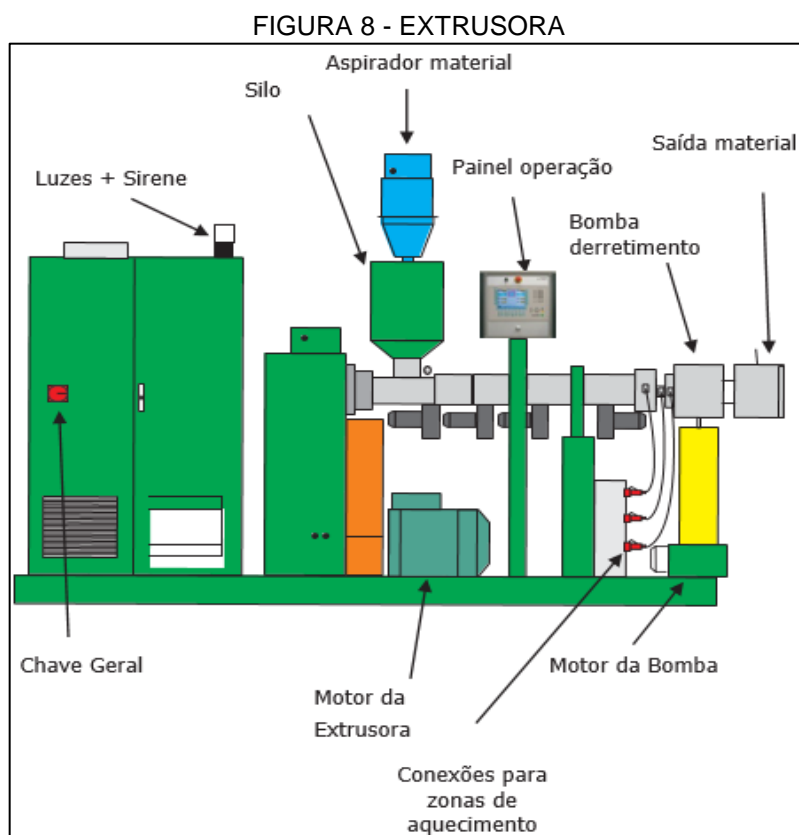
Neste capítulo são descritos a linha de produção e os métodos utilizados para coleta de dados e cálculo de eficiência, bem como a proposta de automação de fluxo de dados conforme observações realizadas na empresa.

#### 3.1 AMBIENTE DE APLICAÇÃO: LINHA DE PRODUÇÃO

A linha de produção de extrusão é composta por três elementos principais: a extrusora, a controlador de velocidade e a máquina de corte. A extrusora, apresentada na FIGURA 8, é o componente que mais consome energia, portanto a que mais realiza trabalho. A extrusora conta com o silo na sua parte traseira, onde o material armazenado, composto geralmente de granulados de polipropileno ou polietileno. O silo possui um funil que direciona o material para uma rosca aquecida que transportará o material até a matriz ou molde. A extrusora possui oito zonas de temperatura responsáveis por transformar o granulado em material derretido e que tomará o formato da matriz posicionado em sua saída. No sistema também há dois motores presentes, o primeiro, é responsável por manter a rosca interna girando enviando material para extremidade, o segundo, é posicionado próximo matriz e controla a saída de material mantendo a pressão interna constante. Este mecanismo evita que ocorram variações dimensionais no produto, mesmo quando há pouco ou há falta de material dentro da extrusora. Nestas situações soam alarmes para que o operador reponha o material faltante dentro do silo. Posicionada no final do canhão da extrusora a matriz será responsável por dar o formato desejado da peça.

Para realizar o resfriamento do material existem dois tanques posicionados após a extrusora. O primeiro é chamado de tanque de calibração e o segundo de tanque de resfriamento. No tanque de calibração ocorre a etapa de ajustes necessários para que a peça atenda as especificações do cliente. Após a saída da matriz o material ainda não possui uma forma bem definida e é necessário a utilização de três calibradores, que possuem o formato desejado da peça, e irão organizar as partículas do material. Também há pontos de água responsáveis por expandir ou fechar as cavidades da peça e “chaves” de ajuste responsáveis por reduzir as imperfeições não corrigidas pelo calibrador. Todos estes ajustes são realizados de forma manual pelo operador, este processo é chamado informalmente de “afinação”.

O segundo tanque, como o próprio nome diz, é responsável por resfriar a linha de material e possui entre 5 a 7 metros.



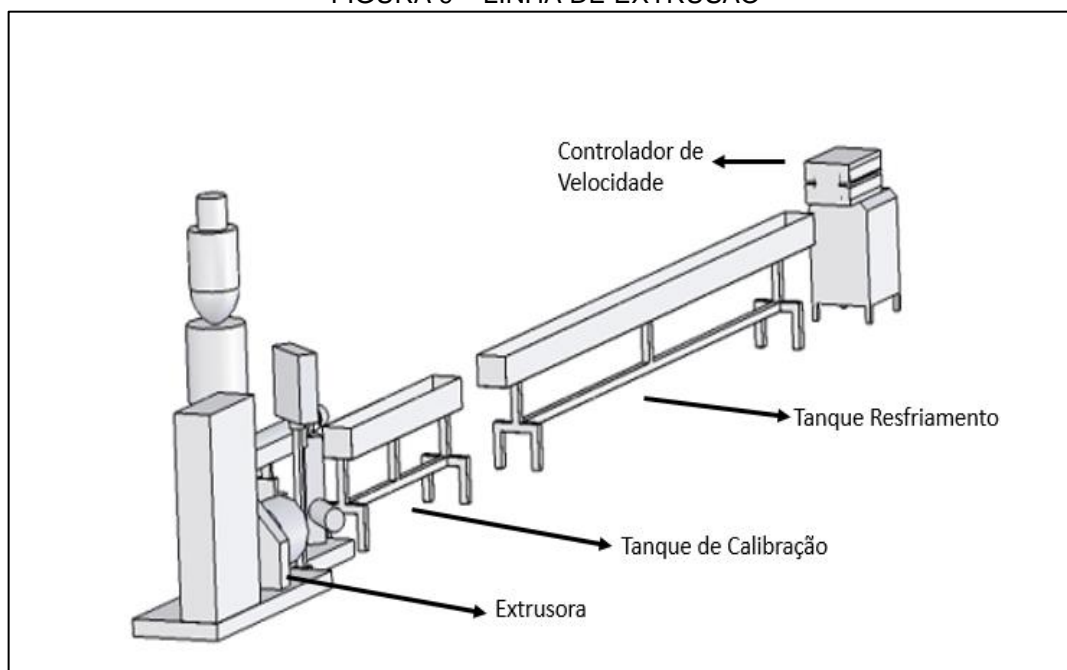
FONTE: O AUTOR (2019)

O segundo elemento é o controlador de velocidade. É ele quem dita o ritmo de velocidade de produção velocidade da linha de perfil plástico. Esta linha se estende desde a saída matriz até a máquina de corte, passando pelas etapas de calibração e resfriamento. O controlador de velocidade possui duas esteiras, posicionadas uma sobre a outra, girando na mesma velocidade. Após o tanque de resfriamento o operador posiciona a linha de material entre as esteiras para que estas puxem automaticamente o plástico que está sendo derretido pela rosca. Cada peça possui uma velocidade pré-estabelecida e informada em um documento que acompanha a ordem de produção, sendo esta a velocidade ideal para operação do controlador de velocidade. As velocidades variam de acordo com o comprimento, formato e complexidade do corte da peça. A extrusora, tanques de calibração e resfriamento, bem como o controlador de velocidade são apresentados na FIGURA 9.

O último elemento, a máquina de corte, é responsável por cortar pedaços da linha de perfil plástico, conforme o comprimento especificado, originando a peça final. Além de cortar no comprimento adequado é possível acoplar outros dispositivos a

máquina de corte para realizarem recortes diferentes no centro ou nas extremidades da peça. A máquina de corte além de permitir o ajuste do comprimento, e controle de outras ferramentas possui através de sua IHM um contador indicando a quantidade de peças por embalagem.

FIGURA 9 – LINHA DE EXTRUSÃO



FONTE: O AUTOR (2019)

### 3.2 METODOLOGIA DE CÁLCULO DE EFICIÊNCIA

Para cálculo de eficiência produtiva são necessários os dados referentes a quantidade de metros reais produzidos e quantidade de metros desejados. Os metros produzidos são calculados considerando a quantidade de peças produzidas e seu comprimento em metros.

$$\text{Metros Produzidos} = \text{Total de Peças} \times \text{Comprimento (m)} \quad (1)$$

Para o cálculo dos metros desejados é necessário o tempo total de produção em minutos, incluindo o tempo de setup e paradas de linha, e a velocidade ideal da linha de produção, que é dada em metros por minuto.

$$\text{Metros Solicitados} = \text{Tempo (min)} \times \text{Velocidade ideal (m/min)} \quad (2)$$



Por fim, a porcentagem de eficiência é dada pela divisão abaixo:

$$Eficiência = \frac{Metros\ Produzidos}{Metros\ Solicitados} (\%) \quad (3)$$

Impactam diretamente na eficiência o tempo de setup da linha de produção e as paradas. Quanto maior o tempo despendido nestas atividades maior será a ineficiência dos processos. A velocidade também constitui um fator importante, pois velocidades de produção abaixo da ideal reduzem a produtividade e aumentam a necessidade de horas por máquina e o consumo de energia. A velocidade é definida com base na complexidade da peça que depende do tipo de material, tipo de corte e formato da peça. Todas estas características e quantidade de diferentes peças produzidas indicam a necessidade de um controle preciso dos dados que impactam diretamente sobre a linha produção e resultados de cada item manufaturado.

### 3.3 AUTOMAÇÃO E FLUXO DE DADOS

Atualmente os dados referentes aos *metros produzidos* e *metros solicitados* são preenchidos em relatórios de produção pelos operadores das linhas de produção. Considerando as linhas de extrusão são gerados diariamente 15 relatórios, divididos em 3 turnos de produção, resultando em 90 relatórios semanais. Os dados são lançados em uma planilha onde é realizado o cálculo de eficiência por linha e posteriormente o cálculo de eficiência geral, descrito no tópico anterior, cuja meta é de 85%. Para os turnos onde a meta de eficiência não foi alcançada são definidas ações com base nos problemas identificados, sejam eles paradas de máquina, elevado tempo de setup ou velocidade abaixo do desejado.

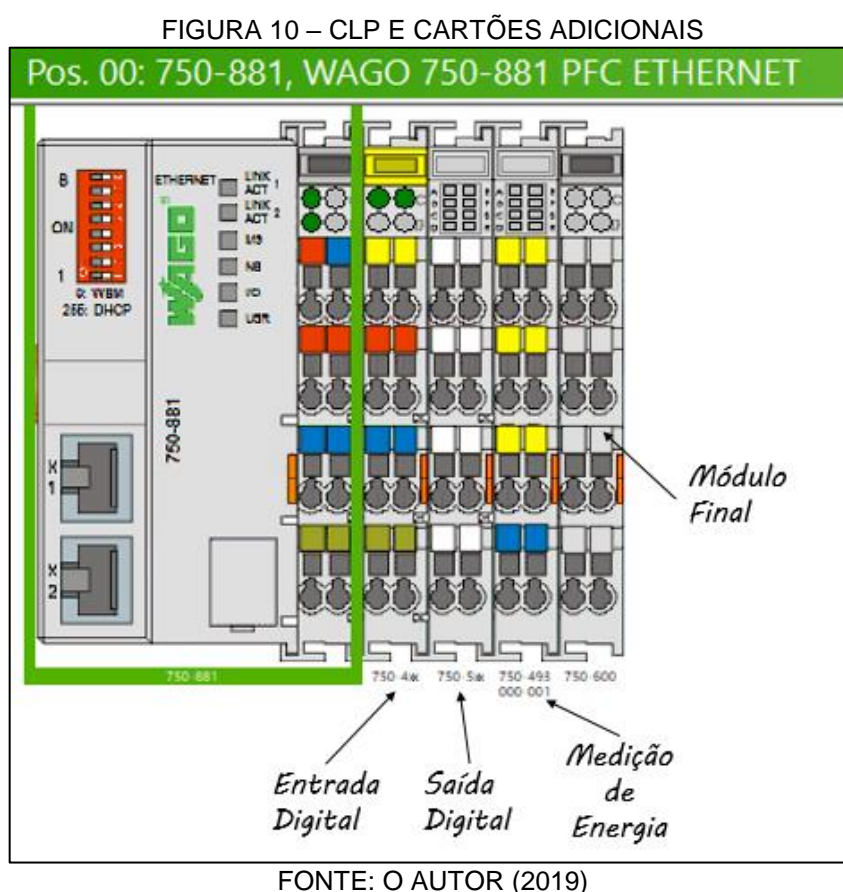
A proposta de automação deste trabalho consiste em coletar estas informações diretamente de uma linha produção, simplificando as etapas de armazenamento e tratamento dos dados. A interface de comunicação com usuário também permitiu o acompanhamento dos resultados da linha de produção, reduzindo assim o tempo de ação sobre os problemas dos profissionais envolvidos neste processo. Além do cálculo de eficiência, foram monitorados a quantidade de metros não utilizados (rejeitos) e o consumo de energia da extrusora, identificando seu comportamento ao longo da produção.

Para criar tal ferramenta de análise de eficiência o controlador lógico programável foi responsável por adquirir os dados da linha de produção e enviar as informações ao banco de dados, compondo os estágios 1 e 2 segundo o modelo de *IIoT* proposto por Boyes (2018). A interface de comunicação com os usuários e disposição dos dados são compostos por uma aplicação em Java em conjunto com software Microsoft Power BI, a serem detalhados no próximo capítulo.

## 4 ETAPAS DE DESENVOLVIMENTO

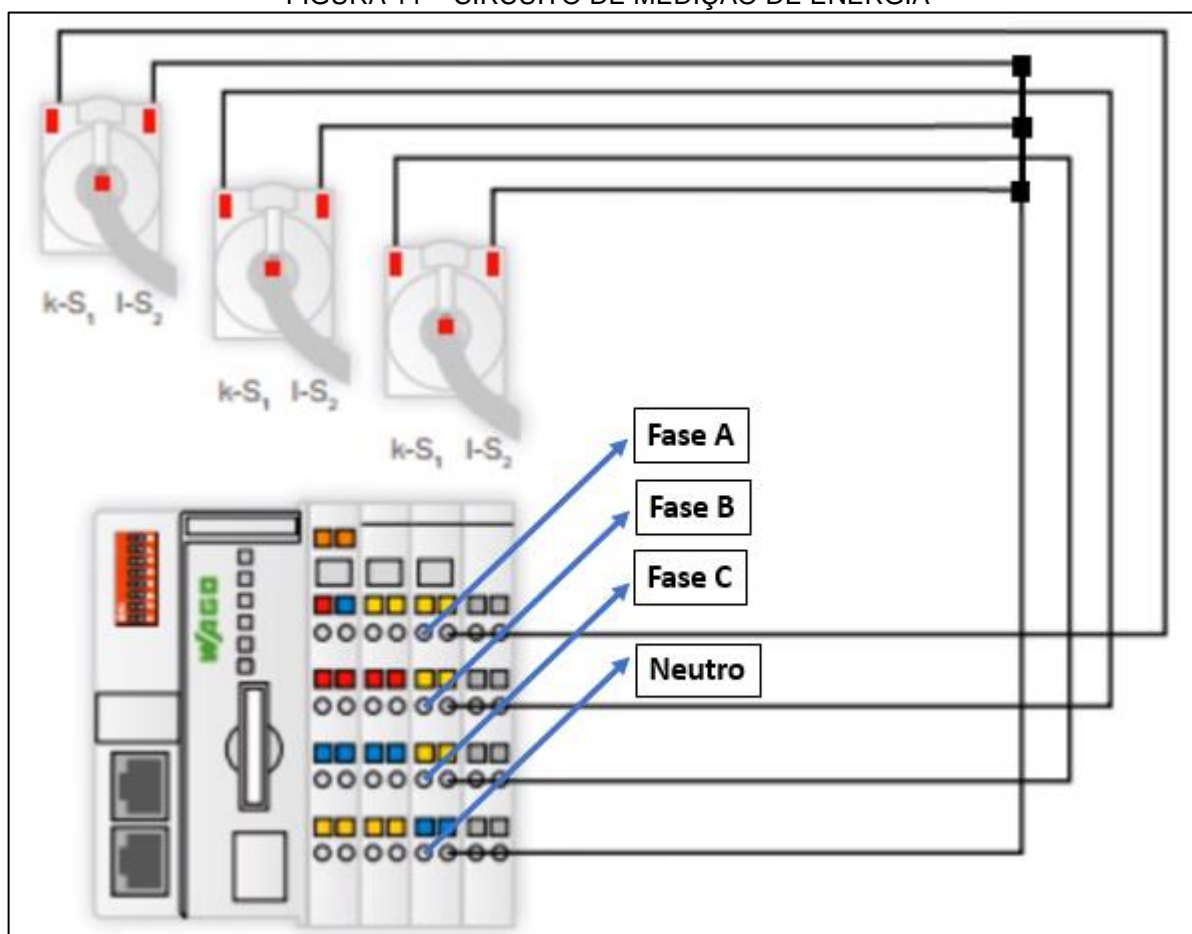
### 4.1 AQUISIÇÃO DE DADOS ENERGIA

Para aquisição de dados o CLP (figura 10) foi acoplado ao painel elétrico da extrusora. Utilizando transformadores de correntes, do mesmo fabricante, foi realizada a conexão das três fases ao cartão de medição de energia 750-493 (FIGURA 10).



A potência e corrente nominal da extrusora é de 115 kW e 143 A respectivamente e a alimentação é trifásica de 380 V. Segundo informado pelo fabricante a capacidade do transformador de corrente é de 2000 A. O cartão de leitura por sua vez suporta até 480 volts e 5 amperes. As ligações foram realizadas conforme apresentada na FIGURA 11.

FIGURA 11 – CIRCUITO DE MEDIÇÃO DE ENERGIA



FONTE: O AUTOR (2019)

O software Codesys, utilizado na programação do CLP, possui bibliotecas prontas para medição de energia, sendo necessário apenas definir os valores de relação de transformação do transformador de corrente para aquisição dos valores corretos. O CLP foi programado para informar a potência instantânea total (kW) e corrente trifásica total (A).

#### 4.2 AQUISIÇÃO DE DADOS PRODUÇÃO

O CLP também foi responsável por obter a informação de metros produzidos. Para efetuar esta operação um encoder incremental (FIGURA 12) foi posicionado próximo a entrada do controlador de velocidade e o seu sinal de saída conectado a entrada de um cartão de entrada digital. Com base no perímetro do encoder é possível determinar quantas voltas são necessárias para alcançar um metro de material produzido. Esta relação foi determinada e inserida no programa principal do CLP.

Os dados são atualizados a cada um minuto, portanto, além de registrar a quantidade de metros é possível determinar a velocidade real de produção (m/min). Também foi adicionado a uma das entradas digitais do CLP um botão de pressão capaz de resetar a contagem de metros sempre que necessário.

FIGURA 12 - ENCODER



FONTE: O AUTOR (2019)

#### 4.3 ARMAZENAMENTO DE DADOS

O software Codesys também dispõe de bibliotecas para comunicação com banco de dados em SQL. Os blocos presentes nesta biblioteca foram utilizados para armazenar no banco de dados XAMPP os valores de potência, corrente e metros. Assim, foi necessário informar ao CLP qual banco de dados, o endereço de IP, porta, login, senha e tabela a serem utilizados para que as informações fossem registradas (FIGURA 13). Os dados são armazenados sequencialmente e identificados através de uma chave primária *ID* configurada para operar com a função AUTO INCREMENT.

FIGURA 13 – PARÂMETROS DE COMUNICAÇÃO COM BANCO DE DADOS XAMPP

```

PROGRAM Comunica
VAR
  oMySQL : MySQL_Context; (* Structure for keeping login and status informations *)

  oMySQL_Login      : MySQL_Login;          (* Structure for keeping login and status informations *)
  sHost              : STRING := '192.168.1.33'; (* IP or hostname where MySQL-server is working *)
  uiPort             : UINT := 3306;         (* TCP-Port where MySQL-server is listening, typ. 3306 *)
  sUsername          : STRING := 'gabriel';  (* Name of database account *)
  sPassword          : STRING := 'gabriel';  (* Password of database account *)
  sDatabase          : STRING := 'WagoDB2';  (* Name of schema to use initially - Optional param *)
  xConnect           : BOOL;                (* Start execution on a rising edge and reset internally when done or on timeout. *)
  diError            : DINT;                (* Numeric return value - more readable in hex *)
  sStatus            : STRING(500);         (* Textual description of current state or error *)
  xConnected         : BOOL;                (* TRUE: when database connection is established *)
  sTable             : STRING := 'data3';

```

FONTE: O AUTOR (2019)

Além disso, um computador foi montado próximo a extrusora e conectado através de um cabo de rede ao CLP servindo como servidor para o armazenamento das informações no banco de dados (FIGURA 14) e transmissão para nuvem. Esta solução foi utilizada pois os servidores da empresa são gerenciados somente pela planta matriz, localizada fora do país.

FIGURA 14 – LAYOUT BANCO DE DADOS

The screenshot shows the phpMyAdmin interface for a MySQL database named 'wagodb2'. The selected table is 'data3'. The interface displays a table with the following columns: Id, timestamp, Corrente, Potencia, and Metros. The table contains 7 rows of data, with the first row having Id 517 and timestamp 2019-02-15 18:59:05. The interface also shows a search bar and a 'Pesquisar' button.

	Id	timestamp	Corrente	Potencia	Metros
<input type="checkbox"/>	517	2019-02-15 18:59:05	22	15	8624
<input type="checkbox"/>	516	2019-02-15 18:57:57	21	15	8607
<input type="checkbox"/>	515	2019-02-15 18:56:48	16	11	8589
<input type="checkbox"/>	514	2019-02-15 18:55:40	22	14	8572
<input type="checkbox"/>	513	2019-02-15 18:54:31	22	15	8555
<input type="checkbox"/>	512	2019-02-15 18:53:23	21	15	8538
<input type="checkbox"/>	511	2019-02-15 18:52:14	21	15	8521

FONTE: O AUTOR (2019)

#### 4.4 STATUS DA LINHA DE PRODUÇÃO - APLICAÇÃO JAVA

A aplicação em Java (FIGURA 15) consiste em uma interface de comunicação com os operadores da fábrica, onde é possível indicar ao sistema qual item está sendo produzido (Part Number), a velocidade ideal (indicada na ordem de produção), o lote de produção e a condição ou *status* atual da linha. Estas informações foram muito importantes pois deram sentido aos dados coletados pelo CLP e permitiram uma análise mais detalhada da eficiência da linha de produção. Os dados são atualizados a cada minuto, assim como no CLP, desta maneira é possível realizar o cruzamento das informações no Microsoft Power BI através do número de *ID* de cada leitura.

FIGURA 15 – APLICAÇÃO JAVA

ID	PART NUMBER	LOTE	VELOCIDADE	STATUS
1	1838-290	MAYQ83	15.1	Produção
3	1838-290	MAYQ83	15.1	Produção
4	1838-290	MAYQ83	15.1	Produção
5	1838-290	MAYQ83	15.1	Produção
6	1838-290	MAYQ83	15.1	Produção
7	1838-290	MAYQ83	15.1	Produção
8	1838-290	MAYQ83	15.1	Produção
9	1838-290	MAYQ83	15.1	Produção
10	1838-290	MAYQ83	15.1	Produção
11	1838-290	MAYQ83	15.1	Produção
12	1838-290	MAYQ83	15.1	Produção
13	1838-290	MAYQ83	15.1	Produção
14	1838-290	MAYQ83	15.1	Produção

FONTE: O AUTOR (2019)

Para facilitar a entrada de dados o campo *status* possui opções pré-definidas, apresentadas na FIGURA 16, facilitando a escolha pelo operador. São elas:

- **Setup:** Indica que operador está preparando a linha de extrusão e seus acessórios para iniciar a produção das peças (não há produção de rejeitos);
- **Afinação:** Após o setup inicia-se o processo de afinação. É nesta etapa onde a linha de perfil plástico começa a ser extrudada e antes do início da produção é necessário realizar as correções do formato da peça no tanque de calibração;
- **Produção:** Peças liberadas para produção e armazenamento no estoque;
- **Qualidade:** Indica anomalias na peça, assim o contador de peças da máquina de corte deve ser desativado e é necessário realizar as correções e ajustes no tanque de calibração;

- **Avaria:** Esta opção é escolhida quando a linha de produção apresenta problemas em algum dos dispositivos. Os problemas são corrigidos através do suporte da equipe de manutenção;
- **Manutenção:** Corresponde às manutenções preventivas que ocorrem durante a produção. Nesta condição, o contador de peças também é desativado por alguns minutos, para a execução de reaperto, lubrificação e realinhamento das ferramentas.

FIGURA 16 – STATUS DA LINHA DE PRODUÇÃO

	PART NUMBER	LOTE	VELOCIDADE	STATUS
1	1838-290	MAYQ83	15.1	Produção
3	1838-290	MAYQ83	15.1	Produção
4	1838-290	MAYQ83	15.1	Produção
5	1838-290	MAYQ83	15.1	Produção
6	1838-290	MAYQ83	15.1	Produção
7	1838-290	MAYQ83	15.1	Produção
8	1838-290	MAYQ83	15.1	Produção
9	1838-290	MAYQ83	15.1	Produção
10	1838-290	MAYQ83	15.1	Produção
11	1838-290	MAYQ83	15.1	Produção
12	1838-290	MAYQ83	15.1	Produção
13	1838-290	MAYQ83	15.1	Produção
14	1838-290	MAYQ83	15.1	Produção

FONTE: O AUTOR (2019)

Após inserir os dados referente a ordem de produção o operador deve utilizar o botão *Atualizar* para inserir as informações no banco de dados. A atualização dos dados a cada um minuto é automática e alimenta o banco de dados (FIGURA 17) através de uma rotina criada dentro da aplicação.



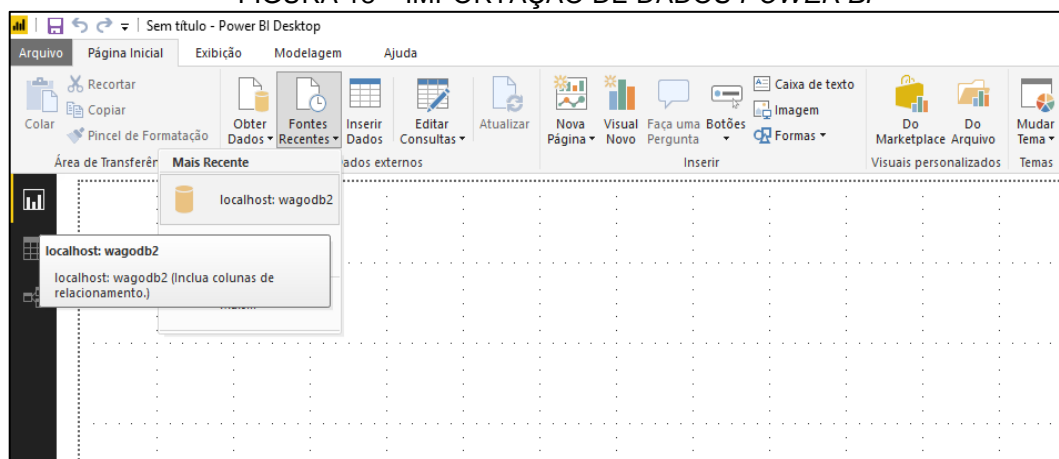
FIGURA 17 – BANCO DE DADOS PARA APLICAÇÃO JAVA

ID	Hora	PartNumber	Lote	Velocidade	Status
532	2019-02-16 08:37:14	1838-100	MAYQ89	15	Produção
531	2019-02-16 08:37:14	1838-100	MAYQ89	15	Produção
530	2019-02-16 08:37:14	1838-100	MAYQ89	15	Produção
529	2019-02-16 08:37:14	1838-100	MAYQ89	15	Produção
528	2019-02-16 08:37:13	1838-100	MAYQ89	15	Produção
527	2019-02-16 08:37:13	1838-100	MAYQ89	15	Produção
526	2019-02-16 08:37:13	1838-100	MAYQ89	15	Produção
525	2019-02-16 08:37:13	1838-100	MAYQ89	15	Produção
524	2019-02-16 08:37:13	1838-100	MAYQ89	15	Produção
523	2019-02-16 08:37:13	1838-100	MAYQ89	15	Produção
522	2019-02-16 08:37:13	1838-100	MAYQ89	15	Produção
521	2019-02-16 08:37:13	1838-100	MAYQ89	15	Produção
520	2019-02-16 08:37:13	1838-100	MAYQ89	15	Produção
519	2019-02-16 08:37:13	1838-100	MAYQ89	15	Produção

FONTE: O AUTOR (2019)

#### 4.5 ANÁLISE DE DADOS

A última etapa é análise das informações. O Power BI dispõe de serviços que permitem realizar a importação (FIGURA 18) direto da base dados do XAMPP.

FIGURA 18 – IMPORTAÇÃO DE DADOS *POWER BI*

FONTE: O AUTOR (2019)

Posteriormente, os dados podem ser manipulados alimentando os gráficos que irão informar a eficiência da linha de produção (FIGURA 19). Ainda é possível remover ou adicionar colunas, gerando novas informações através de funções prontas ou programação, atendendo as necessidades de cada projeto. Outra vantagem do Power BI é que os relatórios podem ser salvos localmente ou enviados para a nuvem.

Também há possibilidade de criar layouts para visualizações relatórios em smartphones através do aplicativo oferecido pelo software.

FIGURA 19 – EDIÇÃO POWER BI

Id	timestamp	Corrente	Potencia	Metros
1	12/02/2019 08:16:49	11	8	7
2	12/02/2019 08:16:49	11	8	7
3	12/02/2019 08:17:57	13	9	20
4	12/02/2019 08:19:06	13	9	34
5	12/02/2019 08:20:14	14	9	48
6	12/02/2019 08:21:23	13	8	61
7	12/02/2019 08:22:31	13	9	75
8	12/02/2019 08:23:40	13	9	89
9	12/02/2019 08:24:48	13	9	102
10	12/02/2019 08:25:56	13	8	116
11	12/02/2019 08:27:05	13	9	130
12	12/02/2019 08:28:13	13	9	143
13	12/02/2019 08:29:22	14	9	157
14	12/02/2019 08:30:30	13	9	171
15	12/02/2019 08:31:39	13	8	184
16	12/02/2019 08:32:47	14	10	198
17	12/02/2019 08:33:56	13	9	212
18	12/02/2019 08:35:04	13	9	225
19	12/02/2019 08:36:13	13	9	239
20	12/02/2019 08:37:21	13	9	253
21	12/02/2019 08:38:30	13	9	266
22	12/02/2019 08:39:39	13	9	280
23	12/02/2019 08:40:47	13	9	294

FONTE: O AUTOR (2019)

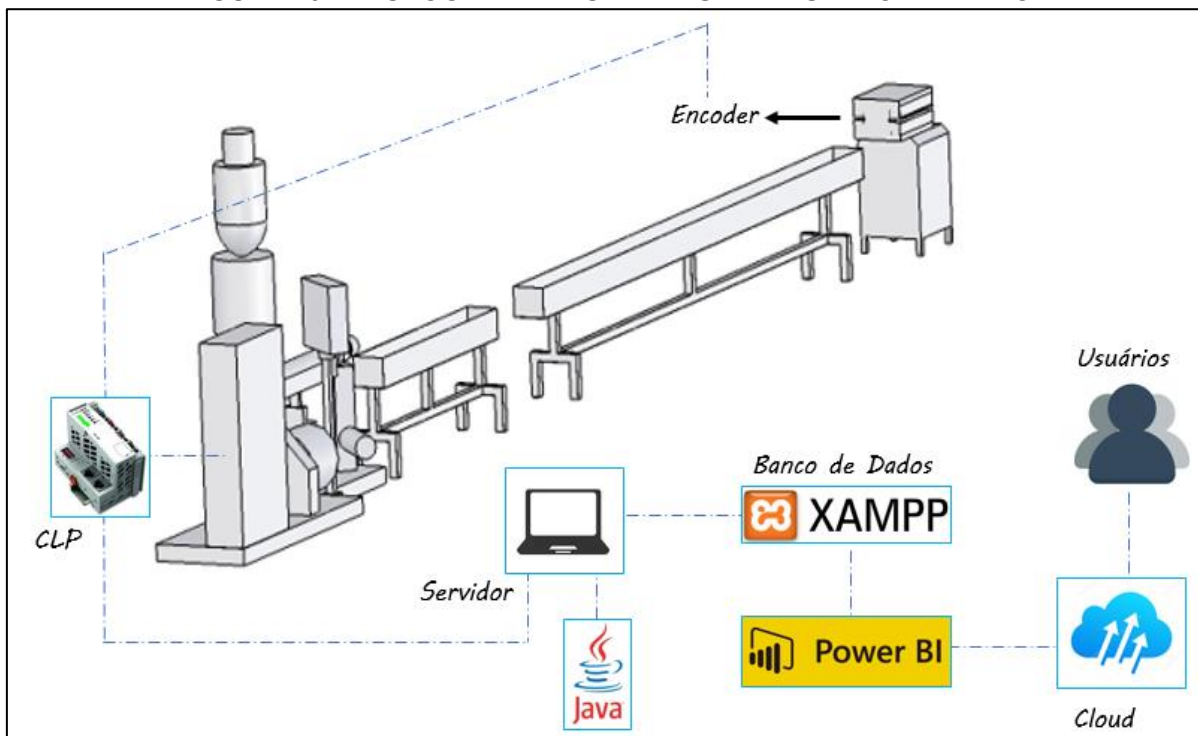
#### 4.6 FLUXOGRAMA DAS ETAPAS DE DESENVOLVIMENTO

A partir das etapas de desenvolvimento descritas acima é possível montar um fluxograma da origem da informação até o usuário final apresentado na FIGURA 20. É interessante observar que apesar dos diferentes tipos de soluções utilizadas na criação da ferramenta a compatibilidade entre as linguagens de programação do CLP (ladder), aplicação java e banco de dados (SQL) permitiram que os dados fossem entregues para análise sem nenhum ruído ao longo do percurso.

Na fase final os dados importados pelo Power BI construíram o espaço essencial para análise de eficiência, consumo de energia e peças descartadas ao longo da medição. Como mencionado, a plataforma do Power BI permite que os

relatórios sejam salvos na nuvem e acessados remotamente pelo smartphone, criando a interface com o usuário final

FIGURA 20 – FLUXOGRAMA DAS ETAPAS DE DESENVOLVIMENTO



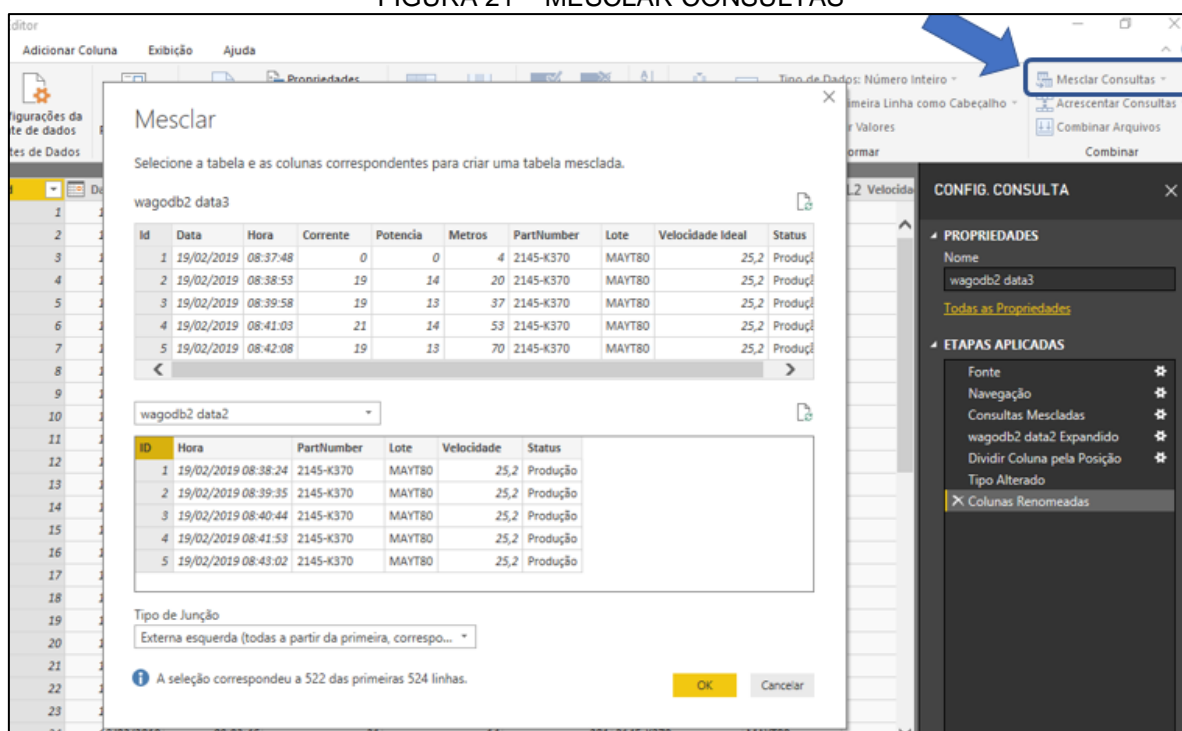
FONTE: O AUTOR (2019)

## 5 ESTUDO DE CASO – APLICAÇÃO DA FERRAMENTA

### 5.1 ANÁLISE DE EFICIÊNCIA, REJEITOS E VELOCIDADE DA LINHA DE PRODUÇÃO

As informações coletadas pelo CLP foram importadas pelo *Power BI* para criação dos relatórios. O primeiro relatório elaborado consistiu na análise de eficiência, scrap (rejeitos) e velocidade da linha de produção durante o intervalo de leitura. O software permite a edição destas informações, e o primeiro processo é o de unir as duas tabelas criadas no banco de dados XAMPP através da função *mesclar consultas* (FIGURA 21), utilizando como referência a chave primária *ID*.

FIGURA 21 – MESCLAR CONSULTAS



FONTE: O AUTOR (2019)

A junção das duas tabelas permite a elaboração de uma análise completa contendo as informações das duas bases de dados (FIGURA 22) fornecidas pelo CLP e pela aplicação Java.

FIGURA 22 – COLUNAS BANCO DE DADOS POWER BI

Id	Data	Hora	Corrente	Potencia	Metros	PartNumber	Lote	Velocidade Ideal	Status	Velocidade Real	% Velocidade
231	20/02/19	12:38:12	13	9	2185	2184-035	MAYU40	15	Produção	9,5	-36,94%
232	20/02/19	12:39:17	13	9	2196	2184-035	MAYU40	15	Produção	9,5	-36,90%
234	20/02/19	12:41:27	13	9	2217	2184-035	MAYU40	15	Produção	9,5	-36,84%
235	20/02/19	12:42:32	13	9	2228	2184-035	MAYU40	15	Produção	9,5	-36,79%
236	20/02/19	12:43:37	13	9	2239	2184-035	MAYU40	15	Produção	9,5	-36,75%
240	20/02/19	12:47:57	13	9	2282	2184-035	MAYU40	15	Produção	9,5	-36,61%
243	20/02/19	12:51:12	13	9	2315	2184-035	MAYU40	15	Produção	9,5	-36,49%

FONTE: O AUTOR (2019)

Foram então criadas novas colunas para determinação da velocidade real da linha de produção e a redução ou acréscimo percentual da velocidade em relação ao solicitado na ordem de produção (FIGURA 23). A velocidade real da linha de produção é calculada através da razão entre os metros produzidos e os valores da coluna ID que representam os minutos de cada leitura.

FIGURA 23 – CÁLCULO DE VELOCIDADE REAL

Id	Data	Hora	Corrente	Potencia	Metros	PartNumber	Lote	Velocidade Ideal	Status	Velocidade Real	% Velocidade
207	20/02/19	12:12:11	13	9	1925	2184-035	MAYU40	15	Produção	9,3	-38,00%
211	20/02/19	12:16:31	13	9	1968	2184-035	MAYU40	15	Produção	9,3	-37,82%
212	20/02/19	12:17:36	13	9	1979	2184-035	MAYU40	15	Produção	9,3	-37,77%

FONTE: O AUTOR (2019)

A última coluna “% Velocidade” (FIGURA 24) é a razão entre a velocidade real e a velocidade ideal. A fórmula criada entrega a diferença entre as velocidades e informa se ouve a acréscimo ou decréscimo de velocidade durante a produção.

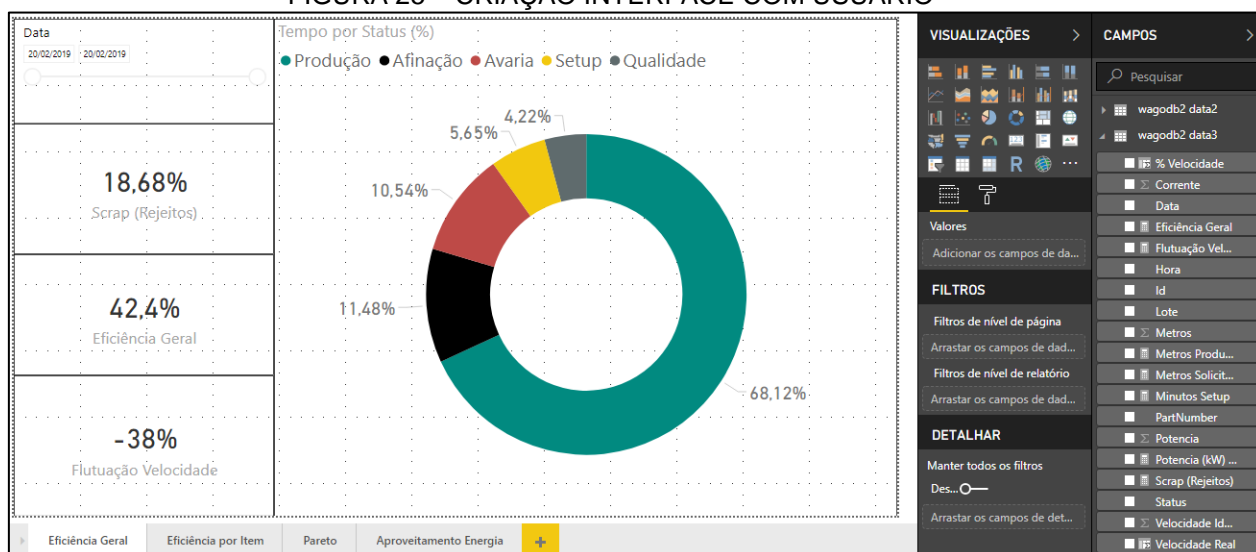
FIGURA 24 – CÁLCULO ALTERAÇÃO DE VELOCIDADE PERCENTUAL

Id	Data	Hora	Corrente	Potencia	Metros	PartNumber	Lote	Velocidade Ideal	Status	Velocidade Real	% Velocidade
230	20/02/19	12:37:07	13	9	2174	2184-035	MAYU40	15	Produção	9,5	-36,99%
231	20/02/19	12:38:12	13	9	2185	2184-035	MAYU40	15	Produção	9,5	-36,94%
232	20/02/19	12:39:17	13	9	2196	2184-035	MAYU40	15	Produção	9,5	-36,90%
234	20/02/19	12:41:27	13	9	2217	2184-035	MAYU40	15	Produção	9,5	-36,84%

FONTE: O AUTOR (2019)

Com a manipulação de dados finalizada é possível gerar os relatórios que irão informar ao usuário de forma visual os resultados da linha de produção (FIGURA 25). O software dispõe de diferentes recursos de visualização (gráficos, filtros, tabelas e textos) e permite que sejam geradas novas variáveis que irão auxiliar para cálculo de eficiência e segmentação dos dados.

FIGURA 25 – CRIAÇÃO INTERFACE COM USUÁRIO

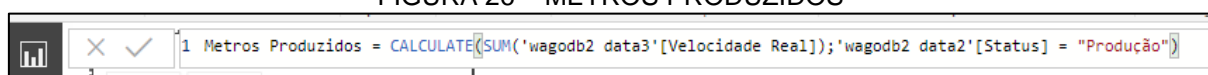


FONTE: O AUTOR (2019)

### 5.1.1 CÁLCULO DE EFICIÊNCIA

O cálculo de eficiência como apresentado no item 3.2 é a razão entre os metros produzidos e metros solicitados durante o intervalo de produção. Os metros produzidos foram calculados a partir da soma dos valores da coluna *Velocidade Real* onde o status fosse igual a *Produção*, retirando assim os metros de material não convertido em peças boas durante os outros status de produção (afinação, qualidade, avaria, etc.). A formula de cálculo é descrita na FIGURA 26.

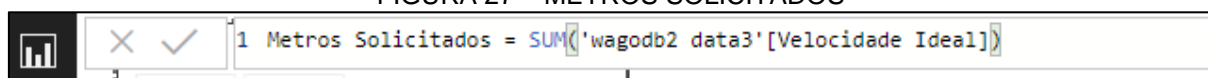
FIGURA 26 – METROS PRODUZIDOS



FONTE: O AUTOR (2019)

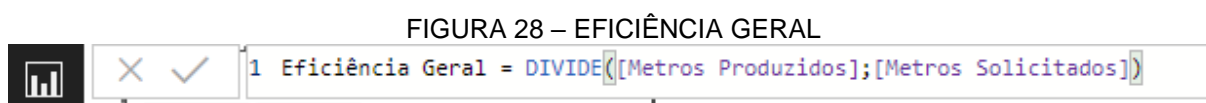
Os metros solicitados (FIGURA 27) utilizaram as informações da coluna *Velocidade Ideal* desta forma é possível identificar a eficiência real da linha considerando todas as paradas durante a produção e a perdas por redução de velocidade.

FIGURA 27 – METROS SOLICITADOS



FONTE: O AUTOR (2019)

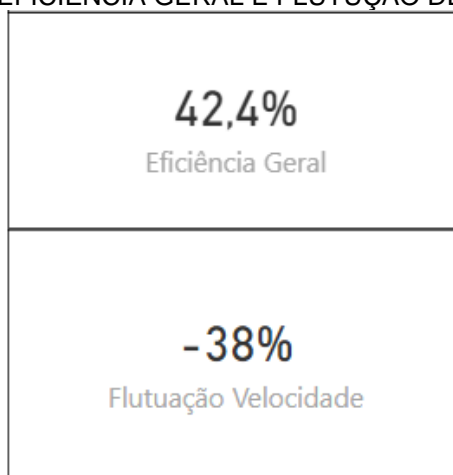
A eficiência resulta da razão entre estas duas *medidas* (FIGURA 28):



FONTE: O AUTOR (2019)

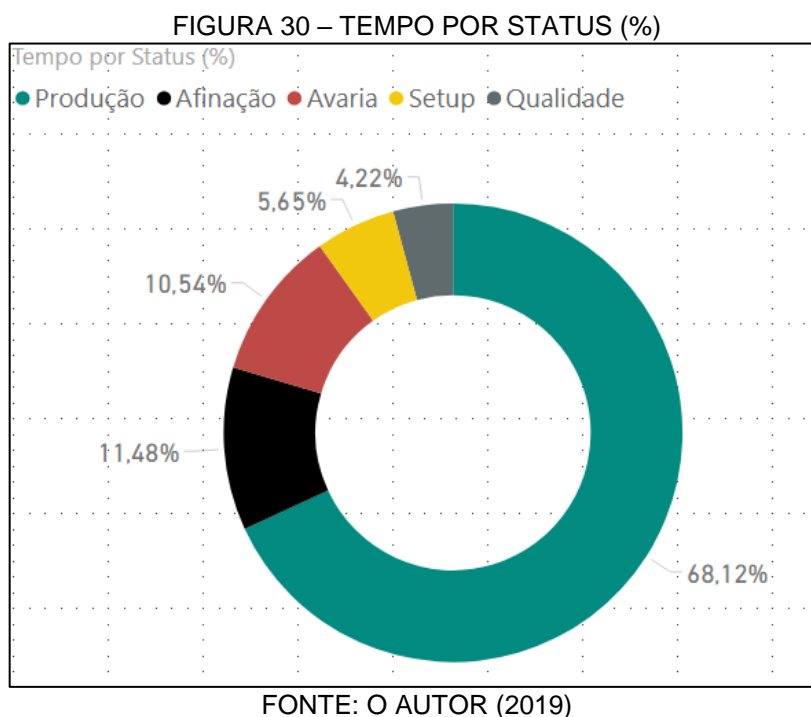
O resultado neste intervalo de leitura, apresentado nas FIGURA 25 e FIGURA 29, foi de 42,4%, principalmente influenciado pela flutuação negativa de velocidade, cuja média foi de -38%. Para os envolvidos nos trabalhos diários de produção será importante investigar o que causou esta flutuação e se é possível trabalhar em melhorias para que a *Velocidade Ideal* seja alcançada ou realizar correção da velocidade para os próximos lotes de produção.

FIGURA 29 – EFICIÊNCIA GERAL E FLUTUAÇÃO DE VELOCIDADE



FONTE: O AUTOR (2019)

A combinação entre os status apontados durante a produção e a leitura de velocidade originou um gráfico de rosca (FIGURA 30) informando o tempo despendido em cada atividade durante as horas trabalhadas. Esta análise também é importante a fim de identificar quais situações impactaram na eficiência produtiva além da redução de velocidade.



O gráfico informa que além das atividades comuns a toda ordem de produção, que são setup, afinação e produção a linha ainda apresentou perdas de 10,54% em avaria e outros 4,22% em problemas de qualidade no produto. Esta instabilidade no processo pode indicar que o setup da linha produção não foi realizado corretamente, a utilização de dispositivos não apropriados ou desgaste de ferramentas.

### 5.1.2 CÁLCULO DE REJEITOS

Os metros de material não aproveitados, ou rejeitos, foram calculados utilizando a linguagem de programação do Power BI. A pequena rotina criada (FIGURA 31) seleciona os metros produzidos, considerando a velocidade real, quando o estado da linha de produção é: *Qualidade*, *Manutenção*, *Avaria* ou *Afinação*. Na linha 9 do código os metros que foram registrados durante o tempo de setup são retirados dos metros totais originado a variável *MetrosTotaisCorrigido*. Esta correção é necessária pois quando o estado é *Setup* não ocorre a geração de rejeitos, apenas a preparação da linha de produção. Ao final, na linha 11, obtém-se o valor percentual de rejeitos, através função *DIVIDE*, calculando a razão entre a soma das variáveis e a variável *MetrosTotaisCorrigido*.



FIGURA 31 – CÁLCULO DE REJEITOS (%)

```

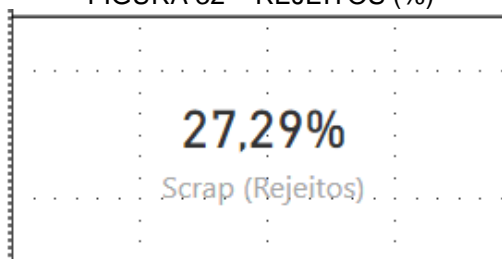
1 Scrap (Rejeitos) =
2
3     var MetrosTotais = SUM('wagodb2 data3'[Velocidade Real])
4     var MetrosQuali = CALCULATE(SUM('wagodb2 data3'[Velocidade Real]);'wagodb2 data3'[Status]
5     = "Qualidade")
6     var MetrosAfina = CALCULATE(SUM('wagodb2 data3'[Velocidade Real]);'wagodb2 data3'[Status]
7     = "Afinação")
8     var MetrosManut = CALCULATE(SUM('wagodb2 data3'[Velocidade Real]);'wagodb2 data3'[Status]
9     = "Manutenção")
10    var MetrosAvaria = CALCULATE(SUM('wagodb2 data3'[Velocidade Real]);'wagodb2 data3'[Status]
11    = "Avaria")
12    var MetrosSetup = CALCULATE(SUM('wagodb2 data3'[Velocidade Real]);'wagodb2 data3'[Status]
13    = "Setup")
14    var MetrosTotaisCorrigido = MetrosTotais - MetrosSetup
15    return
16    DIVIDE((MetrosQuali + MetrosAfina + MetrosManut+ MetrosAvaria);MetrosTotaisCorrigido)

```

FONTE: O AUTOR (2019)

O valor obtido para intervalo de dados coletado foi de 27,29% acima da meta atual que é de 14% (FIGURA 32). Assim como no cálculo de eficiência o tempo despendido na correção do produto, avarias e outras paradas de linha também impacta negativamente no resultado de quantidade peças não aproveitadas. Esta informação também compõe o painel de interface com o usuário.

FIGURA 32 – REJEITOS (%)

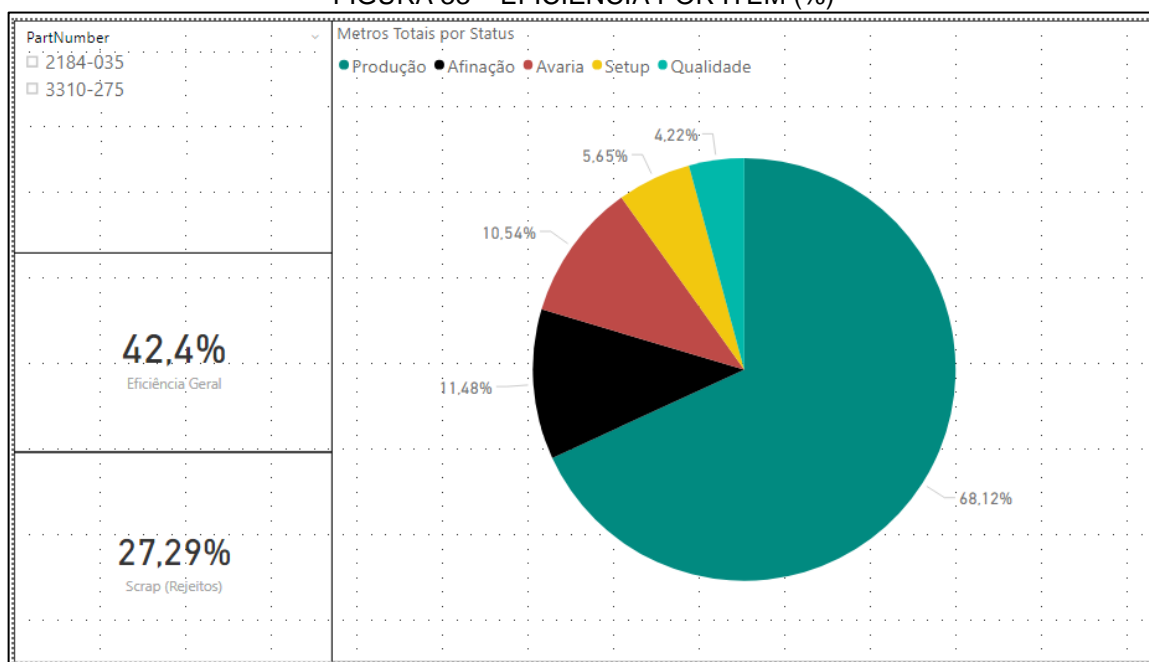


FONTE: O AUTOR (2019)

### 5.1.3 SEGMENTAÇÃO DE DADOS

Uma importante etapa no diagnóstico dos problemas é identificar em quais momentos se obtiveram as maiores perdas e assim definir ações corretivas e preventivas mais robustas. Para isto foi criado um relatório, *Eficiência por Item*, adicional realizando a segmentação dos dados por item produzido, permitindo a avaliação da performance de produção de cada item, indicando o tempo de cada estado, eficiência e rejeitos (FIGURA 33).

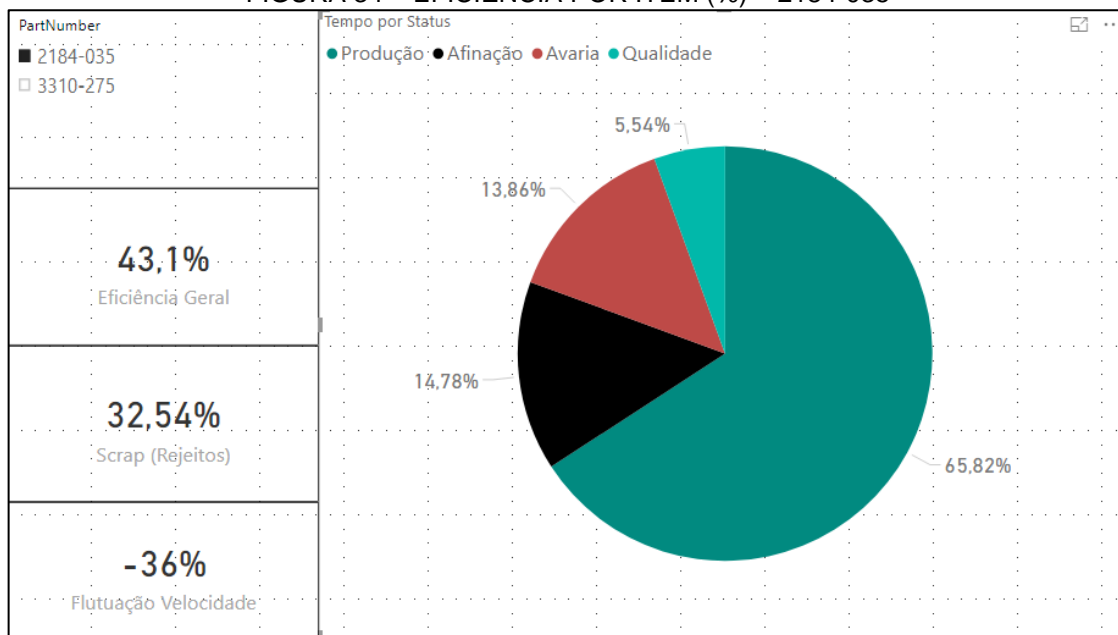
FIGURA 33 – EFICIÊNCIA POR ITEM (%)



FONTE: O AUTOR (2019)

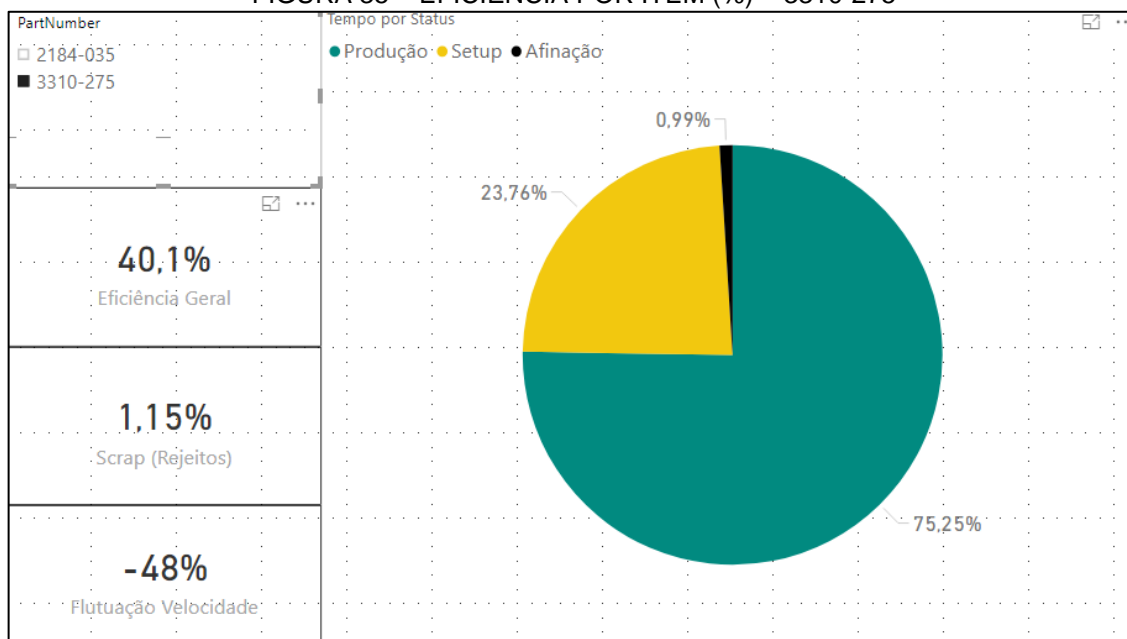
Selecionando item desejado no filtro *PartNumber* o gráfico de pizza entrega os dados separadamente. Os resultados são apresentados nas FIGURA 34 e FIGURA 35.

FIGURA 34 – EFICIÊNCIA POR ITEM (%) – 2184-035



FONTE: O AUTOR (2019)

FIGURA 35 – EFICIÊNCIA POR ITEM (%) – 3310-275



FONTE: O AUTOR (2019)

Como apresentado na FIGURA 34 o produto 2184-035 aliado a perda de eficiência por baixa velocidade apresentou problemas por avaria e qualidade, enquanto o item 3310-125, FIGURA 35, também apresentou redução de velocidade e tempo de setup (preparação da linha) considerável. Ao longo da semana e do mês outros produtos serão produzidos e ao agrupar estes dados é possível montar outros relatórios, como um gráfico de Pareto, orientando assim as ações para as paradas de linhas mais longas.

### 5.3 ANÁLISE DE CONSUMO DE ENERGIA

O relatório de consumo de energia foi criado com base nas medições realizadas a cada minuto através do CLP e o cartão 750-493, conforme descrito no tópico 4.1. Os valores informados pelo CLP são dados em kW (quilowatts) sendo assim é necessária a transformação para kWh (quilowatts-hora) para cálculo do consumo e custos de acordo com o sistema tarifário praticado pela distribuidora de energia. Esta tarefa foi excetuada através uma rotina dentro do Power BI, onde é determinado a quantidade total de quilowatts e o tempo em horas do intervalo de leitura, resultando no quilowatt-hora (FIGURA 36). O resultado entregue pela equação permite assim encontrar o consumo (kWh) para produção das peças.

FIGURA 36 – CÁLCULO DE CONSUMO (KWH) POR METROS PRODUZIDOS

```

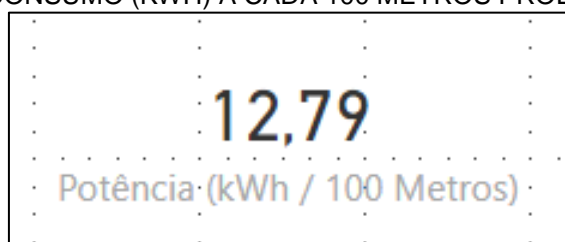
1 Potencia (kWh / 100 Metros) =
2   var Pot =
3     DIVIDE(SUM('wagodb2 data3'[Potencia]);MAX('wagodb2 data3'[ID_Horas]))
4   var Metros =
5     CALCULATE(SUM('wagodb2 data3'[Velocidade Real]);'wagodb2 data2'[Status] =
6                                     "Produção")
7   return
8     100*DIVIDE(Pot;Metros)

```

FONTE: O AUTOR (2019)

A variável *Pot* (linha 2) é obtida pela razão entre a soma dos quilowatts consumidos da coluna *Potência* e valor máximo da coluna *ID\_Horas* obtendo o valor em quilowatts-hora. A coluna *ID\_Horas* transforma a coluna *ID* (minutos) para horas e seu valor máximo indica o tempo total de leitura. Já a variável *metros* retorna à quantidade metros produzidos quando o status for igual a *Produção*, ou seja, quantidade metros de material convertidos em produto acabado. Na linha 7 obtém-se valor do consumo total (kWh) por metros produzidos. Este valor é multiplicado por 100 para facilitar a comparação com os indicadores já existentes na empresa. O resultado (FIGURA 37) obtido para o intervalo de leitura foi de 12,79 kWh a cada 100 metros.

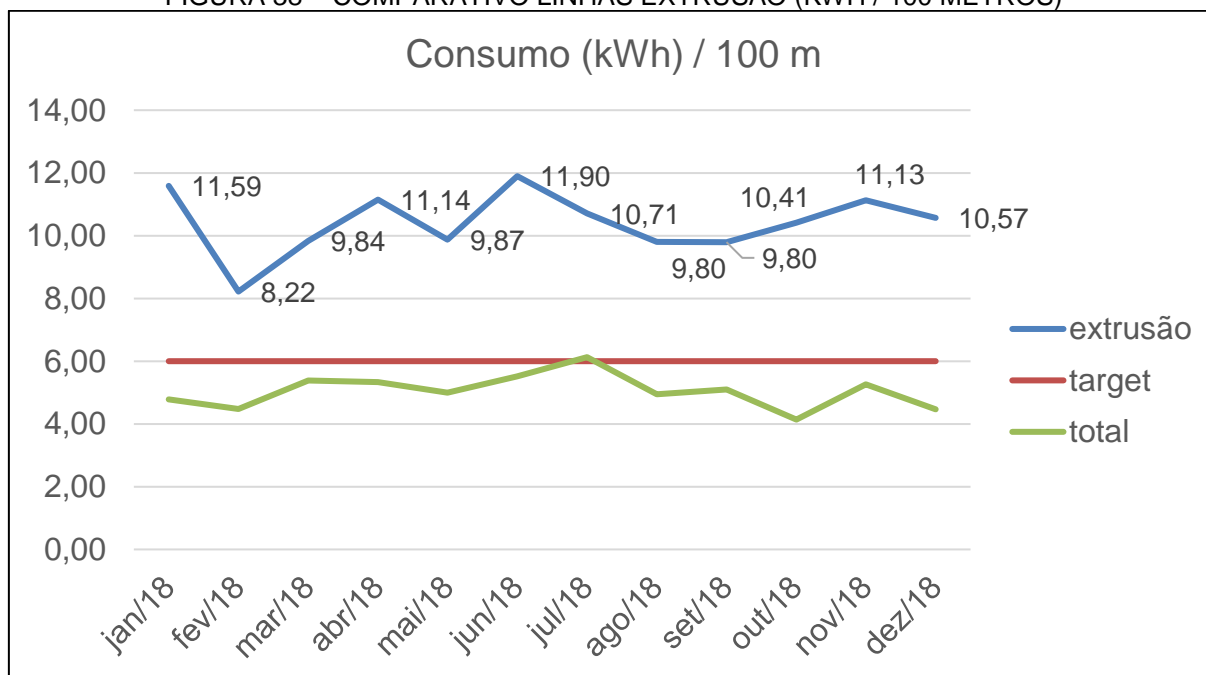
FIGURA 37 – CONSUMO (KWH) A CADA 100 METROS PRODUZIDOS



FONTE: O AUTOR (2019)

O valor está acima dos resultados das linhas de extrusão dos últimos 12 meses e acima da meta de 6 kWh/100 metros. A FIGURA 38 mostra um gráfico comparativo entre os resultados das linhas de extrusão (azul) e resultado total da fábrica (verde).

FIGURA 38 – COMPARATIVO LINHAS EXTRUSÃO (KWH / 100 METROS)

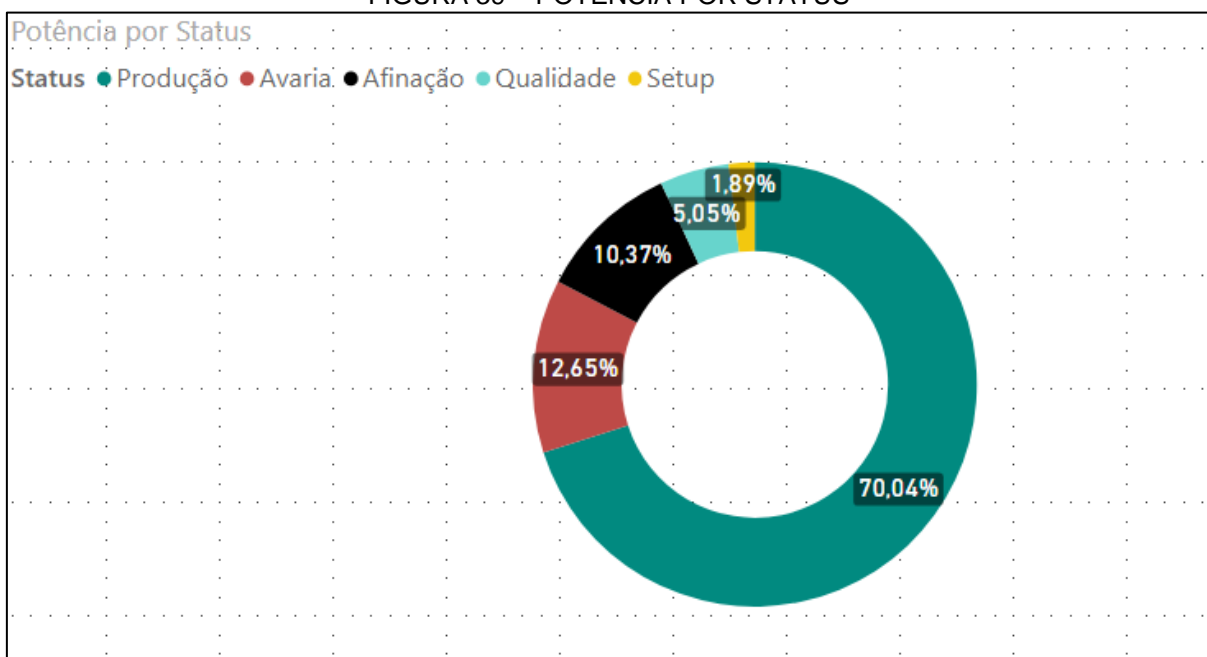


FONTE: O AUTOR (2019)

Com relação a extrusão pode-se observar que o desempenho não tem sido satisfatório no que se refere ao consumo de energia. Isto se deve ao consumo e energia excessivos nas atividades que não agregam valor ao produto como avarias e correções durante a produção. Atividades diretas, como setup e afinação também podem ser aprimoradas reduzindo o consumo. Nesta etapa podem ser desenvolvidas ações como documentação dos parâmetros de linha, treinamentos, conservação do ferramental e padronização dos processos, isto não só reduzirá o consumo de energia, mas também auxiliará nos resultados de eficiência e rejeitos. A FIGURA 39 detalha a energia consumida por status. A eficiência final é de 70,04 %.

Para determinação dos custos foi calculado a média das tarifas aplicadas (TABELA 2) para horário de ponta e fora de ponta, durante o ano de 2018, considerando as bandeiras tarifárias (verde, amarela e vermelha). A empresa atualmente se enquadra na tarifa horo-sazonal verde industrial subgrupo A4 (2,3 a 25 kV).

FIGURA 39 – POTÊNCIA POR STATUS



FONTE: O AUTOR (2019)

TABELA 2 – CUSTO UNITÁRIO (R\$/KWH) - 2018

Mês	Horário	Consumo (kWh)	Custo Total (R\$/Mês)	Custo (R\$/kWh)	Custo Unitário Total (R\$/kWh)
jan/18	Fora	112.962	56499	0,500	0,637
	Ponta	12.564	23418	1,864	
fev/18	Fora	101.589	49895	0,491	0,622
	Ponta	11.130	20166	1,812	
mar/18	Fora	122.976	58113	0,473	0,605
	Ponta	13.837	24601	1,778	
abr/18	Fora	116.346	56414	0,485	0,613
	Ponta	12.724	22646	1,780	
mai/18	Fora	127.080	64831	0,510	0,629
	Ponta	12.891	23173	1,798	
jun/18	Fora	130.482	78310	0,600	0,735
	Ponta	14.922	28611	1,917	
jul/18	Fora	137.766	88043	0,639	0,769
	Ponta	14.403	29027	2,015	

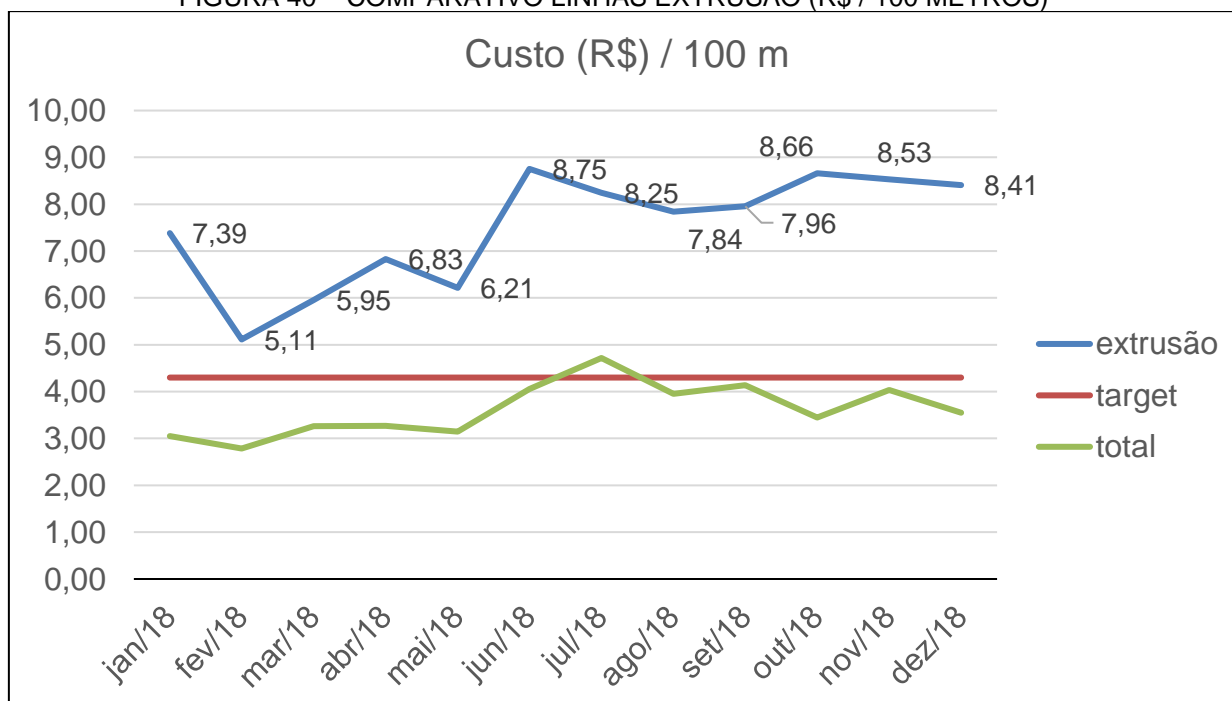
(continua)

Mês	Horário	Consumo (kWh)	Custo Total (R\$/Mês)	Custo (R\$/kWh)	Custo Unitário Total (R\$/kWh)
set/18	Fora	135.697	90941	0,670	0,812
	Ponta	15.227	31571	2,073	
out/18	Fora	101.335	70384	0,695	0,832
	Ponta	10.551	22662	2,148	
nov/18	Fora	116.267	72580	0,624	0,766
	Ponta	12.527	26114	2,085	
dez/18	Fora	64.671	42856	0,663	0,794
	Ponta	6.680	13824	2,069	
Média					0,718

FONTE: O AUTOR (2019)

O custo unitário médio obtido foi R\$ 0,718 por kWh resultando em custo total de R\$ 9,09 por kWh/100 metros. Este valor também é superior à média anual (7,49 kWh/100 metros) do ano de 2018 e acima da meta de 4,30 kWh/100 metros, conforme apresentado na FIGURA 40.

FIGURA 40 – COMPARATIVO LINHAS EXTRUSÃO (R\$ / 100 METROS)

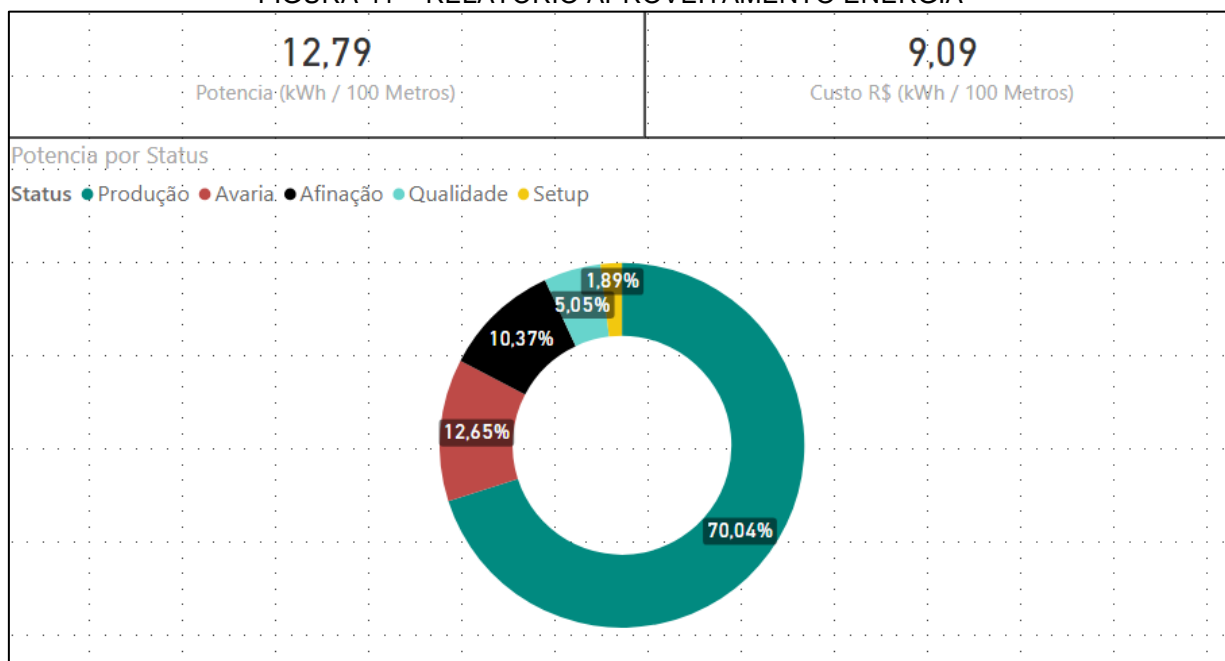


FONTE: O AUTOR (2019)

As informações obtidas acima compuseram o relatório denominado de Aproveitamento de Energia (FIGURA 41) no Power BI. Estas informações são muito pertinentes pois detalham o desempenho da linha de produção ao invés da análise

global já realizada. Isto permite monitorar o desempenho de cada linha, identificando os padrões de comportamento da extrusora, apontando a necessidade de reposição de peças e realização de manutenções preventivas, além de determinar as peças que mais impactam no consumo de energia.

FIGURA 41 – RELATÓRIO APROVEITAMENTO ENERGIA



FONTE: O AUTOR (2019)

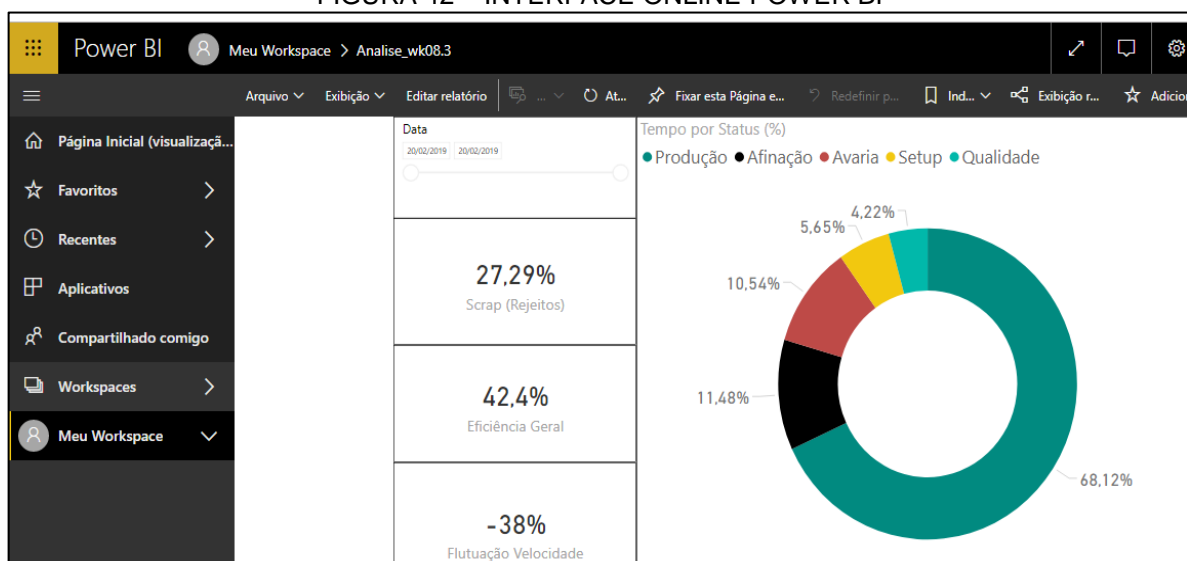
#### 5.4 PUBLICAÇÃO DE ANÁLISES E RELATÓRIO PARA SMARTPHONE

Todos relatórios criados podem ser compartilhados. Na versão utilizada o Power BI permite que os relatórios salvos na nuvem e então acessados remotamente pela conta pessoal cadastrada (FIGURA 42). O serviço funciona como um conta de e-mail e ao acessar a plataforma é possível realizar a visualização interativa dos dados e ainda gerar novos gráficos.

O Power BI também oferece a opção de alterar os relatórios para visualização no smartphone (FIGURA 43). Nesta opção o software simula a tela de um celular é possível arrastar e redimensionar os elementos criados montando um novo layout. Esta configuração será utilizada para apresentar os dados ao usuário no aplicativo.

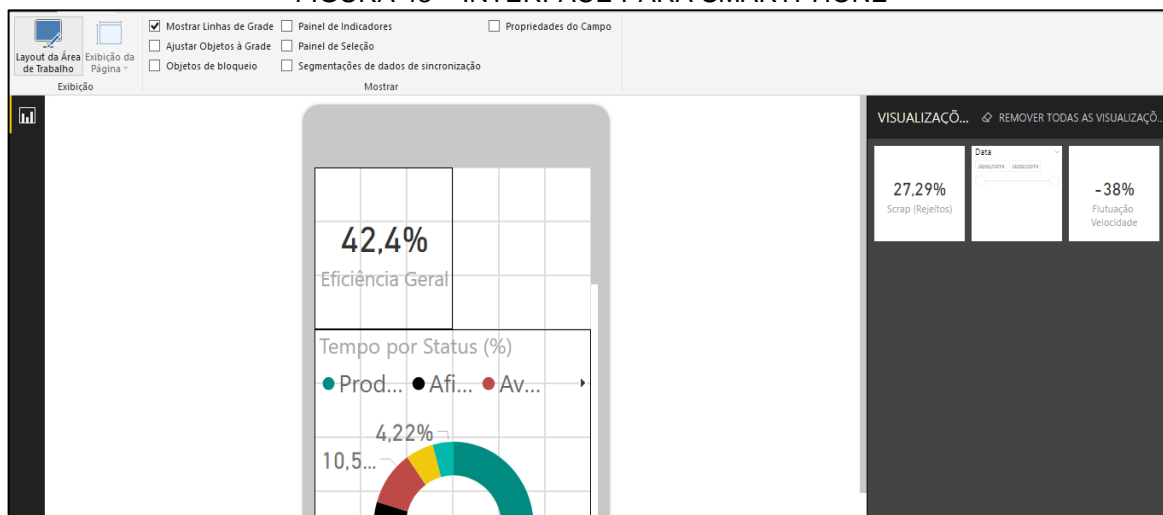


FIGURA 42 – INTERFACE ONLINE POWER BI



FONTE: O AUTOR (2019)

FIGURA 43 – INTERFACE PARA SMARTPHONE



FONTE: O AUTOR (2019)

## 6 CONSIDERAÇÕES FINAIS

### 6.1 DIFICULDADES E DESAFIOS FUTUROS

Durante a execução do trabalho as principais dificuldades superadas foram as de encontrar documentação necessária para realizar a integração de todos os elementos responsáveis pela coleta, armazenamento e processamento dos dados. Em partes a ferramenta desenvolvida neste trabalho reuniu conceitos e aplicações ainda não usuais nas empresas. Normalmente as soluções comerciais são adquiridas em pacotes que fornecem os serviços para todas as etapas, entretanto requerem um alto investimento e o *know-how* dos serviços fica sob controle da empresa contratada para prestação de serviços.

As dificuldades encontradas durante a execução do projeto também auxiliaram na ampliação do escopo inicial resultando na busca por novos recursos para preencher as lacunas observadas na coleta das informações. Foi neste ponto onde a aplicação Java desempenhou o papel mais importante. Não era possível apenas coletar e armazenar os dados, como previsto inicialmente, se estes não pudessem ser convertidos posteriormente em informações relevantes capazes de aprimorar o processo produtivo. As informações enviadas pela aplicação permitiram a segmentação dos dados, a aproximação da linguagem dos relatórios virtuais com os relatórios de produção existentes e uma maior robustez nas análises a serem desenvolvidas.

Apesar dos bons resultados iniciais e aumento da transparência do processo para o futuro é necessário a avaliação da ferramenta aplicada a linha de produção e seu impacto na melhoria dos seus resultados. Se comprovada a eficiência da ferramenta no auxílio da tomada de decisões é possível replicar o modelo para outras linhas de produção, após uma análise de custos. Será preciso a aquisição de um controlador lógico programável e encoder por linha, além de um servidor e serviço de banco de dados mais robustos. A licença do software Power BI também deve ser alterada para a publicação dos relatórios com outros colaboradores da empresa. Há espaço também para melhorias na interface da aplicação Java, como a substituição do lançamento manual das informações da ordem de produção por leitores de QR code. Isto evitaria erros de lançamento e reduziria o tempo gasto nesta atividade. Além disso, os dados coletados poderão gerar outros relatórios como gráficos de Pareto e

análises de produção de lotes econômico, considerando os produtos com maior dificuldade de produção.

## 6.2 CONCLUSÕES

A ferramenta criada obteve resultados positivos. Sua utilização para monitoramento da linha de produção e análise da eficiência produtiva apontou para oportunidades melhorias e identificação dos pontos críticos do processo. Para cálculo de eficiência conforme apresentado no tópico 5.1.1 os dados coletados pelo controlador lógico programável e pela interface Java entregaram a eficiência produtiva. Os metros produzidos, informados pelo encoder ao CLP, representam a quantidade real de peças. Já a quantidade ideal de metros solicitados foi calculada com base no parâmetro de velocidade informado na ordem de produção. A razão entre as duas variáveis, respectivamente, resultou no valor de eficiência da linha de produção durante o intervalo de tempo em que linha estava sendo monitorada.

Considerando os indicadores eficiência, scrap e consumo os resultados ficaram abaixo da meta estabelecida pela empresa. Os resultados negativos foram influenciados principalmente pela flutuação negativa de velocidade (velocidade abaixo do solicitado pela ordem de produção), paradas para setup de linha e avarias. Também houveram paradas menores por incidentes de qualidade devido a instabilidade de processo. A aplicação da ferramenta evidenciou os pontos que merecem atenção para redução dos desperdícios. Além disso será possível criar um histórico e acompanhar a performance de cada item após as ações de correção seja elas de treinamento e documentação, desenvolvimento de novos meios de produção, manutenções preventivas e corretivas ou aperfeiçoamento de ferramentas.

A análise do consumo de energia, adicionada ao longo do desenvolvimento do projeto, enriqueceu o estudo realizado. Foi possível observar o consumo de energia da linha durante as etapas de produção. Ao longo do tempo estes dados mostram as performances obtidas a cada produção do mesmo item e se houveram alteração no consumo máximo de energia, auxiliando no acompanhamento do desempenho dos motores da extrusora (principais consumidores de energia). Além disso, é possível traçar os custos envolvidos, analisando não apenas a eficiência produtiva, mas também comercial. A depender da evolução da ferramenta pode-se criar novos

espaços de análises impactando na política de preços e a alteração para meios mais vantajosos de produção.

Por último os resultados obtidos foram informados na plataforma online do software Power BI criando uma interface de comunicação com o usuário. A flexibilidade deste software permite a criação de diferentes relatórios, além dos criados previamente, a fim de explorar ainda mais as informações fornecidas pelos sensores e o CLP. É possível também combinar informações e criar *dashboards* contendo diferentes componentes do processo (energia, metros, velocidade de produção, entre outros). Os dados podem ser acessados a qualquer momento, reduzindo assim o tempo para toda de decisões, sendo esta sua principal vantagem.

De maneira geral estas informações auxiliaram na otimização do perfil de produção abrindo espaço para novos estudos, determinando, por exemplo, o custo de produção por peça e o lote econômico de produção, além de criar uma espécie de memória do produto contendo sua performance produtiva ao longo do tempo. A sua utilização também irá reduzir a quantidade diária de arquivos físicos (relatórios) gerados entregando ao operador mais tempo para executar outras atividades. É possível adicionar mecanismos para facilitar o lançamento dos dados na interface java como leitores de QR code, que podem reduzir o tempo desta operação e evitar erros de digitação. Outras alternativas como luzes indicação podem auxiliar o operador a identificar o status da linha de produção e aumentar a atenção para as diferentes etapas de produção.

Na quarta revolução industrial a informação e a conectividade desempenham papel central. A abordagem deste trabalho durante o desenvolvimento da ferramenta de análise de eficiência foi guiada por estes dois elementos, buscando aumentar a transparência dos processos, reduzindo as etapas de documentação e o tempo de reatividade para resolução dos problemas. O acesso às informações em uma plataforma online, no computador e celular, com a possibilidade de acesso de diferentes usuários, também representa um grande avanço ao estado atual e favorece ainda mais o aprimoramento das etapas produtivas.

Apesar do início não muito sólido da indústria 4.0 no Brasil, onde ainda restam muitas dúvidas sobre os reais benefícios desta nova revolução, a difusão de tecnologias como *cloud computing*, internet das coisas, big data, robótica avançada, *machine learning* e outras tem aberto espaço para discussão deste temas dentro dos centros acadêmicos preparando engenheiros, técnicos e tecnólogos capazes de

operar estas ferramentas fornecendo mão de obra qualificada. Os custos de implantação também tendem a reduzir com crescimento das alternativas disponíveis no mercado. Fica claro também, através do conteúdo internacional voltado para este tema, que as empresas ao redor do mundo estão migrando para este novo padrão de produção. Apesar de ainda não haver garantias financeiras do retorno do investimento, há uma preocupação crescente com a perda de competitividade. Para pequenas e médias empresas onde é extremamente importante garantir a satisfação do cliente, realizar a gestão de suprimentos e produzir mais com menos estas tecnologias, se bem empregadas, podem fortalecer o negócio. A necessidade de evolução também abre espaço para o surgimento de startups e empresas nacionais capazes de adaptar o maquinário existente nas fábricas para informar dados relevantes de produção a um custo acessível e garantindo a segurança das informações.

É importante ressaltar que os conceitos da indústria 4.0 não substituem as ferramentas tradicionais do *lean manufacturing*. De nada vale um sistema robusto para coleta, armazenamento e análise de dados se o chão de fábrica não opera no mesmo nível de eficiência. As duas metodologias se complementam e as tecnologias habilitadoras atuam para facilitar o processo de identificação dos pontos ainda não maduros dentro das indústrias e que necessitam atenção facilitando a priorização das atividades, a tomada de decisões e aplicação das ferramentas *lean*.

## REFERÊNCIAS

AMAZONAS, J. R. Opportunities, challenges for internet of things technologies. IN: VERMESAN, O.; FRIESS, P. (Orgs). Internet of things - global technological and societal trends from smart environments and spaces to green ICT. **River Publishers**, 2010.

AMERICAN SOCIETY FOR QUALITY. Survey management. **Quality Magazine**. Estados Unidos, 2014. Disponível em: <https://www.qualitymag.com/articles/91688-smart-manufacturing>. Acesso: 23 abr. 2018.

APACHE FRIENDS. **About Apache Friends**. Disponível em: <https://www.apachefriends.org/about.html>. Acesso em: 18 out. 2018.

ARMBRUST, M.; FOX, A.; GRIFFITH, R.; JOSEPH, A. D.; KATZ, R. H.; KONWINSKI, A.; LEE, G.; PATTERSON, D. A.; RABKIN, A.; STOICA, I.; ZAHARIA, M. About the clouds: a Berkeley view of cloud computing. **Electrical Engineering and Computer Sciences**. Technical Report nº UCB/EECS – 2009-28. Fev, 2009.

BRASIL. Competitividade Brasil 2017-2018 – comparação com países selecionados. **Confederação Nacional da Indústria**. Brasília 2018.

BRASIL. Desafios para a indústria 4.0 no Brasil. **Confederação Nacional da Indústria**. Brasília, 2016a.

BRASIL. Special Survey – Indústria 4.0. Indústria 4.0: a new challenge for Brazilian industry. **Confederação Nacional da Indústria. Indicadores CNI**. Ano 17, n. 2. Brasília, 2016b.

BATEMAN, K. Industry 4.0 and the cloud: a match made in heaven. **Information Age**. Topics – Cloud & Edge Computing. Disponível em: <https://www.information-age.com/industry-4-0-cloud-match-heaven-123471806/>. Acesso em: 20 out. 2018.

BOYES, H.; HALLAQ, B.; CUNNINGHAM, J.; WATSON, T. The industrial internet of things (IIoT): An analysis framework. **Computers in Industry**, v. 101, p. 1-12, jun. 2018.

CAPGEMINI CONSULTING. **Industry 4.0 – The Capgemini Consulting View**. Sharpening the Picture beyond the Hype. Paris, 2014.

CASAGRAS. CASAGRAS an **EU framework 7 project**. Disponível em: <https://docbox.etsi.org/zArchive/TISPAN/Open/IoT/low%20resolution/www.rfidglobal.eu%20CASAGRAS%20IoT%20Final%20Report%20low%20resolution.pdf>. Acesso: 7 mai. 2018.

EMPRESA DE PESQUISA ENERGÉTICA. **Consumo Anual de Energia Elétrica por classe (nacional)**. Disponível em: <http://www.epe.gov.br/pt/publicacoes-dados-abertos/publicacoes/Consumo-Anual-de-Energia-Eletrica-por-classe-nacional>. 2017. Acesso em: 20 out. 2018.

CÔRREA, T. S.; ALMEIDA, D. E. C.; NETO, A. F. G. Comparação entre banco de dados relacional e não relacional em arquitetura distribuída. **Instituto nacional de telecomunicações – Inatel**. III Seminário de desenvolvimento em SOA com cloud computing e conectividade. Set, 2017.

CRUZ, N. M. P. **Implementação de ferramentas Lean Manufacturing no processo de injeção de plásticos**. Universidade do Minho. Escola de Engenharia. Dissertação mestrado. Out. 2013.

DELOITTE. Industry 4.0 – Challenges and solutions for the digital transformation and use of exponential Technologies. **Deloitte**. Suíça, 2014.

ERLACH, K. Value stream design: the way towards a lean factory. **New York Springer**. Berlim, 2013.

GEISSBAUER, R.; VEDSO, J.; SCHRAUF, S. **Global Industry 4.0: Building the digital enterprise**. PWC, 2016.

GOOGLE. **Announcing Data Studio: our free, new, data visualization product**. Disponível em: <https://analytics.googleblog.com/2016/05/announcing-data-studio-our-free-new.html>. Acesso em: 20 out. 2018.

GUBBI, J.; BUYYA, R.; MARUSIC, S.; PALANISWAMI, M. Internet of things (IoT): A vision, architectural elements, and future directions. **Future Generation Computer Systems**, v. 29, p.1645-1660, fev. 2013.

KOLBERG, D.; ZÜHLKE, D. Lean Automation enabled by Industry 4.0 Technologies. **Science Direct**. V. 18, N. 3, P. 1870-1875. 2015.

LIKER, J. K. **The Toyota Way: 14 Management Principles from the World's Greatest Manufacturer**. McGraw Hill, 2004.

MELL, P.; GRANCE, T. The NIST definition of cloud computing – Recommendations of the National Institute of Standards and Technology. **National Institute of Standards and Technology - U.S. Department of Commerce** Special Publication 800-145. Gaithersburg, MD 20899-8930. Set. 2011.

MICROSOFT. O que é power bi. Disponível em: <https://docs.microsoft.com/pt-br/power-bi/power-bi-overview>. fev. 2019. Acesso em: 15 fev 2019.

MOREIRA, L. Desconfiança é mais o maior obstáculo da nuvem. **Computerworld**. Disponível em: <https://computerworld.com.br/2016/03/02/desconfianca-nao-e-mais-o-maior-obstaculo-da-nuvem/>. Acesso em: 20 out. 2018.

OLIVEIRA, D. P.; REBOUÇAS. **Sistemas de informações gerenciais: estratégicas, táticas, operacionais**. São Paulo: Atlas, 1992

ONO, T. Toyota production system: beyond large-scale production. **Productivity Press**. Cambridge, Massachusetts, 1988.

PETRUZELLA, F. D. **Controladores lógicos programáveis**. Porto Alegre: Amgh, 2013.

REZENDE, R. **A história do banco de dados**. Disponível em: <https://www.devmedia.com.br/a-historia-dos-banco-de-dados/1678>. 2016. Acesso em: 20 out. 2018.

ROUSE, M. **Big data analytics**. Disponível em: <https://searchbusinessanalytics.techtarget.com/definition/big-data-analytics>. 2017. Acesso em: 15 out. 2018.

ROUSE, M. **Industrial internet of things (IIoT)**. Disponível em: <https://internetofthingsagenda.techtarget.com/definition/Industrial-Internet-of-Things-IIoT>. 2018. Acesso em: 15 out. 2018.

SANTOS, G. **Como aprimorar a gestão de energia na manufatura com a ISO 50001**. Automação industrial. Disponível em:



<https://www.automacaoindustrial.info/como-aprimorar-a-gestao-de-energia-na-manufatura-com-a-iso-50001/>. Acesso em: 12 out. 2018.

SCHRÖDER, C. The challenges of Industry 4.0 for small and medium-sized enterprises. **Friedrich ebertstiftung**. Alemanha, 2017.

SHROUF, F.; ORDIERES, J.; MIRAGLIOTTA, G. Smart factories in Industry 4.0: a review of the concept and of energy management approached in production based on the internet of things paradigm. **International Conference on Industrial Engineering and Engineering Management**. Dec. 2014.

SILVA, E. M. **Lean Manufacturing e Indústria 4.0**. Disponível em: <https://www.linkedin.com/pulse/lean-manufacturing-e-ind%C3%BAstria-40-edson-miranda-da-silva/>. 2006. Acesso em: 20 out. 2018.

SILVEIRA, L.; LIMA, W. Q. Um breve histórico conceitual da Automação Industrial e Redes para Automação Industrial. **Programa de Pós-graduação em Engenharia Elétrica**. Universidade Federal do Rio Grande do Norte. Natal, mai. 2003.

TAURION, Cezar. **Cloud computing: computação em nuvem: transformando o mundo da tecnologia da informação**. Rio de Janeiro: Brasport, 2009. 205p. ISBN:978-85- 7452-423-8

ORACLE. **Big Data SQL**. Disponível em: <https://www.oracle.com/corporate/features/big-data-sql/>. 2016. Acesso em: 20 out. 2018.

ORACLE. The history of Java Technology. Disponível em: <https://www.oracle.com/technetwork/java/javase/overview/javahistory-index-198355.html> 2018. 2016. Acesso em 3 dez. 2018.

UDEN, L.; HE, W. How the internet of things can help knowledge management: a case study from the automotive domain. **Journal of Knowledge Management**, v.21, n.1, p.57-70, 2017.

WAGO. **Minden Headquarters**. Disponível em: <https://www.wago.com/us/minden> Jul. 2017. Acesso em: 19 out. 2018.

WOMACK, J. P., & JONES, D. T. **Lean Thinking: Banish Waste and Create Wealth in your Corporation.** New York, USA: Simon & Schuster, 1996.

## APÊNDICE A – CÓDIGO CLP WAGO

### 1) Configuração MySQL e Variáveis Banco de Dados

```

PROGRAM Comunica
VAR
  oMySQL : MySQL_Context; (* Structure for keeping login and status informations *)

  oMySQL_Login      : MySQL_Login;          (* Structure for keeping login and status informations *)
  sHost              : STRING := '192.168.1.33'; (* IP or hostname where MySQL-server is working *)
  uiPort             : UINT := 3306;         (* TCP-Port where MySQL-server is listening, typ. 3306 *)
  sUsername          : STRING := 'gabriel';  (* Name of database account *)
  sPassword          : STRING := 'andrei';   (* Password of database account *)
  sDatabase          : STRING := 'WagoDB2';  (* Name of schema to use initially - Optional param *)
  xConnect           : BOOL;                (* Start execution on a rising edge and reset internally when done or on timeout. *)
  diError            : DINT;               (* Numeric return value - more readable in hex *)
  sStatus            : STRING(500);        (* Textual description of current state or error *)
  xConnected         : BOOL;               (* TRUE: when database connection is established *)
  sTable             : STRING := 'data3';

  asSqlStatement1   : ARRAY[0..gcMySQL_iSqlUpperBound] OF STRING(gcMySQL_iSqlLength);
  oMySQLExec1       : MySQL_Execute; (* Structure for keeping login and status informations *)
  xExec1            : BOOL;             (* Start execution on a rising edge and reseted internally when done or on timeout. *)
  diErrorExec1      : DINT;            (* Numeric return value - more readable in hex *)
  sStatusExec1      : STRING(500);     (* Textual description of current state or error *)

  oMySQL_Logout     : MySQL_Logout; (* Structure for keeping login and status informations *)
  xDisconnect       : BOOL;           (* Start execution on a rising edge and reset internally when done or on timeout. *)
  diErrorLogout     : DINT;          (* Numeric return value - more readable in hex *)
  sStatusLogout     : STRING(500);   (* Textual description of current state or error *)

  Corrente: INT;
  AUX: REAL;
  Potencia: INT;
  AUX2: REAL;
  AUX3: REAL;
  Temp: BOOL;
  Contador: CTU;

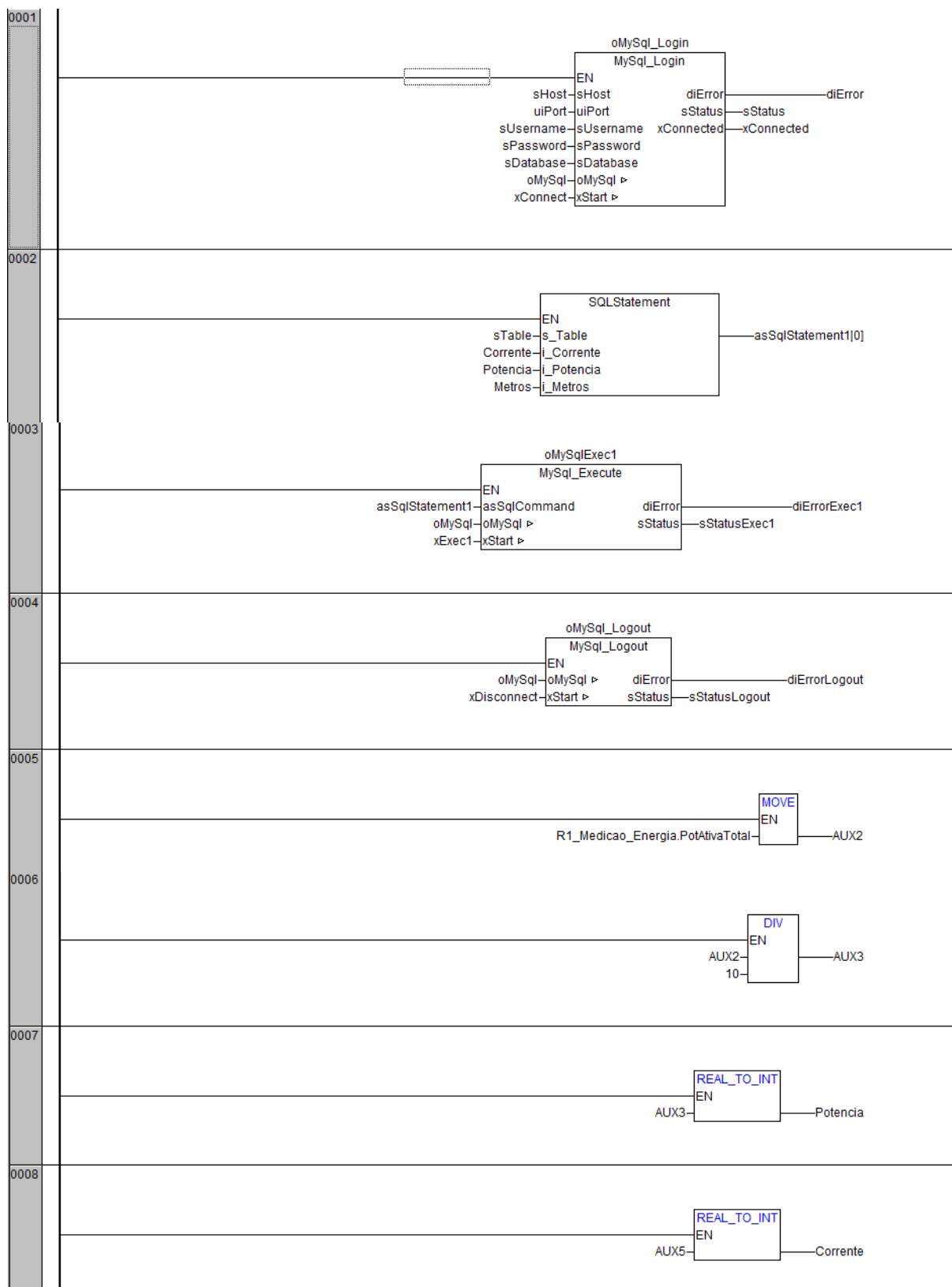
  saida: BOOL;

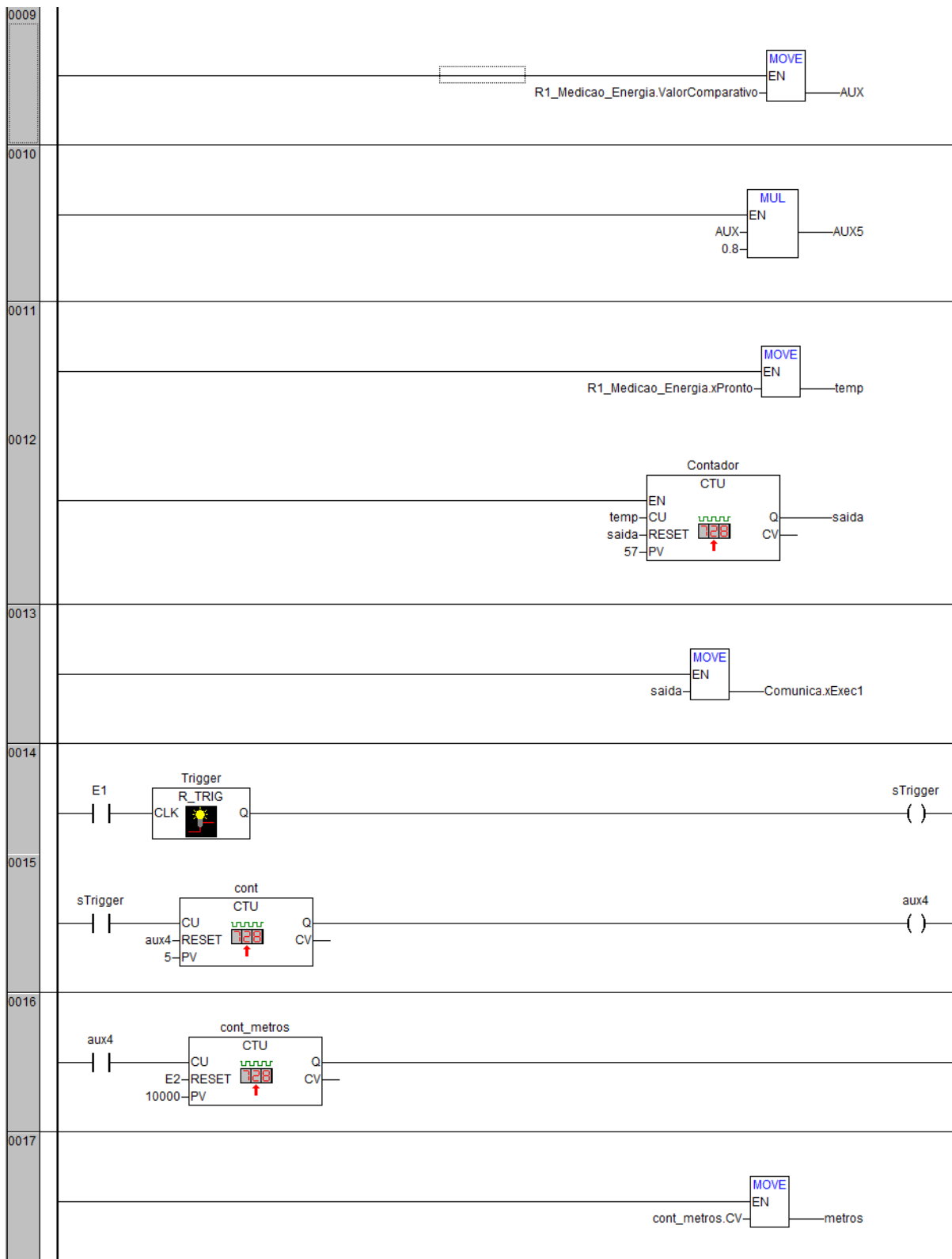
  Trigger: R_TRIG;
  sTrigger: BOOL;
  Metros: INT;
  cont: CTU;
  smetros: BOOL;
  aux4: BOOL;
  cont_metros: CTU;

  AUX5: REAL;
END_VAR

```

## 2) Configuração MySQL e Variáveis Banco de Dados – Diagrama de Blocos





### 3) Configuração Tabela Banco de Dados

```

0001 FUNCTION SQLStatement : STRING
0002 VAR_INPUT
0003     s_Table: STRING;
0004     i_Corrente : INT;
0005     i_Potencia : INT;
0006     i_Metros:INT;
0007
0008 END_VAR
0009 VAR
0010     sTemp   : STRING(500)   := 'INSERT INTO';
0011 END_VAR
0012
0001 sTemp := CONCAT(sTemp, s_Table);
0002 sTemp := CONCAT(sTemp, '(Corrente,Potencia,Metros) VALUES ( ');
0003 sTemp := CONCAT(sTemp, INT_TO_STRING(i_Corrente));
0004 sTemp := CONCAT(sTemp, ', ' );
0005 sTemp := CONCAT(sTemp, INT_TO_STRING(i_Potencia));
0006 sTemp := CONCAT(sTemp, ', ' );
0007 sTemp := CONCAT(sTemp, INT_TO_STRING(i_Metros));
0008 sTemp := CONCAT(sTemp, ') ' );
0009 SQLStatement := sTemp;
0010

```

### 4) Configuração Módulo de Medição de Energia

```

0001 PROGRAM R1_Medicao_Energia
0002 VAR
0003     (*Variáveis para o bloco do módulo trifásico*)
0004     (*Variáveis de entrada*)
0005     ModuloEnergia: Fb750_493_Master3Phase_IPC;
0006
0007
0008     aDadosEntrada AT %IB0: ARRAY [0..11] OF BYTE; (*Array do processamento do cartão*)
0009     aDadosSaida   AT %QB0: ARRAY [0..11] OF BYTE; (*Array do processamento do cartão*)
0010
0011     typConfiguracaoBloco: typConfig3Phase; (*Estrutura de configuração do cartão*)
0012
0013     (*Variáveis de saída*)
0014     adiPotenciaAtiva: ARRAY[1..3] OF DINT; (*0,1 W*)
0015     xPronto: BOOL;
0016     ValorComparativo: REAL;
0017     PotAtivaTotal: REAL;
0018     Grafico: REAL;
0019
0020     adwCurrent: REAL;
0021     corrente: REAL;
0022
0023     PotAtivaTotal := DINT_TO_REAL(adiPotenciaAtiva[1] + adiPotenciaAtiva[2] + adiPotenciaAtiva[3]); (*Soma para resultar na potência total*)
0024     ValorComparativo := DINT_TO_REAL(ModuloEnergia.adwCurrent[1]);
0025 END_VAR
0026
0001 Config();
0002
0003     (*Instância bloco que faz leitura de energia*)
0004     ModuloEnergia(
0005         xEnable:= TRUE,
0006         tCycleTime:= #1s,
0007         wCurrentTransformerRatioL1:= RelacaoTCF1,
0008         wCurrentTransformerRatioL2:= RelacaoTCF2,
0009         wCurrentTransformerRatioL3:= RelacaoTCF3,
0010         abIn_Data_750_493:= aDadosEntrada,
0011         abOut_Data_750_493:= aDadosSaida,
0012         typConfig3Phase:= typConfiguracaoBloco,
0013         xReady=> xPronto,
0014         adiEffectivePower=> adiPotenciaAtiva);

```

## APÊNDICE B – CÓDIGO JAVA

### 1) Connection Factory

```

package connection;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.logging.Level;
import java.util.logging.Logger;

/**
 *
 * @author Samuelson
 */
public class ConnectionFactory {

    private static final String DRIVER = "com.mysql.jdbc.Driver";
    private static final String URL = "jdbc:mysql://localhost:3306/wagodb2";
    private static final String USER = "root";
    private static final String PASS = "";

    public static Connection getConnection() {
        try {
            Class.forName(DRIVER);
            return DriverManager.getConnection(URL, USER, PASS);
        } catch (ClassNotFoundException | SQLException ex) {
            throw new RuntimeException("Erro na conexão: ", ex);
        }
    }

    public static void closeConnection(Connection con) {
        try {
            if (con != null) {
                con.close();
            }
        } catch (SQLException ex) {
            Logger.getLogger(ConnectionFactory.class.getName()).log(Level.SEVERE, null, ex);
        }
    }

    public static void closeConnection(Connection con, PreparedStatement stmt) {

        closeConnection(con);

        try {

            if (stmt != null) {
                stmt.close();
            }
        }
    }

```

```

    } catch (SQLException ex) {
        Logger.getLogger(ConnectionFactory.class.getName()).log(Level.SEVERE, null, ex);
    }
}

public static void closeConnection(Connection con, PreparedStatement stmt, ResultSet rs) {

    closeConnection(con, stmt);

    try {

        if (rs != null) {
            rs.close();
        }

    } catch (SQLException ex) {
        Logger.getLogger(ConnectionFactory.class.getName()).log(Level.SEVERE, null, ex);
    }
}
}

```

## 2) Dados

```

package model.bean;

/**
 *
 */
public class Dados {

    private int id;
    private String PartNumber;
    private String Lote;
    private Double Velocidade;
    private Object Status;

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public String getPartNumber() {
        return PartNumber;
    }

    public void setPartNumber(String PartNumber) {
        this.PartNumber = PartNumber;
    }
}

```



```

    }

    public String getLote() {
        return Lote;
    }

    public void setLote(String Lote) {
        this.Lote = Lote;
    }

    public Double getVelocidade() {
        return Velocidade;
    }

    public void setVelocidade(Double Velocidade) {
        this.Velocidade = Velocidade;
    }

    public Object getStatus() {
        return Status;
    }

    public void setStatus(Object Status) {
        this.Status = Status;
    }
}

```

### 3) DadosDAO

```

package model.dao;

import connection.ConnectionFactory;
import java.sql.SQLException;
import java.util.ArrayList;
import java.util.List;
import java.util.logging.Level;
import java.util.logging.Logger;
import model.bean.Dados;
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.util.TimerTask;
import javax.swing.JOptionPane;
import java.util.Timer;

/**
 *
 */
public class DadosDAO {

    public void create(Dados d) {

```

```

Connection con = ConnectionFactory.getConnection();

PreparedStatement stmt = null;

try {
    stmt = con.prepareStatement("INSERT INTO data2
(PartNumber,Lote, Velocidade, Status)VALUES(?,?,?,?)");
    stmt.setString(1, d.getPartNumber());
    stmt.setString(2, d.getLote());
    stmt.setDouble(3, d.getVelocidade());
    stmt.setObject(4, d.getStatus());

    stmt.executeUpdate();

    JOptionPane.showMessageDialog(null, "Salvo com sucesso!");
} catch (SQLException ex) {
    System.out.println(ex);
} finally {
    ConnectionFactory.closeConnection(con, stmt);
}

/**
 *
 * @param d
 */
}

public List<Dados> read() {

    Connection con = ConnectionFactory.getConnection();

    PreparedStatement stmt = null;
    ResultSet rs = null;

    List<Dados> lista = new ArrayList<>();

    try {
        stmt = con.prepareStatement("SELECT * FROM data2");
        rs = stmt.executeQuery();

        while (rs.next()) {

            Dados info = new Dados();

            info.setId(rs.getInt("id"));
            info.setPartNumber(rs.getString("PartNumber"));
            info.setLote(rs.getString("Lote"));
            info.setVelocidade(rs.getDouble("Velocidade"));
            info.setStatus(rs.getObject("Status"));
            lista.add(info);
        }
    } catch (SQLException ex) {
        Logger.getLogger(DadosDAO.class.getName()).log(Level.SEVERE, null, ex);
    } finally {

```

```

        ConnectionFactory.closeConnection(con, stmt, rs);
    }

    return lista;
}

public void delete(Dados d) {

    Connection con = ConnectionFactory.getConnection();

    PreparedStatement stmt = null;

    try {
        stmt = con.prepareStatement("DELETE FROM data2 WHERE id = ?");
        stmt.setInt(1, d.getId());

        stmt.executeUpdate();

        JOptionPane.showMessageDialog(null, "Excluido com sucesso!");
    } catch (SQLException ex) {
        JOptionPane.showMessageDialog(null, "Erro ao excluir: " + ex);
    } finally {
        ConnectionFactory.closeConnection(con, stmt);
    }
}

public void atualiza (Dados d) {

    Timer timer = new Timer();

    final long TEMPO = (1000 * 64);

    TimerTask tarefa = new TimerTask() {

        @Override
        public void run() {
            Connection con = ConnectionFactory.getConnection();
            PreparedStatement stmt = null;
            try {
                stmt = con.prepareStatement("INSERT INTO data2 (PartNumber, Lote, Velocidade,
Status)\n"
                + "SELECT PartNumber, Lote, Velocidade, Status FROM wagodb2.data2\n"
                + "ORDER by id DESC limit 1");
                stmt.executeUpdate();
                //JOptionPane.showMessageDialog(null, "Atualizado com sucesso!");
            } catch (SQLException ex) {
                System.out.println(ex);
            } finally {
                ConnectionFactory.closeConnection(con, stmt);
            }
        }
    };

    timer.scheduleAtFixedRate (tarefa, TEMPO, TEMPO);
}

```

```

    }
}

```

#### 4) Tela de Cadastro de Ordem

```

package view;

import javax.swing.JOptionPane;
import javax.swing.table.DefaultTableModel;
import javax.swing.table.TableRowSorter;
import model.bean.Dados;
import model.dao.DadosDAO;

/**
 *
 */
public final class TelaCadastroOrdem extends javax.swing.JInternalFrame {

    /**
     * Creates new form TelaCadastroOrdem
     */
    public TelaCadastroOrdem() {
        initComponents();
        DefaultTableModel modelo = (DefaultTableModel) jTdados.getModel();
        jTdados.setRowSorter(new TableRowSorter(modelo));

        readJTable();
    }

    public void readJTable() {

        DefaultTableModel modelo = (DefaultTableModel) jTdados.getModel();
        modelo.setNumRows(0);
        DadosDAO ddao = new DadosDAO();

        for (Dados d : ddao.read()) {

            modelo.addRow(new Object[]{

                d.getId(),
                d.getPartNumber(),
                d.getLote(),
                d.getVelocidade(),
                d.getStatus()
            });

        }

    }

}

```

```

* This method is called from within the constructor to initialize the form.
* WARNING: Do NOT modify this code. The content of this method is always
* regenerated by the Form Editor.
*/
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jTextField3 = new javax.swing.JTextField();
    jPanel1 = new javax.swing.JPanel();
    jPanel2 = new javax.swing.JPanel();
    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    txtStatus = new javax.swing.JComboBox<>();
    txtPartNumber = new javax.swing.JTextField();
    txtLote = new javax.swing.JTextField();
    jLabel3 = new javax.swing.JLabel();
    BtnAtualizar = new javax.swing.JButton();
    jButton2 = new javax.swing.JButton();
    txtVelocidade = new javax.swing.JTextField();
    jLabel4 = new javax.swing.JLabel();
    jPanel3 = new javax.swing.JPanel();
    jScrollPane1 = new javax.swing.JScrollPane();
    jTable1 = new javax.swing.JTable();

    jTextField3.setText("jTextField3");

    setName(""); // NOI18N

    jPanel2.setBorder(javax.swing.BorderFactory.createTitledBorder("INFORMAÇÕES"));

    jLabel1.setText("PART NUMBER");

    jLabel2.setText("LOTE");

    txtStatus.setModel(new javax.swing.DefaultComboBoxModel<>(new String[] { "-",
"Setup", "Afinação", "Avaria", "Produção", "Qualidade", "Manutenção" }));
    txtStatus.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            txtStatusActionPerformed(evt);
        }
    });

    jLabel3.setText("STATUS DA LINHA PRODUÇÃO");

    BtnAtualizar.setText("ATUALIZAR");
    BtnAtualizar.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            BtnAtualizarActionPerformed(evt);
        }
    });

    jButton2.setText("EXCLUIR");
    jButton2.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {

```

```

        jButton2ActionPerformed(evt);
    }
});

jLabel4.setText("VELOCIDADE");

javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);
jPanel2.setLayout(jPanel2Layout);
jPanel2Layout.setHorizontalGroup(
    jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel2Layout.createSequentialGroup()
            .add(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(jPanel2Layout.createSequentialGroup()
                    .addGroup(jPanel2Layout.createSequentialGroup()
                        .addContainerGap()
                    )
                )
            )
        )
        .addGroup(jPanel2Layout.createSequentialGroup()
            .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                .addGroup(jPanel2Layout.createSequentialGroup()
                    .addComponent(txtStatus, javax.swing.GroupLayout.Alignment.LEADING,
                        javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
                        javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jLabel3, javax.swing.GroupLayout.Alignment.LEADING)
                    .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                        .addGroup(jPanel2Layout.createSequentialGroup()
                            .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                                .addComponent(BtnAtualizar)
                                .addGap(18, 18, 18)
                                .addComponent(jButton2))
                            .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                                .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                                    .addComponent(jLabel1)
                                    .addComponent(txtPartNumber,
                                        javax.swing.GroupLayout.PREFERRED_SIZE, 155,
                                        javax.swing.GroupLayout.PREFERRED_SIZE))
                                    .addGap(52, 52, 52)
                                )
                            .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                                .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                                    .addComponent(txtLote, javax.swing.GroupLayout.PREFERRED_SIZE,
                                        155, javax.swing.GroupLayout.PREFERRED_SIZE)
                                    .addComponent(jLabel2))
                                    .addGap(31, 31, 31)
                                )
                            .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                                .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                                    .addComponent(jLabel4)
                                    .addComponent(txtVelocidade,
                                        javax.swing.GroupLayout.PREFERRED_SIZE, 171,
                                        javax.swing.GroupLayout.PREFERRED_SIZE))
                                    .addContainerGap(26, Short.MAX_VALUE))
                                )
                            )
                    )
                )
            )
        )
        .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jLabel4)
                .addComponent(txtVelocidade,
                    javax.swing.GroupLayout.PREFERRED_SIZE, 171,
                    javax.swing.GroupLayout.PREFERRED_SIZE))
            )
            .addContainerGap(26, Short.MAX_VALUE))
        )
    );
jPanel2Layout.setVerticalGroup(
    jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel2Layout.createSequentialGroup()
            .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                        .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                            .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                                .addComponent(jLabel1)
                                .addComponent(txtPartNumber,
                                    javax.swing.GroupLayout.PREFERRED_SIZE, 155,
                                    javax.swing.GroupLayout.PREFERRED_SIZE))
                                .addGap(52, 52, 52)
                            )
                            .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                                .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                                    .addComponent(txtLote, javax.swing.GroupLayout.PREFERRED_SIZE,
                                        155, javax.swing.GroupLayout.PREFERRED_SIZE)
                                    .addComponent(jLabel2))
                                    .addGap(31, 31, 31)
                                )
                            .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                                .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                                    .addComponent(jLabel4)
                                    .addComponent(txtVelocidade,
                                        javax.swing.GroupLayout.PREFERRED_SIZE, 171,
                                        javax.swing.GroupLayout.PREFERRED_SIZE))
                                    .addContainerGap(26, Short.MAX_VALUE))
                                )
                            )
                    )
                )
            )
        )
    );

```

```

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(jLabel1)
    .addComponent(jLabel2)
    .addComponent(jLabel4))
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(txtPartNumber, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(txtLote, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(txtVelocidade, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
.addGap(18, 18, 18)
.addComponent(jLabel3)
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
.addComponent(txtStatus, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 29,
Short.MAX_VALUE)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(BtnAtualizar)
    .addComponent(jButton2)))
);

jTdados.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {

    },
    new String [] {
        "ID", "PART NUMBER", "LOTE", "VELOCIDADE", "STATUS"
    }
) {
    boolean[] canEdit = new boolean [] {
        false, false, false, false, false
    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {
        return canEdit [columnIndex];
    }
});
jScrollPane1.setViewportViewView(jTdados);

javax.swing.GroupLayout jPanel3Layout = new javax.swing.GroupLayout(jPanel3);
jPanel3.setLayout(jPanel3Layout);
jPanel3Layout.setHorizontalGroup(
    jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel3Layout.createSequentialGroup()
        .addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addContainerGap()

```

```

        .addComponent(jScrollPane1, javax.swing.GroupLayout.DEFAULT_SIZE, 592,
Short.MAX_VALUE)
        .addContainerGap()
    );
    JPanel3Layout.setVerticalGroup(
        JPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel3Layout.createSequentialGroup()
            .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 238,
javax.swing.GroupLayout.PREFERRED_SIZE))
        );

    javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
    jPanel1.setLayout(jPanel1Layout);
    jPanel1Layout.setHorizontalGroup(
        jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel1Layout.createSequentialGroup()
            .addContainerGap()

        .addGroup(jPanel1Layout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jPanel3, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(jPanel2, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
            .addContainerGap()
        );
    jPanel1Layout.setVerticalGroup(
        jPanel1Layout.createSequentialGroup()
        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addContainerGap()
                .addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                .addComponent(jPanel3, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
            );

    javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        );
    layout.setVerticalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
        );
    pack();

```



```

} // </editor-fold>

private void txtStatusActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
}

@SuppressWarnings("empty-statement")

private void BtnAtualizarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:

    Dados d = new Dados();
    DadosDAO dao = new DadosDAO();

    d.setPartNumber(txtPartNumber.getText());
    d.setLote(txtLote.getText());
    d.setVelocidade(Double.parseDouble(txtVelocidade.getText()));
    d.setStatus(txtStatus.getSelectedItem());
    dao.create(d);
    dao.atualiza(d);
    readJTable();
    txtPartNumber.setText("");
    txtLote.setText("");
    txtVelocidade.setText("");
    txtStatus.setSelectedItem("");

    // DefaultTableModel dtmTabela = (DefaultTableModel) jTdados.getModel();
    // Object [] tabela = {txtPartNumber.getText(), txtLote.getText(),
txtStatus.getSelectedItem()};
    // dtmTabela.addRow(tabela);

}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:

    if (jTdados.getSelectedRow() != -1){

        Dados d = new Dados();
        DadosDAO dao = new DadosDAO();

        d.setId((int) jTdados.getValueAt(jTdados.getSelectedRow(), 0));

        dao.delete(d);

        txtPartNumber.setText("");
        txtLote.setText("");
        txtVelocidade.setText("");
        txtStatus.setSelectedItem("");

        readJTable();

    } else {

```

```
        JOptionPane.showMessageDialog(null, "Selecione uma linha para excluir");
    }
}

// Variables declaration - do not modify
private javax.swing.JButton BtnAtualizar;
private javax.swing.JButton jButton2;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JPanel jPanel1;
private javax.swing.JPanel jPanel2;
private javax.swing.JPanel jPanel3;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTdados;
private javax.swing.JTextField jTextField3;
private javax.swing.JTextField txtLote;
private javax.swing.JTextField txtPartNumber;
private javax.swing.JComboBox<String> txtStatus;
private javax.swing.JTextField txtVelocidade;
// End of variables declaration
}
```

## APÊNDICE C – BANCO DE DADOS XAMPP

### 1) Estrutura Banco de Dados – Java

#	Nome	Tipo	Agrupamento (Collation)	Atributos	Nulo	Predefinido	Comentários	Extra	Ações
<input type="checkbox"/> 1	<b>ID</b>	int(11)			Não	None		AUTO_INCREMENT	Muda  Elimina  Mais
<input type="checkbox"/> 2	<b>Hora</b>	timestamp			Não	CURRENT_TIMESTAMP			Muda  Elimina  Mais
<input type="checkbox"/> 3	<b>PartNumber</b>	text	latin1_swedish_ci		Não	None			Muda  Elimina  Mais
<input type="checkbox"/> 4	<b>Lote</b>	text	latin1_swedish_ci		Não	None			Muda  Elimina  Mais
<input type="checkbox"/> 5	<b>Velocidade</b>	double			Não	None			Muda  Elimina  Mais
<input type="checkbox"/> 6	<b>Status</b>	text	latin1_swedish_ci		Não	None			Muda  Elimina  Mais

### 2) Estrutura Banco de Dados – CLP

#	Nome	Tipo	Agrupamento (Collation)	Atributos	Nulo	Predefinido	Comentários	Extra	Ações
<input type="checkbox"/> 1	<b>Id</b>	int(11)			Não	None		AUTO_INCREMENT	Muda  Elimina  Mais
<input type="checkbox"/> 2	<b>timestamp</b>	timestamp			Não	CURRENT_TIMESTAMP			Muda  Elimina  Mais
<input type="checkbox"/> 3	<b>Corrente</b>	int(11)			Não	0			Muda  Elimina  Mais
<input type="checkbox"/> 4	<b>Potencia</b>	int(11)			Não	0			Muda  Elimina  Mais
<input type="checkbox"/> 5	<b>Metros</b>	int(11)			Não	0			Muda  Elimina  Mais