

Juliana Martins

**DESENVOLVIMENTO DE UM SISTEMA PARA
MONITORAMENTO E CONTROLE DE MÁQUINAS CNC
APLICANDO CONCEITOS DA INDÚSTRIA 4.0**

Dissertação submetida ao Programa de
Pós-Graduação Mestrado em
Engenharia e Ciências Mecânicas da
Universidade Federal de Santa
Catarina para a obtenção do Grau de
Mestre em Engenharia e Ciências
Mecânicas
Orientador: Prof. Dr. Adriano Fagali
de Souza
Coorientador^a: Prof^a. Dr^a. Aline Durrer
Patelli Juliani

Joinville
2019

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Martins, Juliana

Desenvolvimento de um sistema para monitoramento e controle de máquinas cnc aplicando conceitos da indústria 4.0 / Juliana Martins ; orientador, Adriano Fagali de Souza, coorientador, Aline Durrer Patelli Juliani, 2019.

92 p.

Dissertação (mestrado) - Universidade Federal de Santa Catarina, Campus Joinville, Programa de Pós Graduação em Engenharia e Ciências Mecânicas, Joinville, 2019.

Inclui referências.


1. Engenharia e Ciências Mecânicas. 2. Indústria 4.0. 3. Máquina CNC. 4. Usinagem. 5. Aplicativo móvel. I. Souza, Adriano Fagali de . II. Juliani, Aline Durrer Patelli. III. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Engenharia e Ciências Mecânicas. IV. Título.

Juliana Martins

Desenvolvimento de um sistema para monitoramento e controle de máquinas cnc aplicando conceitos da indústria 4.0

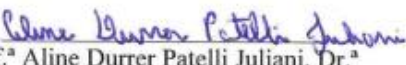
Esta Dissertação/Tese foi julgada adequada para obtenção do Título de Mestre em Engenharia e Ciências Mecânicas e aprovada em sua forma final pelo Programa de Pós-Graduação mestrado em Engenharia e Ciências Mecânicas

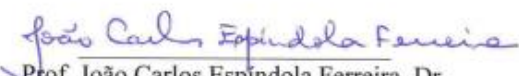
Joinville, 26 de fevereiro de 2019.


Prof. Régis Kovacs Scalice, Dr.
Coordenador do Curso

Banca Examinadora:


Prof. Adriano Fagali de Souza, Dr.
Orientador
Universidade Federal de Santa Catarina


Prof.ª Aline Durrer Patelli Juliani, Dr.ª
Coorientadora
Universidade Federal de Santa Catarina


Prof. João Carlos Espindola Ferreira, Dr.
Universidade Federal de Santa Catarina



Prof. Pablo Andretta Jaskówiak, Dr.
Universidade Federal de Santa Catarina



Prof. Diogo Lôndero da Silva, Dr.
Universidade Federal de Santa Catarina

Dedico este trabalho à minha mãe, por
todo esforço, amor e fé em mim.

AGRADECIMENTOS

Agradeço à empresa Tecnodrill Indústria de Máquina LTDA pela disposição em acreditar, financiar e apoiar todo o desenvolvimento do projeto, contribuindo com ferramentas fundamentais como a máquina CNC DM5 e minha bolsa de mestrado.

Ao professor orientador Dr. Adriano Fagali de Souza, que acreditou no meu potencial mesmo em momentos adversos e contribuiu essencialmente no desenvolvimento do projeto.

À professora coorientadora Dr.^a Aline Durrer Patelli Juliani pela visão analítica do projeto e auxílio nas correções textuais.

Aos colegas do GPCAM – Grupo de Pesquisa em Manufatura Auxiliada por Computador – pela ajuda nos conhecimentos em usinagem, amizade e parceria.

À UFSC e aos professores do Programa de Pós-Graduação em Engenharia e Ciências Mecânicas, pela oportunidade do acesso à educação de qualidade e gratuita.

“I believe that at the end of the century the use of words and general educated opinion will have altered so much that one will be able to speak of machines thinking without expecting to be contradicted.” (Alan Turing, 1950)

RESUMO

A Indústria 4.0 foi desenvolvida na Alemanha com o objetivo de incorporar novas soluções de tecnologia da informação no ambiente industrial. Impulsionando a criação da Indústria 4.0, outros conceitos também surgiram: Internet Industrial, Fábrica Inteligente e Sistemas Cyber Físicos. Neste trabalho esses conceitos e tecnologias de desenvolvimento de *software* foram empregados a fim de desenvolver um sistema para monitoramento e controle de uma máquina CNC (Comando Numérico Computadorizado) através de um dispositivo móvel, em tempo real e com baixo tempo de resposta. Foi realizado um estudo experimental para analisar a velocidade de resposta de diferentes estruturas de bancos de dados. Avaliou-se a estrutura convencional SQL (Linguagem de Consulta Estruturada), onde os dados relacionam-se por tabelas e uma estrutura mais moderna, a NoSQL (Não Somente SQL), na qual os dados são salvos de maneira otimizada em escalabilidade horizontal. Identificou-se que a estrutura não relacional apresentou menor tempo de resposta dentro do sistema criado. Esta estrutura foi utilizada para desenvolver o banco de dados proposto. Foi criado um sistema na linguagem de programação GO que se conecta a uma máquina CNC e a um banco de dados NoSQL, na nuvem. Posteriormente, foi desenvolvido um aplicativo móvel que se conecta a esse banco de dados e sincroniza com os dados da máquina. O sistema desenvolvido demonstrou elevada eficiência, permite monitorar e controlar qualquer variável da máquina CNC remotamente, em tempo real, e representa uma nova perspectiva na área de engenharia de manufatura.

Palavras-chave: Indústria 4.0. Máquina CNC. Fábrica Inteligente. Aplicativo móvel. Usinagem.

ABSTRACT

Industry 4.0 was developed in Germany with the aim of incorporating new information technology solutions into the industrial environment. Boosting the creation of Industry 4.0, other concepts have also emerged: Industrial Internet, Smart Factory and Cyber Physical Systems. In this work, these concepts and technologies of software development were employed in order to develop a system for monitoring and controlling a CNC (Computer Numeric Control) machine through a mobile device, in real time and with a low response time. An experimental study was conducted to examine the response speed of different database structures. The conventional SQL (Structured Query Language) structure was evaluated, in which the data are related by tables and a more modern structure, NoSQL (Not Only SQL), in which the data is optimally saved in horizontal scalability. It was identified that the non-relational structure presented less response time within the created system. This structure was used to develop the proposed database. A system was created in the GO programming language that connects to a CNC machine and the NoSQL database in the cloud. Subsequently, a mobile application was developed that connects to this database and synchronizes with the machine data. The developed system demonstrated high efficiency, allows monitoring and control of any CNC machine variable remotely, in real time, and represents a new perspective in the area of manufacturing engineering.

Keywords: Industry 4.0. CNC Machine. Smart Factory. Mobile App. Machining.

LISTA DE FIGURAS

Figura 1 - Visão da Indústria 3.0.....	9
Figura 2 - Visão da Indústria 4.0.....	11
Figura 4 - Utilização de tecnologias digitais pelas empresas.....	13
Figura 5 - Sistema proposto.....	14
Figura 6 - Possibilidades de Aplicação do Sistema Proposto.....	15
Figura 7 - Funil de conhecimento.....	17
Figura 8 - Parâmetros de corte.....	19
Figura 9 - Malha de Controle Fechada.....	20
Figura 10 - Níveis da Malha de Controle.....	21
Figura 11 - Números de dispositivos conectados na internet.....	23
Figura 12 - Revoluções industriais.....	24
Figura 13 - Sistemas cyber físicos.....	25
Figura 14 - Troca de mensagens em um servidor web.....	27
Figura 15 - Exemplo de mensagem SOAP.....	28
Figura 16 - Tempos de processamento.....	31
Figura 17 - Arquitetura CNC aberta.....	34
Figura 18 - Arquitetura de um Sistema de Supervisão.....	34
Figura 19 - Sistema de processamento de dados.....	37
Figura 20 - Sistema de monitoramento.....	41
Figura 21 - Eixo árvore inteligente.....	44
Figura 22 - Arquitetura de um PCMS.....	46
Figura 24 - Tela com as máquinas e seus status.....	48
Figura 25 - Gráfico com os status das máquinas selecionadas.....	49
Figura 26 - Monitor SmoothLink.....	50
Figura 27 - Tecnologias e Ferramentas.....	54
Figura 28 - Kanban com atividades.....	54
Figura 29 - Máquina DM5.....	56
Figura 30 – Modificação no comando numérico da máquina.....	58
Figura 31 - Sequência Binária.....	60
Figura 32 - Diagrama de Classes do programa GO.....	61
Figura 33 - Banco de dados.....	62
Figura 34 - Diagrama de Sequência.....	63
Figura 35 - Modificação da velocidade de avanço através de serviço na linguagem GO.....	64
Figura 36 - Diagrama de classes do aplicativo.....	66
Figura 37 - Caso de Uso.....	67
Figura 38 - A - Tela de <i>Login</i> B - Lista de máquinas C - Status da máquina selecionada.....	68
Figura 39 - A - Tela de alarmes B - Lista de histórico de usinagem.....	69

Figura 40 - Monitoramento e Controle da velocidade de avanço.	70
Figura 41 - Monitoramento e Controle do nome do programa.	70
Figura 42 - Alteração da velocidade de avanço.	71
Figura 43 - Telas do Aplicativo.	71
Figura 44 - Banco de dados MySQL.	73
Figura 45 - Programas executáveis.	74
Figura 46 - Exemplo de chamada para inserção de dados.	74
Figura 47 - Comparação de tempos para inserção de dados entre Firebase e MySQL.	75
Figura 48 - Rotina GO para inserção no MySQL.	85
Figura 49 - Rotina GO para inserção de dados no Firebase.	86
Figura 50 - Tela com todas as máquinas Tela com as ferramentas tela com o gráfico de uso das máquinas.	87
Figura 51 - Tela com parâmetros da máquina Tela para envio de programa.	87
Figura 52 - Tela inicial Tela com todas as máquinas e status tela com os eventos da máquina.	88
Figura 53 - Tela de manutenção, mostra uma barra progressiva para a próxima manutenção da máquina.	88

LISTA DE ABREVIATURAS E SIGLAS

AE - Largura de corte
AP - Profundidade de corte
ASCII - *American Standard Code for Information Interchange*
BI - *Business Intelligence*
CAD - *Computer Aided Design*
CNC - Comando Numérico Computadorizado
CNI - Confederação Nacional da Indústria
CSS - *Cascading Style Sheets*
DAQ - *Data acquisition*
DDE - *Dynamic Data Exchange*
DDR3 - *Double Data Rate 3*
ERP - *Enterprise Resource Planning*
F – Velocidade de avanço
FFNN - *Feedforward Neural Network*
GB - *Gigabyte*
HD – *Hard Disc*
HSM - *High Speed Machining*
HTML - *Hypertext Markup Language*
HTTP - *HyperText Transfer Protocol*
IHM – Interface Homem Máquina
IP - *Internet Protocol*
IPV - *Internet Protocol Version*
JSON - *JavaScript Object Notation*
Mbps - Megabytes por segundo
MES - *Manufacturing Execution Systems*
NC - *Numeric Control*
NFC - *Near Field Communication*
NoSQL - *Not only SQL*
OPC - *Open Platform Communications*
OPC UA - *Open Platform Communications Unified Architecture*
PA - Posição Atual
PC - Personal Computer
PCA - *Principal Component Analysis*
PCMS - *Personalized Customization Manufacturing System*
PD - Posição Desejada
RAMI 4.0 - *Reference Architectural Model for Industrie 4.0*
REST - *Representational State Transfer*
RFID - *Radio-Frequency IDentification*
SAN - *Storage Area Network*

SCADA - *Supervisory Control and Data Acquisition*
SDRAM - Memória De Acesso Aleatório Dinâmica Síncrona
Sem - Semântica
SOAP - *Simple Object Access Protocol*
SQL - *Structured Query Language*
TB - *Terabyte*
TCP - *Transmission Control Protocol*
TI - Tecnologia da Informação
TPB - Tempo de Processamento de Blocos
UML - *Unified Modeling Language*
VC – Velocidade de Corte
VRAM - *Video Random Access Memory*
W3C - *World Wide Web Consortium*
WSDL - *Web Service Description Language*
XML - *Extensible Markup Language*

SUMÁRIO

1	INTRODUÇÃO	9
1.1	OBJETIVOS	14
1.2	JUSTIFICATIVAS	15
1.3	ORGANIZAÇÃO	15
2	REVISÃO DE LITERATURA	17
2.1	MÁQUINAS DE FRESAMENTO CNC	18
2.1.1	Unidade de Controle	19
2.1.2	Unidade de Processamento.....	21
2.2	INDÚSTRIA 4.0 E TECNOLOGIAS	22
2.2.1	Conexão de sistemas e máquinas no ambiente industrial	26
2.2.2	Protocolos de comunicação entre sistemas.....	26
2.2.3	Engenharia de Software	28
2.2.4	Desenvolvimento de sistemas web e aplicativos móveis ...	30
2.3	MONITORAMENTO E CONTROLE	33
2.3.1	Sistemas de Monitoramento Interno	33
2.3.2	Sistemas de Monitoramento Remoto.....	38
2.3.3	Sistemas para monitoramento (SCADA)	42
2.3.4	Monitoramento e Controle por Dispositivos Móveis.....	45
2.4	PESQUISA DE MERCADO	47
2.4.1	Principais Fabricantes do Mercado.....	47
2.4.2	Outras Aplicações	50
3	METODOLOGIA	53
3.1	DESENVOLVIMENTO DO SISTEMA	53
3.1.1	Comunicação de baixo nível com a máquina CNC	55
3.1.2	Criação de Serviço na Nuvem	56
3.1.3	Desenvolvimento do Aplicativo	56
3.2	AVALIAÇÃO DE DESEMPENHO DO SISTEMA	57
4	RESULTADOS E DISCUSSÕES	58
4.1	DESENVOLVIMENTO DO SISTEMA	58

4.1.1	Criação do Método de Comunicação com a máquina CNC	58
4.1.2	Criação de Serviço na Nuvem.....	61
4.1.3	Desenvolvimento do Aplicativo.....	65
4.2	AVALIAÇÃO DE DESEMPENHO DO SISTEMA.....	72
5	CONCLUSÕES.....	77
	REFERÊNCIAS.....	79
	APÊNDICE A – Rotina GO para Inserção no MySQL.....	85
	APÊNDICE B - Rotina GO para Inserção de Dados no Firebase..	86
	ANEXO A – Telas do Aplicativo CNC Mobile	87
	ANEXO B – Telas do Aplicativo Thermwood CNC Mobile	88

1 INTRODUÇÃO

A indústria 4.0 foi criada na Alemanha com o objetivo de estabelecer a quarta grande revolução industrial. Segundo Posada et al. (2015), na terceira revolução industrial a automação foi introduzida nas fábricas, auxiliada por robôs, máquinas computadorizadas, sensores e demais equipamentos, porém quase todos esses equipamentos trabalham separadamente, conforme mostrado na Figura 1. Os sistemas da indústria 3.0 são geralmente criados especificamente para um equipamento ou processo de produção, dificultando a comunicação e interação entre os diversos sistemas que podem ser empregados em um ambiente fabril.

A quarta revolução industrial adiciona novas tecnologias aos meios de produção e maior comunicação entre todos os objetos fabris. Para isso, utiliza ferramentas e conceitos da computação, como: internet das coisas, banco de dados na nuvem, inteligência artificial, etc. (GRUNDSTEIN; FREITAG; SCHOLZ-REITER, 2017).

Figura 1 - Visão da Indústria 3.0.



Fonte: Adaptado de “Plattform Industrie 4.0” (2017).

Em 2013, foi criado na Alemanha o “Plattform Industrie 4.0”, que é uma plataforma liderada pela Ministra Federal de Assuntos Econômicos e Energia, Brigitte Zypries, pela Ministra Federal de Educação e Pesquisa, Prof. Johanna Wanka, e altos representantes da indústria, da ciência e dos sindicatos. O objetivo principal desta plataforma é tornar a Alemanha o berço da indústria 4.0, respondendo basicamente as questões: Como a Alemanha pode se tornar o principal fornecedor de equipamentos de

fábrica da Indústria 4.0? Como a Alemanha pode melhorar ainda mais sua competitividade através da Indústria 4.0? Que papel a Alemanha pode desempenhar na definição de normas e como a Indústria 4.0 pode beneficiar as pessoas em suas funções de trabalho?

Para auxiliar nas respostas a estes questionamentos e produzir conteúdos relevantes ao meio industrial, governamental e acadêmico, a plataforma foi dividida em grupos de trabalho:

- a) Grupo de trabalho 1: Arquiteturas de referência, padrões e normas;
- b) Grupo de trabalho 2: Pesquisa e inovação;
- c) Grupo de trabalho 3: Segurança dos sistemas em rede;
- d) Grupo de trabalho 4: Quadro jurídico;
- e) Grupo de trabalho 5: Trabalho, educação e treinamento.

Um dos resultados da plataforma foi o desenvolvimento do RAMI 4.0: *Reference Architectural Model for Industrie 4.0*, que é uma arquitetura de referência da Indústria 4.0, conforme mostrado na Figura 2. Nesta arquitetura, uma fábrica inteligente, formada por máquinas, computadores, robôs, sistemas e demais equipamentos é constituída de uma rede interna que permite a troca de dados entre todos os objetos da fábrica, e também se conecta a internet para aquisição de dados pertinentes ao processo de fabricação, envio de dados para um banco de dados externo (que pode ser utilizado para criar um sistema móvel, por exemplo, onde o usuário fique informado de todos os passos para a fabricação de seu produto) e demais ações que ao final produzirão produtos (LANGMANN; ROJAS-PEÑA, 2016).

Diferente do que foi apresentado na Indústria 3.0, nesta nova arquitetura todos os componentes estão conectados e os dados de produção dos produtos são salvos, tratados e resultam em análises e futuras tomadas de decisão da própria cadeia produtiva (“PLATTFORM INDUSTRIE 4.0”, 2017).

Figura 2 – Visão da Indústria 4.0.



Fonte: “Plattform Industrie 4.0” (2017).

Segundo “Industrial Internet Consortium” (2017), em 2014 foi criado nos EUA o “*Industrial Internet Consortium*”, que é um consórcio de Internet Industrial envolvendo diversas empresas, tendo como fundadores a General Electric, IBM e Intel, sendo que seu objetivo é o de disponibilizar seus estudos para que a comunidade entenda como desenvolver a internet industrial. Para isso, foram criados grupos de trabalho em diversas áreas, como estratégia de negócios e ciclo de vida da solução, ligações (com outras empresas, consórcios, etc.), publicidade, segurança, tecnologia e bancos de ensaios.

No Brasil, a Confederação Nacional da Indústria – CNI (2016) – realizou uma pesquisa com 2.225 empresas, de todos os portes e de setores diversos. Na pesquisa, as empresas foram questionadas quanto ao uso de dez tecnologias digitais relacionadas a Internet Industrial:

- 1 Automação digital sem sensores;
- 2 Automação digital com sensores para controle de processos;
- 3 Monitoramento e controle remoto da produção com sistema do tipo MES e SCADA;
- 4 Automação digital com sensores com identificação de produtos e condições operacionais, linhas flexíveis;
- 5 Sistemas integrados de engenharia para desenvolvimento de produtos e manufatura de produtos;
- 6 Manufatura aditiva, prototipagem rápida ou impressão 3D;

- 7 Simulações/análise de modelos virtuais (Elementos Finitos, Fluidodinâmica Computacional, etc.) para projeto e comissionamento;
- 8 Coleta, processamento e análise de grandes quantidades de dados (*big data*);
- 9 Utilização de serviços em nuvem associados ao produto;
- 10 Incorporação de serviços digitais nos produtos (“Internet das Coisas” ou *Product Service Systems*).

Na Tabela 1 é apresentado o resultado da pesquisa, relacionando cada uma das tecnologias quanto ao uso e a importância para a competitividade das empresas entrevistadas. A Figura 3 sintetiza graficamente os resultados e apresenta que menos da metade dessas empresas já utilizou ao menos uma das tecnologias. Os resultados também apontam, por exemplo, que apenas 6% das empresas já utilizaram algum serviço em nuvem associado ao produto e 4% incorporaram serviços de “Internet das Coisas” (CNI, 2016).

Tabela 1 - 10 Tecnologias Digitais.

ESTÁGIO/FOCO	TECNOLOGIA	USO	IMPORTÂNCIA PARA COMPETITIVIDADE
Processo	Automação digital sem sensores	11	3
	Automação digital com sensores para controle de processo	27	20
	Monitoramento e controle remoto da produção com sistemas do tipo MES e SCADA (1)	7	14
	Automação digital com sensores com identificação de produtos e condições operacionais, linhas flexíveis	8	21
Desenvolvimento/ redução time to market	Sistemas integrados de engenharia para desenvolvimento de produtos e manufatura de produtos	19	25
	Manufatura aditiva, prototipagem rápida ou impressão 3D	5	9
	Simulações/análise de modelos virtuais (Elementos Finitos, Fluidodinâmica Computacional, etc.) para projeto e comissionamento	5	5
Produto/novos modelos de negócios	Coleta, processamento e análise de grandes quantidades de dados (<i>big data</i>)	9	15
	Utilização de serviços em nuvem associados ao produto	6	11
	Incorporação de serviços digitais nos produtos (“Internet das Coisas” ou <i>Product Service Systems</i>)	4	12
Projetos de manufatura por computador CAD/CAM (2) (3)		30	9
Nenhuma das listadas		15	3
Não sabe/ não respondeu		31	39

Nota: A soma dos percentuais supera 100% devido a possibilidade de múltiplas respostas.

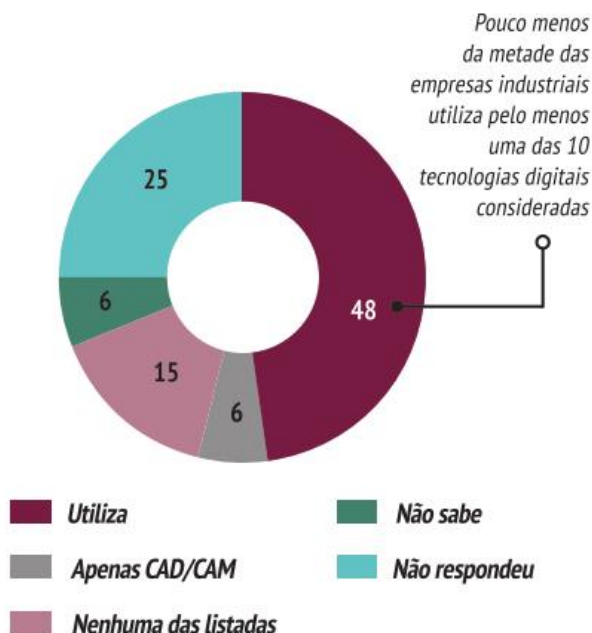
(1) MES - *Manufacturing Execution Systems*; SCADA - *Supervisory Control and Data Acquisition*.

(2) CAD - *Computer-Aided Design*; CAM - *Computer-Aided Manufacturing*.

(3) A opção “Projetos de manufatura por computador CAD/CAM”, isto é, licenças de softwares utilizadas nas etapas de desenvolvimento e de fabricação, não se enquadra como tecnologia digital, apesar de significar maior automação da manufatura. Sua inclusão entre as opções de resposta se deu para deixar mais clara a diferença com “Sistemas integrados de engenharia para desenvolvimento de produtos e manufatura de produtos”.

Fonte: CNI (2016).

Figura 3 - Utilização de tecnologias digitais pelas empresas.



Fonte: CNI (2016).

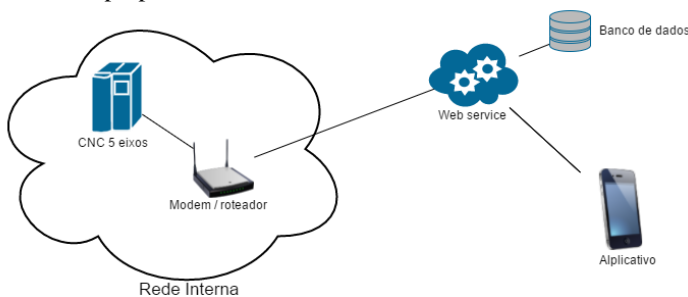
Os baixos números sobre a utilização de tecnologias da informação na indústria brasileira mostram que muitos serviços e sistemas ainda podem ser criados neste sentido. Na indústria metal mecânica, por exemplo, poderiam ser criados sistemas para manutenção de máquinas CNC (indicando ao fabricante da máquina quando houver algum problema), também poderiam ser criados sistemas que informassem aos usuários da máquina quando houvesse algum evento inesperado, etc.

1.1 OBJETIVOS

O objetivo deste trabalho é desenvolver um sistema de Indústria 4.0 aplicado às indústrias de manufatura que empregam máquinas CNC, com baixo tempo de resposta.

Na Figura 4 é mostrado a arquitetura do sistema proposto. A máquina CNC se conecta a nuvem através de uma rede sem fio que possui um sistema hospedado internamente (dentro do ambiente em que a máquina está instalada). Na nuvem fica o banco de dados, que é sincronizado com a máquina e o aplicativo, mantendo salvos os parâmetros atualizados da máquina. O aplicativo móvel busca os dados armazenados no banco de dados e envia requisições para que o serviço controle os parâmetros da máquina.

Figura 4 – Sistema proposto.



Fonte: Autor (2017).

Para atingir este fim são necessárias quatro metas específicas:

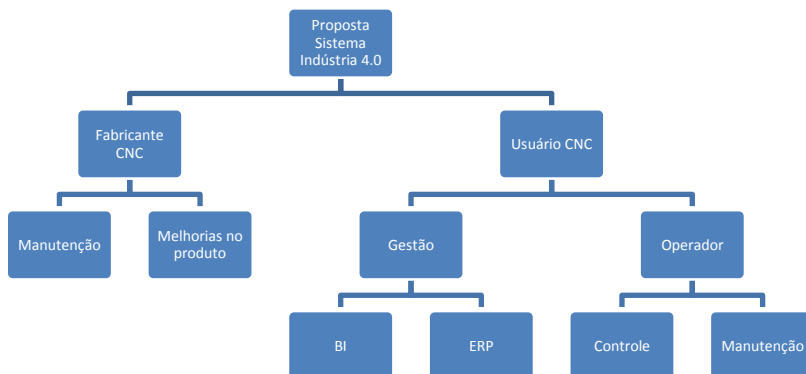
- a) Pesquisa de mercado: realizar um levantamento sobre as tecnologias disponíveis no mercado atual para atender o ambiente de trabalho seguindo padrões da Internet Industrial;
- b) Pesquisar tecnologias de desenvolvimento de sistemas que sejam de baixo custo computacional e financeiro;
- c) Investigar tempos de resposta de bancos de dados para acesso remoto;
- d) Desenvolver um sistema para Internet Industrial que integre uma máquina CNC em uma rede sem fio, com um serviço nas nuvens, junto a um banco de dados e um aplicativo móvel.

1.2 JUSTIFICATIVAS

Este trabalho é um produto nacional e, portanto, auxilia o desenvolvimento dessas tecnologias no Brasil, sendo coerente com as necessidades apresentadas na Figura 3.

A Figura 5 mostra algumas das possibilidades que a aplicação desse sistema pode atingir. Para o fabricante da máquina CNC, as vantagens envolvem a possibilidade de identificar eventuais manutenções na máquina sem precisar ir até o cliente e também a identificação de possíveis melhorias em seu produto. Para os usuários da máquina CNC, existem duas visões: a do operador da máquina, que pode controlar e monitorar a máquina à distância e a visão do gestor, que pode utilizar os dados da máquina em sistemas de inteligência de negócio e gerenciamentos da empresa, produzindo informações que contribuirão para a tomadas de decisões futuras, por exemplo.

Figura 5 – Possibilidades de Aplicação do Sistema Proposto.



Fonte: Autor (2017).

1.3 ORGANIZAÇÃO

Além da introdução, o trabalho possui outros 4 capítulos: revisão de literatura, metodologia, resultados e discussões e conclusões.

Na revisão de literatura são discutidos aspectos gerais de uma máquina de comando numérico e conceitos de Indústria 4.0. Também são apresentados trabalhos já realizados na área de monitoramento e controle.

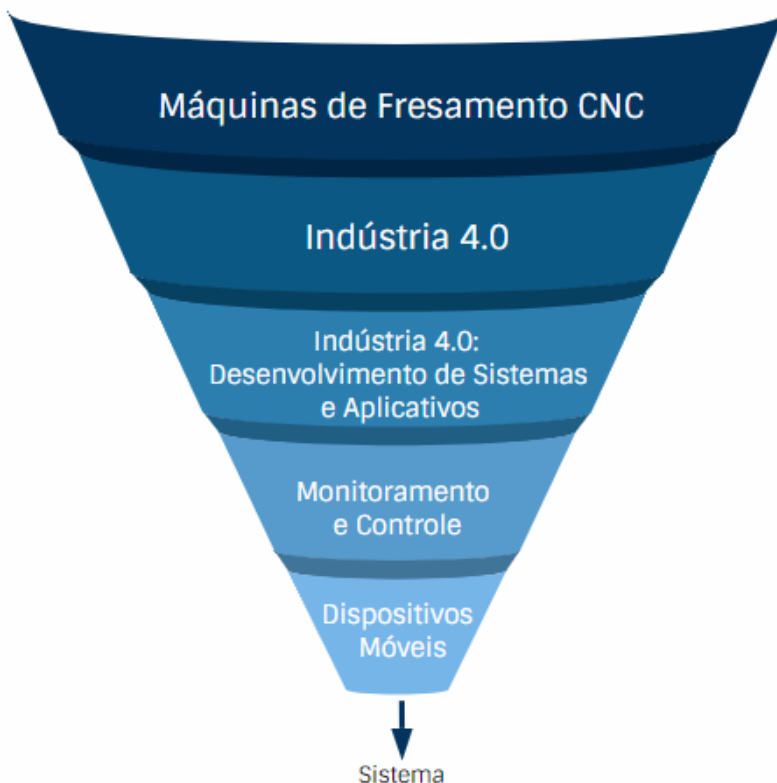
Na metodologia são mostradas as ferramentas e métodos para atingir o objetivo do trabalho.

No penúltimo capítulo são apresentados os resultados obtidos com o desenvolvimento do sistema, assim como propostas de melhorias e demais discussões. Por fim, são mostradas as conclusões, após analisados os resultados.

2 REVISÃO DE LITERATURA

A revisão de literatura deste trabalho foi desenvolvida considerando-se todas as bases de conhecimento necessárias para a criação de um sistema para monitorar e controlar uma máquina CNC. Como pode ser visualizado na Figura 6, a revisão começa com as máquinas de fresamento CNC, contendo um resumo de sua operação e funcionamento. A segunda subseção começa com os conceitos de Indústria 4.0 e tecnologias para o desenvolvimento de um sistema web ou aplicativo. Na última parte, são apresentadas e discutidas pesquisas que realizam o monitoramento e e/ou controle de uma máquina CNC através de variados métodos, inclusive remotamente utilizando dispositivos móveis.

Figura 6 - Funil de conhecimento.



Fonte: Autor (2018).

2.1 MÁQUINAS DE FRESAMENTO CNC

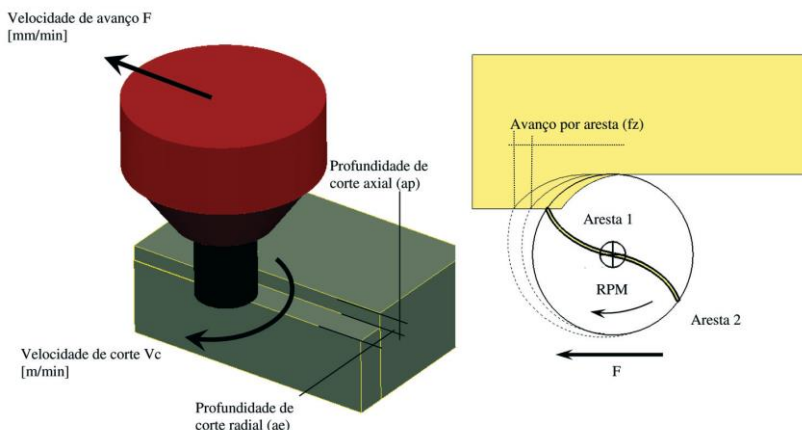
As máquinas de fresamento fabricam peças através de um processo de remoção de material chamado usinagem e são constituídas principalmente de uma mesa de trabalho, um eixo-árvore, motores de acionamento e controles. Essas máquinas podem ser analógicas ou digitais.

Os comandos numéricos computadorizados representam o acionamento de uma máquina por coordenadas numéricas, podendo ser controlado por um computador. Um CNC deve receber dados através de algum dispositivo de inserção de dados, interpretá-los (comandos numéricos) e comandar a máquina para que ela realize a operação. As instruções que são realizadas na máquina CNC são interpretadas de um programa CNC, que é criado por caracteres ASCII e possui blocos de dados com instruções para realizar alguma ação na máquina (SOUZA; ULBRICH, 2013).

O eixo-árvore da máquina é o acionamento que possibilita a rotação de uma ferramenta de corte e possui duas grandezas inversamente proporcionais: rotação e potência. A ferramenta de corte, também chamada fresa, normalmente possui haste cilíndrica e arestas de corte e é a responsável por entrar em contato direto com a peça fixada na mesa e modelá-la conforme os acionamentos (SOUZA; ULBRICH, 2013).

Para remover material das peças, as ferramentas de corte possuem parâmetros geométricos e de velocidade, conforme Figura 7. Os parâmetros geométricos envolvem a profundidade de corte no sentido axial da ferramenta (a_p), a profundidade de corte no sentido radial da ferramenta (a_e) e o avanço por aresta, valores incorretos podem reduzir a vida útil da ferramenta, prejudicar a qualidade superficial da peça usinada ou causar uma ineficiência no processo de usinagem. Os parâmetros de velocidade envolvem a velocidade de corte (V_c) e de avanço (F). Velocidades de corte muito altas podem superaquecer a ferramenta e reduzir sua vida útil.

Figura 7 - Parâmetros de corte.



Fonte: Souza e Ulbrich (2013).

Ainda segundo Souza e Ulbrich (2013) as coordenadas que a fresa deve seguir para realizar a usinagem são instruções previamente definidas em um programa NC (*Numeric Control*), interpretadas em tempo real pelo comando numérico que comanda o *hardware* da máquina para que a fresa se movimente. O esquema de *hardware* básico de uma máquina CNC consiste em uma unidade de controle e uma unidade de processamento de dados.

Nas próximas subseções são detalhados os funcionamentos de cada uma dessas unidades.

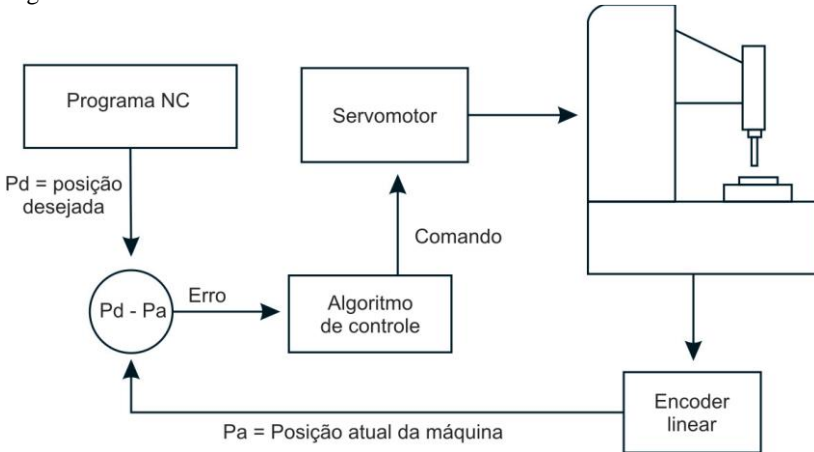
2.1.1 Unidade de Controle

A maioria das máquina-ferramentas CNC possuem um ciclo de controle que fornece *feedbacks* periodicamente ao sistema, com dados necessários para que a máquina trabalhe dentro de sua precisão. A chamada malha de controle representa todo o sistema de monitoramento e comunicação em uma máquina (SUH ET AL., 2008).

A Figura 8 mostra uma malha de controle fechada, que se caracteriza por realimentar seu sistema de monitoramento de posição (para comparar com a especificada no programa) periodicamente. Neste tipo de malha, a posição atual (P_a) é lida pelo CNC, que compara com a posição desejada interpretada no programa NC (P_d) e realiza um cálculo

de erro, que é a diferença entre as duas posições. O sinal de controle (comando) aciona os servo-motores para realizar a correção da posição.

Figura 8 - Malha de Controle Fechada.

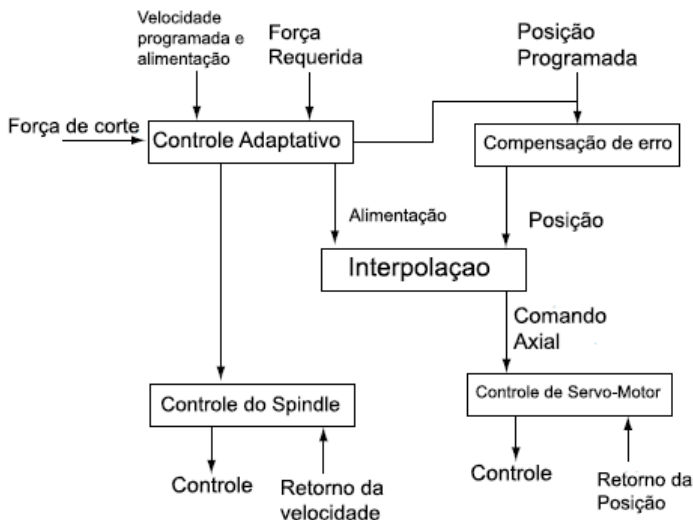


Fonte: Souza e Ulbrich (2013).

Segundo Suh et al. (2008) e conforme Figura 9 as malhas de controle podem possuir uma subdivisão de níveis e tarefas de controle, aumentando a eficiência da usinagem da peça.

No controle adaptativo, mostrado na Figura 9, são ajustadas as variáveis do processo, como velocidade, força e posição. No controle do *Spindle* ocorre a correção da velocidade de avanço desejada conforme a velocidade atual recebida por *feedback*. Na interpolação é verificada a necessidade de corrente. No controle do servo-motor é feita a correção dos eixos, conforme o *feedback* de posição.

Figura 9 - Níveis da Malha de Controle.



Fonte: Adaptado de SUH et al. (2008).

As malhas de controle possibilitam a correção de erros em tempo de execução do programa. Sem elas os erros seriam incrementados a cada execução de bloco e a peça usinada poderia ficar com maior diferença da peça projetada. O sistema proposto nesse trabalho visa não interferir no funcionamento das malhas existentes e trabalhar apenas com as variáveis já salvas na memória da máquina CNC.

Para entender melhor o funcionamento do *hardware* e estudar os impactos que pode gerar em um sistema 4.0, na próxima subseção é apresentada a unidade de processamento.

2.1.2 Unidade de Processamento

Souza e Ulbrich (2013) descrevem que a unidade de processamento é responsável por interpretar bloco a bloco os programas NC e enviar os dados de acionamento para os servo-motores e demais sistemas da máquina.

O bloco de programa equivale a uma linha escrita em linguagem G, com dados da tabela ASCII. Para interpretar cada um dos blocos do programa, é necessário um tempo, chamado de Tempo de Processamento de Blocos (TPB).

Um dos métodos mais utilizados pela indústria para avaliar o desempenho entre diferentes fabricantes de comandos numéricos é a análise do TPB, porém existem outros fatores que contribuem nessa análise.

O tempo que o comando leva para ler um bloco e enviar os acionamentos aos servo-motores deve ser menor do que o tempo que a ferramenta de corte utiliza para percorrer a trajetória descrita por um bloco anterior. Do contrário, a ferramenta terminará a movimentação e o próximo bloco a ser executado ainda não estará pronto, o que acarreta em maior lentidão na usinagem da peça.

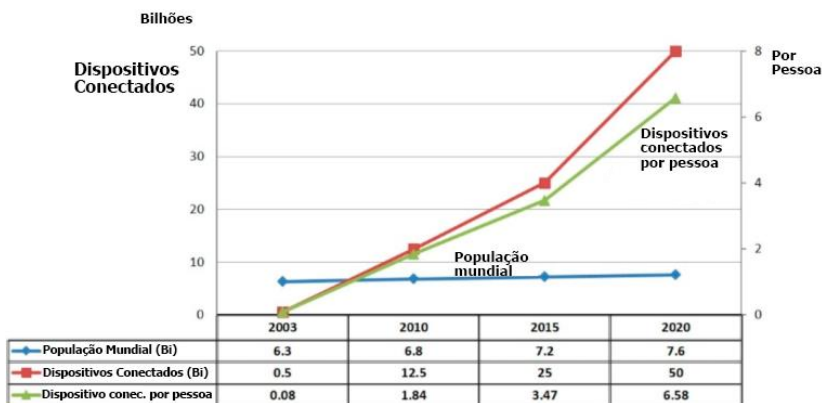
Para o desenvolvimento de um sistema de Indústria 4.0 é necessário entender o funcionamento da unidade de processamento e considerar os tempos de processamento de bloco e trajetória da ferramenta. Um sistema que faça a máquina CNC atrasar a leitura de um bloco ou o acionamento de um servo-motor, poderá acarretar em um maior tempo de usinagem, o que aumenta os custos de produção. A intenção de um sistema deve ser o de auxiliar na produção e não o de impactar negativamente no financeiro.

Na subseção a seguir, são apresentadas a quarta revolução industrial e tecnologias para desenvolvimento de um sistema de monitoramento e controle.

2.2 INDÚSTRIA 4.0 E TECNOLOGIAS

Segundo Lee e Kim (2016) e conforme mostrado na Figura 10, em 2010 o número de dispositivos conectados na internet passou a ser maior do que a população mundial. A perspectiva para 2020 é que esse número alcance, no mínimo, 50 bilhões.

Figura 10 - Números de dispositivos conectados na internet.



Fonte: Adaptado de Bagheri et al. (2015).

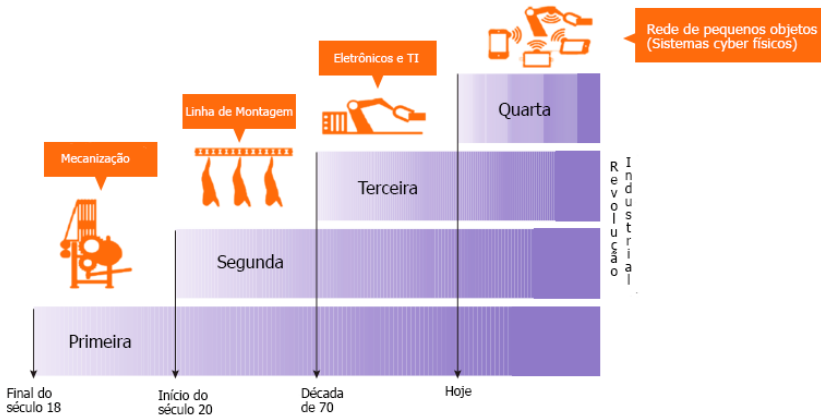
A ideia da internet das coisas, avanças como a versão mais atual do protocolo de internet - IPV6 - e a perspectiva de que cada vez mais as pessoas se comunicarão e tomarão suas decisões baseadas em sistemas computacionais criaram o conceito da indústria 4.0, que também é conhecido como manufatura avançada, internet industrial e demais nomes, apresentando o mesmo objetivo: conectar um ambiente industrial e torná-lo mais eficiente.

Para Monostori (2014) o desenvolvimento da automação industrial sempre esteve em paralelo ao da ciência da computação. A criação de computadores levou o desenvolvimento de comandos numéricos e robôs, a computação gráfica possibilitou a criação de sistemas CAD (Desenho Assistida por Computador), o desenvolvimento de processadores mais rápidos melhorou os comandos numéricos computadorizados, muitas outras tecnologias computacionais possibilitaram e ainda possibilitam os avanços na automação da manufatura.

A Indústria 4.0 surgiu com a ideia da quarta grande revolução industrial, sucessora à inclusão da automação industrial. Conforme Posada et al. (2015), a primeira revolução industrial foi responsável pela mecanização dos meios de produção, a segunda revolução introduziu as linhas de produção, a terceira revolução começou a introdução de equipamentos eletrônicos e de automatização na indústria, enquanto na quarta revolução são utilizados todos os

avanços tecnológicos, sobretudo computacionais, para criar uma fábrica inteligente, utilizando os sistemas cyber físicos (Figura 11).

Figura 11 - Revoluções industriais.

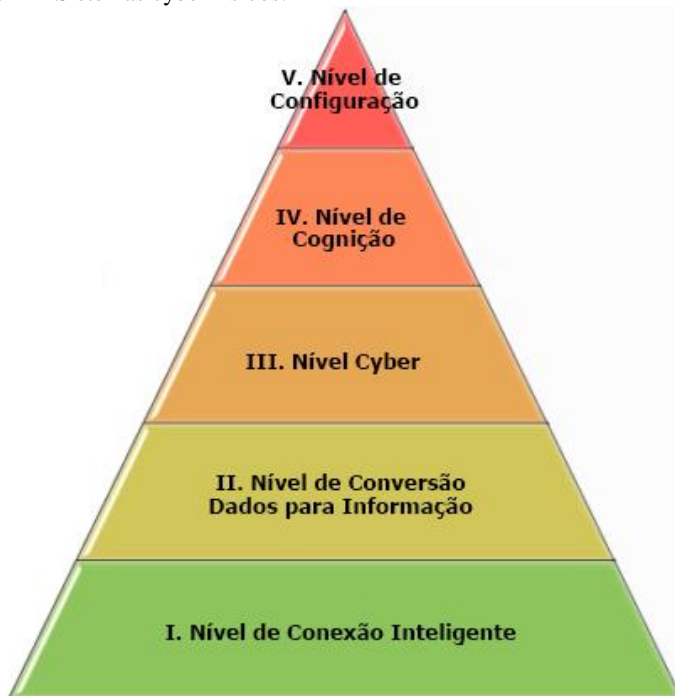


Fonte: Adaptado de Posada et al., (2015).

Para o desenvolvimento de um sistema de indústria 4.0, alguns *frameworks* são utilizados como o de sistemas cyber físicos.

Em Bagheri et al., (2015), os sistemas cyber físicos integram o espaço físico natural, criado pelo homem, aos sistemas computacionais, de comunicação e de controle. Os autores desenvolveram um *framework* mostrado na Figura 12, que define 5 níveis necessários para a criação de um sistema cyber físico. O nível mais baixo chamado “nível de conexão inteligente” define que dados devem ser obtidos através de acesso a sistemas de controle de máquinas, como sensores, sistemas auxiliares e outros, utilizando um protocolo de comunicação padronizado.

Figura 12 - Sistemas cyber físicos.



Fonte: Adaptado de Bagheri et al. (2015).

No segundo nível, informações devem ser adquiridas dos dados previamente obtidos, como a necessidade de manutenção de equipamentos e cálculo de vida útil.

No terceiro nível do *framework*, as informações obtidas de cada máquina são comparadas. No quarto nível, chamado de cognitivo, devem ser realizadas operações de tomadas de decisão, auxiliadas pelas informações previamente obtidas, podendo basear-se em conhecimentos de especialistas.

No último nível, o de configuração, o espaço cyber responde ao espaço físicos, aplicando a medida corretiva ou preventiva que foi desenvolvida no nível cognitivo.

Qin, Liu e Grovesnor (2016) e Wang et al. (2016) também criaram seus *framework* de sistemas cyber físicos, porém basicamente se limitam a conectar os sistemas cyber (banco de dados, aplicações, dispositivos móveis, etc.) com o sistema físico (máquinas, sensores, atuadores), através de uma rede de transmissão de dados (rede sem fio, ethernet, etc.).

2.2.1 Conexão de sistemas e máquinas no ambiente industrial

A tecnologia da informação dispõe de diversos recursos de conectividade para realizar a integração dos objetos no ambiente industrial.

Autores como Li, Hu e Zhang (2011) e Bai (2017) criaram um *framework* onde o protocolo Ethernet Fieldbus foi utilizado como padrão de comunicação no ambiente industrial, pois já é utilizado em sistemas de automação industrial, permitindo uma conexão rápida a rede.

As redes sem fio de curto alcance, como WiFi, Bluetooth, Zigbee e a Rede de Área de Sensores (SAN), são tecnologias comuns para suportar a conexão de sensores, dispositivos e usuários para transmissão de dados, enquanto que o protocolo de internet versão 4 / protocolo de internet versão 6 (IPv4 / IPv6) são padrões utilizados para transportar os dados entre essas redes (XIA et al., 2015).

2.2.2 Protocolos de comunicação entre sistemas

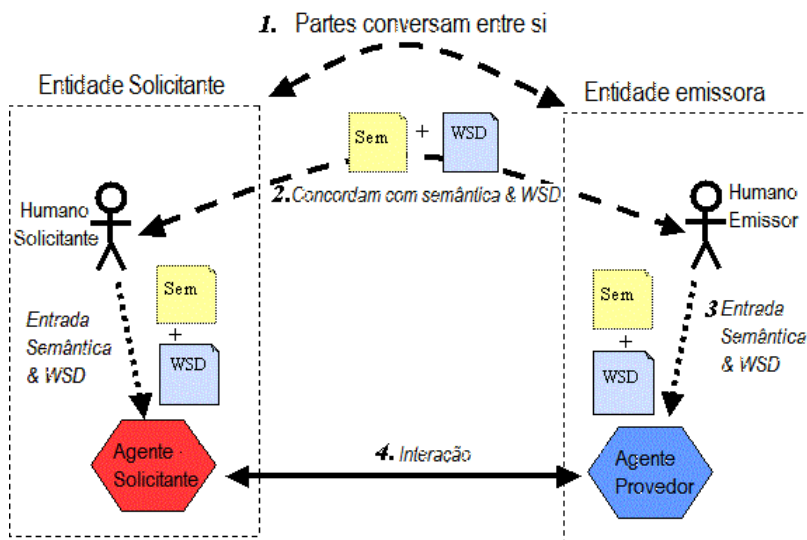
Os diversos sistemas que se integram em uma plataforma de indústria 4.0 precisam utilizar a mesma linguagem. Os sistemas que operam na web possuem alguns padrões de comunicação, como o emprego de serviços web e do protocolo SOAP - *Simple Object Access Protocol*.

Thramboulidis e Christoulakis (2016) utilizam em seus trabalhos serviços web, *Representational State Transfer* (REST) como protocolo de comunicação entre os serviços web e *JavaScript Object Notation* (JSON) como formatação dos dados a serem trocados pelo servidor (linguagem).

Em W3C (2017), os serviços web fornecem um meio padrão de comunicação entre diferentes aplicações de software. Um serviço é um recurso abstrato que representa uma capacidade de executar tarefas que representam uma funcionalidade coerente do ponto de vista das entidades prestadoras e solicitantes. Um serviço da Web é um sistema de software projetado para suportar interações interoperáveis de máquina para máquina, através de uma rede, possuindo uma interface descrita em um formato processável pela máquina.

A Figura 13 mostra o processo de interação de um webservice, sendo a semântica (comportamento das sentenças inseridas no serviço) e a linguagem de descrição do serviço conhecidas por ambas entidades.

Figura 13 - Troca de mensagens em um servidor web.



Fonte: Adaptado de W3C (2017).

SOAP é um protocolo utilizado para a troca de mensagens de um serviço web. Ele tem dois grandes objetivos: ser simples e extensível. É baseado em XML (uma linguagem de marcação recomendada pela W3C para a criação de documentos com dados organizados hierarquicamente).

Na Figura 14, é mostrado um exemplo onde pode ser priorizada a entrega da mensagem com base nas informações de prioridade e expiração do bloco de cabeçalho SOAP. O corpo contém uma mensagem de alerta e o destinatário desta mensagem. Este exemplo demonstra o quanto é simples utilizar este tipo de protocolo para a troca de mensagens entre servidores (W3C, 2017).

Figura 14 – Exemplo de mensagem SOAP.

Example 1: SOAP message containing a SOAP header block and a SOAP body

```
<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope">
  <env:Header>
    <n:alertcontrol xmlns:n="http://example.org/alertcontrol">
      <n:priority>1</n:priority>
      <n:expires>2001-06-22T14:00:00-05:00</n:expires>
    </n:alertcontrol>
  </env:Header>
  <env:Body>
    <m:alert xmlns:m="http://example.org/alert">
      <m:msg>Pick up Mary at school at 2pm</m:msg>
    </m:alert>
  </env:Body>
</env:Envelope>
```

Fonte: Adaptado de W3C (2017).

O protocolo REST possui o mesmo princípio de funcionamento do SOAP, porém facilita a compressão de trocas de mensagens. Neste protocolo as mensagens podem ser escritas em JSON e não apenas em XML.

Segundo Shariatzadeh et al. (2016), o JSON é mais compacto que o XML e, conseqüentemente, precisa de menos espaço para o armazenamento de dados. Além disso, também é utilizado em muitos sistemas empresariais como forma de troca de dados.

2.2.3 Engenharia de Software

A engenharia de software é uma área da computação que visa definir como um software deve ser desenvolvido, aplicando metodologias e utilizando ferramentas que organizem e gerenciem todo o processo de desenvolvimento.

Segundo Silva e Videira (2001), existem diferentes modelos de processos de desenvolvimento de software e os mais clássicos englobam basicamente as seguintes fases:

- a) Planejamento: são definidas necessidades de negócio (conversa inicial com o cliente, para entender quais problemas serão solucionados no projeto);
- b) Análise: os problemas são estudados e detalhados, gerando um documento de requisitos de sistema;
- c) Projeto: é realizada uma especificação técnica do que será necessário para o desenvolvimento do software. Neste momento são definidas arquiteturas de banco de dados e

- servidores, modelagem das saídas do sistema, descrição das lógicas de programação, etc.;
- d) Implementação: é desenvolvido o sistema através das especificações produzidas nas fases anteriores;
 - e) Testes: o sistema é testado, utilizando diferentes técnicas, a fim de identificar se todos os requisitos foram desenvolvidos conforme desejo do cliente;
 - f) Integração e manutenção: o software é instalado no cliente e, caso seja necessário, ajustes são realizados através de manutenção.

O modelo Cascata, idealizado em 1970, propõe que cada uma das fases seja completamente terminada para que outra inicie: a fase de implementação não pode iniciar sem que todos os requisitos sejam especificados, por exemplo, e o software é completamente entregue ao final de todas as fases. Porém, com o desenvolvimento da engenharia de software alguns novos modelos foram criados, como é o caso dos modelos incrementais e do framework SCRUM (integrante das metodologias ágeis de desenvolvimento), em que o software não precisa ser totalmente entregue de uma só vez, podendo ser construído através de diversas interações com o cliente. O cliente recebe uma pequena versão do sistema, podendo testar, verificar se está de acordo com suas expectativas e dar um *feedback* ao time de desenvolvimento (SILVA E VIDEIRA, 2001).

Segundo Sbrocco e Macedo (2012), as metodologias ágeis de desenvolvimento surgiram na década de 90, como alternativa as metodologias mais tradicionais e ao formalismo de documentações e longos ciclos de desenvolvimento como ocorria com o modelo cascata.

Conforme Silva e Videira (2001), incluso nessas metodologias, há a necessidade de se criar documentos e artefatos de desenvolvimento, que auxiliem tanto o cliente quanto o time de desenvolvimento a modelarem corretamente o software, atendendo todas as expectativas. Neste sentido, foi criada a UML (Linguagem de Modelagem Unificada), que é uma linguagem de notação que permite representar um sistema através de um padrão. Um dos padrões utilizados é a criação de diagramas, como o de classes, de caso de uso e de sequência:

- a) No diagrama de classes, todos as entidades de um sistema são definidas, suas estruturas e as interações entre elas;
- b) No diagrama de caso de uso são definidas as relações entre os atores (possíveis utilizadores do sistema) e as funcionalidades do sistema, auxiliando na definição do que cada ator poderá fazer no sistema;

- c) No diagrama de sequência, são definidas as possíveis interações entre os objetos do sistema em relação a um período de tempo.

2.2.4 Desenvolvimento de sistemas web e aplicativos móveis

Para o desenvolvimento de um sistema de indústria 4.0, algumas tecnologias são necessárias, como a utilização de uma linguagem de programação de alto desempenho, um sistema de banco de dados escalável e uma tecnologia que facilite a construção de um aplicativo móvel.

Um dos fatores críticos em um sistema de indústria 4.0 é o tempo. As tecnologias empregadas precisam ser rápidas o suficiente para de certa forma dar a impressão de que o sistema está operando em tempo real.

- Linguagem de programação de alto desempenho

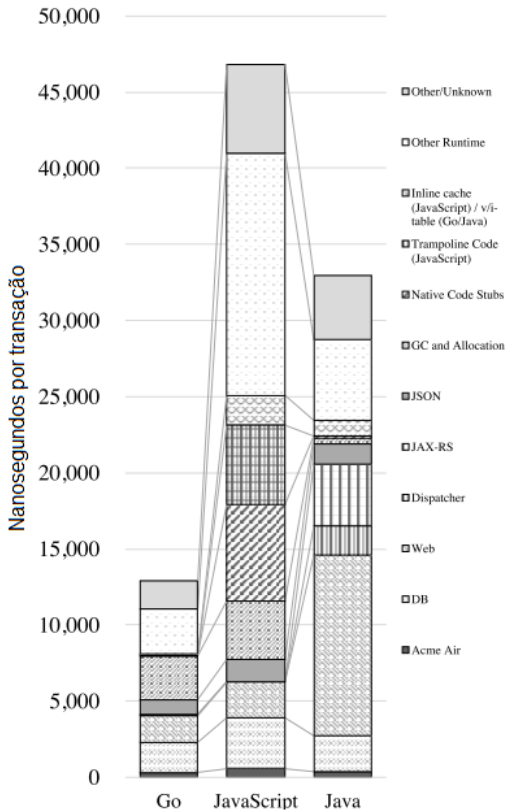
Ueda (2017) desenvolveu um estudo em que verificou o desempenho de três linguagens de programação: GO, Javascript e JAVA. Segundo o autor, a escolha dessas linguagens ocorreu porque a GO é ainda emergente, enquanto Javascript e JAVA são tradicionalmente utilizadas para desenvolver uma aplicação web.

A linguagem GO foi criada em 2007 com o suporte da empresa Google a fim de suprir algumas necessidades que outras linguagens mais antigas possuíam: facilidade na programação (entendimento do código e manutenção futura), alto desempenho e simplicidade em todas as suas funções (KHAITAN; MCCALLEY, 2015).

Ueda (2017) utilizou uma ferramenta para simular requisições em cenários como: realizar login no sistema, atualizar um perfil de cliente e deslogar. Foram analisadas diversas características relacionadas à requisição, como tempo de processamento e quantidades de ciclos de CPU.

A Figura 15 mostra a diferença entre os tempos de processamento nas 3 linguagens, considerando todas as atividades necessárias para realizar uma requisição, como utilização do banco de dados, envio e recebimento de um objeto JSON, acesso ao sistema web e outros.

Figura 15 - Tempos de processamento.



Fonte: Ueda (2017).

Togashi e Klyuev (2012) utilizaram a linguagem GO para desenvolver um sistema de gerenciamento de apontamentos. Os autores concluíram que apesar da pouca idade, a linguagem possui várias bibliotecas e frameworks disponíveis. Segundo os autores, a linguagem é de fácil escrita, o que tornou possível o desenvolvimento de protótipos em um curto espaço de tempo. Além disso, essa simplicidade também torna os custos de aprendizagem menores.

- Aplicativos móveis

Clemente, Esquembre e Wee (2017) citam que existem duas alternativas para a criação de um aplicativo móvel: nativo ou híbrido. Os aplicativos nativos são gerados especificamente para uma plataforma, enquanto os híbridos são desenvolvidos com outras tecnologias como as web (HTML5, CSS e Javascript) e possibilitam a criação de um mesmo código para diferentes plataformas.

Huynh e Truong (2017) citam que, dentre diversas outros frameworks para desenvolvimento de aplicativos híbridos, o IONIC é o mais popular do mercado e por razões como ampla documentação, fácil utilização e gratuidade de licença é provavelmente a primeira escolha de muitos desenvolvedores.

O IONIC é um framework de licença gratuita, utiliza bibliotecas que acessam recursos nativos do dispositivo móvel (como câmera, navegador, telefone) e possui vasta documentação e comunidade de desenvolvedores online (WANG et al., 2017).

- Banco de dados e serviço na internet

O banco de dados é responsável por armazenar e permitir a manipulação de todos os dados que tramitam em um ou entre diversos sistemas. Nos modelos de Indústria 4.0 é essencial que o banco de dados seja acessível pela internet.

Um dos serviços disponíveis para armazenar e manter um banco de dados na internet é o Firebase. Este sistema fornece armazenamento em nuvem em tempo real, permitindo que um aplicativo e um Website, por exemplo, mantenham uma comunicação com o banco de dados usando objetos JSON (KUMAR et al., 2016).

A utilização do formato JSON é comum em muitas aplicações de diversas plataformas, o que torna o Firebase uma ferramenta interessante quando o objetivo é utilizar os mesmos dados entre diferentes contextos. Além disso, também é possível criar *listeners* (em tradução livre “ouvintes”) em seções específicas do documento JSON, que disparam quando dados são criados, alterados ou removidos (BOMAN; TAYLOR; NGU, 2014).

Outros autores como Taherkordi, Eliassen e Horn (2017) e Radu et al. (2015) também citam as vantagens listadas.

Daramas et al. (2016) fez uma análise entre sistemas parecidos com o Firebase e escolheu utilizá-lo em seu projeto. Dentre os motivos,

também acrescentou que este serviço possui autenticação integrada, o que garante a segurança dos dados que são trafegados e diminui esforços no desenvolvimento de autenticações próprias para cada sistema.

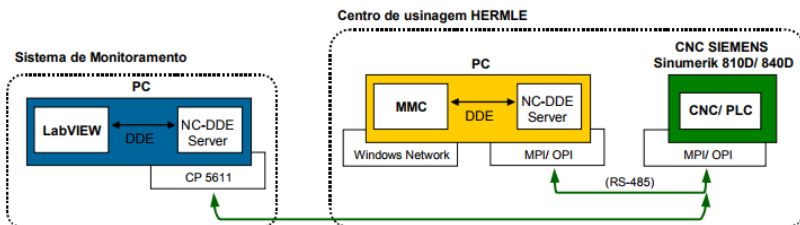
2.3 MONITORAMENTO E CONTROLE

O monitoramento e controle de uma máquina pode ser realizado de diversas formas. Nesta seção são apresentados trabalhos que realizam o monitoramento internamente (acessando as máquinas diretamente ou através de uma rede local) e remotamente (acessando-as através da internet), o sistema SCADA e também são apresentados aplicativos móveis para monitoramento e controle de máquinas.

2.3.1 Sistemas de Monitoramento Interno

Segundo Júnior, Gomes de Oliveira e Fagali de Souza (2005) inicialmente as máquinas CNC possuíam somente arquitetura fechada, o que impossibilitava qualquer tipo de acesso aos seus dados e variáveis de usinagem. Porém, alguns fabricantes de máquinas começaram a desenvolver arquiteturas abertas, onde uma interface de comunicação padronizada é disponibilizada para que aplicações externas possam utilizar os dados da máquina. Existem algumas configurações de arquiteturas abertas e os autores deram ênfase a arquitetura híbrida, onde o cerne do CNC fica fechado, porém sua interface é baseada em PC (computadores pessoais), permitindo uma maior facilidade na integração e implementação de aplicativos. A arquitetura apresentada na Figura 16 mostra a interação entre o sistema de monitoramento e o centro de usinagem: no sistema de monitoramento é necessário o programa LabVIEW para criar a operação que irá monitorar e controlar a máquina, um servidor NC-DDE que se conecta ao comando numérico computadorizado (interface), onde o comando envia os dados a outro servidor NC-DDE para realizar a troca de mensagens entre o computador da máquina e o computador externo através de um protocolo chamado DDE. Os autores exemplificaram a utilização desse conceito na aquisição de dados durante uma usinagem, buscando a velocidade de avanço real e programada e as coordenadas X e Z do perfil da peça.

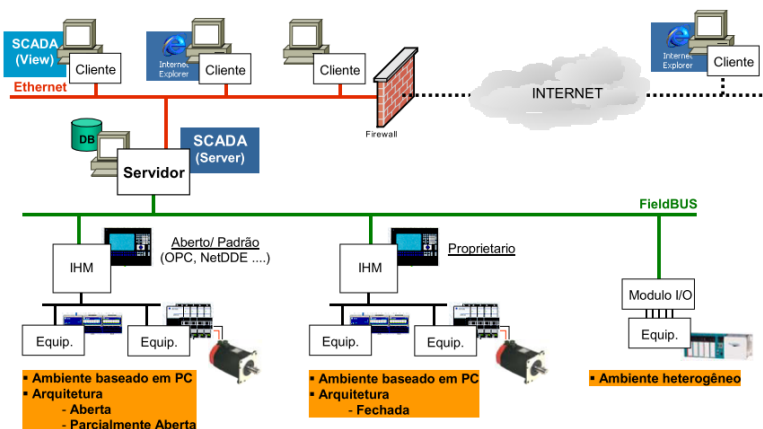
Figura 16 - Arquitetura CNC aberta.



Fonte: Júnior, Gomes de Oliveira e Façali de Souza (2005).

Conforme Figura 17, os autores citam que uma arquitetura típica de um sistema de supervisão é composta por computadores clientes e servidores de dados. Os servidores se conectam ao ambiente industrial através de uma interface IHM – interface humano máquina – e captam os dados dos equipamentos. Os clientes se conectam aos servidores de dados para visualização dos dados e, através de uma rede externa – internet – esses dados podem ser também visualizados na nuvem. No trabalho, os autores realizam o monitoramento pela rede Ethernet (interna), porém não disponibilizam esses dados na internet, que é o diferencial e a proposta deste projeto.

Figura 17 - Arquitetura de um Sistema de Supervisão.



Fonte: Júnior, Gomes de Oliveira e Façali de Souza (2005).

Também utilizando uma CNC de arquitetura aberta híbrida, Souza e Coelho (2007) desenvolveram um experimento com CNC de arquitetura

aberta para monitorar a velocidade de avanço de uma máquina CNC e determinar sua variação durante a operação de fresamento.

Outra pesquisa desenvolvida por Jinsong et al. (2017) mostrou a utilização desse tipo de arquitetura para prever o tempo de vida útil restante da ferramenta de corte em usinagem em altas velocidades (*high speed machining*, HSM). Experimentos foram realizados e dados do processo como a velocidade de rotação, a carga da rotação e a carga dos eixos foram coletados em tempo real. O método *Feedforward neural network* (FFNN) e o método de detecção de anomalias foram utilizados para prever a vida útil restante. O processo de previsão pode ser dividido em três fases:

- a) Aquisição e manipulação de dados: utilizando software SCADA coleta-se conjuntos de dados relacionados. Em seguida, é feita a limpeza, filtragem, seleção de características relevantes e a criação do modelo de mapeamento de todos os tipos de dados;
- b) Fase de análise e modelo: avalia a condição atual da ferramenta monitorada. Em sequência detecta, isola e identifica falhas e estima a vida útil restante. Além disso, recomenda ações de manutenção, quando *off-line*, e configurações de sistema quando *on-line*;
- c) Fase de interação: o método PCA (análise do componente principal) foi usado para filtrar fatores secundários, restando apenas os fatores principais. Então o método FFNN pode ser aplicado para prever a vida útil restante.

O MTConnect e a tecnologia OPC UA foram usados pra transferir dados entre os sensores das máquina-ferramentas e o sistema. MTConnect é um padrão de comunicação entre máquinas, que consiste de alguns adaptadores, um agente, e um programa de cliente. Os adaptadores são responsáveis por realizar a comunicação entre as máquinas conectadas na rede. O agente envia os dados obedecendo os requisitos do MTConnect, via Internet, para aplicativos compatíveis com MTConnect.

Com os testes experimentais notou-se que a carga e a velocidade de rotação aumentaram conforme o desgaste da ferramenta também aumentava. Entretanto, houve pouca mudança na corrente. O resultado mostrou que o tempo de vida restante previsto da ferramenta é próximo do real (JINSONG et al., 2017).

García e Núñez (2018) aplicaram técnicas para processar as forças de corte e monitorar o acabamento superficial em CNCs, em que os dados de força de corte foram coletados através de um dinamômetro que realizou a medição na máquina.

O monitoramento realizado de forma interna, seja através de conexão direta a máquina ou sensores, possui alguns pontos a serem considerados. No trabalho de García e Núñez (2018), por exemplo, foi utilizada uma ferramenta externa para buscar dados que já estão sendo monitorados pela própria máquina, realizando duas vezes o mesmo trabalho.

Schützer et al. (2012) desenvolveram uma pesquisa em que identificaram ser possível suavizar a taxa de avanço da máquina ferramenta na operação de acabamento para melhorar a precisão e textura da superfície. Para encontrar a velocidade de avanço ideal, eles monitoraram as posições dos eixos e a velocidade do eixo árvore, utilizando uma máquina CNC de arquitetura aberta, em um procedimento de aquisição de dados interno que monitora e salva os dados continuamente em um *buffer*, para depois enviá-los a um computador externo. O *buffer* foi desenvolvido utilizando variáveis internas da CNC, chamadas de parâmetros R e é programado diretamente no programa NC. Periodicamente, esses dados são enviados a um PC externo, onde ocorre a análise dos dados. Foi utilizado o programa LabVIEW 8.0, para desenvolver o sistema de aquisição, um cartão flash (CP5511) e um programa de troca de dados dinâmico da Siemens para transmitir e traduzir os dados do cartão para o computador.

No trabalho de Schützer et al. (2012) existe um processo de monitoramento diretamente na máquina CNC, porém os dados visualizados no programa de monitoramento e análise dependem de um método de transmissão com muitos processos, o que pode gerar discrepância de tempo entre os dados reais da máquina e os monitorados.

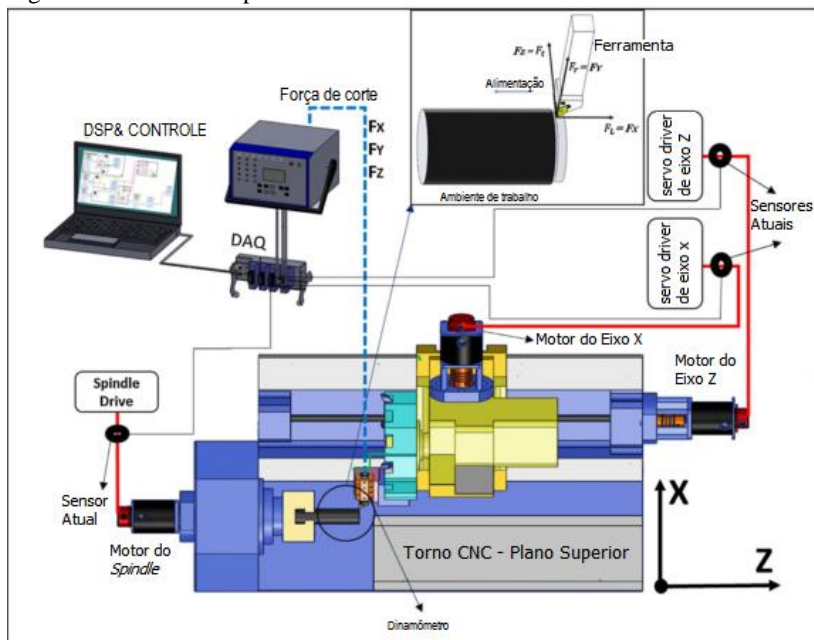
A arquitetura aberta apresentada pelos autores possibilita o monitoramento e controle de uma máquina CNC, porém, como apresentado, a comunicação depende do controle numérico (interface), de um sistema LabVIEW e de servidores para codificar e decodificar a mensagem trocada. Utilizando essas camadas de comunicação, supõem-se que há um tempo maior entre o dado lido na malha de controle da máquina e o produzido pelo sistema de monitoramento, o que pode gerar um falso imediatismo e ações de controle podem ser tomadas tardiamente (SCHÜTZER et al., 2012).

De Santiago-Pérez et al. (2008) desenvolveram um algoritmo capaz de extrair os parâmetros de uma máquina CNC com o objetivo de criar um método que obtenha a dinâmica dos eixos da máquina. Segundo os autores, a dinâmica dos eixos da máquina é um fator importante que influencia o acabamento da peça e interfere no desgaste da máquina e dos atuadores. Os parâmetros que descrevem essa dinâmica são: posição,

velocidade, aceleração e *jerk*. Para a criação do método, os autores utilizaram um codificador óptico para obter a posição dos eixos e um filtro *wavelet* que suaviza os dados recolhidos e obtêm os parâmetros dinâmicos de velocidade, aceleração e *jerk*.

Sparham et al. (2017) aplicaram uma modelagem de inteligência artificial para prever a força de atrito nas guias lineares do CNC, e a corrente dos servomotores no sistema de acionamento de alimentação, quando ocorre a lubrificação a seco. Primeiramente, as forças de atrito são calculadas a partir da análise da força de corte durante a lubrificação a seco. Nas mesmas condições, também são medidas as correntes do servo-motor. Para o experimento, foram utilizados sensores externos acoplados a máquina e não foram feitas interações com o comando numérico. Os dados dos sensores foram conectados a um DAQ – sistema de aquisição de dados - e do DAQ esses valores foram enviados a um computador, como mostra na Figura 18.

Figura 18 - Sistema de processamento de dados.



Fonte: Adaptado de Sparham et al. (2017)

Segundo Sparham et al., (2017), a previsão da força de atrito nas guias lineares e na corrente do servo-motor é possível através de dados de

treinamento (históricos de cortes), utilizando o algoritmo de inteligência artificial ANFIS no software MatLab.

Outros autores como Xu et al. (2002) e Arm, Bradac e Fiedler (2016) também desenvolveram pesquisas baseadas em monitoramento de sensores acoplados a máquina CNC e acesso direto a arquitetura da máquina para monitoramento de dados.

A seguir, são apresentadas pesquisas onde se utiliza o monitoramento remoto de máquinas CNC para manutenção ou controle dos processos de usinagem.

2.3.2 Sistemas de Monitoramento Remoto

Alguns trabalhos vêm sendo desenvolvidos com a proposta de monitoramento e controle remoto de máquinas CNC, suas ferramentas e componentes.

Röschinger, Kipouridis e Günthner (2016) desenvolveram um conceito de um sistema de compartilhamento de dados sobre ferramentas de corte. No sistema, uma ferramenta pode ser identificada através de etiquetas RFID (identificação por radiofrequência), por exemplo, que é única para cada classe de ferramentas e armazena seus dados em uma nuvem compartilhada. Além de dados do fabricante, o modelo também propõe que o sistema atualize um banco de dados na nuvem com o histórico de utilização de cada ferramenta. Essa funcionalidade deverá ser implementada em cada máquina das indústrias que pretendem utilizar o sistema. Segundo os autores, além de outros benefícios, esse sistema poderá ser capaz de informar quanto material uma ferramenta específica já foi capaz de usinar e possibilitará previsões para manutenção e substituição da ferramenta.

Fujishima et al. (2016) desenvolveram um estudo sobre a utilização de sensores em máquinas ferramenta. Um centro de usinagem NVX 5080 foi equipado com 24 sensores adicionais aos já desenvolvidos pelos fabricantes, como:

- a) 6 sensores de aceleração: utilizados no eixo árvore e na mesa para medir vibração e impacto de colisão;
- b) 2 sensores de temperatura;
- c) 1 sensor de corrente e 1 sensor de tensão: utilizados para medir a potência utilizada pela máquina.

Segundo os Fujishima et al. (2016), o uso desses sistemas permitiu o monitoramento remoto de um enorme volume de dados provenientes da máquina, que podem ser utilizados posteriormente para demais

aplicações. As tecnologias de sensoriamento evoluem com a assistência da tecnologia da informação.

Oliveira e Álvares (2016) aplicaram a técnica *Design* axiomático para o levantamento de necessidades no desenvolvimento de sistema para monitoramento de máquinas CNC, através da internet. A seguir são listados alguns dos requisitos encontrados, por ordem de importância:

- a) Monitoramento do status da máquina: ligada ou desligada;
- b) Status da porta da máquina;
- c) Status de refrigeração;
- d) Velocidade de avanço atual;
- e) Posição dos eixos atual;
- f) Alarmes;
- g) Nome do programa sendo usinado;
- h) Velocidade de rotação.

A maioria dos requisitos listados foram implementados neste projeto, mostrando uma concordância com as necessidades reais de clientes e o software proposto.

A pesquisa de Oliveira e Álvares (2016) também mostrou que requisitos como a aquisição de imagens e vídeos do ambiente industrial, ficaram mais abaixo na priorização, o que demonstra a necessidade de aplicações simples, sem demanda de grandes investimentos em hardware e/ou dependência de acessibilidade e velocidade de internet.

Edrington et al. (2014) pesquisaram sobre um sistema de monitoramento remoto de máquinas utilizando a tecnologia MTConnect, que engloba uma máquina CNC da empresa DMG MORI, modelo SEIKI. Essa máquina possui nativamente uma interface humano computador chamada MAPPS, que possui Windows e suporta conexão Ethernet, que viabiliza a conexão a outras máquinas ferramentas, computadores ou sistemas de monitoramento (como o Mori Net e Messenger, da própria DMG MORI).

Segundo os Edrington et al. (2014), inicialmente a DMG possuía o sistema Mori Net, que permitia conexão intranet ou através da instalação de um *software* no sistema operacional Windows. Porém, para se atualizar junto ao mercado tecnológico, a empresa criou o sistema Messenger, que permite monitoramento das máquinas através da internet, utilizando o protocolo MTConnect, que é um protocolo de comunicação aberto que tem por objetivo a padronização da conexão e comunicação com outros equipamentos do ambiente industrial. O protocolo utiliza XML e tecnologia HTTP para realizar as requisições. Ainda conforme os autores, alguns dos requisitos utilizados pela empresa para a criação do Messenger foram:

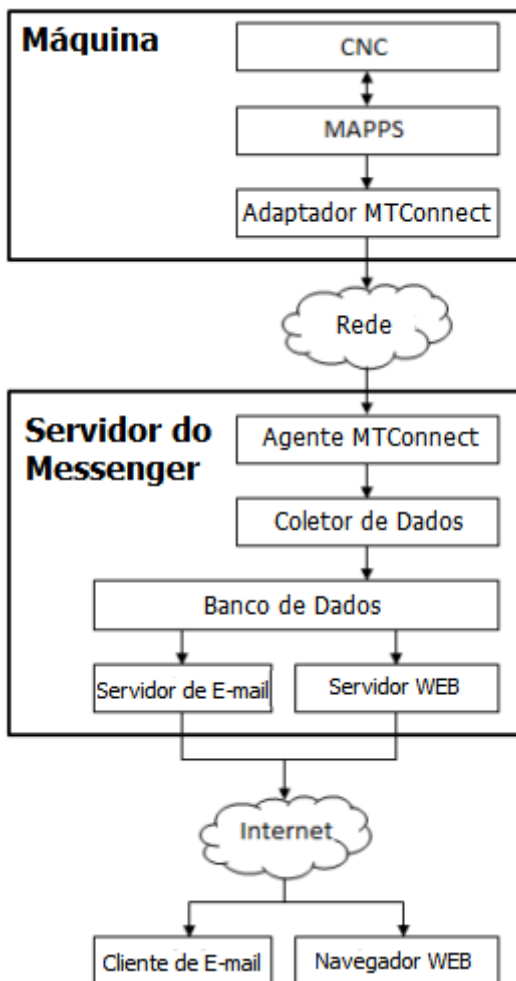
- a) O monitoramento da máquina deve ser acessível ao fabricante em qualquer lugar do mundo que forneça acesso à Internet;
- b) O acesso ao sistema deve ser seguro, a fim de se defender contra ataques maliciosos pela Internet;
- c) Os relatórios apresentados pelo serviço de monitoramento devem fornecer informações suficientes para determinar a condição operacional da máquina;
- d) Os dados coletados devem estar disponíveis aos clientes para que possam utilizá-los em seu sistema de gerenciamento de produção;
- e) O sistema de monitoramento deve permitir que os dados do relatório sejam exportados para formatos de arquivo padrão (como CSV e PDF);
- f) O sistema deve ser capaz de coletar dados de qualquer dispositivo que seja compatível com MTConnect;
- g) O sistema deve fornecer acesso a dados de monitoramento por meio de um navegador da Web e não exigir o uso de um *software* no computador;
- h) O sistema deve ser funcional em dispositivos móveis, como tablets e smartphones;
- i) O sistema deve ser capaz de fornecer níveis variados de acesso, dependendo do nível de acesso do usuário. Por exemplo, os gerentes devem ser capazes de monitorar todas as máquinas sob sua gestão, mas um operador de máquina só deve ter acesso às máquinas que lhes foram atribuídas;
- j) O sistema deve ser capaz de agendar uma notificação aos usuários quando ocorrer um evento de máquina;
- k) O sistema deve ser capaz de detectar e notificar os usuários quando ocorrer uma falha na máquina;
- l) O sistema deve ser tolerante a falhas de rede e recuperar dados quando ocorrerem interrupções na rede;
- m) O sistema deve ser distribuído e escalonável;
- n) O sistema deve funcionar em um servidor de rede local ou na nuvem, de acordo com a política de TI do cliente.

A Figura 19 mostra a arquitetura do sistema Messenger. Dentro da máquina ferramenta, fica o comando numérico, o sistema MAPPS e o adaptador MTConnect. A máquina se conecta a uma rede, que envia os dados ao servidor do Messenger. No servidor, também é implementado o

protocolo MTConnect, para interpretar os dados enviados pela máquina. Além disso, no servidor também está armazenado o banco de dados e os serviços de envio de e-mail e servidor web.

Por outro lado, o cliente pode se conectar ao servidor do Messenger através da interface web ou então receber e-mails de notificação.

Figura 19 - Sistema de monitoramento.



Fonte: Adaptado de Edrington et al. (2014).

Os sistemas de monitoramento remoto apresentados possuem requisitos parecidos com os definidos para o desenvolvimento desta pesquisa, a citar: conectividade, acessibilidade e interação amigável com o cliente. Porém, sistemas como o DMG MORI, por exemplo, são desenvolvidos por grandes fabricantes que não disponibilizam as tecnologias utilizadas para a fabricação do software. Não mencionam qual linguagem de programação foi utilizada, qual o tipo do banco de dados ou de servidor, impossibilitando o entendimento do sistema no âmbito acadêmico. Apesar disso, a disponibilização dos requisitos utilizados pela empresa, viabilizam a validação de um sistema de monitoramento de máquina ferramenta.

2.3.3 Sistemas para monitoramento (SCADA)

Segundo Geng (2017), o termo SCADA começou a ser utilizado em 1960 e é um tipo de tecnologia usada para monitorar e controlar componentes físicos e processos, que foi concebido na terceira revolução industrial, onde os computadores começaram a auxiliar na automação das máquinas. O autor cita também que a quarta revolução industrial é a da “internet das coisas” e que apesar dos sistemas SCADA tradicionais possuírem conceitos parecidos com os sistemas de internet das coisas, muitas das características dos sistemas de internet das coisas nem existiam quando o primeiro sistema SCADA foi desenvolvido. Não havia acesso à internet publicamente como hoje em dia, também não existiam sistemas móveis e, principalmente, os sistemas SCADA eram desenvolvidos com um propósito específico e não compartilhavam dados com demais sistemas e “coisas”, enquanto que um sistema de internet das coisas se comunica com sensores, bancos de dados, recursos da web e usuários.

Verma et al. (2016) introduziu o Avex-SG Technology Inc., um sistema integrador taiwanês focado em componentes de máquinas-ferramentas, com o intuito de obter informações das máquinas e aumentar o valor final de produção para seus clientes, adotou a solução de rede de máquinas-ferramentas CNC fornecido pela empresa Advantech.

Redes de equipamentos permitem ao engenheiro ajustar o modo de operação da máquina CNC remotamente, preservando o tempo que seria necessário para fazer essas alterações manualmente. Como todas as informações são agregadas na mesma plataforma, o sistema automaticamente produz relatórios sobre o processo. Conseqüentemente, é possível analisar o desgaste da ferramenta, viabilizar a criação de modelos de manutenção de equipamentos permitindo que o operador

realize a troca necessária antes que ocorra uma falha na ferramenta ou na peça usinada.

Baseado no sistema WebAccess/SCADA, Advantech introduziu WebAccess/CNC, que possui interfaces para os quatro maiores fornecedores de controle CNC (Fanuc, Mitsubishi, Heidenhain, Siemens), permitindo os usuários coletarem informações diretamente das máquinas CNC conectadas em rede sem precisar customizar as máquinas ou instalar sensores. As principais funções do WebAccess/CNC são: gerenciamento do programa de usinagem, configuração de parâmetros de compensação, histórico de alarmes, status de rotação da ferramenta e manutenção preventiva. Portanto, é possível visualizar as informações da máquina, em tempo real e através de dispositivos móveis. Este sistema vem previamente integrado com os protocolos de rede dos fornecedores de controle CNC mencionados, permitindo acesso direto aos controles de parâmetros e informações do processo através da configuração do software. WebAccess/CNC apresenta as mesmas características de monitoramento e controle de um sistema SCADA, além de sua configuração flexível e capacidade de integração contínua (LIU et al., 2008).

Jinsong et al. (2017) utilizou um sistema SCADA para o monitoramento do estado da ferramenta em fresamento de alta velocidade de placas de titânio. Neste caso, o sistema SCADA recebe como dados de entrada leituras de sensores, dados de medições anteriores e arquivos de processos de cortes e tenta prever o estado da ferramenta.

Cao, Zhang e Chen (2017) propuseram um conceito integrado de eixos árvores inteligentes. Tradicionalmente, os eixos árvores obedecem passivamente aos comandos inseridos pelo operador, porém os eixos inteligentes devem ser capazes de ajustar parâmetros operacionais e otimizar os processos de usinagem e desempenho do eixo através de conhecimento adquirido autonomamente. Os autores consideraram seis funções necessárias para os eixos inteligentes, conforme Figura 20: monitoramento e controle da condição da ferramenta, vibração, colisão do eixo, temperatura, equilíbrio e condição do eixo árvore.

Figura 20 - Eixo árvore inteligente.



Fonte: Adaptado de Cao, Zhang e Chen (2017).

As funções dos eixos árvores são integradas em um sistema de supervisão: na detecção, sinais são monitorados através de sensores integrados, formando uma rede de sensores, esses dados da rede de sensores são processados por um sistema de tomada de decisão que se baseia em conjunto de regras e lógicas (sistema de inteligência artificial). Caso haja alguma anormalidade, um sistema de controle gerencia atuadores, o CNC e/ou notifica o operador para executar as tarefas necessárias para o controle (CAO; ZHANG; CHEN, 2017).

Os autores também demonstram a integração de um eixo árvore em um sistema completo de indústria 4.0: os dados dos sensores e da máquina CNC são obtidos através de um protocolo de comunicação e enviados para uma nuvem de dados que analisa e toma alguma decisão com base nas regras já obtidas anteriormente (com a aplicação de algum algoritmo de inteligência artificial), ao identificar alguma anormalidade, o sistema envia uma notificação ao operador através de sistemas web, onde o operador poderá interferir ou o próprio sistema toma as decisões (se perceber que consegue corrigir sem a intervenção humana). A integração de um volume grande de dados pode facilitar o desenvolvimento dos eixos árvore inteligentes e as fábricas da indústria 4.0 permitirão que as máquinas CNC se conectem como uma comunidade colaborativa, permitindo essa aquisição de dados (CAO; ZHANG; CHEN, 2017).

SCADA é um tipo de sistema, qualquer monitoramento e controle realizado em uma máquina pode ser considerado SCADA. Porém, os sistemas SCADA criados nas décadas anteriores não se comunicavam com o ambiente industrial como atualmente, que atendem as propostas da indústria 4.0.

2.3.4 Monitoramento e Controle por Dispositivos Móveis

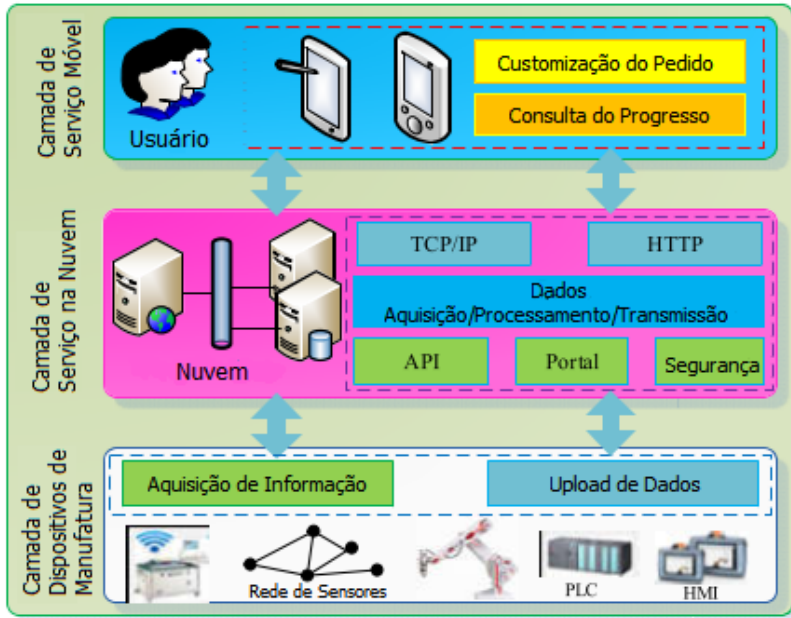
A internet das coisas, onde qualquer coisa pode estar conectada à internet e a um banco de dados, possibilitou a criação dos mais variados tipos de aplicativos para monitoramento. Na área da saúde, trabalhos como os de Iturralde et al. (2016) e Hossain e Mudammad (2016) aplicaram os conceitos de internet das coisas e a utilização de aplicativos móveis para monitorar a saúde de pacientes. Hossain e Mudammad (2016), por exemplo, implantaram um sistema para o monitoramento cardíaco de um paciente, onde o aparelho que realiza o eletrocardiograma envia os sinais de leitura ao celular do paciente e, caso haja alguma variação considerável, o médico consegue visualizar em seu aplicativo.

Considerando a área industrial, Wan et al. (2016) cita que os clientes não são mais apenas compradores passivos de um processo de fabricação existente. Cada vez mais possuem a necessidade de customizar seus produtos e participar do projeto de seus produtos antes da compra.

Com base nisso, o autor desenvolveu uma pesquisa em que utiliza tecnologias móveis, computação em nuvem e demais tecnologias existentes para criar um sistema personalizado de fabricação customizada (*Personalized Customization Manufacturing System - PCMS*).

A Figura 21 mostra as camadas desenvolvidas pelo autor para a conclusão do projeto. Na camada do serviço na nuvem foi utilizada uma chave de segurança para identificar cada usuário e garantir que os dados só trafeguem com usuários devidamente cadastrados no sistema e as mensagens enviadas entre os usuários e o serviço de computação em nuvem utilizam o formato de dados JSON. Esta camada também é responsável por receber os pedidos dos clientes e os estilos dos produtos, analisar as tarefas necessárias de produção, enquanto que o sistema de fabricação recebe as instruções de produção e decodifica os dados JSON para realizar a fabricação.

Figura 21 - Arquitetura de um PCMS.



Fonte: Adaptado de Wan et al. (2016).

Durante o processo de fabricação, é utilizada uma etiqueta RFID em cada produto e, durante a produção do produto, sensores RFID enviam dados para o serviço na nuvem. Na camada de serviço móvel, foi criado um aplicativo que não possui cadastro de usuário nem registro de chaves, sendo que esses dados foram diretamente configurados no aplicativo, que possui a customização de um único produto.

Através da pesquisa, Wan et al. (2016) concluíram que para realizar uma completa automação e tornar os processos de manufatura mais inteligentes, é necessário um estudo mais aprofundado, principalmente em aplicações de computação em nuvem.

Cognigni et al. (2015) desenvolveu um aplicativo que permite a manutenção remota de uma máquina de elevação que auxilia na montagem de peças pesadas. O aplicativo permite que operadores comuns, que não são especialistas em manutenção, possam testar e verificar o status da máquina. Para o funcionamento do sistema foi desenvolvida uma interface para disponibilizar os dados da máquina através de uma conexão em uma rede sem fio.

Os dados de status da máquina que foram incorporados no aplicativo são: curso do motor, tensão do motor, velocidade do motor, comandos inseridos pelo operador, tempos de manipulação, sinais de entrada e saída, alarmes e mensagens de aviso.

Além desses dados, o aplicativo também mostra um vídeo em tempo real da máquina operando. Ao ocorrer alguma necessidade de manutenção, os dados e o arquivo de vídeo podem ser enviados, por e-mail, para que o profissional especializado em manutenção resolva o problema (LIU et al., 2008).

Sipsas et al. (2016) desenvolveu um sistema para Indústria 4.0 em que uma de suas aplicações foi a criação de um aplicativo para o operador de uma linha de produção. Ao ocorrer um erro em uma das máquinas da linha, o operador pode aproximar seu celular para se conectar através de NFC com a máquina e baixar os dados que possivelmente causaram o erro.

Os aplicativos móveis são uma maneira rápida de se obter acesso à informação sem a necessidade de se deslocar até um local com um computador. Os trabalhos mostram que hoje já é possível monitorar e até mesmo controlar alguns dados de máquinas através de aplicativos móveis.

No entanto, a pesquisa também mostrou que a produção desse tipo de conteúdo científico é limitada. Ao realizar uma busca, por exemplo, na base de dados científicos SCOPUS, com as palavras chaves “*Industry 4.0*” e “*mobile app*”, são encontrados apenas 3 resultados de busca.

Para aprimorar o campo de busca e entender o que o mercado de máquinas CNC desenvolveu nos últimos anos com relação a Indústria 4.0, a próxima seção apresenta uma pesquisa de mercado realizada com as principais fabricantes da área.

2.4 PESQUISA DE MERCADO

Nesta pesquisa foram analisados três conceituados fabricantes de máquinas: DMG, Fidia e Mazak. O principal objetivo da pesquisa foi identificar quais recursos da Indústria 4.0 essas empresas empregam ou estão desenvolvendo. Ao final, também são apresentados aplicativos encontrados em lojas online para celular.

2.4.1 Principais Fabricantes do Mercado

A empresa DMG MORI possui um sistema de monitoramento de máquina CNC chamado Messenger. Este sistema pode realizar as seguintes funções:

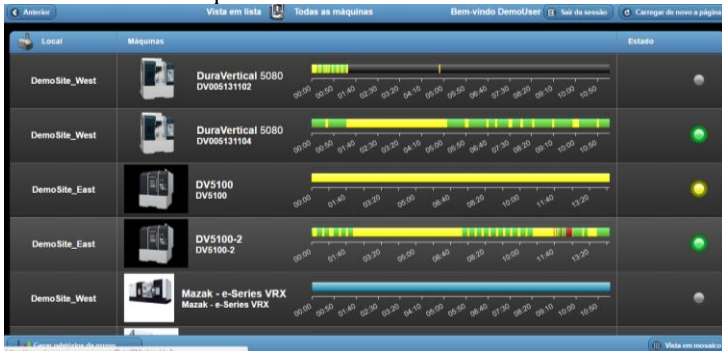
- a) Monitoramento on-line permanente de todas as máquinas;
- b) Compatibilidade com smartphones usuais;
- c) Visualização da estação de controle da máquina;
- d) Visualização direta do estado das máquinas por meio de uma luz de sinalização;
- e) Informações detalhadas do estado;
- f) Função de notificação por e-mail;
- g) Função de programação de turno integrada.

A Figura 22 mostra a visualização de todas as máquinas e um histórico de estados da máquina, assim como seu estado atual. Também é possível gerar relatórios e gráficos de estados da máquina, como o da Figura 23.

Em outra tela do sistema, também é possível visualizar detalhes de uma máquina, com informações ao vivo como:

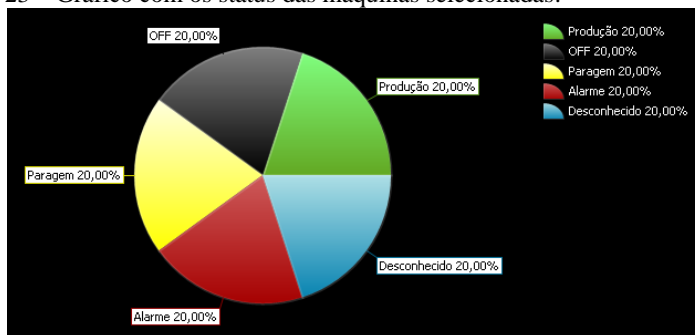
- a) Avanço;
- b) Número da ferramenta usada;
- c) Posicionamento dos eixos;
- d) Número do programa e linha que está sendo executada.

Figura 22 - Tela com as máquinas e seus status.



Fonte: (“DMG MORI Brasil - máquinas CNC de torneamento e fresamento”, 2017).

Figura 23 – Gráfico com os status das máquinas selecionadas.



Fonte: (“DMG MORI Brasil - máquinas CNC de torneamento e fresamento”, 2017).

Os comandos numéricos das máquinas da empresa FIDIA seguem o conceito de “Open CNC”, que possibilita a criação de aplicações baseadas em Windows. Segundo o fabricante, todos os parâmetros e registros da máquina podem ser acessados através de protocolos TCP/IP (“Fidia S.p.A.”, 2017).

A empresa Mazak desenvolveu uma ferramenta chamada “SmartBox”. O objetivo, segundo a fabricante, é monitorar, analisar e proteger os dados de manufatura. O SmartBox é um equipamento físico que pode ser montado ao lado de qualquer máquina, se conectando a sensores para realizar o monitoramento e coleta de dados, que são disponibilizados em uma rede. Este sistema pode ser utilizado em máquinas de outras marcas.

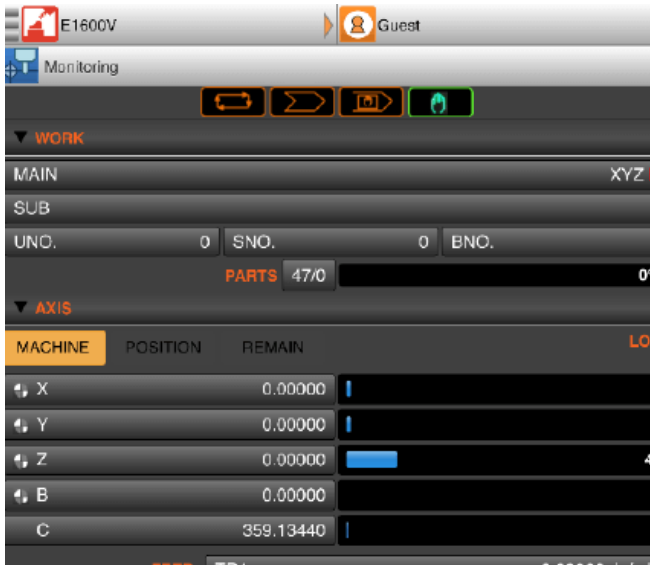
A empresa também desenvolveu o SmoothLink: um sistema que torna possível sincronizar a máquina com um dispositivo móvel para monitorar e gerenciar o status a qualquer momento a partir de um smartphone, tablet ou laptop, conforme Figura 24.

Projetada para sincronizar com os CNCs MAZATROL Smooth, o SmoothLink captura informações em tempo real do controle e transfere com segurança para um dispositivo móvel via Wi-Fi. Algumas de suas características são:

- a) Acesso e monitoramento em tempo real, ferramenta de corte e dados de programa de um dispositivo móvel com acesso Wi-Fi;
- b) Níveis de permissão de administração e gerais que concedem acesso total ou limitado do usuário a certas funções;

- c) Ferramentas e recursos de reescrita de programas de um local remoto para aqueles com acesso de usuário completo.

Figura 24 – Monitor SmoothLink.



Fonte: (“Mazak Sulamericana”, 2017).

2.4.2 Outras Aplicações

A seguir, são apresentados aplicativos encontrados em lojas de celulares, como Google Play e Apple Store.

- CNC Mobile

O aplicativo utiliza uma plataforma chamada Skymars para coletar informações da máquina CNC e disponibilizá-la em tempo real e está disponível para controles FANUC, Mitsubishi e Heidenhain.

Nas telas do aplicativo, que podem ser visualizadas na Figura 48 e Figura 49 do Anexo A, o usuário possui acesso a uma lista de CNC's do seu ambiente industrial. Ao clicar em uma das máquinas da lista, é possível navegar em outras telas que possuem informações como lista de status da máquina, dados de utilização, mostrando o tempo de uso, tempo de alarme, tempo ocioso e desligado, tabela de ferramentas e envio de

programa para a CNC (única ação de controle, as outras interações são de leitura).

- Thermwood CNC Mobile

Foi desenvolvido para máquinas da fabricante Thermwood, realizando apenas interações de leitura e não de controle. Os dados são atualizados a cada 30 segundos.

As telas, cujas imagens podem ser visualizadas na Figura 50 e Figura 51 do Anexo B, contemplam as seguintes informações:

- a) Lista de todas as máquinas;
- b) O estado atual (pronto, em execução ou em parada);
- c) O programa atual e carregado ou executando;
- d) Uma lista de eventos da máquina durante cada programa em ordem cronológica;
- e) A listagem do tempo de ciclo de cada programa em ordem cronológica;
- f) A listagem de conclusão ou interrupção de cada programa;
- g) Todos os eventos da máquina em ordem cronológica, com filtragem;
- h) Monitor de manutenção programada com um tempo de contagem regressiva e uma barra de progresso para a execução da próxima manutenção;
- i) Uma lista de todas as manutenções realizadas anteriormente;
- j) A capacidade de solicitar uma chamada de telefone do serviço de um técnico de suporte Thermwood a partir do dispositivo.

Os principais fabricantes de máquinas ainda estão desenvolvendo seus sistemas para empregar a Indústria 4.0. Por se tratar de um assunto relativamente novo, não há ainda um padrão de mercado.

A empresa DMG desenvolveu um sistema que pode ser acessado através de um website, porém não possui um aplicativo que gera notificações automáticas das máquinas.

A empresa Fidia ainda não possui uma aplicação de internet industrial propriamente dita, porém seus comandos estão disponíveis para acessos via protocolo TCP/IP, possibilitando que empresas terceiras desenvolvam soluções envolvendo suas máquinas.

A Mazak foi a que mais desenvolveu sistemas neste cenário, porém realizam apenas atividades de monitoramento, nenhum deles realiza o controle das máquinas pela internet.

O próximo capítulo apresenta os procedimentos que foram utilizados para a obtenção dos resultados deste trabalho.

3 METODOLOGIA

Neste capítulo são apresentadas as ferramentas e metodologias utilizadas para a criação do sistema monitoramento e controle e para a avaliação de desempenho.

3.1 DESENVOLVIMENTO DO SISTEMA

Para o desenvolvimento do sistema de Indústria 4.0, são necessárias as etapas de análise das funcionalidades desejadas (requisitos), desenvolvimento de método de comunicação de baixo nível com a CNC, desenvolvimento de serviço na nuvem, desenvolvimento do aplicativo e comunicação entre a nuvem, aplicativo e o CNC.

No desenvolvimento deste trabalho, é utilizada uma máquina CNC 5 eixos, modelo DM5, da empresa Tecnodrill (“Tecnodrill Engenharia de Soluções - Fresadora DM5”, 2018). Inicialmente, são controlados e monitorados os seguintes dados da máquina:

- a) Programa relativo à peça usinada (nome e progresso da usinagem);
- b) Velocidade de avanço;
- c) Alarmes.

A Figura 25 mostra um panorama de todas as tecnologias e ferramentas que foram utilizadas para desenvolver o sistema, nomeado de ControlTec.

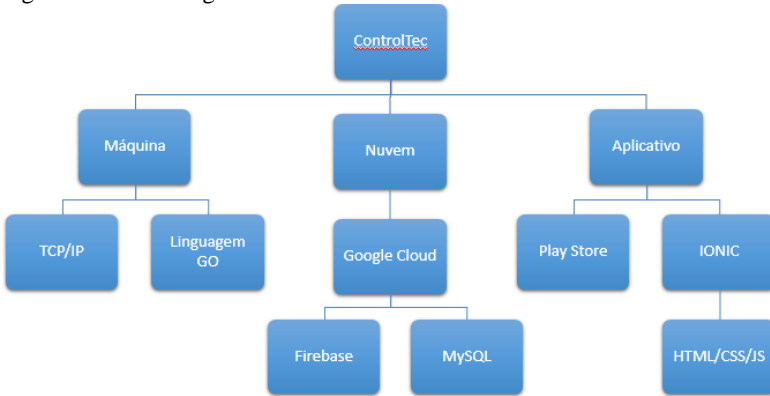
Na máquina é utilizada uma conexão TCP/IP para se conectar diretamente a máquina e poder acessar suas variáveis e funções. Foi utilizada a linguagem GO para criar uma rotina que interaja com a nuvem.

Na nuvem é criada uma conta de serviço no Google Cloud e nela utilizados dois serviços que são comparados neste capítulo: o Firebase e o Google Cloud SQL com banco de dados MySQL. Inicialmente, a empresa Google disponibiliza 300 dólares em crédito por 12 meses para testar os serviços. Como o sistema não foi comercializado, não foi necessário contratar mais do que o valor de teste.

Para o desenvolvimento do aplicativo é utilizado o *framework* IONIC que permite desenvolver aplicativos com o uso de HTML, CSS e Javascript, tecnologias comumente utilizadas para desenvolvimento *web*.

Além disso, pode-se utilizar os serviços do Google Play, que cobra um valor de 25 dólares para publicar o aplicativo na loja Google Play e disponibilizar para download nos dispositivos móveis com sistema operacional Android.

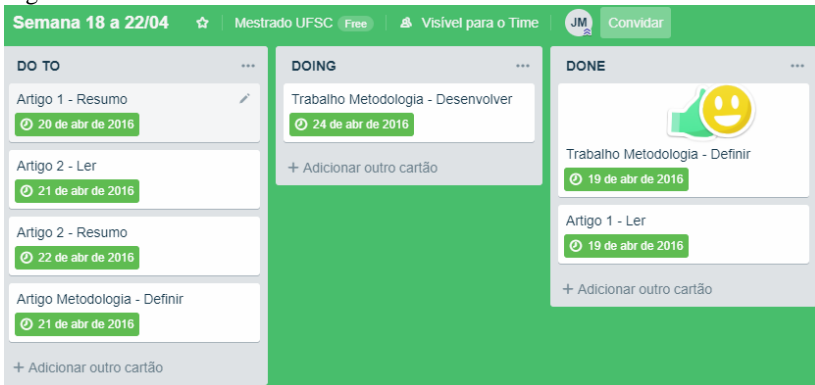
Figura 25 - Tecnologias e Ferramentas.



Fonte: Autor (2018).

Tanto para o desenvolvimento do sistema quanto da dissertação, são empregadas metodologias de engenharia de software e também artefatos do UML. Uma das ferramentas utilizadas é o *software* Trello, que é um quadro Kanban online, em que as tarefas podem ser divididas entre categorias de “a fazer”, “fazendo” e “feitas”. Na Figura 26 pode-se visualizar um exemplo da organização das atividades de uma semana de trabalho.

Figura 26 - Kanban com atividades.



Fonte: Autor (2018).

Além do Kanban, no processo de desenvolvimento do software são seguidas as etapas de análise, desenvolvimento e testes, utilizando o modelo incremental.

Na etapa de análise, realiza-se uma pesquisa de mercado, identificando as funcionalidades implementadas por fabricantes de máquinas CNC conceituadas e também uma entrevista com pessoas da empresa Tecnodrill. O resultado, apresentado no próximo capítulo, é a criação de um diagrama de classes para o programa da linguagem GO, outro diagrama de classes para o aplicativo, uma lista de requisitos para o aplicativo e também um diagrama de sequência para a interação usuário versus máquina.

3.1.1 Comunicação de baixo nível com a máquina CNC

O objetivo desta fase é possibilitar a conexão e interação da máquina CNC pela rede (internet/intranet). Para isto, são desenvolvidas duas abordagens:

- a) Criação de método capaz de obter os dados da máquina e disponibilizá-los;
- b) Criação de métodos que interajam com os dados da máquina, alterando-os em tempo real.

A máquina CNC utilizada, é a DM5 da empresa Tecnodrill, conforme Figura 27. Segundo a empresa, a fresadora DM5 foi concebida para a realização de trabalhos complexos e altamente precisos. Possui tecnologia de 5 eixos simultâneos, estrutura rígida e mecânica de precisão. (“Tecnodrill Engenharia de Soluções - Fresadora DM5”, 2018)

Figura 27 - Máquina DM5.



Fonte: “Tecnodrill Engenharia de Soluções - Fresadora DM5” (2018).

3.1.2 Criação de Serviço na Nuvem

Considerando que o sistema possa ser utilizado por grandes empresas, que pode exigir a movimentação de um alto volume de dados diariamente, nesta aplicação é utilizado um banco de dados não relacional, armazenado em uma plataforma criada pela empresa Google chamada de Firebase.

Além do banco de dados não relacional com formado JSON, o Firebase também possui outros serviços que são utilizados neste trabalho.

3.1.3 Desenvolvimento do Aplicativo

No desenvolvimento do aplicativo são realizadas as atividades de análise, desenvolvimento do protótipo e aplicativo.

- Análise dos requisitos para o aplicativo

Através das pesquisas realizadas, são definidas quais funcionalidades são esperadas pelo cliente do aplicativo, gerando um documento com os requisitos do sistema.

Com a lista de requisitos criada, são definidas as telas do sistema, a disposição dos dados nas telas e também a linguagem de programação e padrões de interface e comunicação entre *backend* e *frontend*.

- Desenvolvimento do protótipo

Para a prototipação, é utilizada a ferramenta online FluidUI. Com a realização da prototipação é possível definir o funcionamento e a usabilidade do sistema antes mesmo de sua programação, além de que, caso seja necessária alguma alteração no decorrer do desenvolvimento, a prototipação ajuda a validá-la e evita retrabalhos.

- Desenvolvimento do aplicativo

Nesta atividade o aplicativo é desenvolvido, respeitando o que foi definido nas atividades de análise e prototipagem. Durante todo o desenvolvimento, são realizados testes no código, garantindo funcionalidade ao sistema.

3.2 AVALIAÇÃO DE DESEMPENHO DO SISTEMA

Para avaliar o tempo de resposta entre as requisições e o servidor na nuvem, é realizado um teste de desempenho entre uma aplicação GO e o servidor da Google, comparando dois tipos de bancos de dados: relacional e não relacional.

Os testes são realizados a partir do mesmo computador, com processador Intel® Core™ i7 3537U /3517U, sistema operacional Windows 8.1, 6 GB de memória SDRAM DDR3 1600 MHz, placa de vídeo NVIDIA® GeForce® GT 740M com 2GB/4GB DDR3 VRAM e HD de 1TB. A velocidade de conexão é de aproximadamente 30Mbps.

4 RESULTADOS E DISCUSSÕES

Este capítulo apresenta os resultados que foram obtidos após a execução dos procedimentos metodológicos.

4.1 DESENVOLVIMENTO DO SISTEMA

O sistema desenvolvido é apresentado detalhadamente nas próximas subseções do capítulo.

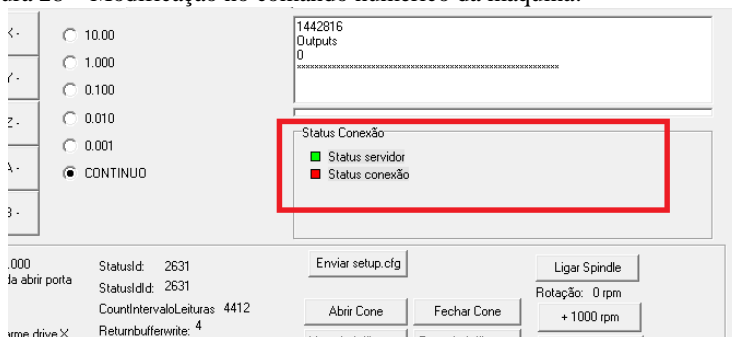
4.1.1 Criação do Método de Comunicação com a máquina CNC

Foi desenvolvido um *socket* TCP-IP que envia e recebe dados da máquina. Através de parceria com a empresa Tecnodrill, também foi alterado o Comando Numérico Computadorizado da máquina com novas funcionalidades que indicam quando o servidor e a conexão estão ativos.

A Figura 28 mostra na região destacada os dois novos campos adicionados à interface do CNC:

- Status servidor: verde caso o *socket* TCP-IP esteja disponível e vermelho se indisponível;
- Status conexão: verde se houver alguma conexão ativa com a máquina e vermelho caso não haja.

Figura 28 – Modificação no comando numérico da máquina.



Fonte: Autor (2017).

O *socket* pode receber uma sequência de bytes de dados que indicam que devem ser executadas na máquina as seguintes funções:

- 1º byte: quando recebe o valor 1, ativa a função de iniciar usinagem na máquina;

- b) 2º byte: quando recebe o valor 1, ativa a função de parar usinagem;
- c) 3º byte: quando recebe o valor 1, ativa a busca de referência da máquina;
- d) 4º byte: quando recebe o valor 1, indica que deve ser alterada a velocidade da máquina;
- e) 5º a 8º byte: recebe o valor da velocidade que deve ser alterada quando indicada a alteração da velocidade da máquina no 3º byte, podendo possuir os valores equivalentes a 100, 75, 50, 25 ou 0;
- f) 9º byte: quando recebe o valor 1, indica que o programa da máquina deve ser alterado;
- g) 10º a 110º byte: recebe o caminho e nome do programa a ser alterado.

Para iniciar uma nova usinagem em velocidade de avanço de 50%, por exemplo, indicando o caminho do programa que deve ser utilizado, pode ser enviado o conjunto de bytes (representado na Figura 29) para o método (em programação, função/ação de uma classe).

Figura 29 - Sequência Binária.



Fonte: Autor (2018).

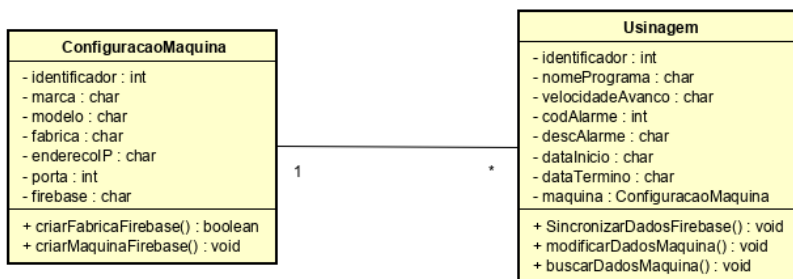
O socket retorna uma sequência de bytes referentes aos dados da máquina:

- 1º byte: se possuir valor decimal 1 indica que existe um programa sendo executado;
- 2º byte: indica se a máquina está em alarme caso possua valor decimal 1;
- 3º byte: caso possua valor decimal 1, indica que a usinagem terminou;
- 4º a 7º byte: retorna a velocidade de usinagem, podendo ser 0, 25, 50, 75 e 100;
- 8º byte: caso possua valor decimal 1, a máquina está em alarme;
- 9º a 12º byte: código do alarme;
- 12º a 15º byte: rotação do eixo árvore da máquina;

- h) 16º a 116º byte: nome do programa relacionado à peça está sendo usinada.

Os métodos são acessados através de um serviço interno desenvolvido na linguagem GO, que serve como uma ponte entre a máquina e o serviço na nuvem. Foi criado o diagrama de classes apresentado na Figura 30. A classe ConfiguracaoMaquina possui os dados da máquina que devem ser configurados para que seja feita a conexão do serviço com a máquina e os dados sejam sincronizados com o Firebase, assim como métodos que criam/atualizam um novo registro de fábrica e máquina no Firebase. A configuração da máquina é única e fica em um arquivo com extensão “.config”, diferente da usinagem que é criada a cada nova identificação de início de usinagem (proveniente da sequência binária). A classe Usinagem mantém todos os dados necessários para a criação e atualização de um registro de usinagem, bem como os métodos de modificar dados na máquina (conexão TCP-IP), buscar dados da máquina (conexão TCP-IP) e sincronização dos dados com o banco de dados Firebase.

Figura 30 - Diagrama de Classes do programa GO.

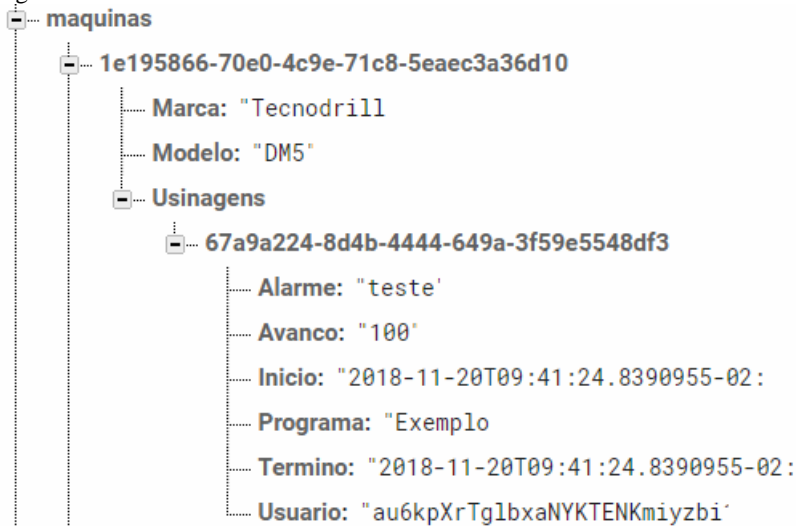


Fonte: Autor (2018).

4.1.2 Criação de Serviço na Nuvem

A Figura 31 mostra um exemplo do banco de dados criado. Não possui tabelas e relacionamentos como os bancos relacionais e foi criado utilizando uma estrutura de objeto JSON, onde os dados podem ser salvos em um arquivo de texto e facilmente transportados entre diferentes sistemas.

Figura 31 - Banco de dados.

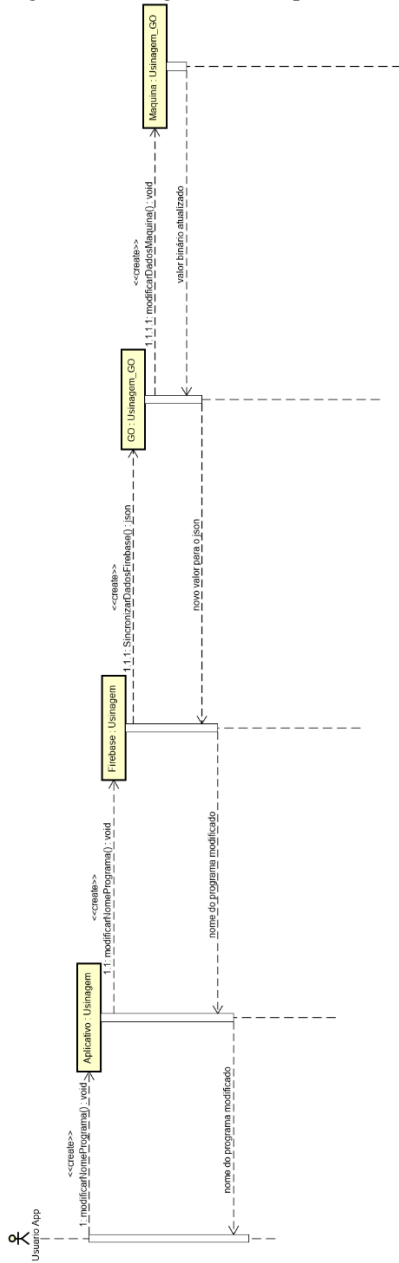


Fonte: Autor (2018).

Foi criado também um serviço na linguagem GO, que é armazenado na intranet da fábrica, para acessar e enviar dados ao banco de dados. Os dados enviados e recebidos por esse serviço seguem o padrão JSON, facilitando a comunicação entre o serviço interno, a máquina e o aplicativo. Este serviço é responsável por receber os dados da máquina, preparar o objeto JSON, enviá-lo ao Firebase e receber os dados JSON do Firebase e enviá-los a máquina.

Um diagrama de sequência, apresentado na Figura 32, foi criado para exemplificar toda a interação do sistema: o usuário realiza a solicitação do nome do programa na tela do aplicativo, a solicitação é enviada para o banco de dados Firebase (no formato JSON), o Firebase, que está conectado ao programa GO, sincroniza o novo valor do nome do programa e, por fim, o programa GO envia o novo valor para a máquina. A máquina retorna uma nova sequência binária (com o valor do nome do programa já alterado) e os dados são sincronizados novamente com o Firebase e com o aplicativo.

Figura 32 - Diagrama de Sequência.



Fonte: Autor (2018).

A Figura 33 exemplifica como é realizada a leitura de dados do Firebase e sua interpretação na linguagem GO. O método “VerificarModificarAvanço” cria uma nova conexão com o Firebase e realiza a leitura constante dos dados, assim que um evento é disparado (dado modificado), é feita uma validação que identifica se o “eventData” (dado oriundo do evento, que é o dado modificado no Firebase) é 100, 75, 50, 25 ou 0. Dependendo do valor, o método “ModificarAvanço()” é chamado para que o valor da nova velocidade de avanço seja alterada na máquina.

Figura 33 – Modificação da velocidade de avanço através de serviço na linguagem GO.

```
func VerificarModificarAvanço(ip string, port int){
    defer wg.Done()
    f := firego.New("https://controltec-a217b.firebaseio.com/acoes/iniciarProg", nil)

    notifications := make(chan firego.Event)
    if err := f.Watch(notifications); err != nil {
        panic(err)
    }

    defer f.StopWatching()

    for event := range notifications {
        eventData, ok := event.Data.(float64)

        if !ok {
            continue
        }
        if eventData == 100{
            ModificarAvanço(ip, port, "1", "100")
        }
        if eventData == 75{
            ModificarAvanço(ip, port, "1", "75")
        }
    }
}
```

Fonte: Autor (2017).

Além do banco de dados, também foram utilizados pelo menos dois outros serviços do Firebase:

- a) Autenticação de usuário: não foi necessário criar nenhum tipo de “tabela” de usuário ou controle de acesso, o Firebase já possui um serviço de autenticação e funciona perfeitamente no aplicativo;
- b) Sincronização/notificações: sempre que um dado é alterado em um cliente (na máquina ou no aplicativo), o Firebase avisa todos os outros clientes de que o dado foi alterado. Este serviço foi utilizado tanto no aplicativo, para verificar se há alguma alteração da máquina, quanto no programa Go, para verificar se há alguma alteração a

ser realizada na máquina. (modificar velocidade de avanço, por exemplo).

Afim de justificar e testar este tipo de banco de dados, foi realizada uma comparação entre o Firebase e um banco de dados relacional MySQL, que é apresentada na próxima subseção.

4.1.3 Desenvolvimento do Aplicativo

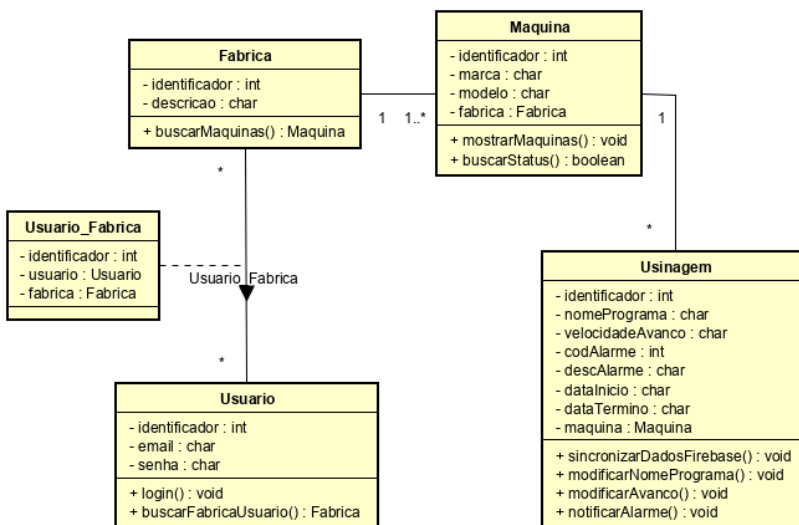
Durante a fase de análise, os seguintes requisitos de sistema foram levantados para o desenvolvimento do aplicativo:

- a) [R1] O aplicativo deve possuir uma tela de *login*, que permite o acesso de usuários previamente cadastrados no sistema;
- b) [R2] A primeira tela que é mostrada ao usuário *logado* precisa ter todas as máquinas cadastradas na fábrica referente ao usuário;
- c) [R3] A lista de máquinas deve indicar o status atualizado de cada máquina: ligada ou desligada;
- d) [R4] Ao clicar em uma máquina, aparecem informações de usinagem em tempo real, alarmes que foram disparados naquela máquina e seu histórico de usinagem;
- e) [R5] Deve ser mostrado o nome do programa que está sendo usinado;
- f) [R6] Deve ser mostrado o nome da ferramenta que está sendo utilizada em tempo real;
- g) [R7] O aplicativo deve mostrar o progresso do programa que está sendo usinado;
- h) [R8] Deve ser possível alterar a velocidade de avanço para 25, 50, 75 e 100%;
- i) [R9] O aplicativo deve mostrar um histórico com nome, data e horário de todos os alarmes disparados na máquina;
- j) [R10] O aplicativo deve possuir um histórico com todos os dados de usinagem que foram coletados em tempo real.

A partir dos requisitos, foi idealizado o diagrama de classes da Figura 34. O aplicativo possui 4 classes principais: Fabrica, Maquina, Usuario e Usinagem. A classe Usuario possui os atributos relacionados aos usuários que poderão utilizar o sistema e também os métodos que são necessários para realizar a ação de *login*. Um usuário pode se relacionar com várias fábricas (ser cadastrados em mais de uma fábrica) e uma

fábrica também pode possuir mais de um usuário, por isso há uma classe de associação entre as classes Usuario e Fabrica. Uma fábrica pode possuir diversas máquinas, porém uma máquina só pode ser cadastrada para uma fábrica, e as duas classes Fabrica e Maquina, respectivamente, possuem os atributos e métodos relacionados as fábricas e máquinas. A classe Usinagem tem como atributos todos os valores que são salvos durante cada nova peça usinada e métodos como “sincronizarDadosFirebase()” para que estes dados sempre sejam sincronizados com o banco de dados na nuvem. Uma máquina pode ter diversas usinagens atreladas a ela, porém um registro de usinagem pertence a apenas uma máquina.

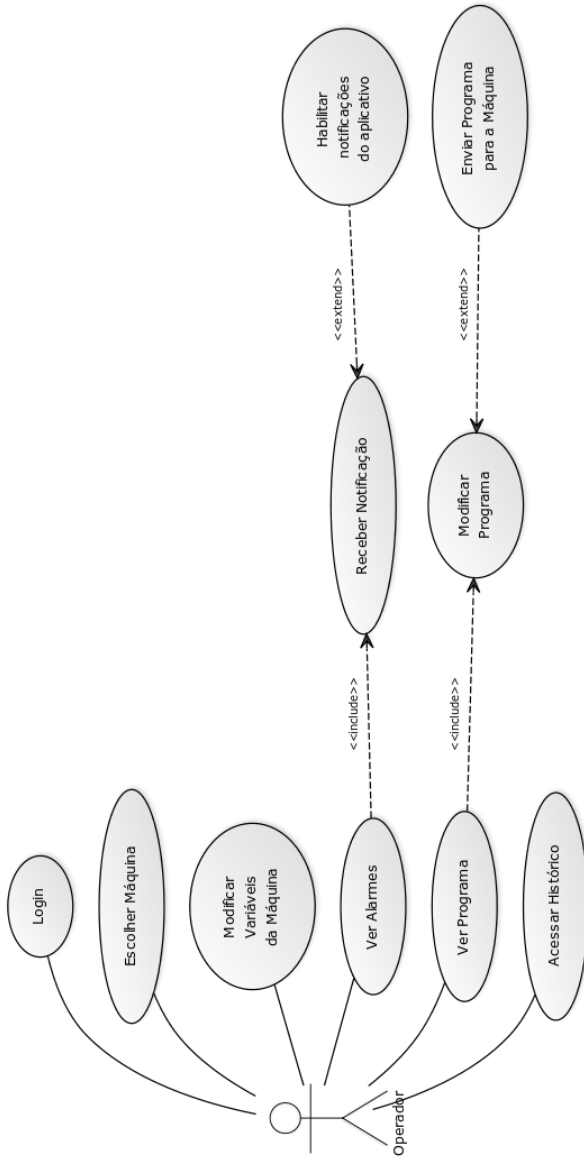
Figura 34 - Diagrama de classes do aplicativo.



Fonte: Autor (2018).

A partir do diagrama de classe, também foi construído o caso de uso da Figura 35, onde o operador do sistema pode realizar o Login, escolher uma máquina, modificar variáveis, ver e receber notificações de alarmes, ver e modificar programas sendo usinados e acessar histórico.

Figura 35 - Caso de Uso.



Fonte: Autor (2018).

Primeiramente, foi desenvolvido um protótipo de tela que abrange todos os requisitos analisados para o aplicativo. Através do protótipo foi possível ter uma visão futura do aplicativo, auxiliando no desenvolvimento das próximas atividades do projeto.

A Figura 36 apresenta o protótipo da tela de *login*, lista de máquinas e status de cada máquina. A tela de *login* possui um campo para inserir o e-mail e outro para a senha do usuário cadastrado. Ao clicar no botão *login*, a fábrica relacionada ao usuário é apresentada, assim como uma lista de máquinas existentes na fábrica. O ícone verde indica que a máquina está ligada, enquanto o cinza indica que está desligada e, ao clicar em “detalhes” na lista de máquinas, o usuário é redirecionado para a página de status de máquina. No status de máquina há um menu com opções de status, alarmes e histórico. Na opção de status são apresentados os dados da máquina em “tempo real” e os botões de alteração da velocidade de avanço.

Figura 36 - A - Tela de *Login* | B - Lista de máquinas | C - Status da máquina selecionada.



Fonte: Autor (2017).

A Figura 37 apresenta o protótipo da tela de alarmes e históricos de usinagens realizadas pela máquina. Na tela de alarme são visualizados os alarmes que já foram disparados pela máquina selecionada. Na tela de histórico são apresentadas as usinagens finalizadas.

Figura 37 – A - Tela de alarmes | B - Lista de histórico de usinagem.

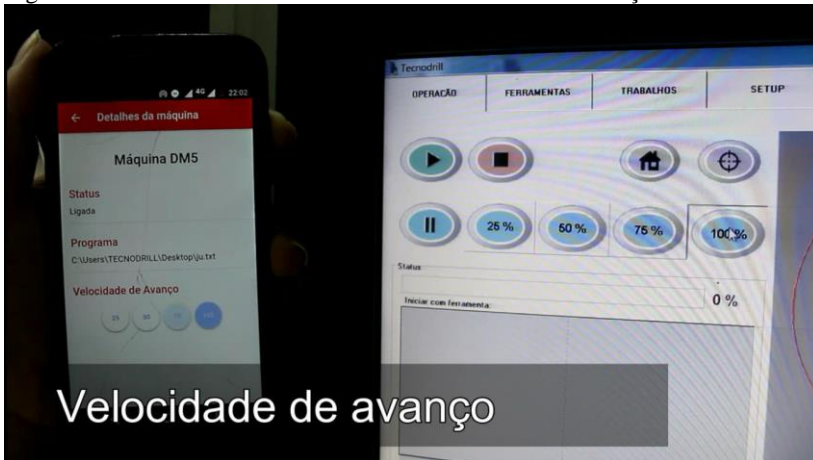
Voltar			Voltar		
Status	Alarmes	Histórico	Status	Alarmes	Histórico
Última atualização: 19/04/2017 - 15:22			Última atualização: 19/04/2017 - 15:22		
Baixa pressão de ar comprimido 15/04/2017 - 13:02			codigonc.txt Início: 19/04/2017 - 12:32		Término: em execução
Porta aberta 14/04/2017 - 12:45			exemplonc1.txt Início: 14/04/2017 - 10:30		Término: 14/04/2017 - 18:00
Fim de Curso Negativo do Eixo 5 13/04/2017 - 09:23			exemplonc2.txt Início: 13/04/2017 - 18:40		Término: 13/04/2017 - 20:00
Alarme4			exemplonc3.txt Início: 13/04/2017 - 12:00		Término: 13/04/2017 - 18:32
Alarme5			exemplonc4.txt		
Alarme6			exemplonc5.txt		
Alarme7			exemplonc6.txt		
Alarme8			exemplonc7.txt		

Fonte: Autor (2017).

A criação do protótipo facilitou a visualização de todos os dados que o sistema irá trabalhar, auxiliando no levantamento dos dados que deveriam ser salvos na nuvem.

Com o aplicativo desenvolvido, foram realizados testes na máquina e gravado um vídeo. A Figura 38 é um *screenshot* do vídeo gravado e mostra o monitoramento da velocidade de avanço sendo realizado. Quando a velocidade de avanço é alterada na máquina, o aplicativo automaticamente modifica a velocidade que está sendo mostrada na interface.

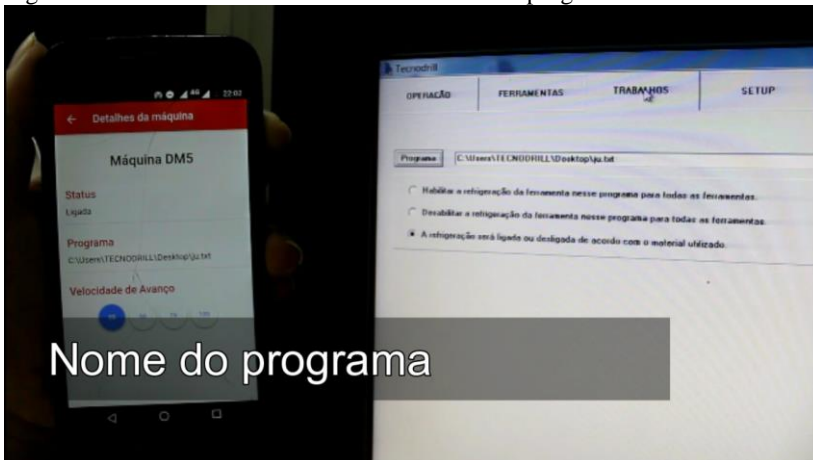
Figura 38 – Monitoramento e Controle da velocidade de avanço.



Fonte: Autor (2018).

A Figura 39 mostra o monitoramento e controle do nome do programa. O aplicativo está em uma rede separada da máquina (4G), porém quando alterado o caminho do programa na máquina, o valor do programa mostrado na interface do aplicativo também é alterado.

Figura 39 - Monitoramento e Controle do nome do programa.



Fonte: Autor (2018).

A Figura 40 mostra um outro *screenshot* do vídeo gravado, onde o usuário altera a velocidade de avanço na interface do celular (conectada através de 4G ao banco de dados Firebase) e a máquina modifica a velocidade de avanço de deslocamento da ferramenta acoplada na máquina.

Figura 40 - Alteração da velocidade de avanço.



Fonte: Autor (2018).

Durante o desenvolvimento, foram produzidas algumas versões de teste do aplicativo e a interface foi alterada. A Figura 41 mostra as principais telas da última versão do aplicativo.

Figura 41 - Telas do Aplicativo.



Fonte: Autor (2018).

Com o aplicativo pronto, conectado a nuvem e a máquina, foram desenvolvidos testes para determinar o tempo de resposta para o envio de dados do servidor.

4.2 AVALIAÇÃO DE DESEMPENHO DO SISTEMA

Um sistema que interaja com uma máquina CNC precisa ser operado no menor tempo possível. Neste trabalho foram utilizadas novas tecnologias desenvolvidas para diminuir o tempo e melhorar a comunicação entre sistemas.

Estas tecnologias já foram comprovadas pelo mercado e são utilizadas por grandes empresas, como a própria Google, porém para comprovar cientificamente que são boas escolhas para este projeto, foi realizada uma avaliação de desempenho comparando o tempo de inserção de dados entre o Firebase (escolhido para o projeto) e um banco de dados MySQL.

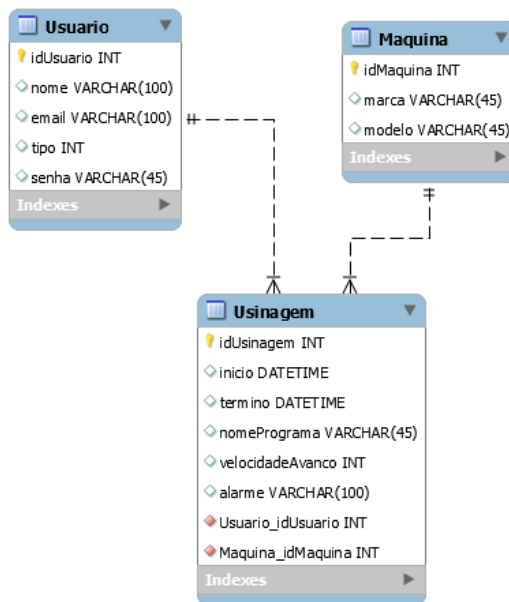
O MySQL foi escolhido por ser o segundo banco de dados mais utilizado no mundo e o primeiro se considerado apenas bancos de dados de utilização livre. (Austrian IT Consulting, 2018).

Os testes realizados simulam requisições ao sistema e tentam estressar o servidor. No cotidiano de uma empresa, uma requisição pode ser considerada qualquer interação que o aplicativo do operador possa realizar com a nuvem – uma busca de máquinas ligadas, por exemplo. O estresse do sistema ocorre quando várias requisições são realizadas ao mesmo tempo, que pode ocorrer quando diversos operadores estiverem conectados na mesma fábrica e, portanto, acessando o mesmo servidor.

- Preparação do ambiente

No banco de dados relacional MySQL, foi criada uma tabela chamada Usinagem, com os campos idUsinagem, inicio, termino, nomePrograma, velocidadeAvanco e relacionamentos com uma tabela de usuários e outra de máquinas, o esquema pode ser visto na Figura 42.

Figura 42 - Banco de dados MySQL.



Fonte: Autor (2018).

Para que o teste fosse realizado em um ambiente coerente, o script do banco de dados proposto foi importado no mesmo servidor Google Cloud utilizado pelo Firebase (mesma conta de usuário e configurações). Para inserir os dados no banco, foi criada uma nova rotina utilizando a linguagem GO, mostrada no APÊNDICE A. No algoritmo, é criada uma nova conexão com o banco de dados e são executados scripts de inserção de uma nova máquina e de uma nova usinagem.

No banco de dados não relacional do Firebase, foi utilizada a lógica de nós como na Figura 31, para acrescentar uma usinagem a uma máquina.

Para a criação do nó “Maquina” foi utilizado o algoritmo mostrado no APÊNDICE B, que cria uma nova conexão com o Firebase, gera um identificador para a nova máquina a ser cadastrada e insere os dados de marca e modelo da máquina. Depois disso, é inserida uma nova “Usinagem” para a máquina recém-criada. A nova usinagem foi criada com valores de teste, que também são utilizados na inserção de valores no banco de dados MySQL.

- Execução dos Programas

A partir dos algoritmos, foram criados dois programas executáveis que rodaram no mesmo servidor local e foram executados sob as mesmas características de *hardware*, descritas no capítulo anterior.

Cada programa, conforme observado na Figura 43, possui 3 variações. A primeira realiza uma chamada para a função de inserção de dados. A segunda realiza três e a última doze chamadas para a função `enviarDados()`, que possui os algoritmos de inserção de dados. A fim de possuir dados amostrais, cada um dos programas foi executado 3 vezes. Um exemplo da segunda inserção pode ser visto na Figura 44.

Figura 43 - Programas executáveis.

Nome	Data de modificaç...	Tipo	Tamanho
firebase-1.exe	21/11/2018 09:28	Aplicativo	5.588 KB
firebase-3.exe	21/11/2018 09:22	Aplicativo	5.588 KB
firebase-12.exe	21/11/2018 09:33	Aplicativo	5.588 KB
mysql.exe	20/11/2018 10:15	Aplicativo	4.257 KB
mysql-1.exe	21/11/2018 09:29	Aplicativo	4.260 KB
mysql-3.exe	21/11/2018 09:21	Aplicativo	4.260 KB
mysql-12.exe	21/11/2018 09:33	Aplicativo	4.260 KB

Fonte: Autor (2018).

Figura 44 - Exemplo de chamada para inserção de dados.

```
func main() {

    start := time.Now()

    enviarValor()
    enviarValor()
    enviarValor()

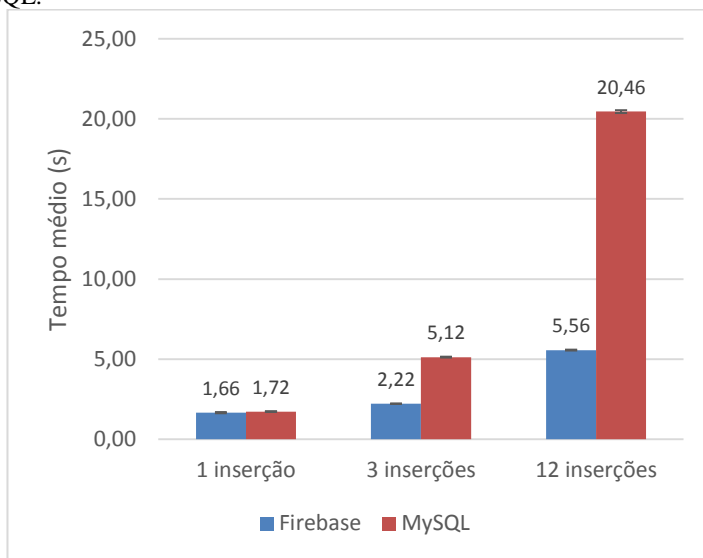
    elapsed := time.Since(start)
    fmt.Printf("Tempo para execução do firebase %s", elapsed)
}
```

Fonte: Autor (2018).

- Medição e Análise

Os valores médios de tempo das execuções dos programas são apresentados na Figura 45.

Figura 45 - Comparação de tempos para inserção de dados entre Firebase e MySQL.



Fonte: Autor (2018).

Conforme Figura 45, nas três execuções do primeiro programa, a média calculada para o Firebase foi de 1,66 segundos e o desvio padrão foi calculado em 0,039 s, versus média de 1,72 s e desvio padrão de 0,0058 s para o MySQL. Nas três execuções do segundo programa, a média do Firebase ficou 2,22 s e desvio padrão de 0,0144 s, enquanto no MySQL a média ficou 5,12 s e o desvio padrão 0,0273 s. Para o terceiro programa, após as três execuções foi encontrada uma média de 5,56 s e um desvio de 0,0345 s para o Firebase, contra média de 20,46 s e desvio padrão de 0,0855 s para o MySQL. Os baixos desvios padrões mostram homogeneidade das amostras.

Nas execuções do primeiro programa, chamando apenas uma vez a função de enviar dados, os valores de tempo entre os dois bancos ficaram aproximados. Sugere-se que isso ocorre devido ao tempo inicial da requisição HTTP, mostrando que em um ambiente com menor fluxo de dados talvez não haveria a necessidade de se escolher um banco de dados mais moderno.

No segundo, com 3 chamadas a função, o Firebase utilizou o tempo médio de 2,2 segundos, enquanto o MySQL utilizou 5,1 segundos. Provando que, considerando um cenário semelhante, o Firebase se mostra

mais rápido em ambientes concorrentes, onde o mesmo recurso precisa ser utilizado várias vezes.

O último teste mostrou uma diferença ainda maior entre o Firebase e o MySQL: média de 5,56 segundos para o Firebase contra 20,46 segundos para o MySQL.

Em um ambiente que exige tomada de decisão rápida, uma espera de 20 segundos, por exemplo, pode penalizar a usinagem de uma peça. Os testes mostraram que um banco de dados como o do Firebase pode ser uma alternativa aos já clássicos e utilizados, quando o objetivo é o menor tempo possível mesmo em ambientes com altas demandas de processamento de dados.

5 CONCLUSÕES

Este trabalho utiliza conhecimentos multidisciplinares entre as Engenharias de Computação e Mecânica. Como abordados na revisão de literatura e nos resultados, será cada vez mais necessário que um profissional entenda de ambas as áreas para desenvolver soluções inteligentes da Indústria 4.0.

Foi desenvolvido um sistema para monitoramento e controle de uma máquina CNC através de um dispositivo móvel. Através do protocolo TCP/IP e da linguagem de programação GO, foi possível acessar as variáveis da máquina, como rotação do eixo árvore, nome do programa usinado e velocidade de avanço afim de sincroniza-las com um banco de dados na nuvem. Para tanto, foi também criado um banco de dados NoSQL na plataforma Firebase, que se conecta ao programa GO, criado previamente, e a um aplicativo móvel. O aplicativo foi criado utilizando o *framework* de desenvolvimento híbrido IONIC, que possibilita o emprego de tecnologias WEB como HTML, CSS e Javascript para geração do aplicativo, sem a necessidade de conhecimentos específicos nas linguagens de desenvolvimento de cada plataforma, como Android e IOS. Foi desenvolvida uma análise de desempenho do banco de dados do Firebase comparado a um banco de dados SQL comumente utilizado, o MySQL.

O programa GO troca dados com a máquina através de uma sequência binária e é capaz de traduzir as mensagens para sincronizar esses dados com o banco na nuvem. No Firebase, além do serviço de banco de dados, também é utilizado o serviço de autenticação para garantir a segurança das conexões com a nuvem e o serviço de notificação, que todos os clientes conectados quando um dado é alterado: se um dado for alterado na máquina, por exemplo, os aplicativos receberão uma notificação de alteração daquele dado. O aplicativo possibilita o controle e a visualização dos dados sincronizados com o banco do Firebase.

As principais conclusões deste trabalho são:

- Na comparação de desempenho entre o banco de dados Firebase e o MySQL sugere-se que em um grande volume de dados, os bancos de dados NoSQL como o do Firebase possuem melhor desempenho. Em um dos cenários, o MySQL chegou a ser 4 vezes mais lento.

- Este projeto reforçou que é possível conectar uma máquina a uma rede com o intuito de monitorar os dados de usinagem e criar um sistema que envie esses dados a um banco de dados e realize um histórico com as usinagens de cada máquina da fábrica.
- A aplicação deste projeto em um ambiente industrial poderá auxiliar: i) o operador da máquina CNC; ii) gestor de fábrica; iii) operadores de manutenção; iv) desenvolvedores e fabricantes da máquina CNC.

Para futuros trabalhos, sugere-se ampliar o número de variáveis a serem disponibilizadas remotamente; analisar e detalhar o tempo de resposta de uma requisição completa do sistema (da máquina para o aplicativo e vice-versa); estudar e desenvolver sistemas de segurança para aplicação destas tecnologias em ambiente industrial; desenvolver esta aplicação para outros processos de manufatura; processar os dados salvos em nuvem, aplicando técnicas de inteligência artificial para analisar e prever possíveis cenários de usinagem.

REFERÊNCIAS

- ARM, J.; BRADAC, Z.; FIEDLER, P. Fault Tolerant CNC Motion Controller. **IFAC-PapersOnLine**, v. 49, n. 25, p. 229–234, 1 jan. 2016.
- BAGHERI, B. et al. Cyber-physical Systems Architecture for Self-Aware Machines in Industry 4.0 Environment. **IFAC-PapersOnLine**, v. 48, n. 3, p. 1622–1627, 2015.
- BAI, Y. Industrial Internet of things over tactile Internet in the context of intelligent manufacturing. **Cluster Computing**, 2017.
- BOMAN, J.; TAYLOR, J.; NGU, A. **Flexible IoT Middleware for Integration of Things and Applications**. Proceedings of the 10th IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing.
- CAO, H.; ZHANG, X.; CHEN, X. International Journal of Machine Tools & Manufacture. v. 112, n. March 2016, p. 21–52, 2017.
- CLEMENTE, F. J. G.; ESQUEMBRE, F.; WEE, L. K. Deployment of physics simulation apps using Easy JavaScript Simulations. **Global Engineering Education Conference (EDUCON), 2017 IEEE**, n. April, p. 1093–1096, 2017.
- CNC Mobile – Apps para Android no Google Play**. Disponível em: <https://play.google.com/store/apps/details?id=demo.CncMobile&hl=pt_BR>. Acesso em: 22 mar. 2017.
- CNI. **Indústria 4.0: novo desafio para a indústria brasileira**. Disponível em: <http://www.portaldaindustria.com.br/relacoesdotrabalho/media/publicacao/chamadas/SondEspecial_Industria4.0_Abril2016.pdf>. Acesso em: 9 mar. 2017.
- COLOGNI, A. L. et al. Smartphone based video-telemetry logger for remote maintenance services. **IFAC-PapersOnLine**, v. 48, n. 3, p. 822–827, 2015.
- DARAMAS, A. et al. **HIVE: Home Automation System for Intrusion Detection**. 2016 Fifth ICT International Student Project Conference (ICT-ISPC), 2016.
- DE SANTIAGO-PÉREZ, J. J. et al. DSP algorithm for the extraction of

dynamics parameters in CNC machine tool servomechanisms from an optical incremental encoder. **International Journal of Machine Tools and Manufacture**, v. 48, n. 12–13, p. 1318–1334, 2008.

DMG MORI Brasil - máquinas CNC de torneamento e fresamento. Disponível em: <<http://br.dmgmori.com/>>. Acesso em: 22 mar. 2017.

EDRINGTON, B. et al. Machine monitoring system based on MTConnect technology. **Procedia CIRP**, v. 22, n. 1, p. 92–97, 1 jan. 2014.

Fidia S.p.A.: Home Page. Disponível em: <http://www.fidia.it/index_en.htm#>. Acesso em: 21 mar. 2017.

FUJISHIMA, M. et al. CIRP Journal of Manufacturing Science and Technology Study of sensing technologies for machine tools. **CIRP Journal of Manufacturing Science and Technology**, v. 14, p. 71–75, 2016.

GARCÍA PLAZA, E.; NÚÑEZ LÓPEZ, P. J. Analysis of cutting force signals by wavelet packet transform for surface roughness monitoring in CNC turning. **Mechanical Systems and Signal Processing**, v. 98, p. 634–651, 2018.

GENG, H. **Internet of Things and Data Analytics**. California: Wiley-Blackwell p. 218–220, 2017.

GRUNDSTEIN, S.; FREITAG, M.; SCHOLZ-REITER, B. A new method for autonomous control of complex job shops – Integrating order release, sequencing and capacity control to meet due dates. **Journal of Manufacturing Systems**, v. 42, p. 11–28, 2017.

HOSSAIN, M. S.; MUHAMMAD, G. Cloud-assisted industrial internet of things (iiot)-enabled framework for health monitoring. **Computer Networks**, v. 101, p. 192–202, 2016.

HUYNH, M.; TRUONG, D. Hybrid app a pproach: could it mark the end of native app domination. **Issues in Informing Science and Information Technology Education** v. 14, p. 49–65, 2017.

Industrial Internet Consortium. Disponível em: <<https://www.iiconsortium.org/>>. Acesso em: 24 mar. 2017.

IONIC. Disponível em: <<http://ionicframework.com/>>. Acesso em: 22 jul. 2017.

ITURRALDE, E. C. et al. **Design and development of a registry system for primary vasculitis**. 2016 7th International Conference on Information, Intelligence, Systems & Applications (IISA). **Anais IEEE**, jul. 2016 Disponível em: <<http://ieeexplore.ieee.org/document/7785429/>>. Acesso em: 11 abr. 2017

JINSONG, B. et al. ScienceDirect A Data Driven Model for Predicting Tool Health Condition in High Speed Milling of Titanium Plates Using Real-Time SCADA. **Procedia CIRP**, v. 61, p. 317–322, 2017.

JÚNIOR, F. F.; GOMES DE OLIVEIRA, J. F.; FAGALI DE SOUZA, A. Tecnologias De Supervisão E Monitoramento Em Máquinas Ferramentas Com Cnc De Arquitetura Aberta. **COBEM**, 2005.

KHAITAN, S. K.; MCCALLEY, J. D. PARAGON: an approach for parallelization of power system contingency analysis using Go programming language. **International Transactions on Electrical Energy Systems**, v. 25, n. 11, p. 2909–2920, 1 nov. 2015.

KUMAR, V. S. et al. Smart Autonomous Gardening Rover with Plant Recognition Using Neural Networks. **Procedia Computer Science**, v. 93, n. September, p. 975–981, 2016.

LANGMANN, R.; ROJAS-PEÑA, L. PLCs as Industry 4.0 Components in Laboratory Applications. **International Journal of Online Engineering (iJOE)**, v. 12, n. 07, p. 37, 2016.

LEE, H.; KIM, T. Prototype of IoT enabled smart factory. **ICIC Express Letters, Part B: Applications**, v. 7, n. 4, p. 955–960, 1 abr. 2016.

LI, P.; HU, T.; ZHANG, C. A unified communication framework for intelligent integrated CNC on the shop floor. **Procedia Engineering**, v. 15, p. 840–847, 2011.

LIU, J. et al. **Automatic control system of intelligent building based on WebAccess**. Proceedings of the World Congress on Intelligent Control and Automation (WCICA). **Anais...2008**

Mazak Sulamericana. Disponível em: <<https://www.mazak.com.br/>>. Acesso em: 24 mar. 2017.

MONOSTORI, L. Cyber-physical Production Systems: Roots, Expectations and R&D Challenges. **Procedia CIRP**, v. 17, p. 9–13, 2014.

OLIVEIRA, L. E. S.; ÁLVARES, A. J. **Axiomatic Design Applied to the Development of a System for Monitoring and Teleoperation of a CNC Machine through the Internet**. *Procedia CIRP. Anais...*2016

Plattform Industrie 4.0 - Homepage. Disponível em: <<http://www.plattform-i40.de/I40/Navigation/EN/Home/home.html>>. Acesso em: 22 mar. 2017.

POSADA, J. et al. Visual Computing as a Key Enabling Technology for Industrie 4.0 and Industrial Internet. **IEEE Computer Graphics and Applications**, v. 35, n. 2, p. 26–40, 1 mar. 2015.

QIN, J.; LIU, Y.; GROSVENOR, R. A Categorical Framework of Manufacturing for Industry 4.0 and Beyond. **Procedia CIRP**, v. 52, p. 173–178, 2016.

RADU, A. et al. **Intercloud platform for connecting and managing heterogeneous services with applications for e-health**. 2015 Conference Grid, Cloud & High Performance Computing in Science (ROLCG). **Anais IEEE**, out. 2015 Disponível em: <<http://ieeexplore.ieee.org/document/7367229/>>. Acesso em: 11 abr. 2017.

RÖSCHINGER, M.; KIPOURIDIS, O.; GÜNTNER, W. A. **A service-oriented cloud application for a collaborative tool management system**. ICIMSA 2016 - 2016 3rd International Conference on Industrial Engineering, Management Science and Applications, 2016.

SBROCCO, José Henrique Teixeira de Carvalho; MACEDO, Paulo Cesar de. **Metodologias ágeis: engenharia de software sob medida - 1. ed.**. São Paulo: Érica, 2012.

SCHÜTZER, K. et al. Improvement of surface accuracy and shop floor feed rate smoothing through open CNC monitoring system and cutting simulation. **Procedia CIRP**, v. 1, n. 1, p. 90–95, 1 jan. 2012.

SHARIATZADEH, N. et al. Integration of Digital Factory with Smart Factory Based on Internet of Things. **Procedia CIRP**, v. 50, p. 512–517, 2016.

SILVA, Alberto Manuel Rodrigues da; VIDEIRA, Carlos Alberto Escaleira. **UML, Metodologias e Ferramentas CASE: Linguagem de Modelação**

UML, Metodologias e Ferramentas CASE na Concepção e Desenvolvimento de Software. Lisboa: Centro Atlântico, 2001.

SIPSAS, K. et al. Collaborative Maintenance in flow-line Manufacturing Environments: An Industry 4.0 Approach. **Procedia CIRP**, v. 55, p. 236–241, 2016.

SOUZA, A. F.; COELHO, R. T. Experimental investigation of feed rate limitations on high speed milling aimed at industrial applications. **International Journal of Advanced Manufacturing Technology**, 2007.

SOUZA, A. F.; ULBRICH, C. B. L. **Engenharia Integrada por Computador e Sistemas CAD / CAM / CNC**. 2ª edição ed. São Paulo: Artliber Editora, 2013.

SPARHAM, M. et al. ANFIS modeling to predict the friction forces in CNC guideways and servomotor currents in the feed drive system to be employed in lubrication control system. **Journal of Manufacturing Processes**, v. 28, p. 168–185, 2017.

SUH, S.-H. et al. **Theory and Design of CNC Systems**. [s.l: s.n.].

TAHERKORDI, A.; ELIASSEN, F.; HORN, G. From IoT big data to IoT big services. **Proceedings of the Symposium on Applied Computing - SAC '17**, p. 485–491, 2017.

Tecnodrill Engenharia de Soluções - Fresadora DM5. Disponível em: <www.tecnodrill.com/fresadora-dm5>. Acesso em: 2 nov. 2018.

Thermwood CNC Mobile – Apps para Android no Google Play. Disponível em: <<https://play.google.com/store/apps/details?id=com.thermwood.cncmobile>>. Acesso em: 22 mar. 2017.

THRAMBOULIDIS, K.; CHRISTOULAKIS, F. UML4IoT—A UML-based approach to exploit IoT in cyber-physical manufacturing systems. **Computers in Industry**, 2016.

TOGASHI, N.; KLYUEV, V. A Novel Approach for Web Development: A Schedule Management System using GAE/Go. 2012.

UEDA, Y. Performance competitiveness of a statically compiled language for server-side Web applications. p. 13–22, 2017.

VERMA, P. K. et al. Machine-to-Machine (M2M) communications: A survey. **Journal of Network and Computer Applications**, v. 66, n. C, p. 83–105, maio 2016.

W3C. Disponível em: < <http://www.w3c.br/>>. Acesso em: 24 mar. 2017.

WAN, J. et al. Mobile services for customization manufacturing systems: An example of industry 4.0. **IEEE Access**, v. 4, p. 8977–8986, 2016.

WANG, N. et al. Design of a New Mobile-Optimized Remote Laboratory Application Architecture for M-Learning. **IEEE Transactions on Industrial Electronics**, v. 64, n. 3, p. 2382–2391, mar. 2017.

WANG, S. et al. Implementing Smart Factory of Industrie 4.0: An Outlook. v. 2016, 2016.

Web Services Architecture. Disponível em: <<https://www.w3.org/TR/ws-arch/>>. Acesso em: 3 jul. 2017.

XIA, M. et al. Closed-loop design evolution of engineering system using condition monitoring through internet of things and cloud computing. **Computer Networks**, v. 101, p. 5–18, 2015.

XU, S. . et al. Research on a fault monitoring system in free-form surface CNC machining based on wavelet analysis. **Journal of Materials Processing Technology**, v. 129, n. 1–3, p. 588–591, 11 out. 2002.

APÊNDICE A – Rotina GO para Inserção no MySQL

Figura 46 - Rotina GO para inserção no MySQL.

```
func enviarValor() {
    db, err := sql.Open("mysql", "root:123456@tcp(35.238.112.12)/tecnodrill")

    if err != nil {
        panic(err.Error())
    }

    //INSERE NOVA MÁQUINA
    defer db.Close()

    marca := "Tecnodrill"
    modelo := "DM5"

    insertMaquina, errMaq := db.Query("INSERT INTO Maquina VALUES ( DEFAULT, ?, ? )",marca,modelo)

    if errMaq != nil {
        panic(errMaq.Error())
    }

    //INSERE NOVA USINAGEM PARA A MÁQUINA COM IDENTIFICADOR = 1 -- Apenas para teste
    defer insertMaquina.Close()

    inicio := time.Now()
    termino := time.Now()
    programa := "Exemplo"
    avanco := "100"
    alarme := "teste"
    usuario := "1"
    maquina := "1"

    insert, err := db.Query("INSERT INTO Usinagem VALUES ( DEFAULT, ?, ?, ?, ?, ?, ? )",inicio,termino,programa,avanco,ala
    if err != nil {
        panic(err.Error())
    }
    defer insert.Close()
}
```

Fonte: Autor (2018).

APÊNDICE B - Rotina GO para Inserção de Dados no Firebase

Figura 47 - Rotina GO para inserção de dados no Firebase.

```
func enviarValor() {
    var err error

    //SIMULA A CRIAÇÃO DE UMA NOVA MÁQUINA, PARA SER UTILIZADO NA CRIAÇÃO DA USINAGEM
    //Gera novo identificador único para a máquina
    idMaquina, err := uuid.NewV4()

    s := []string{"https://controltec-3e22c.firebaseio.com/maquinas", idMaquina.String()};
    url := strings.Join(s, "/")

    maquina := Maquina{
        Marca: "Tecnodrill",
        Modelo: "DMS",
    }

    ref := firebase.NewReference(url)

    if err = ref.Write(maquina); err != nil {
        panic(err)
    }

    //Gera novo identificador único para a máquina
    idUsinagem, err := uuid.NewV4()

    s = []string{"https://controltec-3e22c.firebaseio.com/maquinas", idMaquina.String(), "Usinagens", idUsinagem.String()};
    urlUsinagem := strings.Join(s, "/")

    //CRIAÇÃO DA USINAGEM
    usinagem := Usinagem{
        Inicio: time.Now(),
        Termino: time.Now(),
        Programa: "Exemplo",
        Avanco: "100%",
        Alarme: "teste",
        Usuario: "au6kpXrTgIbxaNYKtENKmiyzi13",
    }

    refUsinagem := firebase.NewReference(urlUsinagem)

    if err = refUsinagem.Write(usinagem); err != nil {
        panic(err)
    }
}
```

Fonte: Autor (2018).

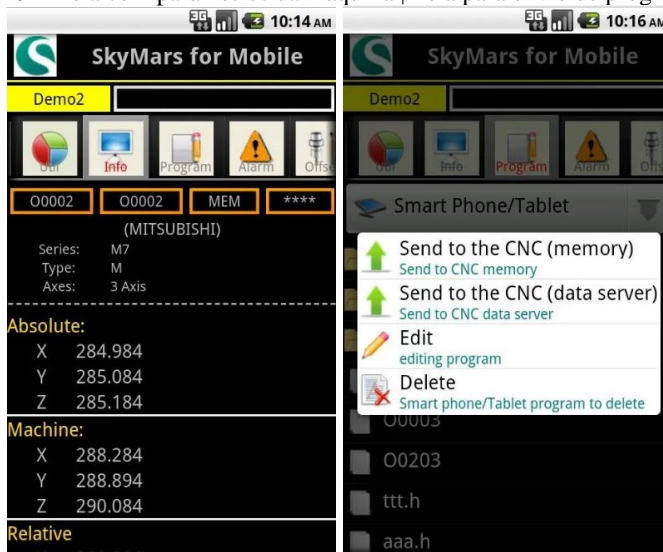
ANEXO A – Telas do Aplicativo CNC Mobile

Figura 48 - Tela com todas as máquinas | Tela com as ferramentas | tela com o gráfico de uso das máquinas.



Fonte: “CNC Mobile – Apps para Android no Google Play” (2017)

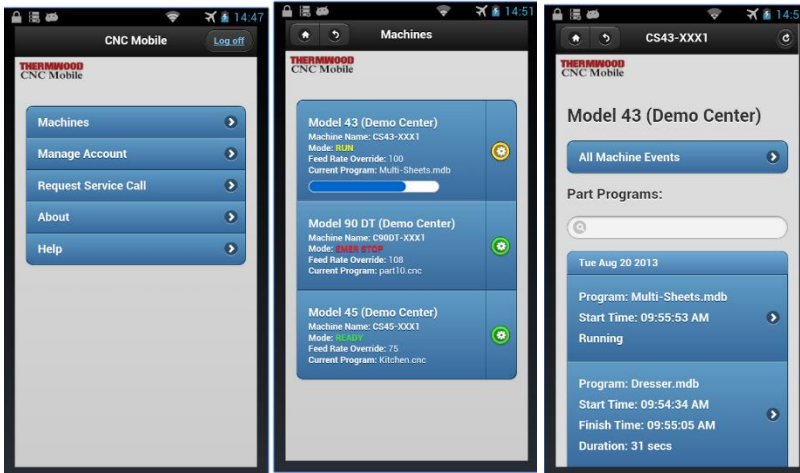
Figura 49 – Tela com parâmetros da máquina | Tela para envio de programa.



Fonte: “CNC Mobile – Apps para Android no Google Play” (2017).

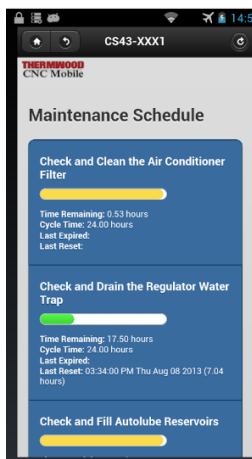
ANEXO B – Telas do Aplicativo Thermwood CNC Mobile

Figura 50 - Tela inicial | Tela com todas as máquinas e status | tela com os eventos da máquina.



Fonte: “Thermwood CNC Mobile – Apps para Android no Google Play” (2017).

Figura 51 - Tela de manutenção, mostra uma barra progressiva para a próxima manutenção da máquina.



Fonte: “Thermwood CNC Mobile – Apps para Android no Google Play” (2017).