



Universidade de Pernambuco
Escola Politécnica de Pernambuco
Programa de Pós-Graduação em Engenharia da Computação

Kayo Henrique de Carvalho Monteiro

Análise de desempenho e disponibilidade de sistemas *e-health* conectados com IoT, *fog* e *cloud*

Dissertação de Mestrado

Recife-PE
Agosto de 2020



Universidade de Pernambuco
Escola Politécnica de Pernambuco
Programa de Pós-Graduação em Engenharia da Computação

Kayo Henrique de Carvalho Monteiro

Análise de desempenho e disponibilidade de sistemas *e-health* conectados com IoT, *fog* e *cloud*

Dissertação de Mestrado

Dissertação apresentada ao Programa de
Pós-Graduação em Engenharia de Computação
da Universidade de Pernambuco como requisito
parcial para obtenção do título de
Mestre em Engenharia de Computação.

Profa. Dra. Patricia Takako Endo
Orientadora

Recife-PE Agosto de 2020

*

*Este trabalho é dedicado a todas as pessoas que se esforçam para tornar sonhos em realidade,
com dedicação e perseverança conseguiremos mudar o mundo.*

Agradecimentos

Agradeço primeiramente a DEUS pelo o dom da vida, pois só foi possível chegar até aqui graças a sua imensa bondade para comigo. Aos meus familiares que durante toda a minha vida acadêmica sempre me deram apoio, e fizeram até o impossível para que esse sonho se realizasse, obrigado e o sonho continua. Um agradecimento especial a minha Mãe por ter me dado todo suporte fazendo dos meus sonhos também seus sonhos. E aos meus amigos que durante essa jornada me apoiaram incondicionalmente e entenderam cada renúncia e ausência (quem um dia ler e for meu amigo sinta-se agradecido)!!!.

Obrigado Comunidade Católica Lumen, por todo acolhimento do corpo e da alma, vocês são o OÁSIS da minha vida. Agradecimento especial aos membros da vida comunitária (Caíke, Lucas Vagner, Felipe, Eduardo e Menandro) por terem me acolhido, Luz da Fé obrigado por perseverarem comigo durante esses longos e turbulentos anos, por fim mas não menos importante OBRIGADO Luís Carlos (Luca) pelo o seu sim, saiba que sua vida é sinal de Deus para muitos e com certeza para mim.

Como agradecer a Ela, que me aguenta a mais de 5 anos, passou de amiga para namorada, namorada para noiva e agora estamos prestes a nos casar. Obrigado PAISHAO (Nirvana) por tanto amor, companheirismo, paciência e compreensão se cheguei até aqui com certeza foi porque você me ajudou muito. Te Amo <3 e obrigado!

CONNECTTA, ah essa empresa <3. Obrigado a todos que fazem parte desse time, obrigado por entenderem e respeitarem minhas ausências vocês são a melhor equipe do mundo e se não for eu cegue! Mikail e Rafael obrigado pela a parceria desde 2011 e se hoje estou conseguindo chegar na reta final de um mestrado devo muito a vocês. Mago, Chocolate, Rhuan obrigado pelo o suporte na caminhada, por escutar meus estresses e por toda a ajuda.

Élisson, obrigado por toda parceria durante as disciplinas e nas ajudas com os códigos tive a "sorte" de me tornar teu amigo durante esses 2 anos de caminhada, obrigado. Arthur Flôr (Tuzinho) e Gerivan, foi massa demais me tornar amigo de vocês e compartilhar todo esse ambiente com vocês, ter amigos durante essa caminhada tornou tudo mais leve.

Obrigado a todos que fazem parte do dotLabBR, aprendi e aprendo muito com todos, juntos somos mais fortes e resistiremos fazendo pesquisa aplicada no interior de Pernambuco. Um agradecimento especial a Guto por todo o suporte durante a dissertação, a Marcelino por me ajudar com a parte gráfica, continuemos porque o trabalho não para o/.

Gratidão eterna a minha orientadora Patricia Takako Endo, que durante minha graduação nos tornamos amigos foi minha orientadora e agora tive a honra de fazer parte das suas primeiras orientações de mestrado. A senhora é FANTÁSTICA, muito muito obrigado por ter aberto as portas desse novo mundo, obrigado por compartilhar seus conhecimentos e por acreditar sempre

em mim.

A Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) -, o presente trabalho foi realizado com apoio da CAPES - Código de Financiamento 001.

*“ A besteira é a base da sabedoria.”
(Falcão - Marcondes Falcão Maia)*

Resumo

Em paralelo ao aumento da média de idade da população mundial, tem-se evidenciado, cada vez mais, a utilização de *smartphones* e dispositivos inteligentes que fazem uso da tecnologia de Internet da Coisas (IoT), dentre eles, destacam-se os dispositivos vestíveis (*wearables*), que são dispositivos que podem ser utilizados como acessórios (relógios e pulseiras, por exemplo). A utilização desses dispositivos traz uma válida oportunidade para prover serviços de saúde, esses sistemas são chamados de sistemas *e-health*; com a advento da IoT é possível fazer o acompanhamento remoto em tempo real de idosos, além de monitoramentos mais complexos como frequência cardíaca, pressão arterial dentre outros. Entretanto os dispositivos IoT apresentam diversas limitações (processamento e armazenamento, por exemplo), essas limitações são pontos críticos para os sistemas *e-health*. Para suprir essas carências e melhorar as respectivas capacidades, podem ser acrescentado duas camadas a essa infraestrutura, a *fog*, na qual tem potencial de trazer o processamento dos dados para mais próximo do usuário; e a *cloud* que agrega armazenamento e processamento "ilimitados". Ainda que o uso dessas camadas auxiliem os sistemas *e-health*, é indispensável garantir também, que tenham bom desempenho e disponibilidade, tendo em vista que qualquer indisponibilidade pode gerar diversos inconvenientes, pois se tratando de sistemas de acompanhamento de pacientes essa indisponibilidade pode gerar alertas e informações inconsistentes. Esta dissertação propõe um protótipo que utiliza como infraestrutura IoT, *fog* e/ou *cloud*, no qual foram realizados experimentos para analisar o desempenho do sistema, além de analisar o custo de implementação e manutenção desse sistema. Os resultados obtidos nos experimentos com o protótipo são utilizados como dados de entrada para os modelos computacionais, também propostos nesta dissertação. Como consequência, é possível estimar a disponibilidade e o desempenho destes sistemas em diferentes combinações da sua infraestrutura. Através dos resultados dos modelos computacionais com os dados reais é possível escolher dentre as configurações analisadas a melhor infraestrutura levando em consideração custo, performance e disponibilidade.

Palavras-chaves: IoT, *fog*, *cloud*, *e-health*, *deep learning*.

Abstract

In parallel with the increase in the average age of the world population, the use of smartphones and smart devices that make use of the Internet of Things (IoT) technology has been evidenced, among them, wearable devices stand out. , which are devices that can be used as accessories (watches and bracelets, for example). The use of these devices provides a valid opportunity to provide health services, these systems are called e-health systems; with the advent of IoT it is possible to do remote monitoring in real time of the elderly, in addition to more complex monitoring such as heart rate, blood pressure, among others. However, IoT devices have several limitations (processing and storage, for example), these limitations are critical points for e-health systems. In order to supply these deficiencies and improve the respective capacities, two layers can be added to this infrastructure, the fog, in which it has the potential to bring the data processing closer to the user; and the cloud that adds "unlimited" storage and processing. Although the use of these layers help e-health systems, it is also essential to ensure that they have good performance and availability, considering that any unavailability can generate several inconveniences, as in the case of patient monitoring systems, this unavailability can generate alerts and inconsistent information. This dissertation proposes a prototype that uses IoT, fog and / or cloud infrastructure, in which experiments were carried out to analyze the system's performance, in addition to analyzing the cost of implementing and maintaining this system. The results obtained in the experiments with the prototype are used as input data for the computational models, also proposed in this dissertation. As a consequence, it is possible to estimate the availability and performance of these systems in different combinations of your infrastructure. Through the results of the computational models with the real data it is possible to choose among the analyzed configurations the best infrastructure taking into account cost, performance and availability.

Key-words: IoT, fog, cloud, e-health, deep learning.

Lista de ilustrações

Figura 1 – Esquema de mapeamento sistemático	16
Figura 2 – Esquema de classificação através de palavras-chaves e resumo	18
Figura 3 – Métodos de avaliação usados para validação da proposta	26
Figura 4 – Sistema IoT integrado	31
Figura 5 – Itens na rede de Petri	33
Figura 6 – Exemplo de rede de Petri antes da transação	33
Figura 7 – Exemplo de rede de Petri depois da transação	34
Figura 8 – Exemplo de modelo RBD	35
Figura 9 – Exemplo de Fila	37
Figura 10 – Proposta: Arquitetura para sistemas de <i>e-health</i>	38
Figura 11 – Configuração: Cenário 1	40
Figura 12 – Estrutura do protótipo	41
Figura 13 – <i>Mockup</i> do aplicativo	42
Figura 14 – Metadado - JSON	43
Figura 15 – Consumo em porcentagem de processamento durante os experimentos	46
Figura 16 – Consumo de memória RAM durante os experimentos	46
Figura 17 – Médias de tempo de serviço durante os experimentos	47
Figura 18 – Modelo RBD do dispositivo IoT	48
Figura 19 – Modelo RBD da <i>fog</i>	49
Figura 20 – Modelo RBD da <i>cloud</i>	49
Figura 21 – Modelo dos experimentos: cenários 1 e 3	50
Figura 22 – Modelo dos experimentos: cenários 2 e 4	52

Lista de tabelas

Tabela 1 – Critérios de inclusão e exclusão para selecionar artigos relevantes	17
Tabela 2 – Classificação dos trabalhos	18
Tabela 3 – Configurações de infraestrutura comumente usadas por aplicações <i>e-health</i> com IoT	20
Tabela 4 – IoT - aplicações e monitoramentos	21
Tabela 5 – <i>Design</i> do experimento	39
Tabela 6 – Tipos de atividades classificáveis	44
Tabela 7 – Resultados dos experimentos, em segundos.	47
Tabela 8 – Resultados do RBD	49
Tabela 9 – Função de guarda de funcionamento do sistema: cenários 1 e 3	50
Tabela 10 – Função de guarda: cenários 1 e 3	51
Tabela 11 – Transições temporizadas em horas: cenários 1 e 3	51
Tabela 12 – Transições temporizadas em horas: cenários 2 e 4	52
Tabela 13 – Função de guarda: cenários 2 e 4	52
Tabela 14 – Função de guarda de funcionamento do sistema: cenários 2 e 4	53
Tabela 15 – Resultados dos modelos SPN	55
Tabela 16 – Valores de dispositivos da <i>fog</i> em dólar	56
Tabela 17 – Valores do serviço de armazenamento (<i>cloud</i>) em dólar	56
Tabela 18 – Produção diretamente relacionada com a dissertação	60
Tabela 19 – Produção como colaborador	60
Tabela 20 – Programa de computador	61
Tabela 21 – Artigos do mapeamento sistemático	71

Lista de abreviaturas e siglas

OMS	Organização Mundial de Saúde
ONU	Organização das Nações Unidas
IoT	<i>Internet of Things</i>
GSMA	<i>Global System for Mobile Communications Association</i>
DL	<i>Deep Learning</i>
QP	<i>Questões de Pesquisa</i>
PDI	Processamento Digital de Imagem
EMG	Eletromiografia
SPN	<i>Stochastic Petri Nets</i>
MTBF	<i>Mean Time Between Failures</i>
MTTF	<i>Mean Time To Failure</i>
MTTR	<i>Mean Time To Repair</i>
RBD	<i>Reliability Block Diagrams</i>
VM	<i>Virtual Machine</i>
SO	<i>Sistema Operacional</i>
CTMC	<i>Continuous Time Markov Chain</i>
NTP	<i>Network Time Protocol</i>
JSON	<i>JavaScript Object Notation</i>
API	<i>Application Programming Interface</i>
SDK	<i>Software Development Kit</i>
TP	<i>Throughput</i>
TS	Tempo de Serviço
PD	Probabilidade de Descarte
RP	Requisições em Processos

VPS *Virtual Private Server*

INPI Instituto Nacional da Propriedade de Industrial

Sumário

1	INTRODUÇÃO	13
1.1	Problemática	14
1.2	Objetivos	15
1.3	Estrutura da dissertação	15
2	REVISÃO SISTEMÁTICA	16
2.1	Mapeamento sistemático	16
2.1.1	Estudos primários	17
2.1.2	Triagem dos artigos	17
2.2	Resultados do mapeamento	18
2.2.1	Infraestrutura	18
2.2.2	Aplicação	20
2.2.3	Tipo de monitoramento	24
2.2.4	Método de avaliação	25
2.3	Desafios encontrados	27
2.4	Considerações	29
3	CONCEITOS BÁSICOS	30
3.1	Definições IoT, <i>fog</i> e <i>cloud</i>	30
3.2	Sistemas <i>e-health</i>	31
3.3	Técnicas de modelagem	32
3.3.1	Redes de Petri	32
3.3.2	<i>Reliability Block Diagrams</i> (RBD)	34
3.4	Modelos computacionais	35
3.4.1	Modelo de performance	36
3.5	Considerações	37
4	PROTOTIPANDO UM SISTEMA <i>E-HEALTH</i>	38
4.1	Arquitetura do sistema <i>e-health</i>	38
4.1.1	<i>Design</i> de experimentos	39
4.2	Prototipação	40
4.2.1	Funcionamento do protótipo	41
4.2.2	Descrição da infraestrutura	45
4.3	Resultados dos experimentos	45
4.4	Considerações	47

5	MODELANDO UM SISTEMA <i>E-HEALTH</i>	48
5.1	Modelos RBD	48
5.2	Modelos SPN	49
5.3	Métricas	53
5.4	Resultados	55
5.4.1	Análise de disponibilidade e desempenho	55
5.4.2	Análise de custo	56
5.5	Considerações finais	57
6	CONCLUSÃO	58
6.1	Contribuições e produções científicas	59
6.2	Limitações e dificuldades	61
	REFERÊNCIAS	62
	APÊNDICES	69
	APÊNDICE A – LISTA DE ARTIGOS DO MAPEAMENTO SISTEMÁTICO	70

1 Introdução

O aumento da população idosa em todo o mundo é uma realidade. De acordo com a Organização Mundial de Saúde (OMS) (ORGANIZATION, 2018), acredita-se que no presente ano de 2020, pela primeira vez na história, o número de pessoas idosas será maior que o de crianças com até cinco anos, com até 80% deles viverão em países de baixa e média renda. A OMS afirma que embora os indivíduos estejam vivendo mais, não significa que encontram-se mais saudáveis que antes. Uma população mais velha é sinônimo de uma população que necessita de mais cuidado e acompanhamento constante, o que desafia os sistemas atuais de saúde.

Durante a assembleia geral na Organização das Nações Unidas (ONU) (NATIONS, 2018), concomitante à afirmação da OMS, o tema de cuidados com os idosos também foi colocado em pauta como um dos desafios globais de saúde. A relevância toma tais proporções porque os idosos têm maior predisposição para apresentar doenças crônicas e sofrer alguma condição de deficiência ou limitações físicas, necessitando de cuidados e monitoramento constantes.

A *Global System for Mobile Communications Association* (GSMA) em seu relatório anual Economia Móvel de 2019 (STRYJAK; SIVAKUMARAN, 2019), apresentou a popularização dos dispositivos móveis, onde em todo o planeta, mais de 5,1 bilhões de pessoas usam algum tipo de *smartphone*. Essa popularização também é vista com a utilização de dispositivos de Internet das Coisas (do inglês *Internet of Things* - IoT), o uso desses dispositivos são uma das principais alternativas acessíveis para o acompanhamento de pacientes, através de ferramentas tecnológicas voltadas ao gerenciamento de cuidados com a saúde (NEAGU et al., 2017).

Com o advento dos dispositivos IoT, dados como os sinais vitais podem ser coletados, analisados e consultados a qualquer momento e a partir de qualquer lugar. Essas informações podem ser enviadas para sistemas de processamento e armazenamento, possibilitando que não somente médicos, mas também toda a equipe multiprofissional e familiares acessem os dados do paciente sempre que desejarem. Esses sistemas que fazem uso de tecnologia como ferramenta principal para prover saúde são chamados de sistemas *e-health*. Eles podem ser compostos por diversas camadas e recursos na sua infraestrutura, como dispositivos e sensores IoT, tecnologias de redes (*edge*, *fog* e *cloud*), técnicas de *machine learning* e *data science*, que de acordo com a finalidade do sistema podem fazer uso de uma infraestrutura mais simples ou complexa (MIOTTO et al., 2017) (IRFAN; AHMAD, 2018).

A composição da estrutura dos sistemas *e-health* possibilita o monitoramento do paciente e caso seja identificada alguma anormalidade, o sistema conseguirá gerar notificações em tempo real sobre o estado do paciente, as quais só são possíveis através do processamento de dados próximo ao usuário, que é garantido pela presença da *fog*, onde por exemplo, poderá ser executado um modelo de *deep learning* (DL) para realizá-lo (MIOTTO et al., 2017).

Com o aumento da demanda por cuidados nestes pacientes, visto que os mesmos podem não ter amplos recursos financeiros para custear essa assistência médica contínua, é necessário propor mecanismos/tecnologias que sejam de baixo custo. Os *smartphones* e os equipamentos *wearables* como *smartwatch* dentre outros dispositivos IoT, são possibilidades plausíveis para esses pacientes, pois são de uso em massa da população mesmo em diferentes classes socioeconômicas (STRYJAK; SIVAKUMARAN, 2019), e tem o seu valor de compra menor que os dispositivos médicos especializados.

Os dispositivos IoT são possíveis coletarem diferentes sinais vitais, tais como: frequência cardíaca, temperatura corporal, pressão arterial, nível de oxigenação no sangue, além do acompanhamento durante as atividades diárias e identificação de possíveis acidentes domésticos, quedas ou até mesmo ataque cardíaco. Porém, os dispositivos IoT possuem algumas limitações, e desenvolver tecnologias que suavizem essas limitações de capacidade computacional e de armazenamento é necessário para uma maior disseminação dos sistemas *e-health*.

Uma forma de suprir as limitações é construir sistemas *e-health*, em que os dispositivos IoT terão somente a função de coletar os dados através dos seus sensores e enviar para *fog* ou *cloud*, que ficarão responsáveis por todo o processamento e armazenamento dos dados. Propor um sistema com alto desempenho e alta disponibilidade são pontos primordiais para essas aplicações.

1.1 Problemática

Os sistemas *e-health*, compostos por dispositivos IoT que fazem o monitoramento de pacientes tendem a sofrer com problemas de capacidade computacional devido ao alto fluxo de dados coletados (KOTRONIS et al., 2017). Para suprir esta limitação, a IoT possui dois grandes aliados: *fog* e *cloud*. No entanto, de encontro aos benefícios trazidos por essa infraestrutura, a utilização dessa composição aumenta a complexidade dos sistemas.

Considerando o cotidiano das instituições e pessoas que precisam dessas ferramentas, a assistência médica-familiar e a disponibilidade de sistemas *e-health* são fundamentais para garantir a detecção e a integridade das informações vitais coletadas do paciente (VARGHEESE; VINIOTIS, 2014), (IRFAN; AHMAD, 2018). Para melhorar a disponibilidade do sistema como um todo, a modelagem computacional pode ser utilizada para prever falhas nos dispositivos IoT, para encontrar gargalos no sistema e sugerir redundância (SANTOS et al., 2018), além de analisar a robustez e a escalabilidade da solução.

O desafio que tange o pleno funcionamento de toda a infraestrutura (IoT, *fog* e *cloud*) suscitou o desenvolvimento deste trabalho, através da proposta de modelos computacionais que sejam capazes de analisar a disponibilidade e o desempenho de sistemas *e-health* visando gerar informações que auxiliem implementações futuras. A construção desses modelos tem um alto grau de complexidade, pois, é preciso identificar as dependências entre todos os componentes do

sistema e compreender como a falha de cada componente afetará o funcionamento do sistema como um todo.

1.2 Objetivos

O objetivo principal deste trabalho é propor e avaliar modelos computacionais para analisar a disponibilidade e o desempenho de um sistema *e-health* que utiliza IoT, *fog* e *cloud* como infraestrutura.

Os objetivos específicos deste trabalho resumem-se em:

1. Apresentar uma revisão sistemática sobre como sistemas *e-health* integrado com IoT, *fog* e *cloud* estão sendo implementados e utilizados na literatura atual;
2. Definir, implementar e avaliar o desempenho de um protótipo de sistema *e-health* integrado com IoT, *fog* e *cloud*;
3. Propor modelos estocásticos para realizar análise de disponibilidade e desempenho de sistema *e-health* integrado com IoT, *fog* e *cloud*;
4. Realizar uma análise de custo de implementação e manutenção do sistema *e-health* integrado com IoT, *fog* e *cloud*.

1.3 Estrutura da dissertação

O presente trabalho está organizado da seguinte maneira: o Capítulo 2 apresenta uma revisão sistemática sobre o tema do projeto; o Capítulo 3 aborda os principais conceitos que foram utilizados para realização do mesmo, assim como, as técnicas de modelagem utilizadas e os modelos computacionais; o Capítulo 4, descreve todo o processo de prototipação, desde a arquitetura do sistema, descrição da infraestrutura até os resultados dos experimentos; o Capítulo 5 apresenta os modelos computacionais, análises e os resultados obtidos; e por fim, o Capítulo 6 expõe as conclusões e considerações finais.

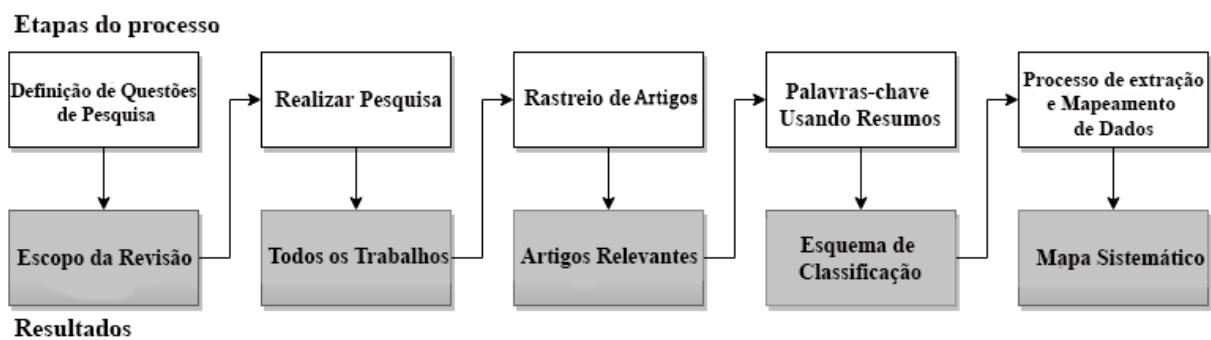
2 Revisão Sistemática

Este capítulo apresenta o mapeamento sistemático e a metodologia utilizada na fase inicial do estudo (Seção 2.1). Além dos resultados 2.2 e desafios encontrados na literatura 2.3 durante essa etapa do estudo.

2.1 Mapeamento sistemático

Este trabalho utilizou a metodologia de mapeamento sistemático apresentada por Petersen et al. (PETERSEN et al., 2008). A Figura 1 mostra as etapas do processo e seus respectivos resultados.

Figura 1 – Esquema de mapeamento sistemático



Fonte: Adaptação de (PETERSEN et al., 2008).

De acordo com Petersen et al. (PETERSEN et al., 2008), o principal objetivo de um mapeamento sistemático é “fornecer uma visão geral de uma determinada área de pesquisa e identificar a quantidade e o tipo de pesquisa disponível nela”. Regularmente, um mapeamento sistemático também pode ser utilizado para observar as frequências de publicação ao longo do tempo para assim identificar tendências. Foram definidas quatro questões de pesquisas (QP), para direcionar as avaliações dos arquivos:

- **QP 1:** Qual é o estado da arte atual em aplicações *e-health* usando a infraestrutura *fog-to-cloud computing*?
- **QP 2:** Quais são as aplicações *e-health* mais comuns que utilizam como infraestrutura *fog-to-cloud computing*?
- **QP 3:** Quais são os métodos mais comuns (simulação, modelagem ou protótipo) usados para avaliar as aplicações *e-health*?
- **QP 4:** Quais são os principais desafios para implementação de aplicações de *e-health* que dependem da infraestrutura *fog-to-cloud computing*?

2.1.1 Estudos primários

De acordo com Petersen et al. (PETERSEN et al., 2008), os estudos primários *"são identificados usando cadeias de pesquisa em bancos de dados científicos ou navegando manualmente através de anais de conferências ou publicações de periódicos relevantes"*.

A identificação dos estudos primários foi realizada através de uma busca junto as bases de dados relevantes, para isso foi utilizada a *string* de busca: *("e – health"OR"ehealth"OR"healthcare")AND("cloud"OR"fog")AND("Internetofthings"OR"IoT")* ; as bases de artigos científico foram: IEEE ¹, ACM ², e Springer ³. O estudo resultou em 83 artigos do IEEE, 1 do ACM e 81 do Springer.

2.1.2 Triagem dos artigos

Alguns dos artigos que resultaram da busca inicial podem não estar estritamente relacionados as perguntas de pesquisa ou não as respondem, sendo criado critérios de inclusão ou exclusão dos mesmos, conforme descrito abaixo na Tabela 1.

Tabela 1 – Critérios de inclusão e exclusão para selecionar artigos relevantes

Inclusão	O resumo menciona explicitamente aplicações <i>e-health</i> no contexto IoT, <i>fog</i> ou <i>cloud</i> . Foram considerados artigos de conferências, periódicos e capítulos de livros.
Exclusão	O artigo está fora do contexto da infraestrutura IoT, <i>fog</i> ou <i>cloud</i> . É uma aplicação <i>e-health</i> , porém, não fazem uso da infraestrutura e, portanto, não são considerados em nosso mapeamento sistemático.

Como IoT, *fog* e *cloud* são termos muito abrangentes, eles geralmente são usados em resumos, mas não significa que o artigo seja relacionado ao tema. Assim, a aplicação dos critérios de inclusão e exclusão é feita manualmente e, após a triagem, mantiveram-se 38 trabalhos do IEEE e 12 do Springer (ver Apêndice A).

Nesta etapa, realizou-se uma pequena adaptação proposta por Petersen et al. (PETERSEN et al., 2008). Originalmente, os autores usam palavras-chaves para *" [...] reduzir o tempo necessário no desenvolvimento do esquema de classificação e garantir que o esquema leve em consideração os estudos existentes "* onde é realizada a leitura do resumo e as palavras-chaves dos trabalhos selecionados. Se o resumo contém pouca informação, é levado em consideração a introdução e a conclusão do artigo.

Neste mapeamento sistemático, não foi considerado os elementos estruturais do texto separadamente, como apenas resumo, palavras-chaves ou introdução. Para a classificação de

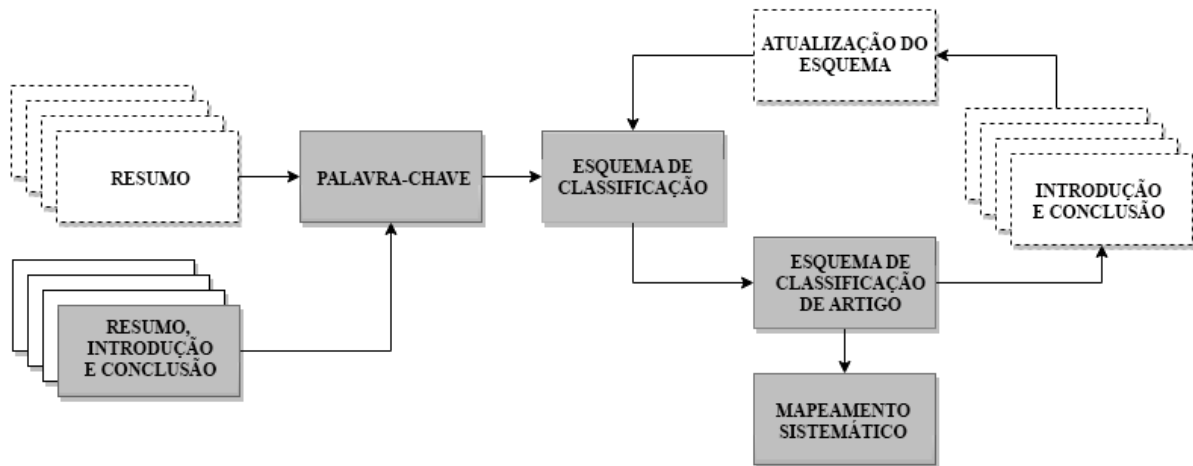
¹ <https://ieeexplore.ieee.org/Xplore/home.jsp>

² <https://dl.acm.org/>

³ <https://www.springer.com/gp>

inclusão foram interpretados todo o conjunto citado, conforme demonstrado na Figura 2, onde apresenta a etapa de palavras-chaves (quadrados cinza) adaptada de Petersen et al. (PETERSEN et al., 2008) (os quadrados de linhas pontilhadas não estão sendo usados neste trabalho).

Figura 2 – Esquema de classificação através de palavras-chaves e resumo



Fonte: Adaptação de (PETERSEN et al., 2008).

Após a elaboração do esquema de classificação, os artigos relevantes são classificados e ocorre a extração de dados. Os resultados do mapeamento sistemático são apresentados na próxima seção.

2.2 Resultados do mapeamento

Com base na metodologia de mapeamento sistemático, esta seção apresenta os trabalhos classificados após a extração de dados e a Tabela 2 segmenta os trabalhos em: Infraestrutura, Aplicação, Tipo de monitoramento e Avaliação.

Tabela 2 – Classificação dos trabalhos

Categoria	Descrição
Infraestrutura	Descreve os tipos de cenário que a aplicação IoT é implantada, bem como a infraestrutura utilizada (combinação de IoT, <i>fog</i> e <i>cloud</i>).
Aplicação	Relata as aplicações IoT encontradas.
Tipo de monitoramento	Apresenta o que está sendo monitorado e qual mecanismo é utilizado pela aplicação IoT.
Avaliação	Relata os tipos de avaliação utilizados nos trabalhos.

Fonte: Autor (2020).

2.2.1 Infraestrutura

Habitualmente, os aplicativos de IoT geram uma grande quantidade de dados que devem ser processados e armazenados para vários propósitos em sistemas *e-health*, onde destacam-

se: auxiliar o médico em diagnósticos, alertar a família de pacientes em casos de emergência, identificar padrões de pacientes com determinada doença, facilitar os estudos de coorte para novos medicamentos, aumentar a capacidade do sistema de gerar prontuários concisos de pacientes, acompanhar a adesão e evolução de tratamentos de doenças crônicas.

No entanto, algumas vezes, os aplicativos não são desenvolvidos para lidar com a grande massa de dados gerada (*big data*), além de requerem infraestruturas com alto desempenho e disponibilidade. Foram encontrados artigos que apresentam sistemas *e-health* que integram dispositivos IoT a infraestruturas de *fog* e *cloud*. Os trabalhos foram classificados de acordo com o contexto em que o sistema de *e-health* é usado: uso pessoal, hospitalar, casas inteligentes e de uso genérico (nenhum contexto específico definido no trabalho), como apresentado na Tabela 3.

Como destacado anteriormente, existem muitas restrições inerentes aos dispositivos de IoT, como recursos limitados de armazenamento, fonte de alimentação e processamento, que influenciam negativamente o desempenho das redes (GHANAVATI et al., 2017) (CHARALAMPIDIS et al., 2017), tornando-os geralmente dependentes da infraestrutura em *cloud*. Esse fato se reflete nos trabalhos encontrados na literatura, onde a configuração predominante da infraestrutura é a IoT e *cloud* (75,51% dos trabalhos). Quando considera-se apenas o contexto dos hospitais, a infraestrutura integrada de IoT e *cloud* é utilizada por todas as soluções encontradas.

Outra observação relevante dos resultados é sobre os estudos recentes da infraestrutura integrada IoT, *fog* e *cloud*. Dos 50 artigos analisados, 56% são datados de 2017 ou 2018, no entanto, ao analisar apenas os artigos que utilizam a infraestrutura IoT, *fog* e *cloud* esse número aumenta para aproximadamente 83%. Com base nessas informações, podemos reafirmar que a computação em *fog* está desempenhando seu papel como a tecnologia que “acopla” a IoT e os recursos da *cloud*, especialmente quando os aplicativos da IoT são o foco da aplicação, fornecendo uma ação rápida quando necessário (como notificação por *push* para queda, anormalidade de pulso, vazamento de gás, dentre outros...) (GUIBERT et al., 2017). De acordo com Corotinschi e Găitan (COROTINSCHI; GĂITAN, 2017), “o processamento local é obrigatório para reduzir o tráfego de dados da rede [...]. Por esse motivo, é realizada a tentativa de executar o processamento o mais próximo possível do dispositivo que está gerando os dados”.

De acordo com Yao et al. (YAO et al., 2018), “[...] o aumento da tecnologia IoT impulsiona a inteligência ambiental, expandindo significativamente a escala e o escopo do domínio da saúde [...]”, criando soluções inovadoras e não invasivas de monitoramento de saúde.

Diante do cenário encontrado na revisão sistemática, este trabalho utilizou como infraestrutura para a aplicação *e-health*: IoT, *fog* e *cloud*, que além de ser uma tendência na academia, os dados do mapeamento sistemático também atenderão a qualquer necessidade de processamento próximo ao dispositivo IoT, como apresentado em (MIOTTO et al., 2017), (COROTINSCHI; GĂITAN, 2017).

Tabela 3 – Configurações de infraestrutura comumente usadas por aplicações *e-health* com IoT

Infraestrutura	Contexto	Artigos
IoT e <i>Cloud</i>	Uso Pessoal	(RADWAN et al., 2017) (FAZIO et al., 2015) (DHANALIYA; DEVANI, 2016) (GERDES et al., 2015) (PIYARE et al., 2013) (RADU et al., 2015) (RENNER et al., 2014) (DANG et al., 2018) (JI et al., 2017) (PESCOSOLIDO et al., 2016) (MATSUI; CHOI, 2017)
		(KYRIAZAKOS et al., 2017) (NEAGU et al., 2017) (CHEN et al., 2018) (JOSHI et al., 2016) (HA; LINDH, 2017) (KO et al., 2015) (BAGULA et al., 2016) (BAO et al., 2016) (HASSAN, 2015) (GHOSH et al., 2016) (PEREIRA et al., 2017)
		(GHANAVATI et al., 2017) (PLAGERAS et al., 2017) (KAROUI; FTIMA, 2018) (YAMIN, 2018) (SINGH; BACON, 2014)
		(LI et al., 2018) (STRADOLINI et al., 2018b) (KOTRONIS et al., 2017) (ULLAH et al., 2016) (HUAXIN et al., 2012) (CABRA et al., 2017) (VARGHEESE; VINIOTIS, 2014) (AHMADI et al., 2018) (BHATIA; SOOD, 2018)
IoT, <i>fog</i> e <i>cloud</i>	Uso Pessoal	(ALI; GHAZAL, 2017) (COROTINSCHI; GĂITAN, 2017) (GUIBERT et al., 2017) (MASIP-BRUIN et al., 2016)
	Casas Inteligentes	(STRADOLINI et al., 2018a) (ELMISERY; ABORIZKA,) (IRFAN; AHMAD, 2018) (CHARALAMPIDIS et al., 2017)
	Genérico	(KOTRONIS et al., 2018) (DAS et al., 2015) (EL-LATIF et al., 2017) (YAO et al., 2018)

Fonte: Autor (2020).

2.2.2 Aplicação

Concomitante à existência de diversidade na infraestrutura de sistemas *e-health*, podem ser encontradas quando são analisadas as aplicações IoT, sendo possível identificar alguns grupos na Tabela 4:

Tabela 4 – IoT - aplicações e monitoramentos

Aplicação	Tipo do Monitoramento	Artigo
Monitoramento de pacientes em hospitais	Temperatura; Pressão arterial; Frequência cardíaca; Nível de glicose.	(GHOSH et al., 2016)(MASIP-BRUIN et al., 2016)(STRADOLINI et al., 2018a)
		(RADWAN et al., 2017)(DHANALIYA; DEVANI, 2016)(PIYARE et al., 2013)(JI et al., 2017)(MATSUI; CHOI, 2017)(KYRIAZAKOS et al., 2017)(JOSHI et al., 2016)(HA; LINDH, 2017)(KO et al., 2015)(GHANAVATI et al., 2017)(LI et al., 2018)(HUAXIN et al., 2012)(VARGHEESE; VINIOTIS, 2014)(BHATIA; SOOD, 2018)(ALI; GHAZAL, 2017)
Acompanhamento de pacientes em atividades diárias	Temperatura; Pressão arterial; Frequência cardíaca.	
Acompanhamento de pacientes com doenças respiratórias	Frequência cardíaca; volume de oxigênio no sangue.	(MASIP-BRUIN et al., 2016)
Acompanhamento de idosos	Temperatura; Pressão arterial; Frequência cardíaca; Nível de glicose; Processamento digital de imagem.	(MATSUI; CHOI, 2017)(LI et al., 2018)(KOTRONIS et al., 2017)(HUAXIN et al., 2012)
Monitoramento de pacientes durante reabilitação	Monitoramento de sensores; Processamento digital de imagem.	(BAO et al., 2016)
Deteção de variação da luz artificial	Processamento digital de imagem.	(DAS et al., 2015)
Deteção de doenças através da íris	Processamento digital de imagem.	(EL-LATIF et al., 2017)
Deteção de ataques do coração	Frequência cardíaca.	(DHANALIYA; DEVANI, 2016)(RENNER et al., 2014)(HA; LINDH, 2017)
Deteção de quedas	Processamento digital de imagem; acelerômetro; giroscópio; monitoramento de sensores.	(BAO et al., 2016)(KOTRONIS et al., 2018)(YAO et al., 2018)
Realização e avaliação de eletrocardiograma	Pressão arterial; Frequência cardíaca;	(JI et al., 2017)(KO et al., 2015)(COROTINSCHI; GÁITAN, 2017)

Fonte: Autor (2020).

Monitoramento de pacientes em hospitais: Foram encontradas várias aplicações de IoT direcionadas aos pacientes hospitalizados, desde o monitoramento no leito, conforme descrito em (GHOSH et al., 2016) e (MASIP-BRUIN et al., 2016), a aplicações mais complexas, onde são monitorados os níveis de sedação dos pacientes no bloco cirúrgico (STRADOLINI et al., 2018a).

Acompanhamento de pacientes em atividades diárias: As aplicações de IoT que fornecem acompanhamento de pacientes durante suas atividades diárias são comumente usadas por aqueles que necessitam de assistência médica pela necessidade constante de verificação de sinais vitais (pressão arterial e frequência cardíaca). Essas aplicações também armazenam os dados coletados na *cloud* para futuras verificações ou análises e, em caso de possíveis anormalidades, a aplicação pode gerar algum alerta para o médico que faz o acompanhamento (RADWAN et al., 2017), (DHANALIYA; DEVANI, 2016), (PIYARE et al., 2013), (JI et al., 2017), (MATSUI; CHOI, 2017), (KYRIAZAKOS et al., 2017), (JOSHI et al., 2016), (HA; LINDH, 2017), (KO et al., 2015), (GHANAVATI et al., 2017), (GHANAVATI et al., 2017), (HUAXIN et al., 2012), (BHATIA; SOOD, 2018), (ALI; GHAZAL, 2017). Entre essas aplicações, é interessante notar uma proposta apresentada por Vargheese e Viniotis (VARGHEESE; VINIOTIS, 2014), em que o acompanhamento diário dos pacientes é realizado durante o período de 30 dias após a alta médica para evitar a recorrência do agravo de saúde.

Acompanhamento de pacientes com doenças respiratórias: Esta aplicação é focada em pacientes que têm doenças respiratórias crônicas e precisam usar cilindros de oxigênio, conforme descrito em Masip-Bruin et al. (MASIP-BRUIN et al., 2016). A solução pretende oferecer uma melhor qualidade de vida às pessoas que apresentam essa enfermidade. Originalmente, essa aplicação foi projetada para ser usada em um contexto pessoal, acompanhando o paciente durante suas atividades diárias; no entanto, também pode ser usado em hospitais, pois os dispositivos IoT são capazes de medir a saturação de oxigênio que circula pelo sangue do paciente.

Acompanhamento de idosos: Com o aumento da expectativa de vida (ORGANIZATION, 2018), surgiram várias aplicações IoT que fazem o acompanhamento de idosos, variando desde a aferição da pressão arterial, (LI et al., 2018), (KOTRONIS et al., 2017), (HUAXIN et al., 2012) ao monitoramento de atividades diárias com o uso de técnicas de processamento digital de imagem (KYRIAZAKOS et al., 2017). Dentre as aplicações encontradas neste mapeamento, os autores Matsui e Choi (MATSUI; CHOI, 2017) apresentam uma solução para evitar o choque térmico, que é uma das maiores causas de mortes de idosos no Japão. Nesta aplicação, os autores utilizam sensores espalhados pela casa e um sensor próximo aos idosos, todos interconectados a uma central que controla a temperatura do ambiente e, durante a locomoção dos idosos pela casa, o sistema aquece o espaço mais próximo para evitar o choque térmico.

Monitoramento de pacientes durante reabilitação: É possível monitorar o paciente durante a reabilitação para medir e acompanhar a sua evolução através de um dispositivo IoT proposto por Bao et al. (BAO et al., 2016), que monitora, através do processamento digital de

imagem e sensores, todos os movimentos realizados durante a execução dos exercícios.

Deteção de variação da luz artificial: Em Das et al. (DAS et al., 2015), os autores apresentam uma aplicação IoT para deteção de riscos à saúde pela variação da luz artificial. A aplicação consiste no processamento de imagens para identificar quais riscos o usuário está sendo exposto à oscilação da luz em frequências de aproximadamente 3 hertz a 70 hertz, que pode causar convulsões em indivíduos altamente sensíveis, enquanto frequências de cintilação moderadas, de cerca de 100 hertz a 500 hertz, podem levar a percepção indireta dos efeitos estroboscópicos.

Deteção de doenças através da íris: Os autores em El-Latif et al. (EL-LATIF et al., 2017) propõem uma solução também baseada no processamento digital de imagens, onde, através de uma imagem da íris, a aplicação pode identificar possíveis doenças, além de fazer o monitoramento do humor e identificar se o paciente está sentindo algum tipo de incômodo ou dor durante o procedimento médico.

Deteção de ataques cardíacos: As aplicações para deteção de ataques cardíacos, conforme apresentado por Ali e Ghazal (ALI; GHAZAL, 2017), demonstram que os pacientes fazem uso de vários tipos de dispositivos portáteis, como *smartwatch* e *smartphone*, assim é possível monitorar o paciente através da aplicação, identificar quando um acometimento está para ocorrer e gerar uma notificação ao hospital mais próximo. Neste estudo sistemático de mapeamento, foram encontradas outras três aplicações que detectam ataques cardíacos: (RENNER et al., 2014), (DHANALIYA; DEVANI, 2016) e (HA; LINDH, 2017).

Deteção de queda: São apresentadas aplicações que fazem deteção de quedas em (BAO et al., 2016), (KOTRONIS et al., 2018) e (YAO et al., 2018). Essas aplicações IoT podem usar fontes de entrada de dados diferentes, como sensores de movimento dos dispositivos conectados ao corpo, dados de acelerômetro e giroscópio, processamento digital de imagens e análise de vídeos. Após a deteção da queda, é gerado um alerta para os contatos de emergência do paciente (familiares, cuidadores ou equipe médica).

Exame de eletrocardiograma: Foram encontradas algumas aplicações de IoT que lidam com dados de eletrocardiograma: (JI et al., 2017), (KO et al., 2015) e (COROTINSCHI; GĂITAN, 2017). Dentre elas, é interessante destacar a apresentada por Corotinschi e Găitan (COROTINSCHI; GĂITAN, 2017); é uma aplicação utilizada para prover serviços médicos de qualidade aos pacientes que moram em locais remotos, onde, por meio de um dispositivo IoT, é possível fazer o eletrocardiograma e, em seguida, os dados são enviados para a *cloud*, na qual médicos especialistas avaliam o exame.

Baseado na análise dos resultados encontrados, foi desenvolvido um sistema *e-health* embasado em modelo de *deep learning* para classificação de atividades diárias. A escolha surgiu tendo em vista que essa aplicação atende a quatro grandes grupos da Tabela 4: monitoramento de pacientes em hospitais, acompanhamento de atividades diárias, acompanhamentos de idosos e

detecção de quedas. Assim, o sistema implementado visa ter um parecer generalista para sistemas *e-health* que utilizam como infraestrutura IoT, *fog* e *cloud* e executam na *fog* um modelo de *deep learning*.

2.2.3 Tipo de monitoramento

Por existir uma grande diversidade de dispositivos IoT, é possível monitorar vários aspectos relacionados à saúde dos pacientes. Durante o mapeamento sistemático foram encontrados os seguintes tipos de monitoramento: acelerômetro e giroscópio, eletromiografia (EMG), frequência cardíaca, nível de glicose, pressão arterial, processamento digital de imagem, sensores de movimento, temperatura, umidade e temperatura ambiente, e volume de oxigênio no sangue.

As aplicações IoT apresentadas na subseção anterior exigem que os dispositivos coletem os dados, os quais são segmentados de acordo com o objetivo do monitoramento, conforme descrito a seguir:

Acelerômetro e giroscópio: Os dados do acelerômetro e do giroscópio são usados para aplicações de detecção de quedas, onde através dos dados é possível identificar se o paciente sofreu uma queda (BAO et al., 2016), (KOTRONIS et al., 2018) e (YAO et al., 2018). A coleta desses dados geralmente é feita por dispositivos portáteis, como *wearables*, *smartwatches* ou *smartphones*.

Eletromiografia (EMG): é comumente usado para diagnosticar distúrbios, como neuropatia muscular e miopatia, conforme proposto em Ghanavati et al. (GHANAVATI et al., 2017).

Frequência cardíaca: A frequência cardíaca é um sinal vital usado no acompanhamento do paciente durante suas atividades diárias (RADWAN et al., 2017), (JI et al., 2017), (KYRIAZAKOS et al., 2017), (HA; LINDH, 2017), (KO et al., 2015), (GHANAVATI et al., 2017), (LI et al., 2018), (BHATIA; SOOD, 2018), (ALI; GHAZAL, 2017) e aplicações de acompanhamento de pacientes em hospitais (GHOSH et al., 2016), (STRADOLINI et al., 2018a). Além disso, autores em Corotinschi e Găitan (COROTINSCHI; GĂITAN, 2017) também coletam frequência cardíaca para realizar exames cardiológicos em pacientes que vivem em locais mais distantes.

Nível de glicose: O monitoramento do nível de glicose foi encontrado em algumas aplicações de IoT, cujo objetivo principal é o monitoramento de pacientes com doenças crônicas que sofrem influência da glicose (FAZIO et al., 2015), (ULLAH et al., 2016), (IRFAN; AHMAD, 2018).

Pressão arterial: Sensores que monitoram a pressão arterial são usados em aplicações de acompanhamento de pacientes em hospitais (GHOSH et al., 2016), (MASIP-BRUIN et al., 2016), na realização e avaliação de eletrocardiograma (COROTINSCHI; GĂITAN, 2017) e no acompanhamento durante as atividades diárias (RADWAN et al., 2017), (DHANALIYA; DEVANI, 2016), (JOSHI et al., 2016), (GHANAVATI et al., 2017).

Processamento digital de imagem (PDI): Algumas aplicações IoT fazem uso de técnicas de PDI a partir de dados coletados por câmeras IP e *Kinect*. Através do processamento dessas imagens, é possível identificar o progresso do paciente durante a reabilitação (YAO et al., 2018), avaliar os riscos de variação artificial da luz (DAS et al., 2015) e também diagnosticar possíveis doenças através da íris (EL-LATIF et al., 2017).

Sensores de movimento: O uso de dados coletados através de sensores de movimento é de suma importância para aplicações de monitoramento de pacientes durante o processo de reabilitação, onde através do movimento capturado é possível identificar a evolução do paciente durante a fase de recuperação (BAO et al., 2016) e também é usado em aplicativos de detecção de queda (YAO et al., 2018).

Temperatura: O monitoramento da temperatura corporal é realizado através dos sensores interconectados junto ao corpo. Esses dados são usados para alimentar aplicações e ficam disponíveis para consulta, podendo ser acessados pela família ou pela equipe médica que realiza o acompanhamento do paciente. Aplicações que fazem o uso desse monitoramento são descritos em (RADWAN et al., 2017), (MATSUI; CHOI, 2017), (HUAXIN et al., 2012), (BHATIA; SOOD, 2018). Esse tipo de monitoramento também é utilizado em aplicações hospitalares, onde é realizado o acompanhamento dos pacientes internados, conforme apresentado em (GHOSH et al., 2016), (MASIP-BRUIN et al., 2016).

Umidade e temperatura ambiente: Os níveis de umidade relativa e de temperatura ambiente são monitorados em sistemas *e-health* implementados em residências e hospitais inteligentes, onde é necessário controle sobre o meio ambiente (SAWAND et al., 2014), (PIYARE et al., 2013), (PESCOSOLIDO et al., 2016), (GHOSH et al., 2016), (ULLAH et al., 2016), (CABRA et al., 2017), (CHARALAMPIDIS et al., 2017). Em Matsui e Choi (MATSUI; CHOI, 2017), os autores usam a temperatura do ambiente como o principal dado de entrada para o sistema de *e-health*, uma vez que a proposta é um sistema inteligente de controle de temperatura para evitar que os idosos no Japão sofram choques térmicos e acabem tendo complicações de saúde que em casos mais graves podem levar a óbitos.

Volume de oxigênio no sangue: Conforme descrito em Masip-Bruin et al. (MASIP-BRUIN et al., 2016), este monitoramento é utilizado em aplicações que fazem o acompanhamento de pacientes que sofrem de doenças respiratórias.

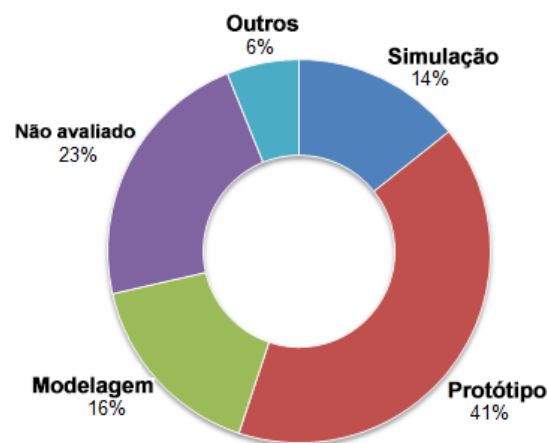
2.2.4 Método de avaliação

Grande parte dos trabalhos analisados utilizam modelagem e simulação como métodos de avaliação. Tais trabalhos geralmente visam ajudar ou criar uma tecnologia para resolver parte do problema; eles não apresentam aplicações que podem ser implementadas e colocadas em produção. As soluções que fazem uso de protótipo como forma de avaliação se preocupam com a possibilidade de reprodução do estudo e acabam descrevendo com maior detalhe todo o processo

realizado durante a pesquisa. Mesmo com a variedade de sensores para construção de protótipos e ferramentas para simulação e modelagem, alguns trabalhos não fazem nenhuma avaliação.

A Figura 3 mostra os principais métodos de avaliação utilizados nos trabalhos encontrados neste estudo de mapeamento sistemático. A maioria dos autores utiliza protótipo para avaliar suas soluções (41%). Esses trabalhos implementam aplicações, como monitoramento de pacientes em hospitais (GHOSH et al., 2016), (MASIP-BRUIN et al., 2016), (STRADOLINI et al., 2018a), monitoramento de pacientes em atividades diárias (RADWAN et al., 2017), (DHANALIYA; DEVANI, 2016), (PIYARE et al., 2013), (JI et al., 2017), (MATSUI; CHOI, 2017), (KYRIAZAKOS et al., 2017), (JOSHI et al., 2016), (HA; LINDH, 2017), (KO et al., 2015), (GHANAVATI et al., 2017), (LI et al., 2018), (HUAXIN et al., 2012), (BHATIA; SOOD, 2018), (ALI; GHAZAL, 2017), (VARGHEESE; VINIOTIS, 2014), acompanhamento de pacientes com doenças respiratórias (MASIP-BRUIN et al., 2016), acompanhamento de idosos (MATSUI; CHOI, 2017), (KYRIAZAKOS et al., 2017), (LI et al., 2018), (KOTRONIS et al., 2017), (HUAXIN et al., 2012), monitoramento de pacientes durante a reabilitação (BAO et al., 2016), detecção de variação da luz artificial (DAS et al., 2015), detecção de doenças pela íris (ALI; GHAZAL, 2017), detecção de ataques cardíacos (DHANALIYA; DEVANI, 2016), (RENNER et al., 2014), (HA; LINDH, 2017), (ALI; GHAZAL, 2017), detecção de quedas (BAO et al., 2016), (KOTRONIS et al., 2018), (YAO et al., 2018), exame de eletrocardiograma (JI et al., 2017), (KO et al., 2015), (COROTINSCHI; GĂITAN, 2017).

Figura 3 – Métodos de avaliação usados para validação da proposta



Fonte: Autor, 2020.

Para analisar o que é implementado nos sistemas estudados, foi levado em consideração a infraestrutura aplicada. Aplicações que utilizaram como forma de avaliação modelagem e simulação fazem uso de infraestrutura composta por IoT, *fog* e *cloud* em 70% das vezes; quando analisados os protótipos, apenas 18% das aplicações utilizaram essa infraestrutura. Isso indica que, em um cenário "real", implementar uma solução *e-health* que integre IoT, *fog* e *cloud* ainda é uma tarefa complexa.

Uma vez identificada essa complexidade, é encontrada também a oportunidade de desenvolvimento de um sistema *e-health*, que seja possível de ser implementado e que tenha bons resultados quando analisados o desempenho e a disponibilidade. O presente trabalho apresenta o desenvolvimento de um protótipo de baixo custo que faz uso da infraestrutura IoT, *fog* e *cloud* afim de coletar os valores obtidos durante experimentos e empregá-los como dados de entrada para modelos computacionais.

2.3 Desafios encontrados

Nesta seção, é apresentado alguns desafios e oportunidades de pesquisa encontrados em aplicações IoT implementados no contexto do sistema de *e-health*.

Capacidade limitada de dispositivos IoT: Os dispositivos IoT são os principais atores nesse contexto e, como supracitado, são comumente caracterizados por heterogeneidade e tamanho compacto, o que limita alguns de seus recursos, como computação, armazenamento e bateria. Apesar dos avanços das tecnologias, a maioria dos dispositivos IoT ainda são computacionalmente limitados, e isso afeta não apenas sua capacidade de executar tarefas em si, mas também aspectos de rede, como latência e comunicação com outros dispositivos (GHANAVATI et al., 2017).

Devido à necessidade de funcionamento constante, a duração da bateria também é uma questão crítica, especialmente ao lidar com aplicações de saúde (PIYARE et al., 2013). Para superar esse problema, o sistema também pode monitorar o uso de recursos e gerar notificações para recarregar ou substituir a bateria. Nesse caso, o sistema *e-health* deve ser aprimorado; em vez de gerar apenas alarmes referentes a saúde do paciente, ele também pode cuidar da saúde dos dispositivos IoT para evitar a indisponibilidade. Em Plageras et al. (PLAGERAS et al., 2017) e Charalampidis et al. (CHARALAMPIDIS et al., 2017), o consumo de energia dos dispositivos IoT é monitorado; no entanto, nenhum sistema de gerenciamento é aplicado.

Disponibilidade do sistema: A disponibilidade dos sistemas *e-health* é fundamental para garantir o monitoramento e a integridade das informações vitais coletadas do paciente (VARGHEESE; VINIOTIS, 2014), (IRFAN; AHMAD, 2018). Em Vargheese e Viniotis (VARGHEESE; VINIOTIS, 2014), os autores se concentraram no uso de sensores para coletar dados vitais assim que o paciente recebe alta do hospital e afirmam que a disponibilidade dos dispositivos de *cloud* e IoT é um aspecto crítico nesse cenário.

Em Santos et al. (SANTOS et al., 2018), os autores afirmam que para melhorar a disponibilidade do sistema *e-health* como um todo, a modelagem computacional pode ser utilizada para prever falhas nos dispositivos de IoT (nos níveis de *hardware* e *software*), encontrando assim gargalos no sistema e sugerindo redundância de recursos. É realizada a combinação de modelos de disponibilidade e desempenho para apresentar uma avaliação mais robusta de um sistema *e-health* IoT-to-cloud (I2C).

Integração, interoperabilidade e padronização: De acordo com Yao et al. (YAO et al., 2018), vários desafios permanecem em relação à interoperabilidade entre diferentes sensores com tecnologias de comunicação específicas e diferentes tipos de informações coletadas por eles. Além disso, cada usuário é único e variam em como, quando e por que usam esses sistemas (SINGH; BACON, 2014).

Geralmente, o *middleware* é apresentado como uma solução flexível de interoperabilidade. Em Yao et al. (YAO et al., 2018), os autores propõem um *middleware* de IoT. O principal objetivo é fornecer a infraestrutura uma colagem transparente em recursos e serviços heterogêneos, acessando sensores e atuadores em diferentes protocolos, plataformas e locais na pilha da Internet. Os autores em Singh e Bacon (SINGH; BACON, 2014) também propuseram um *middleware* como forma de interligar sistemas *e-health* que fazem o acompanhamento em tempo real, com base na produção de dados para vários dispositivos, suporte a múltiplas comunicações, segurança de dados e configuração dinâmica.

Segurança de dados: O uso de um aplicativo IoT baseado na infraestrutura de *fog* e *cloud* permite o armazenamento a longo prazo e o acesso remoto de modo fácil a dados médicos (BAGULA et al., 2016), (VARGHEESE; VINIOTIS, 2014). Com essas informações, é possível fornecer uma visão abrangente do histórico de saúde do paciente, apoiando a tomada de decisões e planejando processos mais personalizados (ELMISERY; ABORIZKA,).

De acordo com Yamin (YAMIN, 2018), a proteção da privacidade e da segurança dos dados médicos assumem maior pronúncia e prioridade, uma vez que os dados podem ser muito sensíveis e estar vinculados a vida dos pacientes. No entanto, os dispositivos IoT apresentam restrições nos recursos de computação e comunicação, e esse recurso apresenta desafios em termos de mecanismos de segurança que podem ser implantados em tais sistemas.

Um relato sobre os sistemas *e-health* e suas criticidades é descrito em Kotronis et al. (KOTRONIS et al., 2017), apresentando os principais desafios na implementação, dentre eles destacam-se os problemas relacionados a confiabilidade, disponibilidade e desempenho; o que, por se tratar de dados críticos de saúde, qualquer falha do sistema pode significar que o paciente esteja necessitando de ajuda.

Após a apresentação dos principais desafios, este trabalho apresenta uma proposta de resolução para dois desses desafios: **capacidade limitada de dispositivos IoT e disponibilidade do sistema**. Uma vez que a infraestrutura IoT, *fog* e *cloud* solucionará o problema de capacidade limitada dos dispositivos IoT, eles serão utilizados apenas como entrada dos dados, além de utilizar o *smartphone* como dispositivo IoT que por padrão já tem implementado os avisos sobre a bateria. Para garantir a disponibilidade do sistema foi construído um protótipo com essa infraestrutura e realizados experimentos cujo os resultados alimentaram os modelos computacionais capazes de garantir a alta disponibilidade do sistema.

2.4 Considerações

Este capítulo apresentou os resultados do mapeamento sistemático onde possibilitou a análise de diversos trabalhos relacionados ao tema da pesquisa, os quais foram segmentados por infraestrutura, aplicação, tipo de monitoramento e método de avaliação. Além disso, foi possível identificar as principais dificuldades da literatura para aplicações de sistemas *e-health*.

O presente trabalho se destaca por apresentar a proposta para alguns dos principais desafios encontrados para os sistemas *e-health*, além de monitorar o uso de um modelo de *deep learning* que garante o processamento dos dados mais próximo do usuário, ou seja, diretamente na *fog*. Outro fator relevante é a realização do desenvolvimento de um protótipo de sistema *e-health* interligando toda infraestrutura IoT, *fog* e *cloud* para a realização de experimentos em condições reais e, analisar os resultados obtidos através de modelos computacionais que visam garantir a disponibilidade e a performance do sistema.

3 Conceitos básicos

Este capítulo faz uma breve apresentação sobre os conceitos da infraestrutura e os sistemas *e-health*. Em suas Seções 3.1, 3.2, 3.3 e 3.4, são apresentadas as definições sobre IoT, *fog* e *cloud*, sistemas *e-health*, além das técnicas de modelagens e os modelos computacionais, respectivamente.

3.1 Definições IoT, *fog* e *cloud*

IoT remete ao conceito de que qualquer coisa pode ser conectada a *Internet* por meio de um protocolo de comunicação, estipulados através de equipamentos, e sensores capazes de coletar dados para realizar troca de informações e comunicações a fim de alcançar: reconhecimento de dados, posicionamento, rastreamento, monitoramento e administração (PATEL et al., 2016). Os autores em Buyya e Dastjerdi (BUYYA; DASTJERDI, 2016) definem o paradigma IoT como a promessa de fazer com que "coisas" como eletrodomésticos e eletrônicos sejam conectados junto a internet, esse paradigma abre portas para inovações que possibilitam a implementação de cidades inteligentes, e melhorias da qualidade de vida dos usuários.

O *National Institute of Standards and Technology* (NIST) define *fog* como um paradigma de recursos horizontal, físico ou virtual que reside entre os dispositivos finais inteligentes (os dispositivos IoT) e a *cloud* tradicional. Este paradigma oferece suporte a aplicações sensíveis à latência isolados verticalmente, fornecendo assim computação, armazenamento e conectividade escalonáveis em camadas (IORGA et al., 2017).

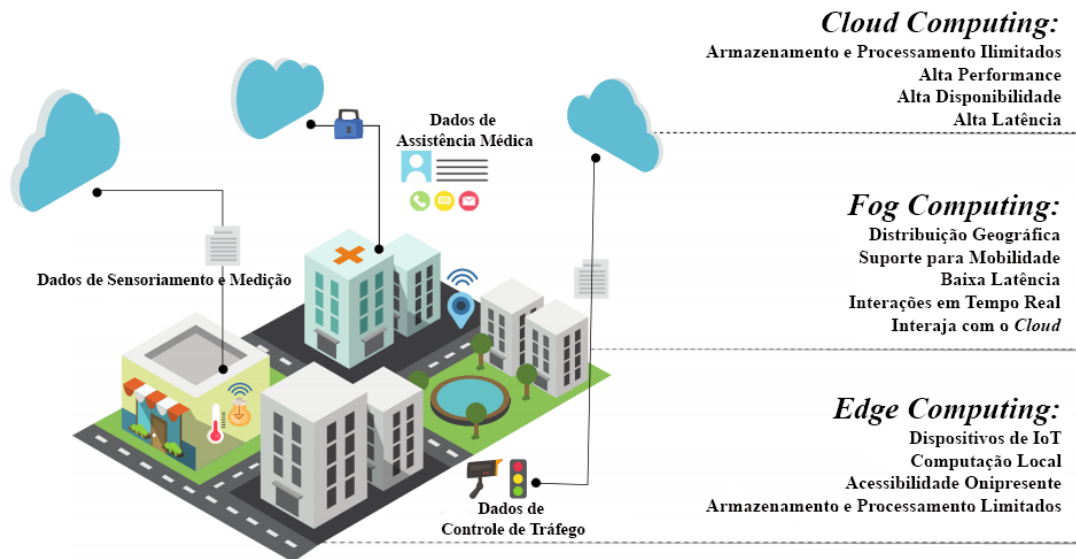
Em outro documento, o NIST define *cloud* como o modelo onipresente, conveniente e *on-demand* à rede para um conjunto de recursos computacionais compartilhados, que podem ser rapidamente provisionados e liberados com o mínimo de esforço de gerenciamento ou interação dos provedores de serviços (MELL et al., 2011).

Essas definições são primordiais para o entendimento de toda a infraestrutura utilizada.

A *fog* e a *cloud* podem ser vistas como tecnologias complementares as aplicações de IoT. A *cloud* pode compensar algumas das restrições computacionais dos dispositivos IoT, tais como limitações de processamento e armazenamento, oferecendo uma solução econômica para casos de uso de IoT. A *fog*, por sua vez, consegue trazer uma parcela do poder computacional da *cloud* para próximo da borda dos dispositivos IoT, atuando como um meio de diminuir a latência gerada na troca de informações entre o IoT e a *cloud* (KNISS et al., 2016), provendo assim aos dispositivos de IoT capacidade computacional mais próximo do usuário final. As aplicações IoT estendem a utilidade e o valor da *cloud* para o mundo real, onde empresas e usuários finais interagem com uma ampla variedade de aplicações. A Figura 4 apresenta um cenário de um

sistema IoT integrado com *fog* e *cloud* aplicado em uma cidade inteligente (SANTOS et al., 2018).

Figura 4 – Sistema IoT integrado



Fonte: Adaptação de (SANTOS et al., 2018)

Essa integração melhora o desempenho, a integridade e a disponibilidade das aplicações de IoT fornecendo um ambiente completo e robusto para esses aplicativos. As infraestruturas integradas de *fog* e *cloud* podem realmente atuar como uma plataforma computacional quase ilimitada e otimizada para aplicativos de IoT. Como tal, a disponibilidade de serviços em *cloud* e dispositivos de *fog* se torna mais crucial nesse cenário.

3.2 Sistemas *e-health*

Sistemas de *e-health* já são uma realidade em vários países do mundo e atualmente já estão em uso diversos sistemas de monitoramento remoto dos sinais vitais: frequência cardíaca, pressão arterial e temperatura do corpo; monitoramento de atividade físicas; além de sistemas de gerenciamento hospitalar onde tem-se o monitoramento dos pacientes em tempo real, esses dados coletados dos pacientes são armazenados na *cloud* e ficam disponíveis para o acompanhamento médico (GHOSH et al., 2016).

Um meio de visualizar como esses sistemas já estão presentes hoje, são o uso de pulseiras inteligentes, capazes de medir batimentos cardíacos, passos, e sono, com o auxílio de aplicativos é possível gerar relatórios e acompanhar a saúde de um paciente, gerando alertas de pouca movimentação, alguns exemplos são encontrados em (RADWAN et al., 2017), (JI et al., 2017), (KYRIAZAKOS et al., 2017), (HA; LINDH, 2017), (KO et al., 2015), (GHANAVATI et al., 2017), (LI et al., 2018), (BHATIA; SOOD, 2018), (ALI; GHAZAL, 2017).

Outra forma de ver esses sistemas, seria por meio de uma rede de sensores em um hospital, por exemplo, onde cada paciente tem algum dispositivo capturando dados vitais e enviando para um servidor processar, sinalizar em caso de alguma emergência e armazenar os dados (RADU et al., 2015) (RENNER et al., 2014) (DANG et al., 2018) (JI et al., 2017) (MATSUI; CHOI, 2017) (KYRIAZAKOS et al., 2017) (NEAGU et al., 2017) (CHEN et al., 2018) (JOSHI et al., 2016) (HA; LINDH, 2017) (KO et al., 2015) (BAGULA et al., 2016) (BAO et al., 2016) (HASSAN, 2015) .

3.3 Técnicas de modelagem

Os modelos computacionais foram construídos baseados nas técnicas de Redes de Petri e *Reliability Block Diagrams* (RBD), afim de avaliar aspectos de disponibilidade e desempenho do sistema baseados em dados obtidos através dos experimentos, e para modelar as dependências dentre todos os componentes da infraestrutura (IoT, *fog* e *cloud*) e representá-los.

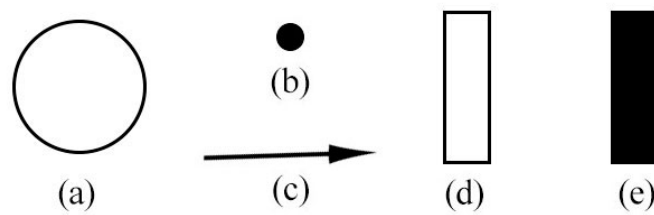
3.3.1 Redes de Petri

As redes de Petri são um meio de representação gráfica e matemática de sistemas computacionais distribuídos, revelando com clareza as relações intrínsecas aos seus componentes (BIASOLI; FONTOURA,) e (MACIEL et al., 1996). Os elementos mais básicos para se construir modelos de redes de Petri são ilustrados na Figura 5, onde cada um tem sua função representativa às ações necessárias dentro do modelo.

Os elementos são: (a) *Places*, usados para representar as entidades que estão sendo estudadas no modelo, nesse estudo presente, é exemplificado como sendo a representação do dispositivo IoT (*smartphone*), *fog* e *cloud*, que são círculos vazios e podem conter um ou mais (b) *Tokens*, que por sua vez são pequenos círculos pretos, atuando como um valor para o *place* naquele momento. Os *tokens* transitam entre os *places* por meio dos (c) Arcos, que ligam às transições aos *places* e vice-versa, controlando a via de quantos *tokens* podem transitar de uma única vez, servindo como um peso. Como se tratam de arcos direcionados, se o arco sai de um *place* em direção a uma transição, este *place* é considerado entrada àquela transição; já se o arco for de uma transição a um *place*, este é entendido como um *place* de saída. São apresentadas dois tipos de transições: (d) temporizada estocástica que atua sob um determinado tempo configurado no modelo, e (e) imediata que é ativada através de uma distribuição de probabilidade.

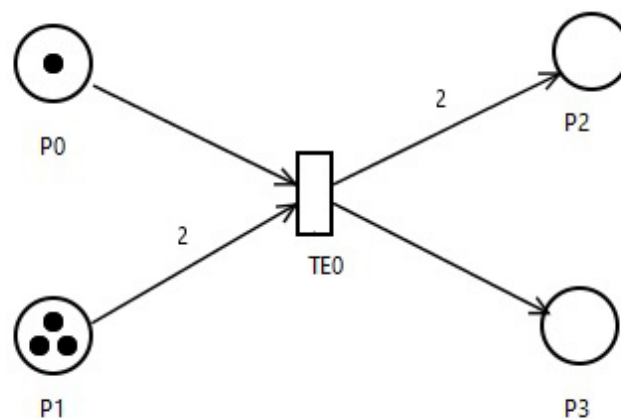
A cada vez que uma transição é disparada, ela consome *tokens* dos *places* de entrada e acrescenta no destino, assim alterando seu estado. Essas transições são habilitadas para disparar quando a quantidade de *tokens* está de acordo com o peso contido no arco presente nessa transição. Por conseguinte, será retirada a quantidade de *tokens* do *place* de entrada necessárias para a transição e, ao fim dessa transição os *tokens* retirados, serão levados para o *place* de saída.

Figura 5 – Itens na rede de Petri



Fonte: Autor, 2020.

Figura 6 – Exemplo de rede de Petri antes da transação



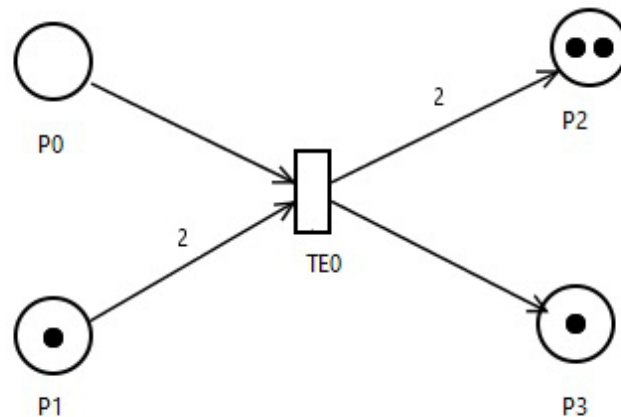
Fonte: Autor, 2020.

A Figura 6 mostra um exemplo de uma rede de Petri simples, onde é possível ver um modelo antes de ser executada a transação. O modelo utilizado no exemplo é constituído por quatro *places*, P_0 , P_1 , P_2 , e P_3 , e uma transação (TE_0). Torna-se notável que dois dos arcos possuem um número pequeno próximo a eles, são os pesos dos mesmos e, caso não possua de forma visível esse número, significa que o peso do arco é um. Para que a transação seja disparada é necessário que o *place* P_1 tenha no mínimo dois *tokens* contidos nele e o *place* P_0 tenha no mínimo um *token*. Os *places* P_0 e P_1 atendendo a esses requisitos mínimos, habilitam assim a realização da transição em questão.

Após o disparo da transição TE_0 , P_2 vai receber dois *tokens* e o P_3 apenas um, de acordo com os pesos dos arcos de saída ligados aos *places* P_2 e P_3 , respectivamente. Ao analisar a Figura 7 pode-se ver como fica o modelo após o disparo da transição. Observa-se que de P_0 e P_1 foram removidos apenas a quantidade de *tokens* que os arcos de entrada necessitavam.

Inicialmente as redes de Petri não tinham o conceito de tempo implementado, mas com o avanço das pesquisas nas áreas de sistemas computacionais, desempenho e problemas com escalonamento, se tornou essencial a utilização de tempo nos modelos computacionais. Um meio de vislumbrar a implementação do tempo dentro desse ambiente é por meio de processos probabilísticos dos disparos de transições (MACIEL et al., 1996). Ainda segundo (MACIEL et

Figura 7 – Exemplo de rede de Petri depois da transição



Fonte: Autor, 2020.

al., 1996), em meio a diversas abordagens de redes de Petri que possuem requisitos temporais, um dos principais requisitos apresentados é que o tempo é associado à transição.

A *Stochastic Petri Net* (SPN), que vem como solução para a abstração da rede não usar em seus modelos o tempo de acionamento das transições, onde tem-se como diferencial levar em conta o tempo de disparo das transições já que o tempo tem uma importância significativa em um sistema. Considerar o tempo dentro da modelagem é algo fundamental, e com SPN é possível ter uma ferramenta para avaliação e modelagem de disponibilidade de um sistema que envolve concorrência (SALES, 2002).

A SPN considera como uma função de acionamento das transições uma distribuição de probabilidade específica, podendo ser uma exponencial (MOLLOY, 1982). Outra característica de uma SPN é que ela pode ser convertida em uma Cadeia de Markov de Tempo Contínuo (do inglês, *Continuous Time Markov Chain* (CTMC)) e resolvido através de métodos numéricos ou analíticos por meio de simulação, (BAUSE; KRITZINGER, 2002).

3.3.2 Reliability Block Diagrams (RBD)

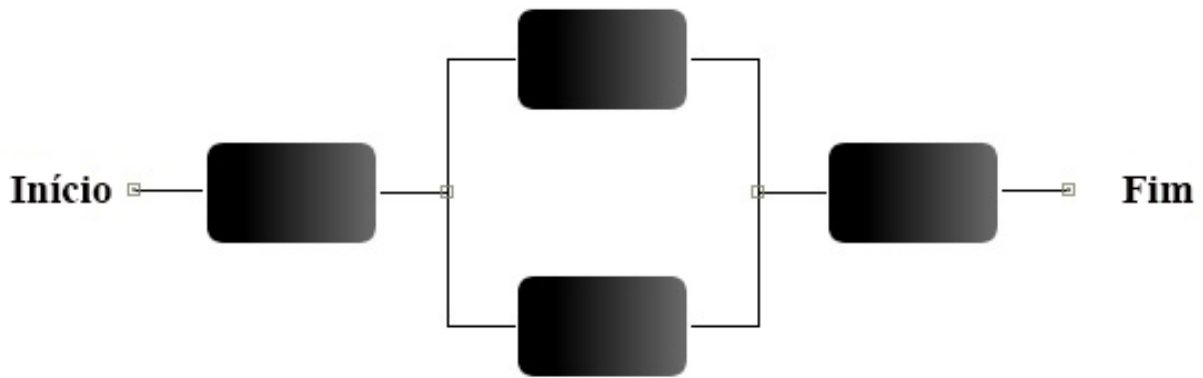
Os *Reliability Block Diagrams*, ou RBD's, são uma ferramenta simples de diagramação de blocos, que usa-os através de caixas pretas para representar componentes de sistemas complexos, supondo dois estados para cada bloco: operacional ou falha. É possível fazer uma análise de confiabilidade do sistema como um todo através dessa técnica (FORCHE, 1990). A abordagem RBD define interações lógicas com componentes que definem quais combinações de elementos em falhas e ativos podem sustentar a operação do sistema. Esse formalismo permite análise de confiabilidade, disponibilidade e manutenção (MACIEL et al., 2010).

Os blocos em um modelo RBD podem ser conectados, de forma paralela ou serial, mas também podem ser estruturados com ambas conexões, que é o caso do modelo de exemplo apresentado na Figura 8. As conexões em série representam as portas lógicas *AND*, as paralelas

representam as portas lógicas *OR*, enquanto que a combinação de ambas conexões representa redundância. Da esquerda para direita, existem dois caminhos que possuem conexões para os blocos seguintes, que são os nós, o que demonstra duas condições para a operação ser bem-sucedida.

O sistema se mantém operando com sucesso até que nenhum caminho válido da extremidade esquerda até o último nó da extremidade direita possua conexão. Através desse mecanismo, a probabilidade de falha do sistema pode ser calculada de acordo com os princípios probabilísticos (GUO; YANG, 2007). A conexão é cortada à medida que ocorrem falhas nos componentes desse processo.

Figura 8 – Exemplo de modelo RBD



Fonte: Autor, 2020.

A disponibilidade de um modelo RBD pode ser dada pela divisão do valor de MTTF (*Mean Time To Failure*) pelo valor de MTBF (*Mean Time Between Failures*) como mostrado na Equação 3.1, que por sua vez, nada mais é do que a soma dos valores de MTTF e MTTR (*Mean Time To Repair*) (VERMA et al., 2010).

$$A_x = \frac{MTTF_x}{MTBF_x} = \frac{MTTF_x}{MTTF_x + MTTR_x} \quad (3.1)$$

3.4 Modelos computacionais

Modelos computacionais representam o comportamento do sistema real, sendo possível observar como a modificação dos parâmetros de entrada altera o estado do sistema (KOROLYUK; KOROLYUK, 2012). Portanto, esses modelos analíticos são uma ferramenta poderosa para analisar sistemas complexos. Neste trabalho foram utilizadas abordagens de modelagem para avaliar uma aplicação de *e-health* baseada em IoT, *fog* e *cloud*. Métricas, como disponibilidade ou relacionadas ao desempenho, são analisadas com essas técnicas e serão apresentadas na Seção 5.3.

Avaliar a disponibilidade dos sistemas é um aspecto relevante para quem é provedor de serviços, e faz-se imprescindível melhorar constantemente esses valores (ENDO et al., 2017).

Um estudo de quais fatores mais impactam a disponibilidade é de suma importância. Entender como reduzir esses fatores e conseguir um aumento na disponibilidade, contudo, é algo que deve ser observado quanto ao custo de implementação da aplicação. Em um contexto de sistemas de monitoramento de *e-health*, alta disponibilidade e alto desempenho são ainda mais essenciais, uma vez que, qualquer atraso da informação pode ser crucial para o bem-estar e a vida do paciente (KOTRONIS et al., 2017), (VARGHEESE; VINIOTIS, 2014), (IRFAN; AHMAD, 2018). A modelagem da disponibilidade e desempenho de tal sistema é também um desafio pois é preciso identificar as dependências entre os componentes e entender como as falhas individuais causam impacto no pleno funcionamento da aplicação, além de se procurar alternativas de implementação com baixo custo.

Existem vários métodos que podem ser aplicados a esse objetivo, como o *Stochastic Petri Nets* (SPN), Cadeia de Markov, *Reliability Block Diagrams* (RBD) entre outros, porém os métodos escolhidos para fazer a modelagem do sistema foi o SPN e RBD, onde através deles é possível estimar a disponibilidade e o desempenho do sistema a fim de encontrar os seus principais gargalos. A modelagem do sistema é descrita no Capítulo 5.

3.4.1 Modelo de performance

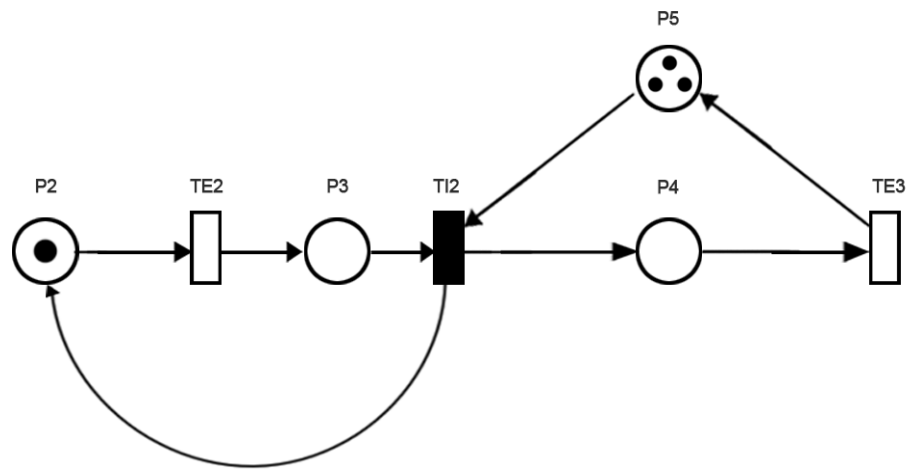
Em paralelo ao desenvolvimento e implementação do protótipo, foi realizada a modelagem do sistema. Para tanto, utilizou-se os conceitos de SPNs, e a ferramenta escolhida para criar os modelos e realizar a análise foi o *Mercury*¹. Foram realizadas análises estacionárias para calcular a disponibilidade, através desse método é criado uma CTMC subjacente definida pelo modelo SPN e pode ser resolvido para obter uma solução analítica (SANTOS, 2018), os resultados serão apresentados em seguida na Seção 5.2.

Para representar os clientes que solicitam um serviço, é considerado um sistema de fila simples $M/M/1/K$, como apresentado na Figura 9. O processo de chegada é uma distribuição de Poisson com taxa λ (M), o tempo de serviço é independente e exponencialmente distribuído com o parâmetro μ (M) e, há apenas um único servidor para processar os pedidos (1), considerando que a capacidade do sistema é limitado (K). Esta configuração de fila permite a avaliação de aspectos relevantes, tais como o impacto de diferentes tempos de chegadas e capacidades de atendimento das filas (SANTOS et al., 2018), (RAHMATI et al., 2014).

Para ilustrar como essas métricas de performance foram modeladas usando uma abordagem SPN, na Figura 9 é apresentado o modelo de fila simples. A transição $TE2$ representa a chegada de solicitações (com taxa λ (M)), enquanto a transição $TE3$ representa o tempo de serviço. Considera-se que o tempo de serviço é uma distribuição exponencial (YANG et al., 2009) (KHAZAEI et al., 2011). O *place* $P4$ representa a solicitação que está em processamento pelo único servidor disponível 1. O *place* $P5$ representa a capacidade total K que o sistema pode suportar, ou seja, é o número de solicitações que podem ser enfileiradas no sistema.

¹ <http://www.modcs.org/?p=2264/>

Figura 9 – Exemplo de Fila



Fonte: Autor, 2020.

3.5 Considerações

O presente capítulo apresentou um conjunto de conceitos fundamentais para entender os modelos propostos neste trabalho. Foram apresentados os conceitos de IoT, *fog* e *cloud*, modelos computacionais e suas técnicas de modelagem. Os conceitos apresentados são essenciais para um melhor entendimento do Capítulo 4.

4 Prototipando um sistema *e-health*

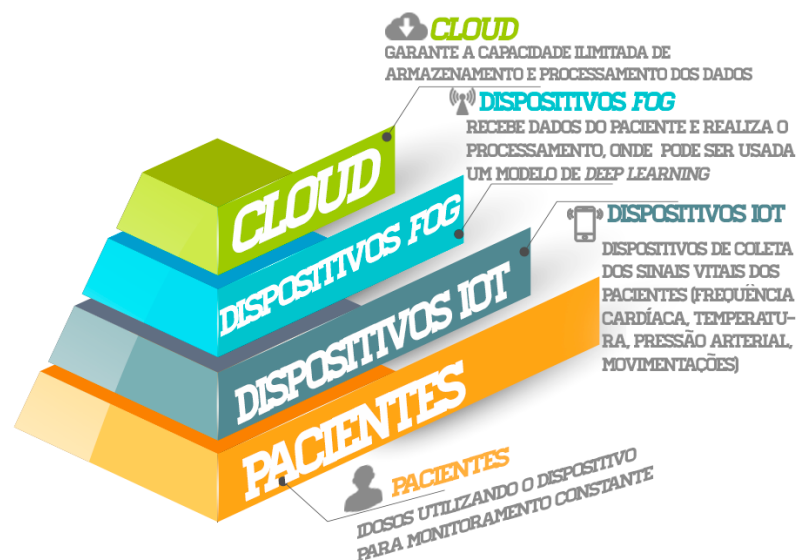
Este capítulo apresenta o processo de prototipação de um sistema *e-health*, em sequência, a arquitetura do sistema é apresentada na Seção 4.1, os componentes do protótipo em 4.2 e por fim são demonstrados os resultados dos experimentos em 4.3.

4.1 Arquitetura do sistema *e-health*

Este trabalho apresenta uma arquitetura básica generalista que foi idealizada e concretizada através de um protótipo para um sistema *e-health*. Ela é composta por quatro camadas integradas (pacientes, dispositivos IoT, *fog* e *cloud*) (MONTEIRO et al., 2018), porém, de acordo com a aplicação podem acontecer diferentes combinações.

Concentrando-se em uma solução de baixo custo foi escolhido como dispositivo IoT o *smartphone*. A *fog* executa um modelo *deep learning* utilizado para classificação de atividades diárias baseado em dados provenientes de um dispositivo IoT (NETO et al., 2020). A arquitetura proposta para o sistema *e-health* é respaldada em aplicações que fazem uso da infraestrutura composta por IoT, *fog* e *cloud* onde é representada na Figura 10.

Figura 10 – Proposta: Arquitetura para sistemas de *e-health*



Fonte: Autor, 2020.

Os sensores do dispositivo IoT coletam os dados do paciente de forma contínua e enviam diretamente para a *fog* na qual acontecerá o processamento dos dados e classificação a atividade realizada pelo o usuário no último segundo, garantindo assim a rápida resposta do sistema, tornando-o apto caso seja identificado algum tipo de anormalidade (MIOTTO et al., 2017). Em

situação elucidada, caso o mesmo sofra alguma queda, um alerta poderá ser emitido através do sistema possibilitando, desta forma, uma maior agilidade no atendimento.

Diante disso, após a classificação, a *fog* poderá eventualmente enviar os dados do paciente para a *cloud*, no intuito de catalogar e armazenar em larga escala os mesmos para análise posterior, e se oportuno, esses dados poderão ser utilizados para retreinar o modelo de *deep learning*. Em alguns casos, a *cloud* pode não fazer parte da infraestrutura, sendo assim, o dispositivo da *fog* irá armazenar localmente esses dados e após chegar ao limite da sua capacidade de armazenamento começará a sobrescrever os dados mais antigos.

Definiu-se nessa perspectiva duas possibilidades: (a) a *fog* realizar a classificação e o armazenamento dos dados e (b) a *fog* fazer a classificação dos dados e enviar para a *cloud* onde transcorrerá o armazenamento e a disponibilização dos dados para consultas.

4.1.1 Design de experimentos

Segundo (JAIN, 1990), um experimento computacional se inicia através da definição de um projeto experimental. O *design* do experimento é fatorial completo, pois, faz uso de todas as combinações possíveis, e em todos os níveis de todos os fatores, cada um dos k fatores é utilizado em dois níveis diferentes o que requer k experimentos. Este é um projeto comumente chamado *2k design* (JAIN, 1990).

A Tabela 5 apresenta o design dos experimentos, com a variação do *hardware* da *fog*. Por conseguinte, com o emprego dos dispositivos diferentes, a *fog* vai assumir dois papéis distintos. Em alguns cenários, a mesma será vista como unidade de processamento (classificação) e armazenamento, já em outros cenários, será reputada apenas como processamento, enviando os dados para serem armazenados na *cloud*. Cada cenário foi executado 30 vezes, durante 30 minutos, e todos os experimentos foram feitos sob as mesmas condições, prezando a variabilidade do mesmo.

Tabela 5 – Design do experimento

Cenário	Tipo do dispositivo	Tipo da aplicação da fog	Tipo da aplicação da cloud
Cenário 1	<i>Raspberry pi II</i>	Classificação	Armazenamento
Cenário 2	<i>Raspberry pi II</i>	Classificação & Armazenamento	-
Cenário 3	<i>Raspberry pi III</i>	Classificação	Armazenamento
Cenário 4	<i>Raspberry pi III</i>	Classificação & Armazenamento	-

Fonte: Autor (2020).

A Figura 11 retrata o cenário 1, composto pelo dispositivo IoT responsável por realizar coleta dos dados; a *Raspberry Pi II* sendo utilizada como *fog*, que executa a aplicação de classificação dos dados; e a *cloud* que, por sua vez, faz o armazenamento dos dados, e também tem a função de servidor NTP (do inglês *Network Time Protocol*) da rede. O cenário 3 tem

infraestrutura similar ao cenário 1, entretanto, o mesmo utiliza como dispositivo da *fog* a *Raspberry Pi III*, em ambos os cenários a *cloud* é usada para armazenamento dos dados.

Figura 11 – Configuração: Cenário 1



Fonte: Autor, 2020.

Os cenários 2 e 4 não fazem uso da *cloud* e tem sua infraestrutura composta apenas pelo o dispositivo IoT e a *fog*. A diferença dentre esses dois cenários é o tipo do dispositivo utilizado na *fog*, *Raspberry Pi II* e *Raspberry Pi III*, respectivamente.

Para avaliar o desempenho do protótipo, foi escolhido monitorar o consumo de recursos de *hardware* (processamento e memória) do dispositivo da *fog*, além de coletar o tempo de serviço de cada requisição. O objetivo principal do protótipo é obter através dos experimentos, dados reais para alimentar os modelos estocásticos de desempenho e disponibilidade do sistema nos diversos cenários (SANTOS, 2018).

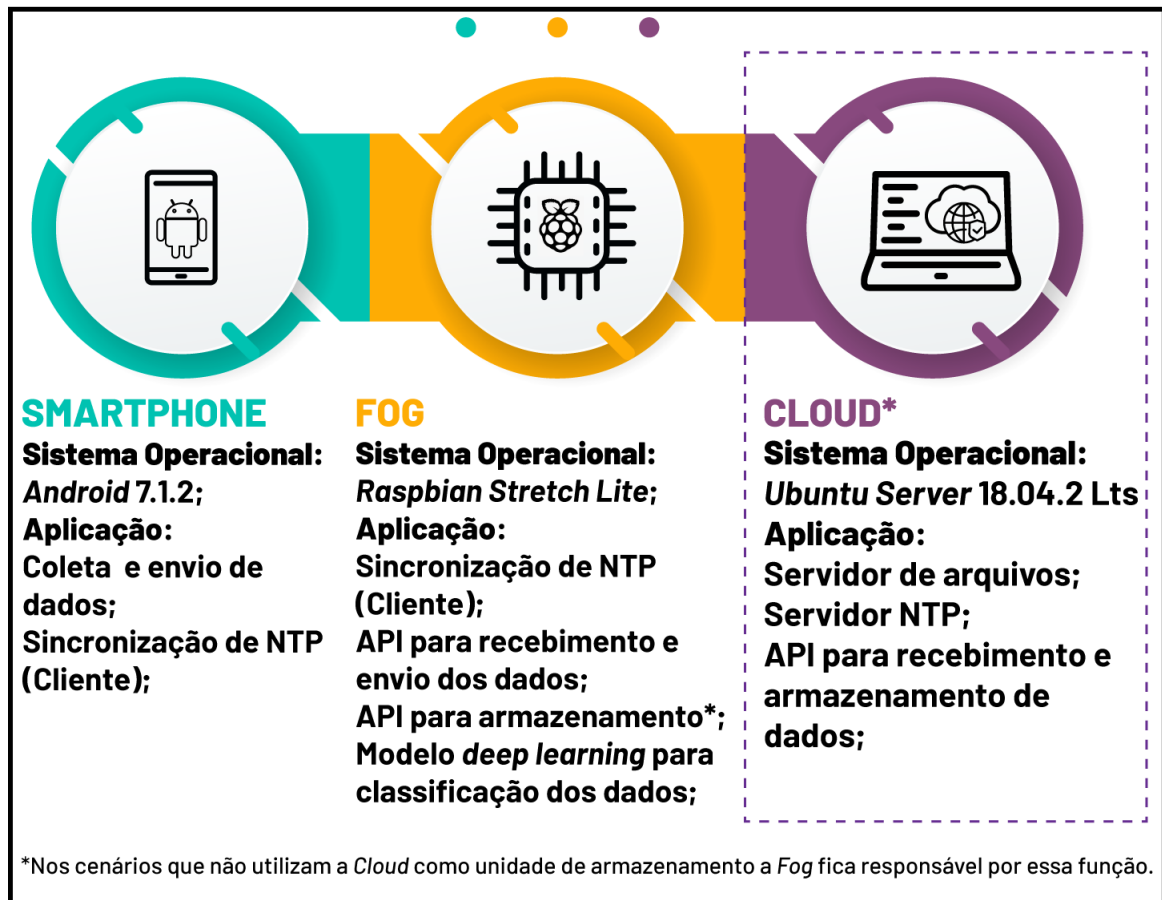
4.2 Prototipação

Baseado na arquitetura apresentada na Figura 10, foi construído um protótipo, tendo como direcionamento as seguintes definições de funcionamento: (a) Os pacientes devem utilizar sensores para monitoramento de atividades diárias; (b) Estes sensores estão no dispositivo IoT onde é executado um aplicativo para coleta e envio dos dados para a *fog*; (c) na *fog*, acontece o processamento, onde os dados são classificados, pelo o modelo de *deep learning* e em alguns cenários são armazenados localmente ou são enviados para serem armazenados na *cloud*; e (d) a *cloud* armazena os dados que estarão disponíveis para realização de análises em larga escala, além de ser o servidor NTP da rede.

A Figura 12 apresenta a configuração da estrutura de *software* necessária para criação do protótipo de um sistema *e-health* que utiliza a infraestrutura integrada IoT, *fog* e *cloud*. Essa infraestrutura sofre variações e em alguns cenários a *cloud* é retirada para que seja possível avaliar o desempenho da *fog* executando as duas tarefas (o conteúdo do retângulo de linhas pontilhadas não faz parte dos cenários 2 e 4).

Para os experimentos, foi utilizado um *smartphone* como dispositivo IoT, dois tipos de *hardware* para a *fog* e uma máquina virtual como *cloud*, todas as especificações técnicas de

Figura 12 – Estrutura do protótipo



Fonte: Autor, 2020.

hardware são apresentadas na subseção 4.2.2 e, a subseção 4.2.1 apresentará todos os *softwares* utilizados para implementação do protótipo, bem como o seu funcionamento.

4.2.1 Funcionamento do protótipo

O dispositivo IoT executa o aplicativo desenvolvido para coletar os dados referentes aos sensores de acelerômetro e giroscópio. Antes de iniciar a coleta dos dados, o aplicativo faz a sincronização com o servidor NTP da rede (nesse caso a *cloud*), ela é realizada em todos os componentes do protótipo para garantir que não haverá diferença nos horários de cada dispositivo. Após essa etapa, é iniciada a coleta dos dados e enviada via JSON (do inglês *JavaScript Object Notation*) a cada segundo para API (do inglês *Application Programming Interface*) de processamento que está na *fog*.

O *smartphone* utiliza o sistema operacional *Android* na versão 7.1.2 N2G47H, que foi escolhido por ofertar maior viabilidade técnica para o desenvolvimento de aplicativos. O aplicativo foi construído através da ferramenta *Android Studio*¹ com SDK (do inglês *Software Development Kit*) versão 3.6.2 e a linguagem utilizada foi JAVA com bibliotecas nativas para

¹ <https://developer.android.com/studio>

Android. O desenvolvimento foi baseado em (NETO et al., 2020), os dados do acelerômetro e giroscópio (coordenadas dos eixos X, Y, e Z) são coletados a cada 40 milissegundos e, após completar um conjunto de 22 coletas, esses dados são enviados para a *fog* afim de serem a entrada do modelo de classificação de atividades diárias.

A Figura 13 (a) apresenta a tela inicial do aplicativo onde é realizada toda a configuração da rede, informando os endereços do servidor NTP e da *fog*, além dos parâmetros para envio do JSON (porta e rota). Após essa fase, inicia-se a coleta dos sensores (acelerômetro e giroscópio) e com a realização do conjunto (Figura 13 (b)) são encapsulados juntamente ao tempo atual (*timestamp*) em uma mensagem (JSON) para assim serem enviadas.

Figura 13 – *Mockup* do aplicativo



Fonte: Autor, 2020.

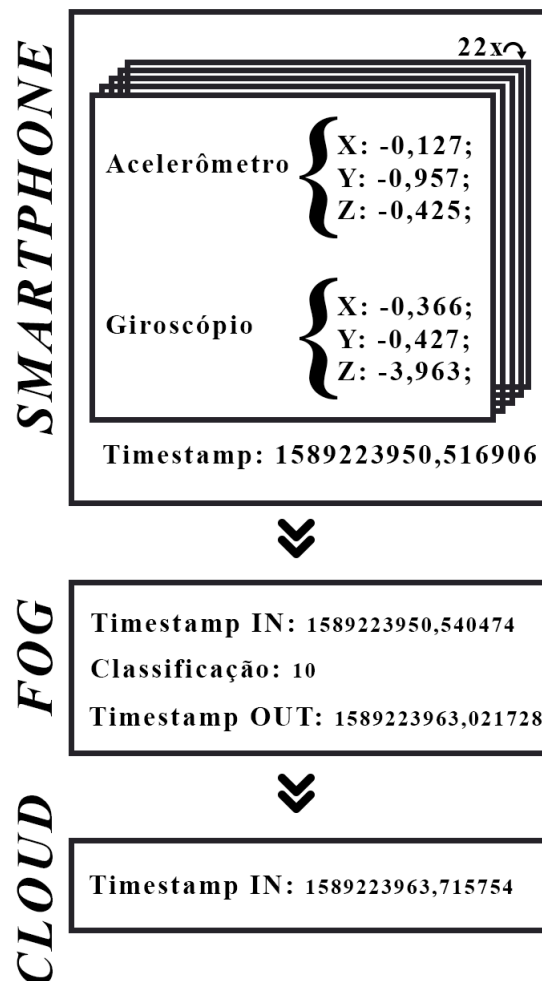
Os dados têm o seguinte fluxo: o *smartphone* faz a coleta dos dados durante o ciclo de um segundo e em seguida faz a captura do tempo concatenando as informações em um único JSON e envia para a API de classificação que está sendo executada na *fog*.

Após receber o JSON, ela irá coletar o tempo de chegada da mensagem e acionará a aplicação de *deep learning* para a classificação dos dados e ao finalizar, o resultado junto ao tempo

final é incrementado ao JSON, assim a mensagem está completa para ser enviada para API de armazenamento. Consequentemente, o JSON após o processamento (classificação da atividade) será composto das seguintes etapas: *Smart phone*(*Acelerometro_Giroscopio* + *Tempo_App*) + *Fog*(*Tempo_Chegada* + *Classificacao* + *Tempo_Saida*).

A Figura 14 apresenta o processo de incremento do JSON que deverá ser armazenado, nele é possível identificar as 22 coletas dos dados do acelerômetro e giroscópio (coordenadas dos eixos X, Y, e Z), os tempos de serviço de cada etapa além do resultado da classificação da atividade realizada naquele segundo pelo usuário.

Figura 14 – Metadado - JSON



Fonte: Autor, 2020.

Foram criadas as API's de (a) classificação e (b) armazenamento, a linguagem de desenvolvimento foi *Python 3*², com o *framework* chamado *flask*³. A API (a) classificação recebe o JSON enviado pelo aplicativo e aciona o modelo *deep learning* para classificação dos dados.

² <https://www.python.org/>

³ <https://github.com/pallets/flask>

Tabela 6 – Tipos de atividades classificáveis

Identificador da Atividade	Descrição
1	Cair para frente usando as mãos
2	Cair para a frente usando os joelhos
3	Cair para trás
4	Cair para o lado
5	Queda tentando sentar em uma cadeira vazia
6	Caminhando
7	Em pé
8	Sentado
9	Pegando um objeto
10	Deitado
11	Pulando

Fonte: Autor, 2020.

Após essa etapa o JSON é enviado para a API (b) armazenamento, onde será salvo em um arquivo.

O sistema operacional instalado na *fog* foi o *Raspbian Stretch Lite*⁴, sem interface gráfica para otimizar o desempenho. A aplicação de *deep learning* que realiza a classificação de atividades diárias, foi desenvolvido e apresentado em (NETO et al., 2020). Onde os autores criaram um modelo Bi-LSTM com uso do *framework Keras*⁵ e como *backend TensorFlow*⁶ capaz de identificar dentre 11 atividades diárias o que o usuário fez no último segundo, a Tabela 6 apresenta as 11 atividades que são identificadas através do modelo utilizado.

Para os experimentos, a *cloud* foi implementada em uma máquina virtual (do inglês Virtual Machine (VM)) fazendo uso do programa Microsoft Hyper-V⁷ e foi atribuída duas funções: servidor de armazenamento e servidor de NTP. O servidor de NTP garante que todos os relógios dos dispositivos estejam sincronizados, o que possibilita calcular o tempo serviço que é uma das métricas escolhida para avaliação do protótipo e será apresentada no decorrer do trabalho. O sistema operacional da VM é o *Ubuntu Server* na versão 18.04.2⁸ *Long Term Support* (LTS), que assim como a *fog* não faz uso de interface gráfica.

Todos os códigos desenvolvidos durante a implementação do protótipo estão disponíveis para consulta em repositório público do GitHub⁹, viabilizando a reprodutibilidade e fomentando o estudo.

⁴ <https://www.raspberrypi.org/downloads/raspbian/>

⁵ <https://keras.io/>

⁶ <https://www.tensorflow.org/>

⁷ <https://docs.microsoft.com/pt-br/virtualization/hyper-v-on-windows/about/>

⁸ <https://ubuntu.com/download/desktop>

⁹ https://github.com/K4y0/Mestrado_Ecomp

4.2.2 Descrição da infraestrutura

Para atuar como dispositivo IoT, foi escolhido o *smartphone* modelo Redmi 4X¹⁰, processador *Octa-Core* Max 1.40GHz, processador gráfico Adreno 505 e memória RAM 3Gb. A conexão de rede *wireless*, (padrão IEEE 802.11b/g/n), e com sensores de acelerômetro e giroscópio implementados a nível de *hardware*.

Para os experimentos a *fog* utiliza dois tipos de modelos diferentes de dispositivo: (a) *Raspberry Pi* II modelo B v1.1¹¹, processador *Quad-Core* ARM Cortex-A7 900MHz; (b) *Raspberry Pi* III modelo B V1.2¹², processador *Quad-Core* 1.2GHz Broadcom BCM2837, ambas com 1Gb de memória RAM e conexão de rede via *Ethernet*.

O dispositivo da *cloud* é uma máquina virtual com três processadores virtualizados (hospedeiro AMD Ryzen 5 3600X), 2Gb de memória RAM e acesso dedicado a placa de rede.

A conexão dentre *fog* e *cloud* é realizada via *Ethernet* e o *smartphone* é conectado através da rede *wireless*. Durante os experimentos o *smartphone* ficou em modo avião, isto é, apenas com a rede *wireless* ativada, porém sem internet. Todo o ambiente de rede foi isolado da internet para uma melhor análise do comportamento,

4.3 Resultados dos experimentos

Durante a execução dos experimentos, foi utilizado um *script*¹³ em *Python 3* para monitorar o consumo de processamento e o consumo de memória RAM do dispositivo da *fog*.

O *boxplot* contendo os resultados de consumo de processamento e consumo de memória RAM podem ser observados nas Figuras 15 e 16, respectivamente.

Os cenários 1 e 3 utilizaram menos recursos que os cenários 2 e 4, uma vez que nesses cenários (2 e 4) o dispositivo da *fog* executa duas tarefas, classificação e armazenamento. Outra observação relevante é a diferença de consumo dos recursos dentre os dispositivos *Raspberry Pi* II e *Raspberry Pi* III. Analisando as Figuras 15 e 16 é possível notar que os cenários (1 e 2), que utilizaram a *Raspberry Pi* II (1 e 2), consumiram mais recursos, tanto de memória quanto de processamento, para execução das tarefas destinadas, quando comparado com o modelo *Raspberry Pi* III dos cenários (3 e 4) essa diferença pode ser explicada pelo o maior poder computacional da *Raspberry Pi* III.

Ao analisar somente o uso de processamento e memória RAM, de acordo com (KOTRONIS et al., 2017) é possível afirmar que a *Raspberry Pi* II consegue atuar bem como dispositivo de *fog* em uma infraestrutura integrada IoT, *fog* e *cloud* de um sistema *e-health* que utiliza um modelo de *deep learning* para processamento dos dados.

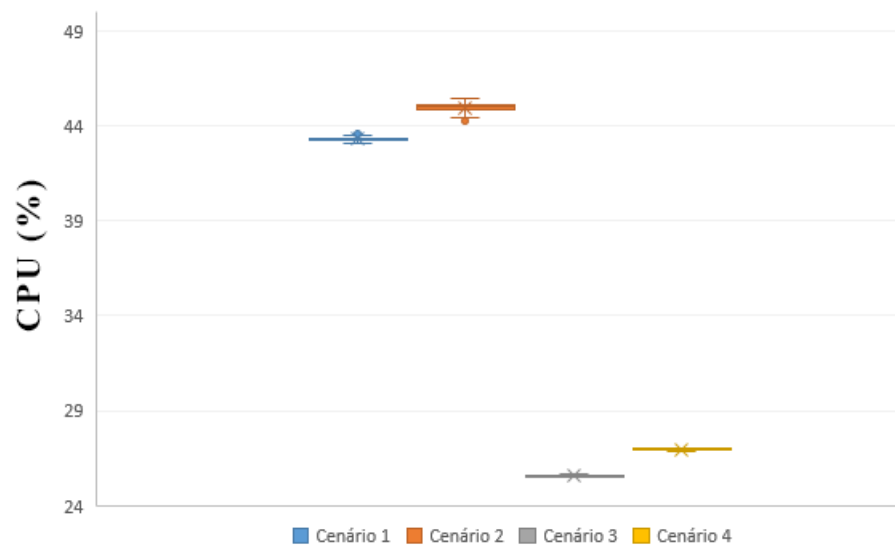
¹⁰ <https://canaltech.com.br/produto/xiaomi/redmi-4x/>

¹¹ <https://www.raspberrypi.org/products/raspberry-pi-2-model-b/>

¹² <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>

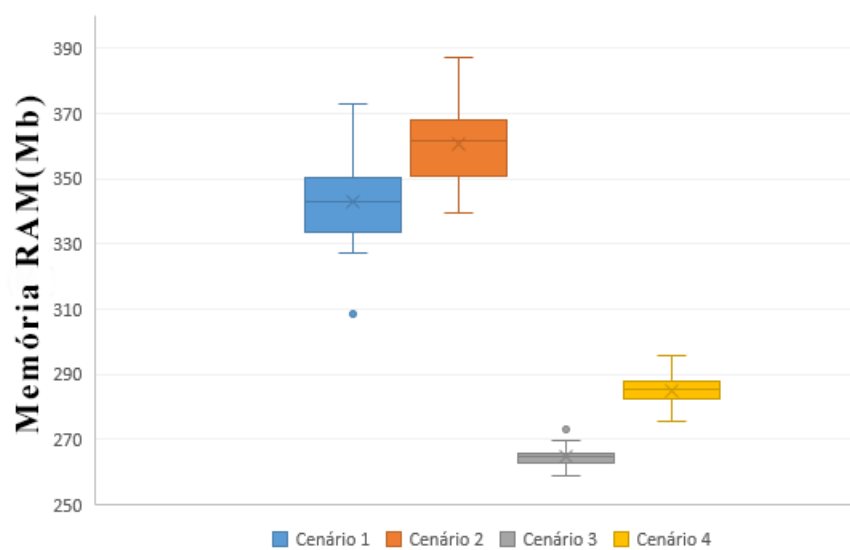
¹³ https://github.com/K4y0/Mestrado_Ecomp/

Figura 15 – Consumo em porcentagem de processamento durante os experimentos



Fonte: Autor, 2020.

Figura 16 – Consumo de memória RAM durante os experimentos



Fonte: Autor, 2020.

A Tabela 7 apresenta as médias de tempo de serviço (em segundos) de cada cenário para cada tarefa distinta (classificação e armazenamento). Sobre estes valores é interessante ressaltar que o apresentado de Armazenamento é referente a todo o tempo após a classificação da atividade, ou seja, uma vez que é finalizada a tarefa de classificação dos dados começa a ser contabilizado o tempo de armazenamento que inicia ao incremento do JSON até o seu arquivamento, esses valores apresentados são utilizados posteriormente pelos modelos SPN (seção 5.2).

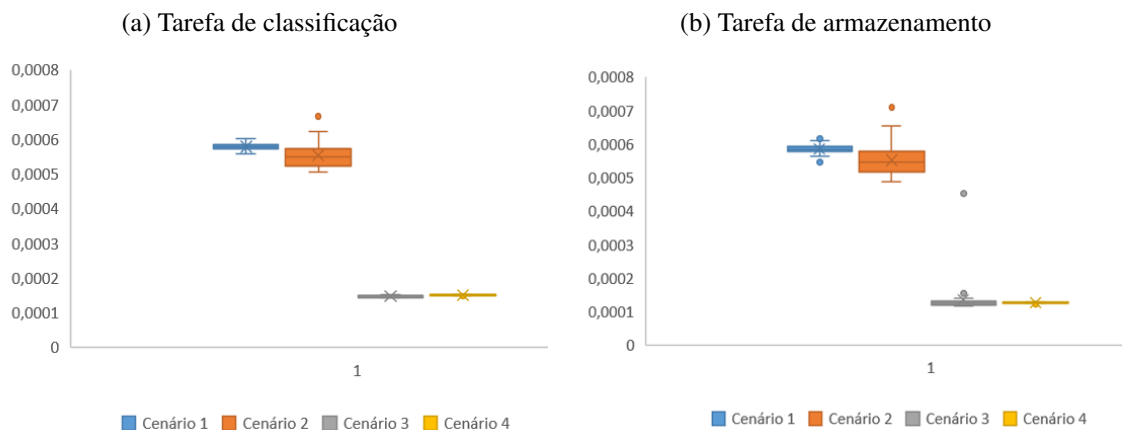
Os cenários 1 e 2 que utilizam a *Raspberry Pi II* como dispositivo de *fog* obtiveram tempos maiores na execução de cada tarefa, chegando a ser quatro vezes maior se comparados as médias dos cenário 1 (*Raspberry Pi II*) e cenário 4 (*Raspberry Pi III*), como podem ser observados na Figura 17. Faz-se necessário destacar que a *fog* não faz uso de todo o recurso

Tabela 7 – Resultados dos experimentos, em segundos.

Cenários	Tempo de Serviço Classificação	Tempo de Serviço Armazenamento
Cenário 1	2,0808	2,1060
Cenário 2	1,9872	1,9836
Cenário 3	0,5292	0,4932
Cenário 4	0,5436	0,4572

Fonte: Autor (2020)

Figura 17 – Médias de tempo de serviço durante os experimentos



Fonte: Autor, 2020.

computacional disponível do dispositivo, porém o tempo médio de serviço é algo que deve ser levado em consideração para escolha do dispositivo.

4.4 Considerações

Neste capítulo foi apresentado todo o processo de prototipação de um sistema *e-health*, desde a concepção de uma arquitetura básica, o *design* de experimentos utilizados para realização dos experimentos e análise dos cenários, além da descrição técnica de todos os dispositivos da infraestrutura e construção do protótipo.

Os resultados do tempo médio de serviço de cada cenário obtidos através dos experimentos são utilizados pelos modelos computacionais apresentados no próximo capítulo.

5 Modelando um sistema *e-health*

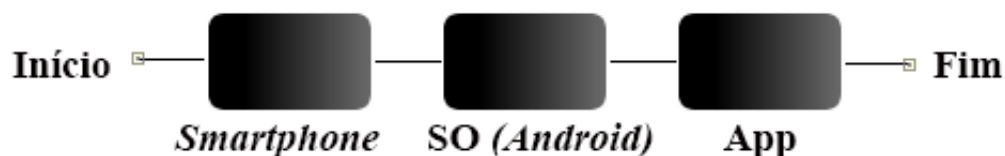
O processo de concepção dos modelos computacionais são divididos em modelos RBD e modelos SPN mencionados nas Seções 5.1 e 5.2, posteriormente, são apresentadas as métricas e a discussão dos resultados dos modelos, respectivamente nas Seções 5.3 e 5.4.

5.1 Modelos RBD

Para a execução dos modelos SPN e mensuração da disponibilidade do sistema é imprescindível ter os dados de *Mean Time To Failure* (MTTF) e *Mean Time To Repair* (MTTR) de todos os componentes da infraestrutura (WEBER, 2003). Esses dados são obtidos através de modelos RBDs para todos os componentes do experimento. Todos os modelos são construídos por blocos simples e interligados em série e estão representados nas Figuras 18, 19 e 20.

O modelo do dispositivo IoT é constituído por três blocos: um para o *smartphone*; um para o sistema operacional (SO); e outro para o aplicativo. É possível visualizar o modelo RBD utilizado na Figura 18.

Figura 18 – Modelo RBD do dispositivo IoT



Fonte: Autor, 2020.

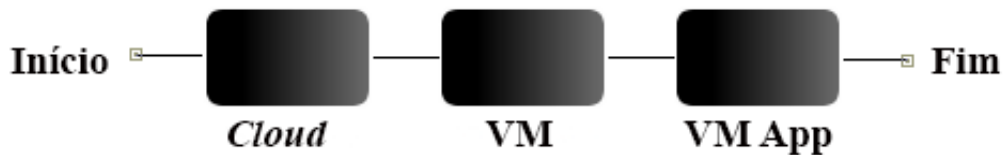
O modelo da *fog* é constituído por dois blocos, um para o *hardware* e outro para o *software* (sistema operacional). Pela escassez de dados na literatura referente aos valores de MTTF e MTTR para aplicação de *deep learning* executada na *fog*, foi considerado apenas o de *hardware* e o *software*. O modelo utilizado é apresentado na Figura 19.

Assim como o modelo construído para o dispositivo IoT, o modelo da *cloud* também faz uso de três blocos na sua estrutura (ver Figura 20): um para o sistema operacional da *cloud*; um para máquina virtual; e por fim o bloco da aplicação da máquina virtual.

Os valores que foram utilizados para construção dos modelos RBD do dispositivo IoT e *cloud* foram obtidos a partir dos dados apresentados em (MENDONÇA et al., 2015). Da mesma maneira a *fog* também teve seus valores obtidos a partir de (SANTOS et al., 2018). Todos os valores estão convertidos em horas e são apresentados na Tabela 8.

Figura 19 – Modelo RBD da *fog*

Fonte: Autor, 2020.

Figura 20 – Modelo RBD da *cloud*

Fonte: Autor, 2020.

Tabela 8 – Resultados do RBD

Dispositivo	Componentes (RBD)	MTTF	MTTR
IoT	<i>Smartphone + SO + App</i>	272,594	0,091
<i>Fog</i>	<i>HW Fog + SO Fog</i>	1105,860	1,573
<i>Cloud</i>	<i>Cloud + VM + VM App</i>	415,641	0,816

Fonte: (MENDONÇA et al., 2015) e (SANTOS et al., 2018).

5.2 Modelos SPN

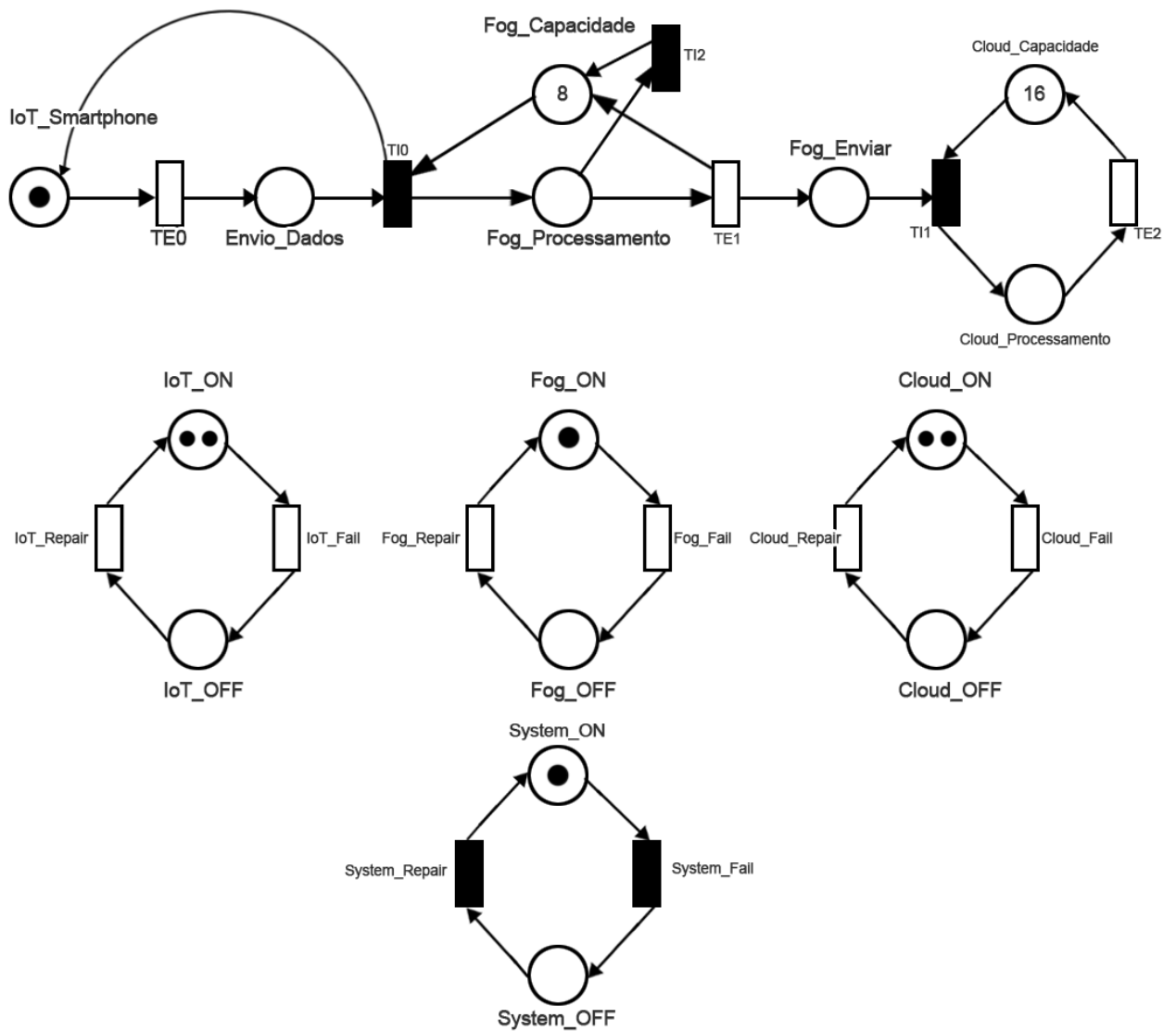
Os modelos SPNs propostos são representados nas figuras 21 e 22. Para compreensão dos modelos é necessário reforçar que todas as descrições de funções de guardas e métricas seguem a sintaxe da ferramenta Mecury.

A Figura 21 apresenta o sistema de uma aplicação *e-health* que utiliza a infraestrutura completa IoT, *fog* e *cloud*, que contemplam os cenários 1 e 3 (ver Tabela 5). A Figura 22 é referente a modelagem dos cenários 2 e 4 onde o dispositivo de *fog* realiza as tarefas de processamento e armazenamento dos dados, nesses cenários não tem a presença da *cloud*.

Na construção dos cenários os *places Fog_Capacidade* e *Cloud_Capacidade* receberam a quantidade de *tokens*, 8 e 16 respectivamente, para limitar o capacidade da fila de processamento desses dispositivos, esses valores foram obtidos a partir dos dados encontrados em (RODRIGUES et al., 2019). Para todos os cenários foi aceito como premissa que caso o dispositivo da *fog* venha a falhar, todos os dados que estão em processamento são descartados, para isso se empregou a transição imediata *TI2* com função de guarda apresentada nas Tabelas 10 e 13.

Para melhor entendimento do modelo, o mesmo pode ser segmentado em três partes.

Figura 21 – Modelo dos experimentos: cenários 1 e 3



Fonte: Autor, 2020.

A primeira parte se refere aos componentes do sistema e como eles estão conectados, são os *places* que representam o dispositivo IoT (*smartphone*), *fog*, *cloud* e suas respectivas transições. A seguinte, estão os *building blocks* do funcionamento de cada parte da infraestrutura, IoT, *fog* e *cloud* onde a presença do token junto ao *place ON* significa que a camada está ativa, ou seja, funcionando corretamente. Por fim, tem-se o *building block* que representa o estado geral de funcionamento do sistema (*ON* ou *OFF*), que nesses cenários, caso alguma camada venha a falhar, todo o sistema irá falhar. O *building block* de funcionamento do sistema (*system*) utiliza as funções de guarda apresentadas na Tabela 9.

Tabela 9 – Função de guarda de funcionamento do sistema: cenários 1 e 3

Experimento	Transição	Função de guarda
1 e 3	<i>System_FAIL</i>	$(\#IoT_ON = 0) \vee (\#Fog_ON = 0) \vee (\#Cloud_ON = 0)$
	<i>System_REPAIR</i>	$(\#IoT_ON > 0) \wedge (\#Fog_ON = 1) \wedge (\#Cloud_ON > 0)$

Fonte: Autor (2020)

Para o sistema ser considerado disponível é necessário que os *places IoT_ON*, *Fog_ON*, e *Cloud_ON* possuam pelo menos um *token*. Caso algum deles não tenha *token*, o sistema reconhece o estado de falha e utilizará os dados de MTTR (ver Tabela 11) como tempo de reparo do sistema. Os *places IoT_ON* e *Cloud_ON* possuem dois *tokens* para representar redundância da aplicação, ou seja, mais de um aplicativo pode enviar os dados para o sistema; e a redundância da *cloud* por ser um dispositivo mais robusto.

Tabela 10 – Função de guarda: cenários 1 e 3

Transição	Função de Guarda
<i>TE0</i>	$\#IoT_ON > 0$
<i>TI0</i>	$\#Fog_ON > 0$
<i>TE1</i>	$\#Fog_ON > 0$
<i>TI1</i>	$\#Cloud_ON > 0$
<i>TE2</i>	$\#Cloud_ON > 0$
<i>TI2</i>	$\#Fog_OFF = 1$

Fonte: Autor (2020)

As transições temporizadas estocásticas podem fazer uso de dois parâmetros de entrada, função de guarda (como apresentados na Tabela 10) e o tempo médio de atraso (ver Tabela 11). O tempo médio de atraso para as transições *TE0*, *TE1*, *TE2* utilizam os valores obtidos através do protótipo e os dados para as transições de funcionamento de cada componente da infraestrutura *IoT_Repair*, *IoT_Fail*, *Fog_Repair*, *Fog_Fail*, *Cloud_Repair*, *Cloud_Fail* foram conseguidos através dos modelos RBDs (Seção 5.1).

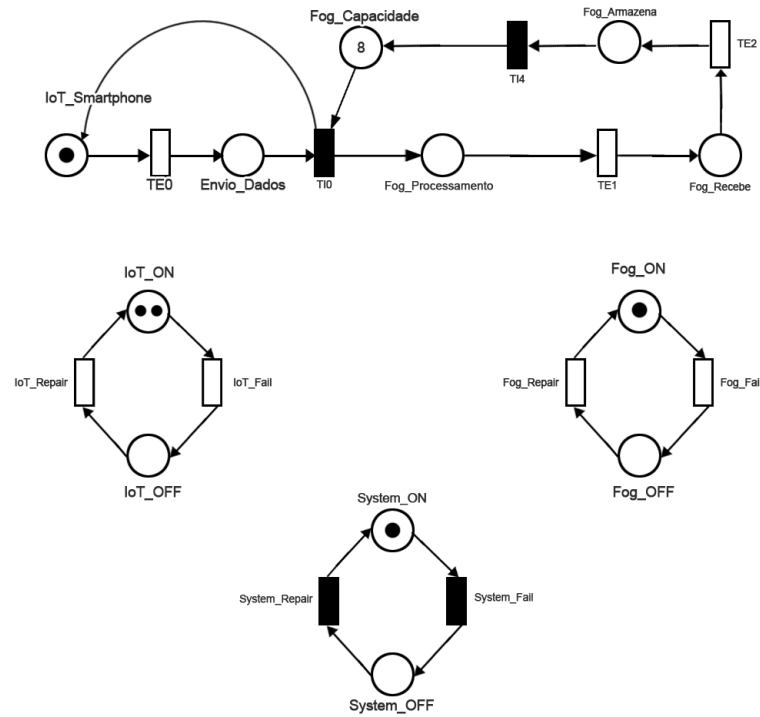
Tabela 11 – Transições temporizadas em horas: cenários 1 e 3

Transição	Cenário 1	Cenário 3
<i>TE0</i>	0,000277	0,000277
<i>TE1</i>	0,000578	0,000147
<i>TE2</i>	0,000585	0,000137
<i>IoT_Repair</i>	0,0911	0,0911
<i>IoT_Fail</i>	272,594	272,594
<i>Fog_Repair</i>	1,57377	1,57377
<i>Fog_Fail</i>	1105,86062	1105,86062
<i>Cloud_Repair</i>	0,8168	0,8168
<i>Cloud_Fail</i>	415,641	415,641

Fonte: Autor (2020)

O modelo que representa os outros cenários 2 e 4 dos experimentos é apresentado na Figura 22; na qual a disponibilidade é dependente dos dois componentes do sistema (*IoT* e *fog*) e, caso algum dos dois venham a falhar, ou seja, *places IoT_OFF*, *Fog_OFF*, receberem um *token* o sistema é considerado indisponível. Assim como no modelo anterior, as transições temporizadas estocásticas *TE0* e *TE1*, fazem uso dos valores obtidos através do protótipo e apresentados na Tabela 12. Todas as transições do sistema tem a função de guarda apresentadas na Tabela 13, a *fog* nesses cenários executa a tarefa de processamento e armazenamento dos

Figura 22 – Modelo dos experimentos: cenários 2 e 4



Fonte: Autor, 2020.

dados, sendo assim a transição *TE1* terá como tempo a soma das tarefas de cada cenário. Os dados para as transições de funcionamento de cada componente da infraestrutura *IoT_Repair*, *IoT_Fail*, *Fog_Repair* e *Fog_Fail*, permanecem os mesmos dos cenários anteriores.

Tabela 12 – Transições temporizadas em horas: cenários 2 e 4

Transição	Cenário 2	Cenário 4
<i>TE0</i>	0,000277	0,000277
<i>TE1</i>	0,0013124	0,00030
<i>IoT_Repair</i>	0,0911	0,0911
<i>IoT_Fail</i>	272,594	272,594
<i>Fog_Repair</i>	1,57377	1,57377
<i>Fog_Fail</i>	1105,86062	1105,86062

Fonte: Autor (2020)

Tabela 13 – Função de guarda: cenários 2 e 4

Transição	Função de Guarda
<i>TE0</i>	$\#IoT_ON > 0$
<i>TI0</i>	$\#Fog_ON > 0$
<i>TE1</i>	$\#Fog_ON > 0$
<i>TI2</i>	$\#Fog_OFF = 1$

Fonte: Autor (2020)

A presença de um *token* no *place IoT_ON* significa que o dispositivo IoT está em funcionamento, quando o dispositivo IoT falha, o *token* do *place IoT_ON* é transportado através

da transição *IoT_FAIL*, para o *place IoT_OFF* sinalizando assim a falha no dispositivo. Quando o dispositivo IoT é reparado após o tempo de MTTR (ver Tabela 12), o *token* do *place IoT_OFF* é enviado pela transição *IoT_REPAIR* até o *place IoT_ON*. O mesmo se aplica a todos os *building blocks* da *fog* e do *system*.

No *building block* do *system*, as transições são imediatas, e para serem ativadas, elas dependem de uma função de guarda, que é apresentada na Tabela 14, que sinaliza as condições de falha e reparo do sistema.

Tabela 14 – Função de guarda de funcionamento do sistema: cenários 2 e 4

Experimentos	Transições	Funções de guarda
2 e 4	<i>System_FAIL</i>	$(\#IoT_ON = 0) \text{OR} (\#Fog_ON = 0)$
	<i>System_REPAIR</i>	$(\#IoT_ON > 0) \text{AND} (\#Fog_ON = 1)$

Fonte: Autor (2020)

Quando um dos dispositivos falha, a transição *System_FAIL* consome o *token* do *place System_ON*, e produz um *token* no *place System_OFF*, sinalizando que o sistema está em falha. O sistema é reparado quando ambos os dispositivos estão em pleno funcionamento, ou seja, é necessário que tenha pelo menos um *token* nos *places IoT_ON* e *Fog_On*.

5.3 Métricas

As métricas escolhidas para avaliação dos modelos são apresentadas logo abaixo utilizando como exemplo os componentes dos modelos apresentados na Figura 21, referente aos cenários 1 e 3. As funções para obter os resultados foram retirados da literatura ((SANTOS et al., 2018), (MACIEL et al., 2010) e (RODRIGUES et al., 2019)):

- **Disponibilidade:** É a capacidade do sistema prover serviços sem interrupção. (MACIEL et al., 2010), descreve como a probabilidade de um determinado sistema estar em condições plena de funcionamento, a Equação 5.1 apresenta a fórmula utilizada para mensurar a disponibilidade onde é a probabilidade de todos os *building blocks* dos componentes dos sistema (*IoT*, *fog* e *cloud*) estejam em funcionamento;

$$Disponibilidade = P\{(\#IoT_ON > 0) \text{AND} (\#Fog_ON > 0) \text{AND} (\#Cloud_ON > 0)\} \quad (5.1)$$

- **Throughput (TP):** É a taxa de solicitação que pode ser atendida pelo sistema; O *Throughput* tende a aumentar à medida que aumenta a carga do sistema. Para o experimento essa métrica é calculada através da probabilidade de ter *tokens* em processamento (*fog* e *cloud*) e que o dispositivo esteja em pleno funcionamento multiplicado pelo o tempo médio

de serviço de cada dispositivo (obtidos através dos experimentos), como apresentado na Equação 5.2.

$$TP = \frac{P\{(\#Cloud_Processamento > 0)AND(\#Cloud_ON > 0)\} * (1/TimeCloud) + P\{(\#Fog_Processamento > 0)AND(\#Fog_ON > 0)\} * (1/TimeFog)}{(5.2)}$$

- **Tempo de Serviço (TS):** É o intervalo entre a solicitação de um usuário e a resposta do sistema, o cálculo dessa métrica é considerado que o tempo de serviço é uma distribuição exponencial (YANG et al., 2013) (KHAZAEI et al., 2011); para o nosso sistema podemos entender como Tempo de Serviço entre o envio do JSON do aplicativo até o armazenamento da mensagem. Essa medição é realizada através da Equação 5.3, que pode ser dividida pela quantidade de *places* envolvidos, para esse caso será dividido em duas partes *fog* e *cloud*:

$$TS = \frac{(E\{\#Fog_Processamento\} / (P\{\#Fog_Processamento > 0\} * (TimeFog))) + (E\{\#Cloud_Processamento\} / (P\{\#Cloud_Processamento > 0\} * (TimeCloud)))}{(5.3)}$$

- **Probabilidade de Descarte (PD):** É a probabilidade de que exista uma solicitação (*token*) em um *place* porém, não existe recursos disponíveis para atender a solicitação; a probabilidade do aplicativo enviar os dados para *fog* e serem descartados por não ter recursos disponíveis na fila da *fog*. A Equação 5.4 é a probabilidade de envio de dados vezes a probabilidade de não ter recurso disponível, multiplicado por 100.

$$PD = \frac{((P\{\#Envio_Dados = 1\}) * P\{\#Fog_Processamento = 0\}) * 100 + ((P\{\#Fog_Enviar = 1\}) * P\{\#Cloud_Processamento = 0\}) * 100}{(5.4)}$$

- **Requisições em Processos (RP):** É o número de solicitações em andamento no sistema, ou seja, é a soma da esperança de que tenha um JSON sendo classificado mais a esperança de que tenha outro JSON sendo armazenado, ambas as tarefas acontecendo simultaneamente; o cálculo pode ser realizado através da Equação 5.5, onde tem-se a esperança de ter *token* no *place Fog_Processamento* somado a esperança de ter um *token* no *place Cloud_Processamento* ;

$$RP = (E\{\#Fog_Processamento\}) + (E\{\#Cloud_Processamento\}) \quad (5.5)$$

5.4 Resultados

5.4.1 Análise de disponibilidade e desempenho

As análises estacionárias foram realizadas com os modelos de desempenho apresentados na subseção 5.2. O tempo de serviço usado na análise foi configurado com os dados obtidos através dos experimentos com o protótipo visto na Tabela 7, enquanto o tempo de chegada das solicitações foi definido em um segundo, de acordo com a configuração do modelo de *deep learning*.

Tabela 15 – Resultados dos modelos SPN

Cenários	Disponibilidade	Throughput (Requisições por ano)	Tempo de Serviço (em segundos)	Probabilidade de Descarte	Requisições em Processos
Cenário 1	0,99813	2826,84906	4,932	2,87450E-4	4,19021
Cenário 2	0,99858	760,85536	5,512	75,20705	7,93184
Cenário 3	0,99865	3933,46816	0,648	1,29061E-8	1,02370
Cenário 4	0,99835	3044,90827	1,120	4,46655	5,17988

Fonte: Autor (2020)

É interessante enfatizar que nos cenários 2 e 4 a infraestrutura tem um componente a menos o que influencia diretamente na disponibilidade do sistema, uma vez que foi retirado um dispositivo que poderia falhar em algum momento durante os experimentos. No entanto, mesmo com um componente a mais os cenários 1 e 3 obtiveram bons resultados. O Cenário 3 obteve o maior valor de disponibilidade 99,86%, o que pode ser explicado pela a robustez do dispositivo de *cloud*.

O *Throughput* apresenta os maiores valores com os cenários que fazem uso da *Raspberry Pi III*, o valor mais expressivo chega a ser cinco vezes maior que o menor valor, Cenário 3 e Cenário 2, respectivamente.

Ao analisar o tempo de serviço dos quatro cenários é possível identificar que a *Raspberry Pi II* mesmo não utilizando por completo todo o recursos de *hardware* disponível tem um tempo de serviço maior comparado a *Raspberry Pi III*, como supracitado o tempo de resposta para sistemas *e-health* é algo essencial.

Verificando as requisições em processos é possível identificar que com a utilização do modelo *Raspberry Pi II* para dispositivo de *fog* acontece fila no processamento dos dados, já nos cenários que utilizam o modelo *Raspberry Pi III*, esses valores são bem menores e no Cenário 3 chegam bem próximo a apenas uma requisição por vez, ou seja, o tempo médio de processamento é bem próximo a taxa de chegada.

Outro fator relevante a ser analisado é a probabilidade de descarte onde no Cenário 2 a taxa é bastante alta se comparada aos outros três cenários, acarretando assim uma grande quantidade de descartes dos dados enviados pelo aplicativo. O que para sistemas de *e-health* é algo preocupante uma vez que, as requisições que são descartadas poderiam conter informações cruciais sobre o estado do paciente.

Analisando os dados obtidos através dos modelos SPN, pode-se concluir que o modelo que tem melhores resultados de performance é o *Raspberry Pi III* e o cenário que obteve os melhores resultados na maioria das métricas foi o cenário 3, obtendo o maior *Throughput*, os menores tempo de serviço, probabilidade de descarte e requisições em processos.

5.4.2 Análise de custo

A Tabela 16 apresenta o valor de três lojas *online* que vendem os dispositivos *Raspberry Pi II* e *Raspberry Pi III*, que foram utilizados nos experimentos, são elas Allied¹, Microcenter² e Newark³.

Tabela 16 – Valores de dispositivos da *fog* em dólar

Lojas	Raspberry Pi II	Raspberry Pi III
Allied	\$35,00	\$34,99
Microcenter	\$29,99	\$24,99
Newark	\$42,31	\$35,00

Fonte: Autor (2020)

A Tabela 17 apresenta o valor mensal em dólar de três empresas *online* que comercializam plataformas de armazenamento em servidores virtuais privados (do inglês, *Virtual Private Server* - VPS) de mesma configuração. Para realizar a cotação dos valores foi escolhido um pacote mensal de 1Tb de armazenamento nas empresas Amazon⁴, Microsoft Azure⁵ e Google Cloud⁶. É interessante ressaltar que todas as plataformas pesquisadas disponibilizam armazenamento gratuito em pequena escala.

Tabela 17 – Valores do serviço de armazenamento (*cloud*) em dólar

Lojas	Cloud (por mês)
Amazon	\$23,00
Azure	\$25,00
Google Cloud	\$22,00

Fonte: Autor (2020)

O presente trabalho considera que o paciente já possui um *smartphone*, e por isso não foi realizado orçamento para o mesmo. Uma vez que um dos objetivos propostos é a construção de um sistema com bons resultados de disponibilidade e custo, sendo o dispositivo recomendado para *fog* é a *Raspberry Pi III* modelo B V1.2 com custo de \$24,99⁷, além de ter um preço mais

¹ <https://www.alliedelec.com/raspberry-pi-boards-view-all/>

² <https://www.microcenter.com/brand/4294818256/raspberry-pi>

³ <https://www.newark.com/c/embedded-computers-education-maker-boards/raspberry-pi/embedded-single-board-computers-sbc-raspberry-pi>

⁴ <https://aws.amazon.com/pt/s3/pricing/>

⁵ <https://azure.microsoft.com/en-us/services/storage/disks/>

⁶ <https://cloud.google.com/storage/pricing>

⁷ <https://www.microcenter.com/brand/4294818256/raspberry-pi>

acessível obteve os melhores resultados de performance e bons resultados de disponibilidade. Para a *cloud* a empresa que tem os melhores valores é a Google Cloud com custo de \$22,00 por mês, sendo esse o único custo recorrente para manter o sistema em funcionamento.

5.5 Considerações finais

A medição de um sistema real, simulação e modelagem analítica são as três técnicas fundamentais existentes para a avaliação de desempenho de sistemas (JAIN, 1990). Neste capítulo foi realizada a medição de um sistema real, protótipo, com dados de entrada para os modelos computacionais.

Foi apresentado os modelos RBD e SPN de um sistema *e-health*, descrição das métricas de avaliação dos modelos e discussão dos resultados obtidos. Por fim, foi realizada uma análise de custo para os dispositivos de *fog* e *cloud* idealizando a melhor performance com o menor custo possível.

Como resultados das duas análises o dispositivo para *fog* que obteve os melhores resultados e tem o menor custo é a *Raspberry Pi III* modelo B e o serviço de *cloud* recomendado é o Google Cloud.

6 Conclusão

O aumento da expectativa de vida da população mundial é uma realidade desafiadora, principalmente no que tange melhores condições de vida e saúde para os idosos (NATIONS, 2018), (ORGANIZATION, 2018). Em paralelo a esse aumento tem-se evidenciado a popularização dos dispositivos móveis em todo o planeta, mais de 5,1 bilhões de pessoas usam algum tipo de *smartphone*, equivalendo a cerca de 67% da população mundial (STRYJAK; SIVAKUMARAN, 2019). Como consequência dessa realidade e da necessidade iminente de aprimoramento no acompanhamento do bem estar de idosos, pesquisas do setor tecnológico, estão sendo impulsionadas para possibilitar o monitoramento de forma remota e principalmente, continua de pacientes.

Sistemas *e-health* que usam como porta de entrada dados de dispositivos IoT sofrem com algumas limitações, dentre elas destacam-se a limitação de armazenamento e processamento, a disponibilidade e o desempenho do sistema. Portanto, a fim de solucionar essas limitações a *fog* e *cloud* são inseridas na infraestrutura do sistema com intuito de garantir armazenamento e processamento ilimitados, além de melhorar o desempenho e a disponibilidade do sistema.

No presente trabalho, foi apresentado a implementação de um protótipo de sistema *e-health* que utiliza um modelo de *deep learning* para classificação de atividades diárias com base em informações coletadas de pacientes e tem como infraestrutura IoT, *fog* e *cloud*. Através do protótipo foi possível realizar experimentos em quatro cenários distintos, alternando o dispositivo da *fog* e retirando a *cloud* da infraestrutura. A partir dos resultados dos experimentos foram criados modelos SPN para que por meio deles seja possível analisar o funcionamento de cada cenário, para avaliação além da disponibilidade foi escolhido métricas de performance: *throughput*, tempo de serviço, probabilidade de descarte e requisições em processos.

Analizando a disponibilidade dos sistemas, todos os quatro cenários obtiveram valores acima de 99,8%, a diferença dentre o menor e maior valor de disponibilidade foi de 0,00052%. Em relação ao desempenho o cenário 3 composto pelo o dispositivo IoT, a *Raspberry Pi III* como dispositivo da *fog* e a *cloud* obteve o melhor resultado com a maior taxa de *throughput*, e os menores valores de probabilidade de descarte, requisições em processos e tempo de serviços.

O cenário 3 obteve o tempo de serviço de 0,65 segundos para executar toda as tarefas, o cenário 1 gastou 4,93 segundos para executar as mesmas tarefas, ambos os cenários são compostos pela a mesma infraestrutura, o que difere é o dispositivo da *fog* onde no cenário 1 é utilizado a *Raspberry Pi II*.

Em relação a análise de custo é considerado que o usuário já possui o dispositivo de IoT, sendo assim, é necessário apenas a aquisição do dispositivo de *fog* e contratação do serviço de *cloud*. Para *fog* o dispositivo que tem o menor valor de aquisição é a *Raspberry Pi III* custando

\$24,99 e a para serviço de *cloud* é o Google Cloud com mensalidade no valor de \$22,00.

Após a realização das análises de desempenho, disponibilidade e custo, o dispositivo que apresentou o melhor custo benefício para ser utilizado como *fog* foi a *Raspberry Pi* III modelo B V1.2, pois além de obter os melhores resultados de desempenho e disponibilidade, atualmente tem valor de compra menor que o outro modelo testado.

Como esperado, os cenários que utilizam a infraestrutura completa IoT, *fog* e *cloud* obtiveram melhores resultados quando analisado as métricas de tempo de serviço, probabilidade de descarte e requisições em processos. Se destacando o cenário 3 que tem custo inicial de \$46,99 para aquisição da *Raspberry Pi* III e assinatura mensal do Google Cloud, após o primeiro mês o custo fixo passa a ser de \$22,00.

6.1 Contribuições e produções científicas

As principais contribuições desta dissertação são:

1. Realização de uma revisão sistemática sobre como sistemas *e-health* integrado com IoT, *fog* e *cloud* estão sendo implementados e utilizados na literatura;
2. Implementação e disponibilização de um protótipo de sistema *e-health* integrado com IoT, *fog* e *cloud*;
3. Modelos estocásticos para realizar análise de disponibilidade e desempenho de sistema *e-health* integrado com IoT, *fog* e *cloud*;

Além dessas contribuições, esse trabalho apresenta grandes colaborações em produções científicas. A seguir são apresentadas duas tabelas contendo as produções, título, tipo e o status das mesmas. A Tabela 18 apresenta os artigos, os quais, o discente foi o primeiro autor ou a produção está diretamente relacionada a sua dissertação; já a Tabela 19 apresenta os artigos cuja a participação foi como colaborador.

A partir da definição do protótipo, dois artigos foram produzidos, sendo aceitos e publicados em congresso e conferência, sendo um deles na *Utility and Cloud Computing* (UCC) 2018 (Qualis A2) com título: *Developing an e-health system based on IoT, fog and cloud computing* (MONTEIRO et al., 2018), o qual apresenta uma arquitetura generalista de infraestrutura para sistemas *e-health*, e é feito um levantamento de aplicações IoT com foco em disponibilidade dos sistemas. O artigo citado ainda foi aceito para apresentação no *workshop Latin American Student Workshop on Data Communication Networks - 2019*¹, durante o Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos 2019.

¹ <http://sbrc2019.sbc.org.br/lancomm-student-workshop-2019/>

A segunda produção relevante para a pesquisa foi o artigo de título: *Accelerometer-based Human Fall Detection Using Convolutional Neural Networks* (SANTOS et al., 2019), onde apresenta um modelo de *deep learning* em um ambiente IoT e *fog* para detecção de quedas. Esse artigo foi aceito e publicado em um periódico (Qualis A1) e gerou o registro de programa de computador apresentado na Tabela 20, a continuação desse estudo resultou em outra publicação aceita em periódico: *Accelerometer-Based Human Fall Detection*.

Durante o processo de levantamento bibliográfico, foi identificado uma lacuna na literatura em relação a um modelo de arquitetura padrão para sistemas de *e-health* que a infraestrutura fosse composta por IoT, *fog* e *cloud*. Este processo resultou em um capítulo de livro contendo uma revisão sistemática da área, que foi produzido e aceito para publicação com título: *Internet of Medical Things (IoMT) applications in e-health systems context*.

Outro capítulo de livro construído e aceito para publicação tem com o título: *Living at the Edge? Optimising availability in IoT* (SANTOS et al., 2020), onde apresenta estratégias de tomada de decisão para maximizar a disponibilidade dos sistemas utilizando infraestrutura composta por IoT, *edge*, *fog* e *cloud*.

Tabela 18 – Produção diretamente relacionada com a dissertação

Título	Tipo	Status
<i>Developing an e-health system based on IoT, fog and cloud computing</i>	Conferência A2	Publicado
Desenvolvimento de uma arquitetura para sistemas <i>e-health</i> baseada em IoT, <i>fog</i> e <i>cloud</i>	Conferência	Publicado
<i>Accelerometer-based Human Fall Detection Using Convolutional Neural Networks</i>	Periódico A1	Publicado
<i>Internet of Medical Things (IoMT) applications in e-health systems context</i>	Capítulo de livro	Aceito para publicação
<i>Living at the Edge? Optimising availability in IoT</i>	Capítulo de livro	Publicado
<i>Accelerometer-Based Human Fall Detection</i>	Periódico	Publicado

Fonte: Autor (2020)

Tabela 19 – Produção como colaborador

Título	Tipo	Status
Detecting epileptic seizures using deep learning with cloud and fog computing	Conferência A2	Publicado
Detecção de crises epiléticas utilizando deep learning com fog e cloud computing	Conferência	Publicado
Optimizing Resource Availability in Composable Data Center Infrastructures	Conferência	Publicado
Prototipação de sistema e-health para avaliação de desempenho.	Conferência	Publicado
Medidas de distanciamento social e mobilidade na América do Sul durante a pandemia por COVID-19: Condições necessárias e suficientes?	Preprint	Publicado
StayHome: Monitoring and benchmarking social isolation trends in Caruaru and the Região Metropolitana do Recife during the COVID-19 pandemic	Revista	Publicado
FiqueEmCasa: Monitorando as tendências de isolamento social na cidade de Caruaru	Nota Técnica	Publicado Online
Modelando a curva de contágio da COVID-19 em Caruaru, Pernambuco	Nota Técnica	Publicado Online

Fonte: Autor (2020)

A Tabela 20 apresenta o título, número de registro e instituição de registro do programa de computador resultado do artigo (SANTOS et al., 2019).

Tabela 20 – Programa de computador

Título	Número de Registro	Instituição de Registro
Detecção de queda humana baseada em acelerômetro usando redes neurais convolucionais	BR5120200009994-8	INPI - Instituto Nacional da Propriedade de Industrial

Fonte: Autor (2020)

6.2 Limitações e dificuldades

Durante o trabalho foram observadas limitações literárias e desafios na implementação do protótipo, inicialmente pela grande complexidade do desenvolvimento do aplicativo e após essa etapa a configuração do modelo de *deep learning* que fosse otimizado para ser executado no dispositivo da *fog*.

Para os experimentos, o ambiente de rede foi isolado e sem conexão com a internet, para avaliar o desempenho de toda a infraestrutura. Foi assumido que a *cloud* é conectada localmente junto a infraestrutura. Em outras palavras, é utilizada *cloud* privada. A conexão de toda infraestrutura a internet poderá acarretar em diferentes valores de tempo de serviços, disponibilidade e desempenho do sistema se comparado com os valores obtidos em ambiente controlado, uma vez que o *smartphone* estará executando diversos aplicativos e a *cloud* será pública.

Por fim, para construção do modelo RBD da *fog* não foi encontrado na literatura os valores de MTTF e MTTR para aplicação de *deep learning*, assim considerando apenas os valores de *hardware* e sistema operacional para construção do modelo.

Referências

AHMADI, H. et al. The application of internet of things in healthcare: a systematic literature review and classification. *Universal Access in the Information Society*, Springer, p. 1–33, 2018. Citado 2 vezes nas páginas 20 e 71.

ALI, S.; GHAZAL, M. Real-time heart attack mobile detection service (rhamds): An iot use case for software defined networks. In: IEEE. *Electrical and Computer Engineering (CCECE), 2017 IEEE 30th Canadian Conference on*. [S.l.], 2017. p. 1–6. Citado 8 vezes nas páginas 20, 21, 22, 23, 24, 26, 31 e 71.

BAGULA, A. et al. Cloud based patient prioritization as service in public health care. In: IEEE. *ITU Kaleidoscope: ICTs for a Sustainable World (ITU WT), 2016*. [S.l.], 2016. p. 1–8. Citado 4 vezes nas páginas 20, 28, 32 e 71.

BAO, J.; YE, M.; DOU, Y. Mobile phone-based internet of things human action recognition for e-health. In: IEEE. *Signal Processing (ICSP), 2016 IEEE 13th International Conference on*. [S.l.], 2016. p. 957–962. Citado 9 vezes nas páginas 20, 21, 22, 23, 24, 25, 26, 32 e 71.

BAUSE, F.; KRITZINGER, P. S. *Stochastic petri nets*. [S.l.]: Citeseer, 2002. v. 1. Citado na página 34.

BHATIA, M.; SOOD, S. K. Exploring temporal analytics in fog-cloud architecture for smart office healthcare. *Mobile Networks and Applications*, Springer, p. 1–19, 2018. Citado 8 vezes nas páginas 20, 21, 22, 24, 25, 26, 31 e 71.

BIASOLI, D.; FONTOURA, L. M. Auxílio de redes de petri coloridas na implantação de projetos scrum. *Anais do VII Simpósio Brasileiro de Sistemas de Informação*, p. 498–502. Citado na página 32.

BUYYA, R.; DASTJERDI, A. V. *Internet of Things: Principles and paradigms*. [S.l.]: Elsevier, 2016. Citado na página 30.

CABRA, J. et al. An iot approach for wireless sensor networks applied to e-health environmental monitoring. In: IEEE. *2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*. [S.l.], 2017. p. 578–583. Citado 3 vezes nas páginas 20, 25 e 71.

CHARALAMPIDIS, P.; TRAGOS, E.; FRAGKIADAKIS, A. *A fog-enabled IoT platform for efficient management and data collection*. 2017. 1–6 p. Citado 5 vezes nas páginas 19, 20, 25, 27 e 71.

CHEN, X. et al. Thriftyedge: Resource-efficient edge computing for intelligent iot applications. *IEEE Network*, IEEE, v. 32, n. 1, p. 61–65, 2018. Citado 3 vezes nas páginas 20, 32 e 71.

COROTINSCHI, G.; GĂITAN, V. G. The use of iot technologies for providing high-quality medical services. In: IEEE. *System Theory, Control and Computing (ICSTCC), 2017 21st International Conference on*. [S.l.], 2017. p. 285–290. Citado 7 vezes nas páginas 19, 20, 21, 23, 24, 26 e 71.

- DANG, L. et al. Resource-efficient secure data sharing for information centric e-health system using fog computing. In: IEEE. *2018 IEEE International Conference on Communications (ICC)*. [S.l.], 2018. p. 1–6. Citado 3 vezes nas páginas 20, 32 e 71.
- DAS, S.; BALLAV, M.; KARFA, S. Application of iot in detecting health risks due to flickering artificial lights. In: IEEE. *Advances in Computing, Communications and Informatics (ICACCI), 2015 International Conference on*. [S.l.], 2015. p. 2331–2334. Citado 6 vezes nas páginas 20, 21, 23, 25, 26 e 71.
- DHANALIYA, U.; DEVANI, A. Implementation of e-health care system using web services and cloud computing. In: IEEE. *Communication and Signal Processing (ICCSP), 2016 International Conference on*. [S.l.], 2016. p. 1034–1036. Citado 7 vezes nas páginas 20, 21, 22, 23, 24, 26 e 71.
- EL-LATIF, A. A. A.; HOSSAIN, M. S.; WANG, N. Score level multibiometrics fusion approach for healthcare. *Cluster Computing*, Springer, p. 1–12, 2017. Citado 5 vezes nas páginas 20, 21, 23, 25 e 71.
- ELMISERY, A. M.; ABORIZKA, M. A new computing environment for collective privacy protection from constrained healthcare devices to iot cloud services. Citado 3 vezes nas páginas 20, 28 e 71.
- ENDO, P. T. et al. Highly available clouds: System modeling, evaluations, and open challenges. In: *Research Advances in Cloud Computing*. [S.l.]: Springer, 2017. p. 21–53. Citado na página 35.
- FAZIO, M. et al. Exploiting the fiware cloud platform to develop a remote patient monitoring system. In: IEEE. *2015 IEEE Symposium on Computers and Communication (ISCC)*. [S.l.], 2015. p. 264–270. Citado 3 vezes nas páginas 20, 24 e 71.
- FORCHE, R. Analysis of reliability block diagrams with multiple blocks per component. In: IEEE. *Annual Proceedings on Reliability and Maintainability Symposium*. [S.l.], 1990. p. 145–148. Citado na página 34.
- GERDES, M. et al. Reference design for smart collaborative telehealth and telecare services based on iot technologies. In: IEEE. *2015 International Conference on Computational Science and Computational Intelligence (CSCI)*. [S.l.], 2015. p. 817–820. Citado 2 vezes nas páginas 20 e 71.
- GHANAVATI, S. et al. Cloud-assisted iot-based health status monitoring framework. *Cluster Computing*, Springer, v. 20, n. 2, p. 1843–1853, 2017. Citado 9 vezes nas páginas 19, 20, 21, 22, 24, 26, 27, 31 e 71.
- GHOSH, A. M.; HALDER, D.; HOSSAIN, S. A. Remote health monitoring system through iot. In: IEEE. *2016 International Conference on Informatics, Electronics and Vision (ICIEV)*. [S.l.], 2016. p. 921–926. Citado 8 vezes nas páginas 20, 21, 22, 24, 25, 26, 31 e 71.
- GUIBERT, D. et al. Cc-fog: Toward content-centric fog networks for e-health. In: IEEE. *e-Health Networking, Applications and Services (Healthcom), 2017 IEEE 19th International Conference on*. [S.l.], 2017. p. 1–5. Citado 3 vezes nas páginas 19, 20 e 71.
- GUO, H.; YANG, X. A simple reliability block diagram method for safety integrity verification. *Reliability Engineering & System Safety*, Elsevier, v. 92, n. 9, p. 1267–1273, 2007. Citado na página 35.

- HA, M.; LINDH, T. Distributed performance management of internet of things as a service for caregivers. In: IEEE. *e-Health Networking, Applications and Services (Healthcom), 2017 IEEE 19th International Conference on*. [S.l.], 2017. p. 1–6. Citado 9 vezes nas páginas 20, 21, 22, 23, 24, 26, 31, 32 e 71.
- HASSAN, M. M. Cost-effective resource provisioning for multimedia cloud-based e-health systems. *Multimedia Tools and Applications*, Springer, v. 74, n. 14, p. 5225–5241, 2015. Citado 3 vezes nas páginas 20, 32 e 71.
- HUAXIN, S. et al. Constructing ideas of health service platform for the elderly. In: IEEE. *e-Health Networking, Applications and Services (Healthcom), 2012 IEEE 14th International Conference on*. [S.l.], 2012. p. 526–529. Citado 6 vezes nas páginas 20, 21, 22, 25, 26 e 71.
- IORGA, M. et al. *The NIST definition of fog computing*. [S.l.], 2017. Citado na página 30.
- IRFAN, M.; AHMAD, N. Internet of medical things: Architectural model, motivational factors and impediments. In: IEEE. *Learning and Technology Conference (L&T), 2018 15th*. [S.l.], 2018. p. 6–13. Citado 7 vezes nas páginas 13, 14, 20, 24, 27, 36 e 71.
- JAIN, R. *The art of computer systems performance analysis: techniques for experimental design, measurement, simulation, and modeling*. [S.l.]: John Wiley & Sons, 1990. Citado 2 vezes nas páginas 39 e 57.
- Jl, C. et al. Mobile cloud ecg intelligent monitoring and data processing system. In: IEEE. *e-Health Networking, Applications and Services (Healthcom), 2017 IEEE 19th International Conference on*. [S.l.], 2017. p. 1–6. Citado 9 vezes nas páginas 20, 21, 22, 23, 24, 26, 31, 32 e 71.
- JOSHI, J. et al. Health monitoring using wearable sensor and cloud computing. In: IEEE. *Cybernetics, Robotics and Control (CRC), International Conference on*. [S.l.], 2016. p. 104–108. Citado 7 vezes nas páginas 20, 21, 22, 24, 26, 32 e 71.
- KAROUI, K.; FTIMA, F. B. New engineering method for the risk assessment: Case study signal jamming of the m-health networks. *Mobile Networks and Applications*, Springer, p. 1–20, 2018. Citado 2 vezes nas páginas 20 e 71.
- KHAZAEI, H.; MISIC, J.; MISIC, V. B. Performance analysis of cloud computing centers using m/g/m/m+ r queuing systems. *IEEE Transactions on parallel and distributed systems*, IEEE, v. 23, n. 5, p. 936–943, 2011. Citado 2 vezes nas páginas 36 e 54.
- KNISS, J.; CUZIK, C.; RÜCK, D. B. Reduzindo a latência de aplicações web com fog computing. In: *Conferência: ERAD-SP-VII Escola Regional de Alto Desempenho-SP*. [S.l.: s.n.], 2016. Citado na página 30.
- KO, Y.-J. et al. A patient-centered medical environment with wearable sensors and cloud monitoring. In: IEEE. *Internet of Things (WF-IoT), 2015 IEEE 2nd World Forum on*. [S.l.], 2015. p. 628–633. Citado 9 vezes nas páginas 20, 21, 22, 23, 24, 26, 31, 32 e 71.
- KOROLYUK, V. S.; KOROLYUK, V. V. *Stochastic models of systems*. [S.l.]: Springer Science & Business Media, 2012. v. 469. Citado na página 35.
- KOTRONIS, C. et al. Managing criticalities of e-health iot systems. In: IEEE. *2017 IEEE 17th International Conference on Ubiquitous Wireless Broadband (ICUWB)*. [S.l.], 2017. p. 1–5. Citado 9 vezes nas páginas 14, 20, 21, 22, 26, 28, 36, 45 e 71.

- KOTRONIS, C. et al. A model-based approach for managing criticality requirements in e-health iot systems. In: IEEE. *2018 13th Annual Conference on System of Systems Engineering (SoSE)*. [S.l.], 2018. p. 60–67. Citado 6 vezes nas páginas 20, 21, 23, 24, 26 e 71.
- KYRIAZAKOS, S. et al. ewall: An open-source cloud-based ehealth platform for creating home caring environments for older adults living with chronic diseases or frailty. *Wireless Personal Communications*, Springer, v. 97, n. 2, p. 1835–1875, 2017. Citado 8 vezes nas páginas 20, 21, 22, 24, 26, 31, 32 e 71.
- LI, Y. et al. A health gateway for mobile monitoring in nursing home. *Wireless Personal Communications*, Springer, p. 1–15, 2018. Citado 7 vezes nas páginas 20, 21, 22, 24, 26, 31 e 71.
- MACIEL, P.; TRIVEDI, K.; KIM, D. Dependability modeling in: Performance and dependability in service computing: Concepts, techniques and research directions. *Hershey: IGI Global, Pennsylvania, USA*, v. 13, 2010. Citado 2 vezes nas páginas 34 e 53.
- MACIEL, P. R.; LINS, R. D.; CUNHA, P. R. *Introdução às redes de Petri e aplicações*. [S.l.]: UNICAMP-Instituto de Computacao, 1996. Citado 3 vezes nas páginas 32, 33 e 34.
- MASIP-BRUIN, X. et al. Fog-to-cloud computing (f2c): the key technology enabler for dependable e-health services deployment. In: IEEE. *Ad Hoc Networking Workshop (Med-Hoc-Net), 2016 Mediterranean*. [S.l.], 2016. p. 1–5. Citado 7 vezes nas páginas 20, 21, 22, 24, 25, 26 e 71.
- MATSUI, K.; CHOI, H. Temperature management system to prevent heat shock in households for elderly people. In: IEEE. *2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCOM/IOP/SCI)*. [S.l.], 2017. p. 1–6. Citado 7 vezes nas páginas 20, 21, 22, 25, 26, 32 e 71.
- MELL, P.; GRANCE, T. et al. The nist definition of cloud computing. Computer Security Division, Information Technology Laboratory, National ... , 2011. Citado na página 30.
- MENDONÇA, J. et al. Assessing performance and energy consumption in mobile applications. In: IEEE. *2015 IEEE International Conference on Systems, Man, and Cybernetics*. [S.l.], 2015. p. 74–79. Citado 2 vezes nas páginas 48 e 49.
- MIOTTO, R. et al. Deep learning for healthcare: review, opportunities and challenges. *Briefings in bioinformatics*, 2017. Citado 3 vezes nas páginas 13, 19 e 38.
- MOLLOY, M. K. Performance analysis using stochastic petri nets. *IEEE Transactions on computers*, IEEE, n. 9, p. 913–917, 1982. Citado na página 34.
- MONTEIRO, K. et al. Developing an e-health system based on iot, fog and cloud computing. In: IEEE. *2018 IEEE/ACM International Conference on Utility and Cloud Computing Companion (UCC Companion)*. [S.l.], 2018. p. 17–18. Citado 2 vezes nas páginas 38 e 59.
- NATIONS, G. A. U. *General Assembly United Nations*. 2018. <http://www.un.org/en/ga/search/view_doc.asp?symbol=A/RES/67/81>. Accessed: 2018-09-27. Citado 2 vezes nas páginas 13 e 58.

NEAGU, G. et al. A cloud-iot based sensing service for health monitoring. In: IEEE. *E-Health and Bioengineering Conference (EHB)*, 2017. [S.l.], 2017. p. 53–56. Citado 4 vezes nas páginas 13, 20, 32 e 71.

NETO, S. R. de A. et al. Detecting human activities based on a multimodal sensor data set using a bidirectional long short-term memory model: A case study. In: *Challenges and Trends in Multimodal Fall Detection for Healthcare*. [S.l.]: Springer, 2020. p. 31–51. Citado 3 vezes nas páginas 38, 42 e 44.

ORGANIZATION, W. H. *World Health Organization*. 2018. <<https://goo.gl/XAYvnq>>. Accessed: 2018-09-27. Citado 3 vezes nas páginas 13, 22 e 58.

PATEL, K. K.; PATEL, S. M. et al. Internet of things-iot: definition, characteristics, architecture, enabling technologies, application & future challenges. *International journal of engineering science and computing*, v. 6, n. 5, 2016. Citado na página 30.

PEREIRA, C. et al. Experimental characterization of mobile iot application latency. *IEEE Internet of Things Journal*, IEEE, v. 4, n. 4, p. 1082–1094, 2017. Citado 2 vezes nas páginas 20 e 71.

PESCOSOLIDO, L. et al. An iot-inspired cloud-based web service architecture for e-health applications. In: IEEE. *Smart Cities Conference (ISC2), 2016 IEEE International*. [S.l.], 2016. p. 1–4. Citado 3 vezes nas páginas 20, 25 e 71.

PETERSEN, K. et al. Systematic Mapping Studies in Software Engineering. In: *12th Int. Conference on Evaluation and Assessment in Software Engineering*. Italy: BCS Learning & Development Ltd., 2008. p. 10. Disponível em: <<http://dl.acm.org/citation.cfm?id=2227115.2227123>>. Citado 3 vezes nas páginas 16, 17 e 18.

PIYARE, R. et al. Integrating wireless sensor network into cloud services for real-time data collection. In: IEEE. *ICT Convergence (ICTC), 2013 International Conference on*. [S.l.], 2013. p. 752–756. Citado 7 vezes nas páginas 20, 21, 22, 25, 26, 27 e 71.

PLAGERAS, A. P. et al. Efficient large-scale medical data (ehealth big data) analytics in internet of things. In: IEEE. *Business Informatics (CBI), 2017 IEEE 19th Conference on*. [S.l.], 2017. v. 2, p. 21–27. Citado 3 vezes nas páginas 20, 27 e 71.

RADU, A. et al. Intercloud platform for connecting and managing heterogeneous services with applications for e-health. In: *2015 Conference Grid, Cloud High Performance Computing in Science (ROLCG)*. [S.l.: s.n.], 2015. p. 1–4. Citado 3 vezes nas páginas 20, 32 e 71.

RADWAN, A.; DOMINGUES, M. F.; RODRIGUEZ, J. Mobile caching-enabled small-cells for delay-tolerant e-health apps. In: IEEE. *Communications Workshops (ICC Workshops), 2017 IEEE International Conference on*. [S.l.], 2017. p. 103–108. Citado 8 vezes nas páginas 20, 21, 22, 24, 25, 26, 31 e 71.

RAHMATI, S. H. A. et al. A multi-objective model for facility location–allocation problem with immobile servers within queuing framework. *Computers & Industrial Engineering*, Elsevier, v. 74, p. 1–10, 2014. Citado na página 36.

RENNER, T.; KLIEM, A.; KAO, O. The device cloud-applying cloud computing concepts to the internet of things. In: IEEE. *Ubiquitous Intelligence and Computing, 2014 IEEE 11th Intl Conf on and IEEE 11th Intl Conf on and Autonomic and Trusted Computing, and IEEE*

14th Intl Conf on Scalable Computing and Communications and Its Associated Workshops (UTC-ATC-ScalCom). [S.l.], 2014. p. 396–401. Citado 6 vezes nas páginas 20, 21, 23, 26, 32 e 71.

RODRIGUES, L.; ENDO, P. T.; SILVA, F. A. Stochastic model for evaluating smart hospitals performance. In: IEEE. *2019 IEEE Latin-American Conference on Communications (LATINCOM)*. [S.l.], 2019. p. 1–6. Citado 2 vezes nas páginas 49 e 53.

SALES, A. H. C. de. Um estudo sobre redes de petri estocásticas generalizadas. 2002. Citado na página 34.

SANTOS, G. L. *Modeling the availability and performance of the integration between edge, fog and cloud infrastructures*. Dissertação (Mestrado) — Universidade Federal de Pernambuco, 2018. Citado 2 vezes nas páginas 36 e 40.

SANTOS, G. L. et al. Analyzing the availability and performance of an e-health system integrated with edge, fog and cloud infrastructures. *Journal of Cloud Computing*, Springer, v. 7, n. 1, p. 16, 2018. Citado 7 vezes nas páginas 14, 27, 31, 36, 48, 49 e 53.

SANTOS, G. L. et al. Accelerometer-based human fall detection using convolutional neural networks. *Sensors*, Multidisciplinary Digital Publishing Institute, v. 19, n. 7, p. 1644, 2019. Citado na página 60.

SANTOS, G. L.; MONTEIRO, K. H. d. C.; ENDO, P. T. Living at the edge? optimizing availability in iot. In: *The Cloud-to-Thing Continuum*. [S.l.]: Palgrave Macmillan, Cham, 2020. p. 79–94. Citado na página 60.

SAWAND, A. et al. Multidisciplinary approaches to achieving efficient and trustworthy ehealth monitoring systems. In: IEEE. *Communications in China (ICCC), 2014 IEEE/CIC International Conference on*. [S.l.], 2014. p. 187–192. Citado 2 vezes nas páginas 25 e 71.

SINGH, J.; BACON, J. M. On middleware for emerging health services. *Journal of Internet Services and Applications*, Springer, v. 5, n. 1, p. 6, 2014. Citado 3 vezes nas páginas 20, 28 e 71.

STRADOLINI, F. et al. Iot for telemedicine practices enabled by an androidTM application with cloud system integration. In: IEEE. *Circuits and Systems (ISCAS), 2018 IEEE International Symposium on*. [S.l.], 2018. p. 1–5. Citado 6 vezes nas páginas 20, 21, 22, 24, 26 e 71.

STRADOLINI, F. et al. Live demonstration: An iot cloud-based architecture for anesthesia monitoring. In: IEEE. *Circuits and Systems (ISCAS), 2018 IEEE International Symposium on*. [S.l.], 2018. p. 1–1. Citado 2 vezes nas páginas 20 e 71.

STRYJAK, J.; SIVAKUMARAN, M. The mobile economy 2019. *GSMA Intelligence*, 2019. Citado 3 vezes nas páginas 13, 14 e 58.

ULLAH, K.; SHAH, M. A.; ZHANG, S. Effective ways to use internet of things in the field of medical and smart health care. In: IEEE. *Intelligent Systems Engineering (ICISE), 2016 International Conference on*. [S.l.], 2016. p. 372–379. Citado 4 vezes nas páginas 20, 24, 25 e 71.

- VARGHEESE, R.; VINIOTIS, Y. Influencing data availability in iot enabled cloud based e-health in a 30 day readmission context. In: IEEE. *Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom), 2014 International Conference on*. [S.l.], 2014. p. 475–480. Citado 9 vezes nas páginas 14, 20, 21, 22, 26, 27, 28, 36 e 71.
- VERMA, A. K. et al. *Reliability and safety engineering*. [S.l.]: Springer, 2010. v. 43. Citado na página 35.
- WEBER, T. S. Tolerância a falhas: conceitos e exemplos. *Apostila do Programa de Pós-Graduação–Instituto de Informática-UFRGS. Porto Alegre*, p. 24, 2003. Citado na página 48.
- YAMIN, M. It applications in healthcare management: a survey. *International Journal of Information Technology*, Springer, p. 1–7, 2018. Citado 3 vezes nas páginas 20, 28 e 71.
- YANG, B.; TAN, F.; DAI, Y.-S. Performance evaluation of cloud service considering fault recovery. *The Journal of Supercomputing*, Springer, v. 65, n. 1, p. 426–444, 2013. Citado na página 54.
- YANG, B. et al. Performance evaluation of cloud service considering fault recovery. In: SPRINGER. *IEEE International Conference on Cloud Computing*. [S.l.], 2009. p. 571–576. Citado na página 36.
- YAO, L. et al. Wits: an iot-endowed computational framework for activity recognition in personalized smart homes. *Computing*, Springer, v. 100, n. 4, p. 369–385, 2018. Citado 9 vezes nas páginas 19, 20, 21, 23, 24, 25, 26, 28 e 71.

Apêndices

APÊNDICE A – Lista de artigos do mapeamento sistemático

Tabela 21 – Artigos do mapeamento sistemático

Título	Referência
Mobile caching-enabled small-cells for delay-tolerant e-Health apps	(RADWAN et al., 2017)
Exploiting the FIWARE cloud platform to develop a remote patient monitoring system	(FAZIO et al., 2015)
Implementation of e-health care system using web services and cloud computing	(DHANALIYA; DEVANI, 2016)
Reference design for smart collaborative telehealth and telecare services based on iot technologies	(GERDES et al., 2015)
Integrating wireless sensor network into cloud services for real-time data collection	(PIYARE et al., 2013)
Intercloud platform for connecting and managing heterogeneous services with applications for e-health	(RADU et al., 2015)
The device cloud-applying cloud computing concepts to the internet of things	(RENNER et al., 2014)
Resource-Efficient Secure Data Sharing for Information Centric E-Health System Using Fog Computing	(DANG et al., 2018)
Mobile cloud ECG intelligent monitoring and data processing system	(JI et al., 2017)
An IoT-inspired cloud-based web service architecture for e-health applications	(PESCOSOLIDO et al., 2016)
Temperature management system to prevent heat shock in households for elderly people	(MATSUI; CHOI, 2017)
eWALL: An Open-Source Cloud-Based eHealth Platform for Creating Home Caring Environments for Older Adults	(KYRIAZAKOS et al., 2017)
A Cloud-IoT based sensing service for health monitoring	(NEAGU et al., 2017)
Remote health monitoring system through IoT	(GHOSH et al., 2016)
Experimental characterization of mobile iot application latency	(PEREIRA et al., 2017)
ThriftyEdge: Resource-Efficient Edge Computing for Intelligent IoT Applications	(CHEN et al., 2018)
Health monitoring using wearable sensor and cloud computing	(JOSHI et al., 2016)
Distributed performance management of Internet of Things as a service for caregivers	(HA; LINDH, 2017)
A Patient-centered medical environment with wearable sensors and cloud monitoring	(KO et al., 2015)
Cloud based patient prioritization as service in public health care	(BAGULA et al., 2016)
Mobile phone-based internet of things human action recognition for E-health	(BAO et al., 2016)
Cost-effective resource provisioning for multimedia cloud-based e-health systems	(HASSAN, 2015)
Cloud-assisted IoT-based health status monitoring framework	(GHANAVATI et al., 2017)
Efficient large-scale medical data (ehealth big data) analytics in internet of things	(PLAGERAS et al., 2017)
New Engineering Method for the Risk Assessment: Case Study Signal Jamming of the M-Health Networks	(KAROU; FTIMA, 2018)
IT applications in healthcare management: a survey	(YAMIN, 2018)
On middleware for emerging health services	(SINGH; BACON, 2014)
A Health Gateway for Mobile Monitoring in Nursing Home	(LI et al., 2018)
Live Demonstration: An IoT Cloud-Based Architecture for Anesthesia Monitoring	(STRADOLINI et al., 2018b)
Managing criticalities of e-health iot systems	(KOTRONIS et al., 2017)
Effective ways to use Internet of Things in the field of medical and smart health care	(ULLAH et al., 2016)
Constructing ideas of health service platform for the elderly	(HUAXIN et al., 2012)
An IoT approach for Wireless Sensor Networks applied to e-health environmental monitoring	(CABRA et al., 2017)
Influencing data availability in IoT enabled cloud based e-health in a 30 day readmission context	(VARGHEESE; VINIOTIS, 2014)
The application of internet of things in healthcare: a systematic literature review and classification	(AHMADI et al., 2018)
Exploring Temporal Analytics in Fog-Cloud Architecture for Smart Office HealthCare	(BHATIA; SOOD, 2018)
Real-time heart attack mobile detection service (RHAMDS): An IoT use case for software defined networks	(ALI; GHAZAL, 2017)
The use of IoT technologies for providing high-quality medical services	(COROTINSCHI; GĂITAN, 2017)
CC-fog: Toward content-centric fog networks for E-health	(GUIBERT et al., 2017)
Fog-to-cloud Computing (F2C): the key technology enabler for dependable e-health services deployment	(MASIP-BRUIN et al., 2016)
IoT for Telemedicine Practices enabled by an Android™ Application with Cloud System Integration	(STRADOLINI et al., 2018a)
A new computing environment for collective privacy protection from constrained healthcare devices to IoT cloud services	(ELMISERY; ABORIZKA,)
Internet of medical things: Architectural model, motivational factors and impediments	(IRFAN; AHMAD, 2018)
A fog-enabled IoT platform for efficient management and data collection	(CHARALAMPIDIS et al., 2017)
A Model-based Approach for Managing Criticality Requirements in e-Health IoT Systems	(KOTRONIS et al., 2018)
Application of IoT in detecting health risks due to flickering artificial lights	(DAS et al., 2015)
Score level multibiometrics fusion approach for healthcare	(EL-LATIF et al., 2017)
WITS: an IoT-endowed computational framework for activity recognition in personalized smart homes	(YAO et al., 2018)
Multidisciplinary approaches to achieving efficient and trustworthy eHealth monitoring systems	(SAWAND et al., 2014)