UNIVERSIDADE FEDERAL DE SANTA CATARINA
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA

Bruno Machado Agostinho

SMART COMM: UM MIDDLEWARE PARA O SUPORTE À INTEROPERABILIDADE DE DISPOSITIVOS HETEROGÊNEOS EM SMART HOMES

Florianópolis

2019

Bruno Machado Agostinho

SMART COMM: UM MIDDLEWARE PARA O SUPORTE À INTEROPERABILIDADE DE DISPOSITIVOS HETEROGÊNEOS EM SMART HOMES

Proposta de Dissertação submetida ao Programa de Pós-Graduação em Ciência da Computação para a obtenção do Grau de Mestre.
Orientador: Prof. Dr. Alex Sandro Roschildt Pinto
Coorientador: Prof. Dr. Mario Antônio Ribeiro Dantas

Florianópolis

2019

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

Agostinho, Bruno Machado
Smart Comm: Um Middleware Para o Suporte à Interoperabilidade de Dispositivos Heterogêneos em Smart Homes / Bruno Machado Agostinho ; orientador, Alex Sandro Roschildt Pinto, coorientador, Mario Antônio Ribeiro Dantas, 2019.
77 p.

Dissertação (mestrado) - Universidade Federal de Santa Catarina, Centro Tecnológico, Programa de Pós Graduação em Ciência da Computação, Florianópolis, 2019.

Inclui referências.

1. Ciência da Computação. 2. Sistema de Computação. 3. Arquitetura de Computação. I. Pinto, Alex Sandro Roschildt. II. Dantas, Mario Antônio Ribeiro. III. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Ciência da Computação. IV. Título.

Bruno Machado Agostinho

SMART COMM: UM MIDDLEWARE PARA O SUPORTE À INTEROPERABILIDADE DE DISPOSITIVOS HETEROGÊNEOS EM SMART HOMES

Esta Dissertação foi julgada aprovada para a obtenção do Título de "Mestre", e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciência da Computação.

Florianópolis, 18 de Fevereiro 2019.

Prof. Dr. José Luís Almada Güntzel
Coordenador do Curso

Prof. Dr. Alex Sandro Roschildt Pinto
Orientador

Banca Examinadora:

Prof. Dr. Carlos Barros Montez

Prof. Dr. Douglas Dyllon Jeronimo de Macedo

Prof. Dr. Roberto Willrich

Este trabalho é dedicado aos meus Professores do PPGCC, meus orientadores, meus pais, minha namorada e ao Laboratório Bridge, que me permitiu dar o primeiro passo para me dedicar integralmente ao mestrado.

AGRADECIMENTOS

Ao Professor Mario Antônio Ribeiro Dantas pela confiança e oportunidade de trabalhar em conjunto, pelas aulas ministradas, pela experiência compartilhada no decorrer dos últimos 3 anos e principalmente pela paciência e compreensão das dificuldades por mim encontradas.

Ao Professor Alex Sandro Roschildt Pinto, pela contribuição em um momento decisivo deste trabalho, pelos recursos cedidos para viabilizar os experimentos e pela parceria que se inicia.

Ao Professor Douglas Dyllon Jeronimo de Macedo pelos feedbacks e conselhos dados na qualificação do meu mestrado, fazendo-o tomar outro rumo. E pela participação na banca avaliadora deste trabalho.

Aos Professores Roberto Willrich e Carlos Barros Montez, pela participação da banca avaliadora.

Aos Professores, Carlos Becker Westphall, Alexandre Gonçalves Silva, Vania Bogorny, Elder Rizzon Santos e Jean Martina, pelas aulas ministradas e conhecimento compartilhado.

Ao colega de Caravela, Cauê Baasch de Souza, pela imensa ajuda nos experimentos, viabilizando a finalização de todos os testes dentro do previsto.

Ao colega Gabriel Geraldelli, pela oportunidade que viabilizou minha dedicação integral ao mestrado e aos meus experimentos.

Ao colega Geomar Schreiner, pelas dicas e experiência compartilhada dentro do mundo acadêmico.

À meus pais, pelo apoio e por uma vida inteira de dedicação, possibilitando que eu pudesse superar mais esta etapa da minha vida.

E principalmente, agradeço à minha namorada, Fernanda Oliveira Gomes, pelo apoio e incentivo nas horas mais complicadas, pelas dicas e ajuda na pesquisa, fazendo com que eu seguisse sempre em frente, independente das dificuldades encontradas.

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

RESUMO

Nos últimos anos tem sido verificado um maior interesse na área acadêmica e indústria com relação a adoção da Internet of Things (IoT) e Smart Homes. Embora existam muitas abordagens relativas a esses ambientes, ainda não existe um consenso comum de tecnologias e nem uma padronização para comunicação entre dispositivos, coleta e processamento de dados gerados pelos mesmos. É baseado nestes desafios que este trabalho tem como proposta o middleware Smart Comm. Este cenário de software foi concebido para a viabilização de interoperabilidade entre dispositivos heterogêneos dentro de ambientes IoT, com ênfase em Smart Homes. O Smart Comm foi desenvolvido em uma arquitetura utilizando microsserviços e visando a utilização de um paradigma de Fog Computing para armazenamento e processamento dos dados. Foram realizados testes com dispositivos utilizando os protocolos Bluetooth LE, ZigBee e Wi-Fi, de maneira isolada e em conjunto para medir a interferência gerada. Os dados gerados pelo ambiente experimental mostraram que embora em alguns casos a média de tempo de resposta tenha crescido em até 26%, na maioria dos testes os valores encontrados nos testes isolados e de interoperabilidade permaneceram próximos. Com uma porcentagem de timeouts de 0,02% e de 0,014% de pacotes inválidos, considera-se viável a utilização o Smart Comm para um ambiente conforme o simulado.

Palavras-chave: Middleware, Smart Home, IoT, Interoperabilidade, Heterogeneidade

ABSTRACT

In the research it has been noticed a large interest in academia and industry related to the adoption of Internet of Things (IoT) and Smart Homes. Although there are many approaches to these environments, there is still no common consensus on technology and no standards for device-to-device communication, data collection and processing. It is based on these challenges that this work proposes the Smart Comm middleware. This software environment is designed to enable interoperability between heterogeneous devices within IoT environments, focused on Smart Homes. Smart Comm was developed in an architecture using microservices aiming the use of a Fog Computing paradigm for data storage and processing. Tests were performed with devices using the Bluetooth LE, ZigBee and Wi-Fi protocols, first isolated and then together to measure the interference generated. The data generated by the experimental environment showed that although in some cases the average response time increased by up to 26%, in most tests the values found in both scenarios remained close. With a timeouts percentage of 0.02% and 0.014% of invalid packets, the use of Smart Comm in a environment like the simulated one was considered feasible.

Keywords: Middleware, Smart Home, IoT, Interoperability, Heterogeneity

LISTA DE FIGURAS

Figura 1 Microsserviços x Monolítico. Fonte: (FOWLER, 2014) .. 27
Figura 2 Smart Comm. Fonte: (AGOSTINHO et al., 2018) 33
Figura 3 Gateway Central - Entidades..... 34
Figura 4 Gateway Secundário - Entidades..... 35
Figura 5 Diagrama de Sequência - Caminho 1..... 36
Figura 6 Topologia proposta 1..... 37
Figura 7 Topologia proposta 2..... 38
Figura 8 Arquitetura Smart Comm..... 44
Figura 9 Requisição a um microsserviço. Fonte: (AGOSTINHO et al., 2018)..... 48
Figura 10 Topologia Bluetooth..... 50
Figura 11 Bluetooth - Total de Detecções..... 51
Figura 12 Bluetooth - Média por Dispositivo..... 52
Figura 13 Bluetooth - Média do Tempo de Resposta..... 53
Figura 14 Topologia ZigBee..... 54
Figura 15 ZigBee - Total de Requisições..... 55
Figura 16 ZigBee - Média por Dispositivo..... 56
Figura 17 ZigBee - Média do Tempo de Resposta..... 57
Figura 18 Topologia Wi-Fi..... 58
Figura 19 Wi-Fi - Total de Requisições..... 59
Figura 20 Wi-Fi - Média por Dispositivo..... 59
Figura 21 Wi-Fi - Média do Tempo de Resposta..... 60
Figura 22 Topologia Interoperabilidade..... 61
Figura 23 Resultado Bluetooth - Total de Detecções..... 62
Figura 24 Resultado Bluetooth - Média por Dispositivo..... 63
Figura 25 Resultado Bluetooth - Média do Tempo de Resposta... 63
Figura 26 Resultado ZigBee - Total de Detecções..... 64
Figura 27 Resultado ZigBee - Média por Dispositivo..... 65
Figura 28 Resultado ZigBee - Média do Tempo de Resposta..... 65
Figura 29 Resultado Wi-Fi - Total de Detecções..... 66
Figura 30 Resultado Wi-Fi - Média por Dispositivo..... 67
Figura 31 Resultado Wi-Fi - Média do Tempo de Resposta..... 67

LISTA DE TABELAS

Tabela 1 Configuração dos Testes de Interoperabilidade 61

LISTA DE ABREVIATURAS E SIGLAS

IoT	Internet of Things	29
SOA	Arquitetura Orientada a Serviços	29
CoAP	Constrained Application Protocol	30
HTTP	HyperText Transfer Protocol	30
MQTT	Message Queue Telemetry Transport	30
BLE	Bluetooth Low Energy	30
USB	Universal Serial Bus	43

SUMÁRIO

1 INTRODUÇÃO 21
1.1 MOTIVAÇÃO 21
1.2 OBJETIVOS 22
1.2.1 Objetivos Específicos 22
1.3 PROBLEMA DE PESQUISA 23
1.4 METODOLOGIA 23
1.5 ORGANIZAÇÃO DO TRABALHO 24
2 SMART HOMES, FOG COMPUTING E MICROSSERVIÇOS 25
2.1 SMART HOMES 25
2.2 FOG COMPUTING 26
2.3 MICROSSERVIÇOS 27
3 INTERNET OF THINGS (IOT) 29
3.1 PROTOCOLOS IOT 30
3.1.1 Bluetooth Low Energy (BLE) e ZigBee 31
4 SMART COMM 33
4.1 TRABALHOS CORRELATOS 39
4.2 PROTÓTIPO 42
4.3 CONSIDERAÇÕES 44
5 AMBIENTE EXPERIMENTAL 47
5.1 CRIAÇÃO DE USUÁRIO, AUTENTICAÇÃO E CHAMADA A MICROSSERVIÇO 47
5.2 PROTOCOLOS ISOLADOS 48
5.2.1 Bluetooth LE 49
5.2.2 ZigBee 53
5.2.3 Wi-Fi 57
5.3 INTEROPERABILIDADE 60
5.3.1 Bluetooth LE 62
5.3.2 ZigBee 64
5.3.3 Wi-Fi 66
5.4 CONSIDERAÇÕES 68
6 CONCLUSÕES E TRABALHOS FUTUROS 69
REFERÊNCIAS 71
APÊNDICE A – Artigo Publicado 77

1 INTRODUÇÃO

1.1 MOTIVAÇÃO

Nos últimos anos, tem sido verificado um grande crescimento do número de dispositivos dentro do conceito de Internet of Things (IoT). Espera-se que essa seja uma das maiores revoluções desde a Internet, trazendo inúmeras oportunidades no mundo inteiro (NGU et al., 2017). O crescimento na IoT, também impulsiona algumas áreas correlatas, tais como VANETs, Smart Cities e Smart Homes. Este fato permite hoje em dia termos um conjunto de dispositivos cooperando para criar facilidades para os usuários, seja dentro de seu carro, de sua casa, ou em em contexto mais abrangente, como uma cidade. A variedade de tipos de dispositivos que podem ser integrados é muito grande, possibilitando uma atuação desde a área de saúde até uma casa totalmente controlada por smartphones.

Se, por um lado, a grande variedade de dispositivos pode trazer diversas facilidades para os usuários, a forma com que a heterogeneidade dos mesmos deve ser tratada ainda é um desafio. Além disso, existem questões em aberto relacionadas à melhor forma de comunicação entre os dispositivos, segurança de acesso à informação, armazenamento e utilização dos dados. Segundo (XIAO et al., 2014), dois dos maiores desafios de aplicações dentro do contexto de IoT são os dispositivos altamente heterogêneos e a cooperação entre milhões de dispositivos distribuídos.

A diversidade de tipos de dispositivos e protocolos torna muito difícil uma abordagem visando a padronização ou a utilização de todos eles. Levando em consideração que mais tipos de dispositivos e protocolos vêm surgindo, e analisando que a médio e longo prazo a tendência pode ser de ainda mais dificuldade, fica visível a falta de consenso sobre qual é a melhor forma de viabilizar a utilização de múltiplos protocolos de comunicação dentro de um ambiente IoT. Essa comunicação se faz necessária visto que cada vez mais tipos de aplicações vem surgindo, como é o caso dos trabalhos de (DUBEY et al., 2015; GIA et al., 2015), onde são propostas aplicações para monitoramento de saúde em Smart Homes. Em aplicações como essas os usuários não podem ficar suscetíveis a interferências entre dispositivos, pois nesse contexto um problema na aplicação pode acarretar em problemas mais graves, relacionados à saúde do usuário.

(BOTTARO; GÉRODOLLE, 2008) propuseram a utilização de uma

arquitetura de serviços para lidar com problemas de interoperabilidade. Embora a utilização de serviços seja uma abordagem promissora, uma arquitetura SOA tradicional pode reduzir a flexibilidade quanto às tecnologias adotadas. Já em (BEDHIEF; KASSAR; AGUILI, 2016), foi proposta a utilização de containers em conjunto com tecnologias SDN para lidar com uma rede de dispositivos heterogêneos, mas ainda não haviam realizado testes em relação a conectividade utilizando múltiplos protocolos. Em (RAHMAN; CHAKRABORTY, 2018) os autores propuseram a utilização de um gateway para tratar da heterogeneidade na comunicação entre dispositivos ZigBee e Bluetooth LE, enviando os dados para consumo na nuvem.

Existem diversos subproblemas e abordagens dentro do contexto de interoperabilidade e heterogeneidade em IoT, como a utilização de dispositivos com especificidades diferentes e utilizando múltiplos protocolos de comunicação. A proposta do Smart Comm apresenta uma abordagem de utilização de microsserviços para lidar com diferentes tipos de dispositivos e protocolos. Ao invés de uma abordagem SOA tradicional, preferiu-se utilizar microsserviços pela sua flexibilidade e independência em relação a outros serviços. Divergindo de algumas propostas apresentadas, o Smart Comm foi projetado para utilizar os dados coletados localmente, viabilizando a utilização de sua infraestrutura dentro do contexto de Fog Computing. Dentro do trabalho apresentado, o termo *Middleware* foi utilizado para denominar a arquitetura e abordagens propostas assim como o protótipo desenvolvido, diferente do conceito de middleware usualmente utilizado em outros trabalhos na área de ciência da computação.

1.2 OBJETIVOS

Este trabalho tem como objetivo o desenvolvimento de um middleware para Smart Homes que seja capaz de lidar com os atuais desafios em relação a heterogeneidade e interoperabilidade de dispositivos.

1.2.1 Objetivos Específicos

1. Desenvolver gateway destinado ao controle de dispositivos IoT.

2. Desenvolver gateway para centralização dos dados e cadastro de dispositivos disponíveis.

3. Desenvolver microsserviços para validar a utilização de múltiplos protocolos.
4. Validar a utilização do Smart Comm através da utilização de tipos diferentes de dispositivos e protocolos.

1.3 PROBLEMA DE PESQUISA

Este trabalho busca responder a seguinte pergunta: **É possível viabilizar a interoperabilidade de dispositivos heterogêneos em ambientes IoT como Smart Homes?**

1.4 METODOLOGIA

- (a) Levantamento bibliográfico sobre o estado da arte e identificação de trabalhos correlatos
- (b) Validação do uso de microsserviços para diferentes tipos de dispositivos.
- (c) Desenvolvimento do gateway secundário para comunicação com dispositivos de borda utilizando microsserviços.
- (d) Desenvolvimento do gateway principal para integração entre os secundários.
- (e) Validação das funcionalidades de acesso ao gateway principal e listagem dos gateway secundários e seus respectivos dispositivos.
- (f) Simulação da chamada de dispositivos.
- (g) Escrita e publicação de artigo com os resultados preliminares do trabalho.
- (h) Qualificação do mestrado.
- (i) Redefinição do escopo do trabalho.
- (j) Nova Revisão da literatura.
- (k) Desenvolvimento de ambiente experimental com dispositivos heterogêneos.
- (l) Validação do uso do Smart Comm.

- (m) Escrita de artigo com os resultados finais.
- (n) Escrita da dissertação.
- (o) Defesa da dissertação.

1.5 ORGANIZAÇÃO DO TRABALHO

A estrutura deste trabalho foi organizada de modo a tentar prover um bom fluxo de leitura e entendimento. Inicia-se o trabalho utilizando os dois primeiros capítulo para apresentar alguns conceitos relacionados à proposta. No capítulo 4 é apresentada a proposta do Smart Comm, detalhando todos os componentes e o protótipo desenvolvido. Ainda neste capítulo, são apresentados algum trabalhos relacionados de propostas anteriores que visam solucionar problemas referentes a interoperabilidade ou que utilizaram algum tipo de abordagem semelhante ao que está sendo proposto. Na seção subsequente são mostrados os resultados obtidos através do ambiente experimental e as considerações sobre os resultados. O trabalho se encerra apresentando as conclusões e trabalhos futuros.

2 SMART HOMES, FOG COMPUTING E MICROSSERVIÇOS

2.1 SMART HOMES

Segundo (ALAM; REAZ; ALI, 2012), uma Smart Home pode ser definida como uma aplicação de computação ubíqua onde o ambiente é monitorado por algum tipo de inteligência a fim de fornecer serviços orientados a contexto e facilitar o controle da casa pelos usuários. Já para (GAIKWAD; GABHANE; GOLAIT, 2015), são casas ou ambientes domésticos que tem tecnologia para permitir que todos os dispositivos sejam controlados automaticamente e remotamente. No futuro, as smart homes controlarão muitos aspectos dentro de uma rotina diária. Como exemplo, podemos ter sensores de controle de luz, temperatura, entre outros. Deverão também aprender os hábitos dos seus usuários, conseguindo uma eficiência maior no consumo de energia e gerando um maior conforto (SOUZA; AMAZONAS, 2013).

Entre diversos tipos de aplicações voltadas para smart homes, sistemas voltados para a saúde dentro de casa, como os trabalhos de (HELAL; COOK; SCHMALZ, 2009) e (LIU et al., 2016), têm sido propostos de maneira a utilizar a tecnologia que está sendo disponibilizada para monitorar e prover funcionalidades, principalmente no cuidado de pessoas mais velhas. Sistemas esses que necessitam utilizar diversos tipos de dispositivos e que não podem estar sujeitos a interferências.

Ainda existem diversos desafios a serem resolvidos dentro dessa área. A interoperabilidade de dispositivos heterogêneos é um deles. Em um ambiente que promete ser totalmente controlado remotamente, com boa parte automatizada, o que não faltam são tipos de dispositivos diferentes. Além do aumento no conforto, a integração dessa diversidade de dispositivos pode gerar uma grande complexidade.

Outro desafio que pode ser listado é o armazenamento e processamento de dados. Boa parte dos dispositivos utilizados em smart homes podem gerar dados. Estes, podem ser armazenados localmente ou enviados para algum lugar, como por exemplo, um ambiente na nuvem. Ainda não existe uma padronização no sentido de como os dados serão tratados e utilizados. Novos paradigmas, como Fog Computing, vêm surgindo como alternativa às arquiteturas tradicionais de armazenamento em nuvem, podendo trazer abordagens diferentes.

2.2 FOG COMPUTING

Fog Computing pode ser descrita como uma grande quantidade de dispositivos heterogêneos, descentralizados e conectados de maneira ubíqua, que potencialmente cooperam entre si para fornecer serviços e atividades sem intervenção de terceiros (VAQUERO; RODERO-MERINO, 2014). Já para (BONOMI et al., 2012), Fog Computing é uma plataforma altamente virtualizada que provê computação, armazenamento e serviços de rede em uma camada localizada entre os dispositivos finais e a de computação em nuvem.

O paradigma de Fog Computing tem como objetivo trazer o processamento para a borda da rede, utilizando diversos dispositivos heterogêneos conectados a fim de realizar o processamento. Fog Computing é tido como o paradigma do futuro para a utilização de Internet das Coisas e de Cidades Inteligentes, pois oferece uma estrutura escalável. Os dispositivos que interagem entre si são chamados de Fog Nodes, e podem ser desde sensores até veículos inteligentes. Um dispositivo nodo utilizado em Fog deve aplicar a abordagem de localização consciente (VAQUERO; RODERO-MERINO, 2014), sendo capaz de inferir sua localização e de descobrir dispositivos próximos a ele. Uma rede de Fog Nodes é considerada fortemente distribuída geograficamente falando.

Dentro do contexto de IoT, segundo (DASTJERDI; BUYYA, 2016), Fog Computing vem para solucionar problemas relacionados a serviços oferecidos pela computação em nuvem e que também não puderam ser resolvidos trazendo todo o processamento para a borda da rede (Edge Computing), devido a baixa capacidade de processamento dos dispositivos. Sendo assim, cria-se uma camada intermediária, solucionando questões relacionadas a latência e processamento mais próximo ao usuário final.

Além da utilização de Fog Computing em um ambiente de Smart Homes, como o que está sendo proposto neste trabalho, esse paradigma está sendo utilizada em variados tipos de aplicações. (DUBEY et al., 2015) e (GIA et al., 2015) propuseram a utilização de uma arquitetura de Fog Computing para processamento de dados de saúde localmente, (HOU et al., 2016) propôs a utilização em computação veicular e (TANG et al., 2015) propôs uma arquitetura hierárquica dentro de um contexto de smart cities. Por ser uma área de pesquisa recente, diversos tipos de aplicações ainda estão sendo propostos.

2.3 MICROSSERVIÇOS

Segundo (DRAGONI et al., 2017), microsserviços podem ser definidos como processos independentes interagindo através de mensagens. E uma arquitetura de microsserviços é uma aplicação distribuída, onde todos os módulos são microsserviços. Já para (THöNES, 2015), um microsserviço é um pequeno aplicativo, de responsabilidade única, que pode ser implantado, dimensionado e testado de forma independente. E para (FOWLER, 2014), arquitetura de microsserviços é uma abordagem para desenvolver um único aplicativo como um conjunto de pequenos serviços, cada um executando em seu próprio processo e comunicando-se com mecanismos leves.

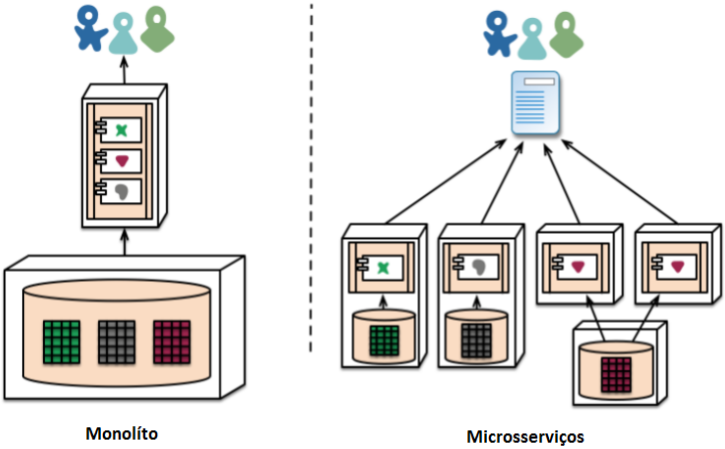

Figura 1 – Microsserviços x Monolítico. Fonte: (FOWLER, 2014)

A Figura 1 mostra a diferença entre um sistema monolito e um sistema utilizando uma arquitetura de microsserviços. No primeiro (esquerda) todos os módulos do sistema fazem parte de um componente maior, se conectando a um mesmo banco de dados. Na arquitetura com microsserviços, cada módulo se torna independente, tendo sua própria base de dados (direita). Um dos princípios de sua utilização é que um microsserviço não pode interferir na execução do outro. Em outras palavras, em qualquer problema relacionado a um serviço, o mesmo deve ficar isolado, e por mais que este serviço deixe de funcionar, os outros permanecem inalterados.

Além das características citadas anteriormente, um sistema utilizando microsserviços traz bastante flexibilidade em relação a tecnologias utilizadas. Uma vez que cada serviço acessa sua própria base de dados, essa arquitetura viabiliza o uso de múltiplos banco de dados de maneira isolada, permitindo que cada funcionalidade de adéque ao paradigma que seja mais apropriado.

3 INTERNET OF THINGS (IOT)

O crescimento do número de dispositivos conectados na Internet nos últimos anos e a interação dos mesmos entre si, onde tudo pode estar conectado e considerado um objeto capaz de interagir é chamado de Internet of Things (IoT) . O conceito de IoT permite que dispositivos vejam, escutem e interajam entre si para desempenhar atividades, compartilhar informações e coordenar decisões (AL-FUQAHA et al., 2015). Segundo (ATZORI; IERA; MORABITO, 2010), Internet of Things é um paradigma inovador que vem crescendo rapidamente e que consiste na ideia da presença pervasiva de diversos objetos, ou coisas (*things*) ao nosso redor, sejam eles sensores, atuadores, smartphones, entre outros.

Espera-se que até o final do ano de 2020, o número de dispositivos inteligentes no mundo alcance a quantidade de 212 bilhões (AL-FUQAHA et al., 2015). Ainda assim, a forma com que os mesmos se comunicam, coletam dados, armazenam e a segurança em cima dessa grande massa de sensores não é um consenso. Embora o crescimento de IoT seja algo natural, é necessário que exista algum tipo de padronização que facilite a utilização de tantos tipos de dispositivos.

A aplicação de Internet of Things pode ser encontrada em diversos contextos do dia a dia, seja em uma rede de sensores médicos, em uma casa inteligente ou em fábricas, onde sua aplicação é denominada Industria 4.0. A maior força na ideia do uso de IoT, é o alto impacto que a mesma pode causar em aspectos do dia a dia e no comportamento de potenciais usuários (ATZORI; IERA; MORABITO, 2010). Tendo uma abrangência variada, os tipos de utilização para os usuários devem seguir a mesma tendência. Para uso pessoal, os usuários vão possuir aplicações dentro do contexto de IoT em suas casas, como smart homes, ou seus carros, como VANETs. Já para uso em negócios, a tendência é a aplicação dos conceitos na área da indústria e serviços, onde a primeira já vem sendo chamada de Indústria 4.0 e no segundo caso já existem exemplos, como pode ser visto no site Slock.it, que fornece serviços de fechaduras eletrônicas com desbloqueio através do uso de tokens e criptomoedas.

Uma vez que dentro de um contexto de IoT existe a necessidade de comunicação com diversos tipos de dispositivos e protocolos, ainda existem alguns desafios sobre como padronizar ou lidar com a comunicação. (XU; HE; LI, 2014) sugere a utilização de uma arquitetura orientada a serviços (SOA) e propõe uma arquitetura utilizando 4 camadas para lidar com a comunicação. As camadas sugeridas foram: Sensores,

onde ficam localizados os objetos IoT, Rede, responsável pode fornecer a comunicação entre os objetos, Serviços, responsável pela oferta de serviços pelos objetos IoT e Interface, responsável por fornecer métodos de interação entre os usuários e serviços.

3.1 PROTOCOLOS IOT

Embora ainda não exista um consenso ou padronização para a comunicação de dispositivos dentro do contexto de IoT, existem diversas propostas de protocolos que podem ser utilizadas. A nível de aplicação, temos protocolos com diferentes tipos de abordagens, como o Constrained Application Protocol (CoAP) , implementado em cima do protocolo HTTP e que utiliza o padrão REST para comunicação. Por utilizar o protocolo UDP, o CoAP apresenta menor consumo computacional e energético. Seu uso permite um menor tempo de resposta, pois mantém uma conexão ativa entre nodos (ROTTA; DANTAS, 2017). Outro protocolo muito utilizado é o Message Queue Telemetry Transport (MQTT)(Luzuriaga et al., 2015), que consiste na utilização do padrão produtor/consumidor, onde os sensores produzem informações e enviam para um broker, que disponibiliza os dados para serem consumidos. Pode ser descrito como um protocolo many-to-many e tem como sua grande vantagem a eficiência energética (ROTTA; DANTAS, 2017). A nível de aplicações, ainda existem outros protocolos utilizados, como XMPP(SAINT-ANDRE et al., 2009), AMQP(Luzuriaga et al., 2015), entre outros. Também existem propostas para protocolos para descoberta de novos dispositivos. Os dois mais utilizados são o Multicast DNS (mDNS) e DNS Service Discovery (DNS-SD)(Jara; Martinez-Julia; Skarmeta, 2012).

Além dos tipos de protocolos citados anteriormente, existe uma variedade de protocolos que podem ser utilizados na comunicação entre dispositivos. Além do protocolo TCP/IP que é normalmente utilizado em redes cabeadas, destacam-se dentro do contexto de IoT, os protocolos que utilizam comunicação sem fio, como Bluetooth Low Energy (BLE) , Wi-Fi (IEEE 802.11) e ZigBee. Ainda existem outros protocolos menos utilizados, como o Sigfox e o LoRa/LoraWAN (AUGUSTIN et al., 2016).

3.1.1 Bluetooth Low Energy (BLE) e ZigBee

O BLE é um protocolo desenvolvido para comunicação em curta distância e monitoramento focado em economia de energia que visa ser utilizado por bilhões de dispositivos já nos próximos anos (GOMEZ; OLLER; PARADELLS, 2012). Tendo sido desenvolvido pelo Bluetooth Special Interest Group, esta tecnologia sempre foi apresentada como uma solução de baixo consumo energético, podendo ter um tempo de vida útil de mais de 10 anos (GOMEZ; OLLER; PARADELLS, 2012). Segundo (SIEKKINEN et al., 2012), o BLE é um dos candidatos promissores para as futuras demandas dentro do contexto de IoT.

Embora utilize a mesma frequência de 2.4 Ghz que as especificações anteriores dos protocolos bluetooth, o BLE não possui uma retrocompatibilidade com com versões anteriores. Ainda assim, existem dispositivos que são chamados de Dual Mode, abrangendo a arquitetura do bluetooth clássico e do BLE. Umas principais características dos dispositivos BLE, é que estes ficam a maior parte do tempo em modo inativo, despertando de tempos em tempos para realizar a troca de informações. Essa permanência faz com que o dispositivo consuma menos energia. Embora possua uma taxa menor de transmissão de dados em relação ao bluetooth clássico, o BLE apresenta um consumo de energia menor e um custo mais baixo na aquisição de dispositivos.

Segundo (BARONTI et al., 2007), o ZigBee foi desenvolvido para ser uma tecnologia de rede wireless confiável, de baixo custo e de baixo consumo energético. Baseado no padrão IEEE 802.15.4, ele ainda implementa mais duas camadas (rede e aplicação), além dos objetos da rede. Embora possa ser utilizado em frequências diferentes em outros países, no Brasil o protocolo opera na frequência de 2.4 Ghz. Projetados para serem dispositivos de baixo consumo energético, o objetos ZigBee, assim como no caso do BLE, permanecem a maior parte do tempo inativos.

4 SMART COMM

A proposta apresentada nesse trabalho consiste no middleware Smart Comm, que foi desenvolvido voltado para a solução de problemas de interoperabilidade na comunicação entre dispositivos heterogêneos dentro de um ambiente IoT, tendo sido escolhido como foco do trabalho um ambiente de smart home. O ambiente foi escolhido devido ao alto grau de dificuldade de simulação e testes em ambiente mais complexos, como uma smart city.

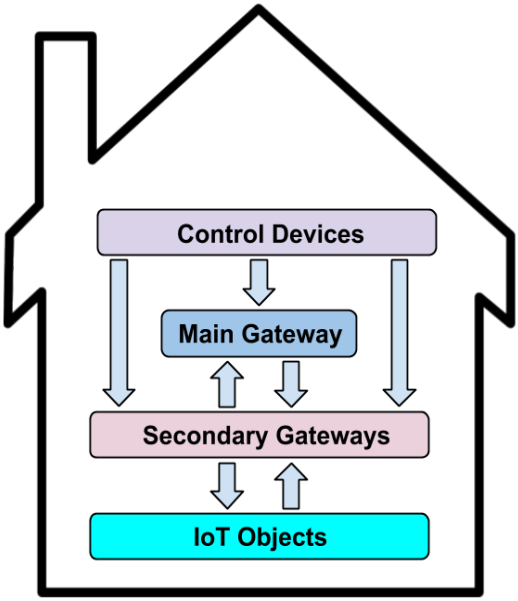

Figura 2 – Smart Comm. Fonte: (AGOSTINHO et al., 2018)

O Smart Comm foi desenvolvido para atuar sobre 4 grupos de componentes: Gateway Central, Gateways Secundários, Objetos IOT e Dispositivos de Controle. Este capítulo apresenta detalhadamente cada um destes. A Figura 2 apresenta a visão geral dos componentes do Smart Comm interagindo entre si.

Gateway Central: Sua principal função é fornecer informações

para os dispositivos de controle sobre quais objetos estão disponíveis e quais gateways devem ser chamados para o serviço desejado. As requisições geralmente são compostas por um comando responsável por realizar alguma função dentro da casa, como acender uma lâmpada ou abrir uma persiana. A comunicação do gateway central com os secundários será feita através de requisições, no modelo cliente/servidor. O gateway central pode fazer também o papel de um gateway secundário. No entanto, por ser um dispositivo muito requisitado na rede, é recomendável que sua principal atividade seja na função de um centralizador. A Figura 3 ilustra uma abstração das entidades projetadas para o desenvolvimento do Gateway Central.

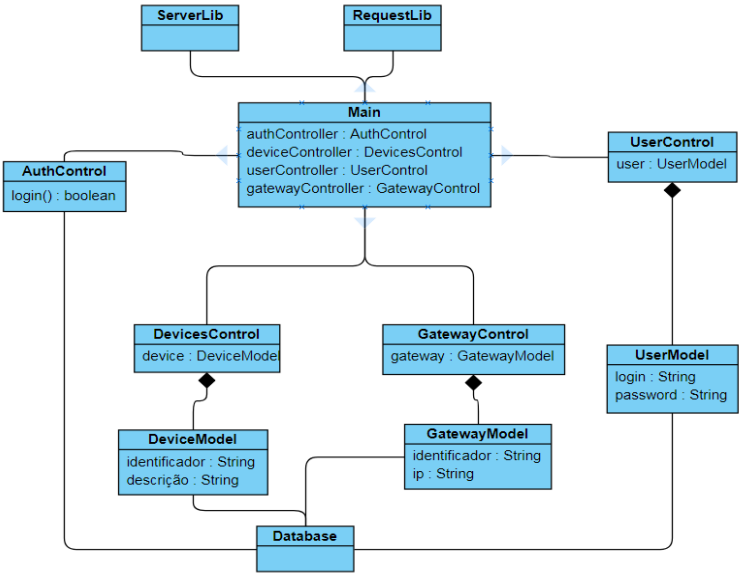

Figura 3 – Gateway Central - Entidades.

Gateways Secundários: São os atores principais dessa arquitetura. A proposta é possuir uma quantidade de gateways que atendam a todos os objetos e sensores da casa sem se sobrecarregarem. Esses dispositivos serão comandados (ativados/desativados ou informação recolhida) através de microsserviços. Os serviços enviam ou recebem mensagens para os objetos ou simplesmente recolhem informações dos sensores através das portas programáveis de entrada e saída da placa ou em sensores remotos. Os dados recolhidos são enviados para o gateway

central. Os gateways secundários podem possuir diversas funcionalidades, como por exemplo, media center, comandos de voz e fornecimento de conectividade Wi-Fi à dispositivos da casa. A Figura 4 ilustra uma abstração das entidades projetadas para o desenvolvimento do Gateway Secundário.

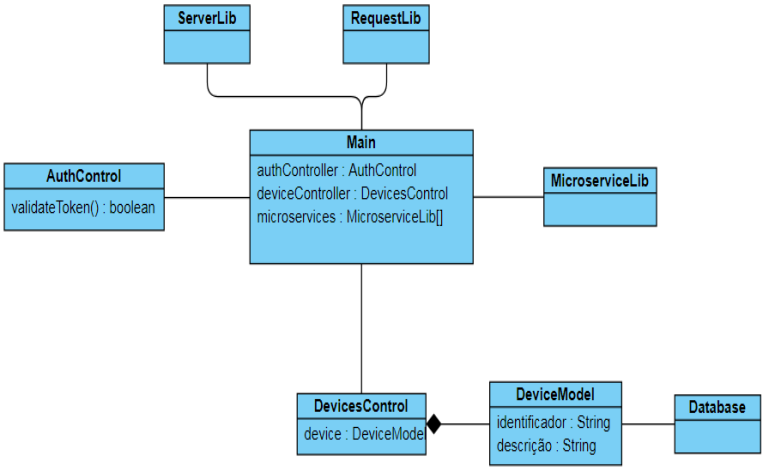

Figura 4 – Gateway Secundário - Entidades.

Dispositivos de Controle: O dispositivo de controle é responsável por mostrar aos usuários do sistema as opções de controle da casa. As opções são determinadas pelos objetos disponíveis. Os dispositivos irão se comunicar com o gateway central para login e aquisição de tokens para acesso aos gateways secundários. A menos que seja feita uma requisição externa, os comandos enviados para objetos IoT via dispositivo de controle sempre serão feitos diretamente a um gateway secundário.

Objetos IoT: Podem ser tratados como atuadores e sensores em um sistema de controle. Eles possuem uma capacidade computacional moderada e necessitam de baixo consumo energético. Sua comunicação é feita através de protocolos IoT. Na proposta foram utilizados o ZigBee, Bluetooth LE e Wi-Fi. Para isso ocorrer, é necessário que o Gateway esteja equipado com um transceiver com a modulação e frequência de rádio especificada pelos padrões de cada comunicação.

Uma das características mais importante do sistema proposto é a comunicação entre os diferentes dispositivos. O caminho que um comando segue, desde a chamada do usuário até o objeto IoT, podem

seguir 3 percursos diferentes, explicados a seguir:

Comando a um gateway secundário para uma ação de um dispositivo de borda conectado ao mesmo gateway (Figura 5): Neste caso o gateway recebe do dispositivo de controle um comando para realizar uma ação em um objeto IoT e então, retransmite o comando de forma direta.

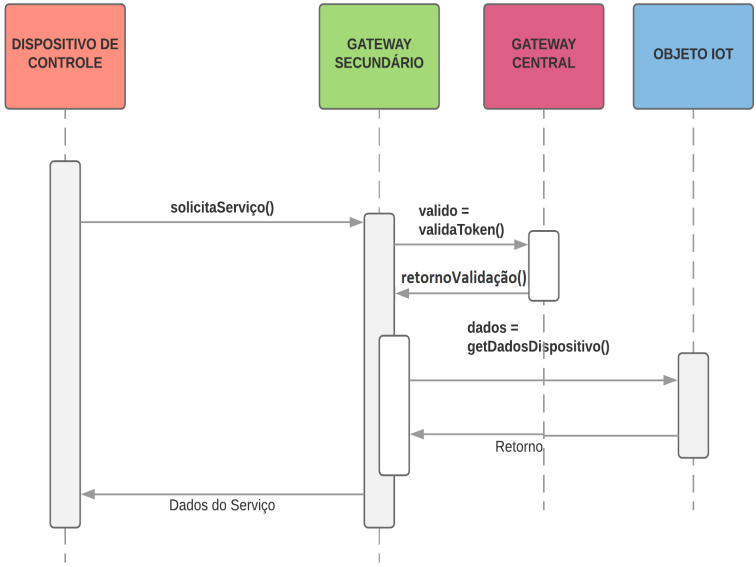

Figura 5 – Diagrama de Sequência - Caminho 1.

Comando a um gateway secundário para uma ação de um dispositivo de borda conectado a outro gateway: Neste caso o gateway recebe do dispositivo de controle um comando para realizar uma ação em um objeto IoT que não consta em sua rede. Desta forma, o gateway envia o comando ao servidor central, que mapeia qual é o gateway correspondente e retransmite o comando ao mesmo, que por sua vez envia o comando ao objeto IoT.

Comando externo para uma ação de um dispositivo de borda: Neste caso o comando vem de uma entidade externa (Internet). O gateway central recebe a requisição e mapeia qual é o gateway secundário correspondente e retransmite o comando a este, que por sua vez, envia o comando ao objeto IoT.

Para evitar problemas de sobrecarga nos gateways secundários, dois tipos de topologias para utilização do Smart Comm são sugeridas.

A primeira, como mostra a Figura 6, utiliza um gateway central para a casa inteira, e um gateway secundário para cada tipo de protocolo de comunicação do cômodo. Como é possível ver na figura, todos os objetos IoT de um cômodo que se comunicam via BLE enviam os dados para o mesmo gateway. O mesmo ocorre para os protocolos ZigBee e Wi-Fi. Além de evitar a sobrecarga, essa topologia visa causar menos interferências, deixando cada gateway secundário encarregado de apenas um tipo de comunicação.

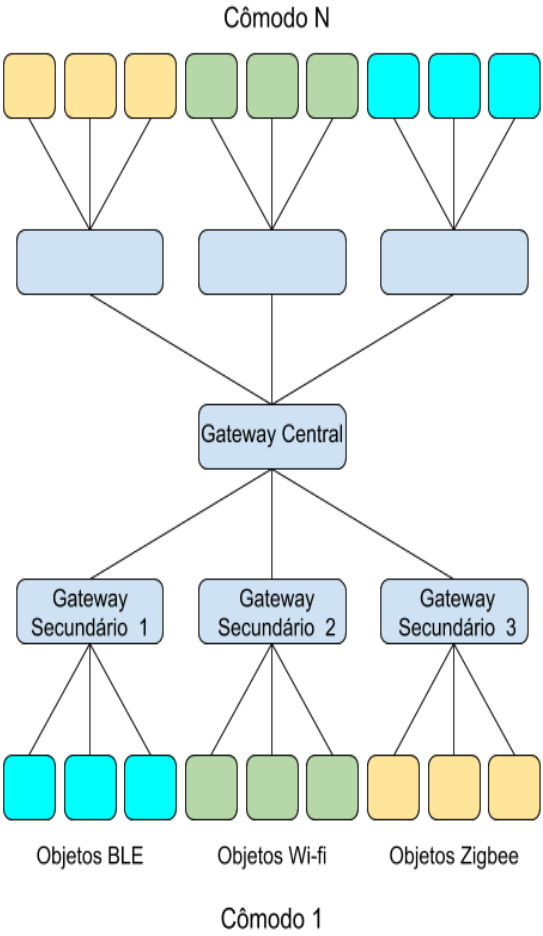

Figura 6 – Topologia proposta 1.

Na Figura 7, é apresentada uma topologia onde cada cômodo, além dos gateways secundários que atuam da mesma forma da Figura 6, possui também um gateway central. Nesse caso os gateways secundários do cômodo enviam seus dados apenas para este gateway e todos os cômodos teriam uma topologia igual à sugerida na imagem.

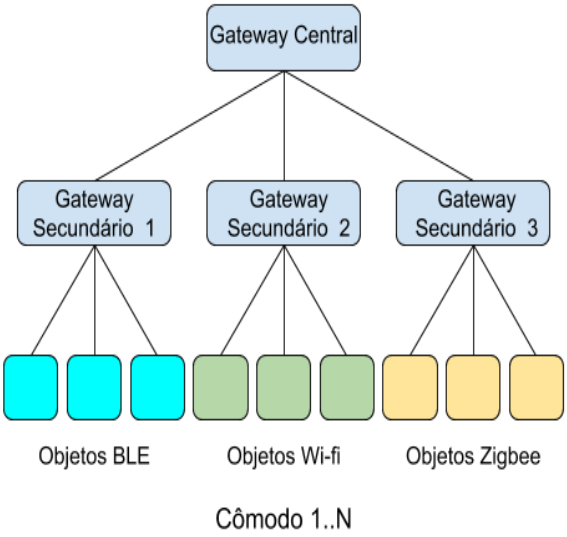

Figura 7 – Topologia proposta 2.

Para o desenvolvimento do Smart Comm, decidiu-se pela utilização de uma arquitetura orientada a serviços, utilizando microsserviços para tal. O motivo da decisão foi em razão da flexibilidade que os microsserviços trazem em comparação com outras arquiteturas possíveis. Uma vez que existem diversos tipos de dispositivos e de diversos fabricantes, a possibilidade de um fabricante enviar seus próprios serviços de acesso a cada tipo de dispositivo pode ser considerado um facilitador na inserção de um novo dispositivo. Outra razão levantada foi que a grande variedade de dispositivos que vem surgindo pode dificultar no futuro a utilização de um tipo específico em um sistema que obrigue a utilização de uma determinada linguagem de programação fixa, tendo nos microsserviços a flexibilidade para utilização que bem entender.

Uma das alternativas ao uso de microsserviços seria a utilização de um sistema modular tradicional. O problema encontrado nesse

tipo de sistema é que normalmente é necessário que todo o sistema esteja em funcionamento para os módulos ficarem ativos, diferente de microsserviços, que podem ser executados de maneira isolada.

A proposta apresentada neste capitulo visa ilustrar uma arquitetura capaz de lidar com os desafios de interoperabilidade e heterogeneidade de dispositivos dentro do contexto de IoT e Smart Homes. Foi proposta a utilização de microsserviços para lidar com as peculiaridades de cada tipo de dispositivo e para viabilizar a utilização de múltiplos protocolos de comunicação. A interoperabilidade citada neste trabalho pode ser definida como a interação entre estes dispositivos de maneira transparente para o usuário, posto que os dispositivos utilizados podem ter diversos fabricantes, formato de dados e formas de acessos diferentes.

4.1 TRABALHOS CORRELATOS

Para realizar a proposta do Smart Comm, uma foi realizada uma busca em trabalhos anteriores que visaram solucionar os mesmos problemas que o Smart Comm ou que tiveram abordagens semelhantes à proposta. Uma vez que a área de interoperabilidade dentro do contexto de IoT é bem abrangente, podendo seguir por diversos caminhos e abordagens, foram selecionados os trabalhos considerados mais relevantes.

Os diversos desafios a serem superados dentro do conceito de Smart Homes fazem com que cada vez mais pesquisas na área ocorram. Em (BOTTARO; GÉRODOLLE, 2008), foi proposto um modelo a fim de solucionar o problema heterogeneidade dentro do contexto de Smart Homes. A variedade de protocolos é gerenciada por drivers orientados a serviços, aproveitando o conceito de "plataforma de serviços", enquanto a dinamicidade da rede é tratada localmente na plataforma. Por meio de alguns casos de uso implementados, são mostrados como as escolhas tecnológicas e os padrões orientados a serviços facilitam a composição dinâmica de serviços de dispositivos domésticos distribuídos e heterogêneos. Embora a solução funcione bem em relação a heterogeneidade, a utilização de uma arquitetura SOA tradicional pode tirar um pouco da flexibilidade para compartilhamento de serviços e variação de tecnologia.

(BEDHIEF; KASSAR; AGUILI, 2016) propuseram a junção das tecnologias para Software Defined Networks (SDN) e containers to tipo Docker com o objetivo de resolver os problemas criados pela heteroge-

neidade dos dispositivos IoT. Foi proposta uma arquitetura para redes heterogêneas de dispositivos. Foram realizados experimentos utilizando esta arquitetura com um controlador SDN centralizado. Os principais pontos observados foram relacionados a conectividade entre os dispositivos heterogêneos ligados à redes por meio do desempenho de diferentes fluxos de tráfego. Com a utilização desses duas tecnologias o trabalho propõe uma arquitetura e foca em provar a sua viabilidade em ambientes com rede e dispositivos heterogêneos. Até então os autores ainda não tinham realizados testes mais a fundo na conexão dos dispositivos. A utilização de um controlador SDN centralizado pode vir a ser um problema em caso de falha. A substituição desse tipo de dispositivos normalmente não ocorre de maneira tão simples.

Em (SARKAR et al., 2014) foi proposta uma arquitetura distribuída em camadas chamada Distributed Internet-like Architecture for Things (DIAT). O autores tiveram como objetivo resolver os problemas relacionados à heterogeneidade, escalabilidade e interoperabilidade além de automatizar os recursos dentro do contexto de IoT. A arquitetura visa trabalhar com a associação de funcionalidades de maneira hierárquica. Para validar esta arquitetura foi planejado um caso de uso que consiste em vários aplicativos de IoT, como smart home, smart transportation e smart healthcare. Com base nos fundamentos genéricos de um ambiente IoT, foram validados casos de utilização de um sistema real. Embora pareça solucionar problemas de integração e heterogeneidade, a arquitetura proposta não parece ser flexível o suficiente para uma fácil adaptação em caso de mudanças necessárias.

Com base nesses avanços recentes na padronização no ambiente IoT com objetivo de melhorar a interoperabilidade, (BELLAVISTA et al., 2013) propôs a utilização de MANETs (Mobile Ad-Hoc Network) para acelerar a coleta de dados de WSNs (Wireless Sensors Network). A proposta explora as oportunidades de interação permitidas pelos protocolos padrões para habilitar o roteamento de rede cruzada com baixa latência. A solução proposta reduziu a latência de entrega de pacotes de dados urgentes em todos os casos testados. A proposta apresentou uma integração de dois tipos de redes diferentes em um contexto IoT mas fica limitada apenas a redes com protocolo de comunicação sem fio.

O trabalho de (MOAZZAMI et al., 2016) elenca os aspectos da heterogeneidade em dispositivos de IoT e as deficiências de soluções disponíveis. É introduzida então a plataforma para smart-homes SPOT, construída sobre um modelo de abstração de drivers de dispositivo dinâmico que visando solucionar problemas de heterogeneidade. O trabalho

sugere que a principal tendência neste domínio de aplicativo IoT é fornecer mais APIs abertas e criar uma plataforma unificada do sistema. A plataforma SPOT é uma plataforma aberta, orientada pela comunidade e extensível para realização da rápida adoção e o suporte a novos dispositivos. Uma das suas principais características é a de oferecer uma camada de abstração para através de uma API aberta e unificada para todos os dispositivos independente do fabricante. Ao tornar as características heterogêneas dos dispositivos inteligentes transparentes, o SPOT minimiza a carga dos desenvolvedores de aplicativos de automação residencial e os esforços dos usuários que teriam que lidar com aplicativos e interface específicas de um certo dispositivo. Embora uma API aberta para solucionar problemas com múltiplos dispositivos pareça eficaz, os autores não deixaram claro o funcionamento dos mesmos em relação aos protocolos de comunicação.

Em (ZHILIANG et al., 2011), os autores propuseram um middleware utilizando a arquitetura SOA em conjunto com sistemas multiagentes para tratar da heterogeneidade dos dispositivos dentro do contexto de IoT. A proposta consiste na utilização de agentes para conversão do acesso aos dados em serviços, que são agrupados em novos serviços, gerando assim uma camada homogênea. Embora tenham lidado com os dispositivos usando uma abordagem de serviços, a transformação dos mesmos em uma camada homogênea, onde existem dependência entre serviços podem tirar um pouco da flexibilidade e da independência dos mesmos.

O trabalho de (RAHMAN; CHAKRABORTY, 2018) propõe a utilização de um gateway conectado aos dispositivos e disponibilização das informações na nuvem para que os mesmos consumam. A proposta tem como objetivo resolver problemas de comunicação entre dispositivos IoT, através do uso dos protocolos BLE e ZigBee. Embora a proposta seja semelhante ao trabalho aqui proposto, os autores focaram no envio das informações para a nuvem, divergindo da proposta de utilização de uma Fog Computing. Além disso, os autores até o momento da publicação não chegaram a desenvolver os testes para validar o que estavam propondo.

O trabalho de (PERUMAL; RAMLI; LEONG, 2011) propõe a criação de um framework para lidar com a heterogeneidade no acesso e formato de dados dentro de um ambiente de smart homes. Através da utilização do protocolo SOAP, os autores propõem a utilização de uma interface de acesso, que utiliza serviços para se comunicar com os objetos da smart home através de uma rede ethernet. Com um foco apenas na interoperabilidade dos dados, e utilizando apenas uma rede

de comunicação cabeada, o trabalho proposto não aborda problemas de interoperabilidade entre dispositivos com múltiplos protocolos de comunicação.

4.2 PROTÓTIPO

O desenvolvimento do protótipo do Smart Comm foi dividido em componentes, um para cada grupo de dispositivos apresentados anteriormente. Este aborda uma série de funcionalidades necessárias para o bom funcionamento do middleware, como por exemplo:

- Segurança de acesso aos dispositivos, utilizando tokens e criptografia para autenticação de usuários: A segurança é fundamental para garantir que os comandos e serviços disponíveis na casa inteligente sejam restritos somente a usuários com autorização, evitando assim usuários mal intencionados.
- Listagem de dispositivos: Necessário para informar aos usuários quais dispositivos estão disponíveis para serem utilizados.
- Gerência de serviços: Permite a inserção e remoção de serviços em tempo real. Isso é fundamental para quando o usuário do sistema decide instalar um novo dispositivo de borda e precisa adicionar um novo serviço no sistema.
- Mapeamento de serviços: Fundamental para que os serviços sejam executados. É necessário antes saber o endereço de acesso correto de cada um.

Responsável pelo encaminhamento de requisições, autenticação e cadastro dos serviços, a implementação do Gateway Central foi realizada utilizando exclusivamente a linguagem Javascript. A escolha da linguagem foi devido ao fato desta ter sido projetada em uma arquitetura orientada a eventos, ideal para lidar com entradas e saídas, comportamento esse, esperado do middleware. Foi escolhido o interpretador Node.js, que utiliza o event-loop, um gerenciador de eventos responsável por prover a funcionalidade de E/S assíncrona. O código consiste em um servidor API monolítico, utilizando como base o framework web Koa.js. A escolha do framework se deve ao fato deste utilizar a versão mais recente do Node.js, podendo aproveitar todas as funcionalidades existente no interpretador. A estrutura do servidor pode ser dividida em arquivos de controle, modelo e roteamento.

Para o armazenamento de dados, foi utilizado o banco de dados NoSQL Mongo DB. Sua escolha ocorreu por sua flexibilidade referente aos schemas de dados dinâmicos, eficiência na utilização de recursos e por sua compatibilidade no armazenamento de dados de objetos Javascript.

Os gateways secundários ficam responsáveis por alocar os microsserviços. A implementação também foi realizada utilizando exclusivamente a linguagem JavaScript, para se aproveitar dos mesmos benefícios do Servidor Central, utilizando também o interpretador Node.js. O código consiste em um servidor principal, utilizando como base o framework web "Express.js". Este é responsável por instanciar os microsservidores utilizando o framework "Microcule". A razão da escolha do framework do servidor principal do Gateway Secundário ser diferente do Gateway Central foi devido a sua facilidade de integração com o framework de microsserviços. A principal função do protótipo do Gateway é a gerência de serviços. Isso significa instanciar os diferentes serviços, verificar a inclusão e remoção dos mesmos e testar a conexão com os dispositivos e as interferências. Toda vez que um novo serviço é adicionado ao diretório, o servidor principal sobe uma nova instância de servidor para atender aquele código novo. E toda vez que um serviço é retirado, o servidor remove aquela instância, não permitindo um novo acesso.

O servidor principal do Gateway Secundário, ao ser instanciado, verifica o diretório de serviços, e então percorre cada um dos arquivos (serviços) encontrados, criando uma instância de cada. Isso é feito utilizando o "Microcule". A inserção de novos serviços é feito utilizando as portas disponíveis, que foram configuradas para começarem em 4000 e incrementar a cada nova instância. O servidor principal do Gateway também possui um observador para este mesmo diretório. A cada inclusão de um novo serviço, o mesmo é instanciado, e a cada remoção, o serviço é desligado. Cada um deles, além do código de execução, possui uma descrição informando as características e também possíveis parâmetros que podem ser enviados através do método POST. Ao instanciar um novo serviço, o servidor inicializa também uma instância da autenticação, forçando a requisição a ser feita por um usuário cadastrado. O funcionamento ocorre da mesma maneira que o Gateway Central, utilizando tokens. Para validar o token, é utilizada uma chave compartilhada. O token é obtido através do Gateway Central.

O protótipo permite três maneiras distintas de operar um dispositivo de borda (Objeto IoT). Na primeira, o objeto IoT realiza uma chamada ao microsserviço, enviando seus dados. Na segunda, os objetos enviam se comunicam com o adaptador USB relativo a seu protocolo.

Neste caso, os dados enviados são encaminhados para um microsserviços responsável. No terceiro caso, o adaptador fica aberto a novas detecções e encaminha os dispositivos que encontra para um microsserviço responsável.

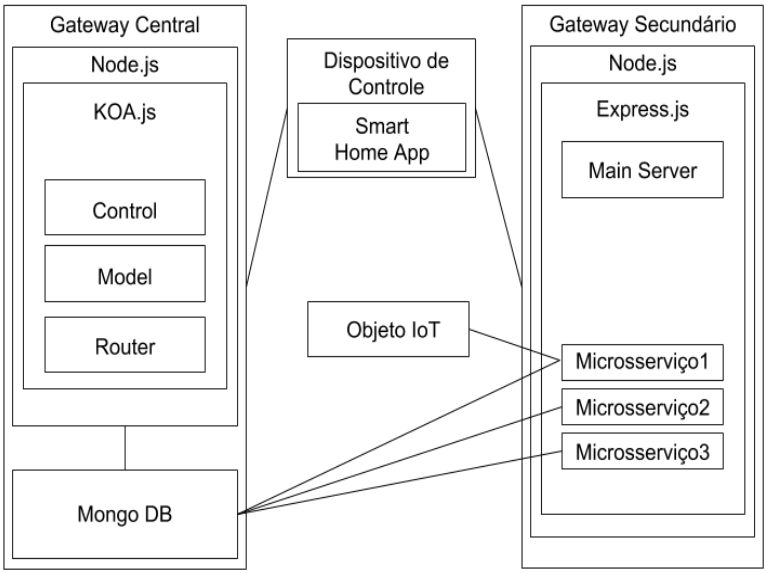

Figura 8 – Arquitetura Smart Comm.

A Figura 8 ilustra a interação entre os componentes do Smart Comm. Embora apareça na figura, a aplicação de controle da smart home não teve seu prototipo desenvolvido. As funcionalidades foram testadas através de requisições HTTP, simulando a atuação esperada da aplicação.

4.3 CONSIDERAÇÕES

Embora tenha sido escolhida uma arquitetura de microsserviços para o desenvolvimento do Smart Comm, foi necessário adaptar o conceito à realidade da estrutura para qual o middleware foi projetado. Visando utilizar dispositivos de baixo custo e baixo poder computacional, optou-se pela centralização do banco de dados para não precisar gastar poder de processamento com múltiplas instâncias de banco de

dados em cada gateway. No entanto, nada impede que no futuro, o Smart Comm seja utilizado com microsserviços totalmente independentes, cada um com sua própria base de dados isolada.

Um diferencial proposto e que o Smart Comm pode alcançar, é a utilização de sua própria estrutura para armazenamento e processamento de dados. Enquanto existe uma certa tendência de envio de dados para nuvem, o Smart Comm foi projetado pensando na utilização do paradigma de Fog Computing. Embora tenha sido desenvolvido pensando em dispositivos de baixo custo, o poder de processamento de uma estrutura de gateways em conjunto deve ser testada. Por mais que o escopo do trabalho seja mais focado na interoperabilidade de dispositivos, toda arquitetura foi pensada para utilização local dos dados. A arquitetura utilizando microsserviços pode vir a ser utilizada não só para comunicação com os objetos IoT, mas também para prover funcionalidades para a casa, como por exemplo, análise de dados dos moradores, previsão do tempo, entre outros.

5 AMBIENTE EXPERIMENTAL

Os experimentos realizados neste trabalho foram elaborados visando medir o nível de interferência que o uso de múltiplos protocolos de comunicação pode gerar. Para tal, foram escolhidos para o teste os protocolos ZigBee, BLE e Wi-Fi. Também foram realizados casos de uso para testar funcionalidades de administração do middleware, como cadastro de usuário e simulação de uma chamada de um microsserviços com validação de token de acesso.

5.1 CRIAÇÃO DE USUÁRIO, AUTENTICAÇÃO E CHAMADA A MICROSSERVIÇO

A primeira ação realizada para poder utilizar o sistema é o cadastro de um novo usuário. Sem ele, nenhuma funcionalidade poderá ser acessada. A restrição dos serviços do protótipo à usuários cadastrados é essencial para garantir a segurança dentro da casa. Sem essa restrição, qualquer visitante poderia fazer requisições, algumas delas podendo ser feitas por usuários mal intencionados.

O cadastro do usuário é feito através de uma requisição à API do Gateway Central. É necessário enviar uma requisição POST com os seguintes parâmetros: email, firstname, lastname, password e username. É possível ver a listagem dos usuários cadastrados através de uma chamada GET ao servidor. Neste caso, é possível realizar a chamada pelo próprio navegador web.

A partir do usuário cadastrado, é possível realizar a autenticação no sistema. O usuário deve realizar uma requisição POST ao endereço http://localhost:3000/api/auth passando como parâmetro no corpo da mensagem o nome de usuário e senha. Em caso de sucesso, o sistema retornará o endereço de ip de um gateway disponível e o token de acesso para realizar chamadas aos serviços oferecidos.

Neste protótipo, não foi desenvolvida nenhuma restrição ao cadastro de usuários. O ideal é que apenas o primeiro usuário consiga fazer seu próprio cadastro. Para os próximos, o sistema deve requerer a autenticação.

A realização de um comando para um dispositivo de borda é feita a partir de um serviço oferecido por um Gateway Secundário. Para efetuar uma requisição, o usuário necessita antes possuir um token de autenticação e uma lista com os endereços para chamada de

cada serviço. Com o token de acesso obtido na autenticação do gateway central, o usuário consegue, através de uma requisição à API do servidor principal do Gateway Secundário, a listagem de todos os serviços oferecidos por ele. Considerando um host qualquer para o Gateway Secundário, é possível ter acesso a um JSON da lista de serviços através de uma requisição POST ao endereço URL http://host:4000/services. Após obter a porta correspondente ao serviço desejado e obtido token de autenticação, é possível fazer a requisição ao serviço. O serviço usado como exemplo é o "hello.js", cuja porta é a 4004. Na figura 5, podemos verificar o resultado de uma requisição POST ao endereço http://192.168.1.2:4004. Foi utilizado o parâmetro "oauthtoken"para poder realizar a autenticação e o IP do Gateway Secundário utilizado foi o 192.168.1.2. Como resposta, o serviço foi programado para retornar a mensagem "Hello World!"

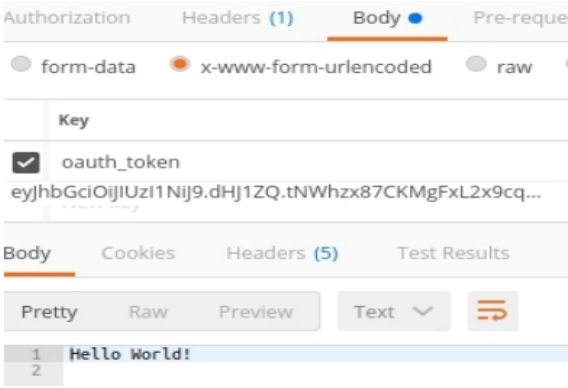

Figura 9 – Requisição a um microsserviço.
Fonte: (AGOSTINHO et al., 2018)

5.2 PROTOCOLOS ISOLADOS

Para os casos de uso de interoperabilidade, foram realizados testes voltados para comparação da utilização dos dispositivos de maneira isolada, com os mesmos cenários em uma simulação de utilização real do middleware.

Os testes foram realizados inicialmente por tipo de protocolo de comunicação, aumentando o número de objetos IoT conectados ao gateway secundário correspondente. A intenção dos testes isolados foi

obter parâmetros de comparação com a utilização do middleware com múltiplos protocolos de comunicação em paralelo. Todos os testes foram repetidos 5 vezes, sendo utilizado a média dos resultados como resultado final.

Para os testes, foi delimitado como escopo a utilização dos padrões de comunicação BLE, ZigBee e Wi-Fi, por atualmente serem as tecnologias mais utilizadas para comunicação em ambientes semelhantes ao apresentado nessa proposta.

Para a infraestrutura utilizada para simulação dos gateways centrais e secundários no ambiente experimental, foram utilizados SoCs do tipo Raspberry Pi 2. As placas contam com um processador de quatro núcleos de 900 MHz e 1 Gb de memória cada.

5.2.1 Bluetooth LE

Na execução dos testes do Smart Comm com objetos IoT com comunicação através do BLE foi montada uma arquitetura com um gateway central, um único gateway secundário equipado com um adaptador Bluetooth 4.0 USB e os objetos IoT, que foram simulados através de 3 Smartphones. A Figura 10 mostra a topologia utilizada para esse caso de teste.

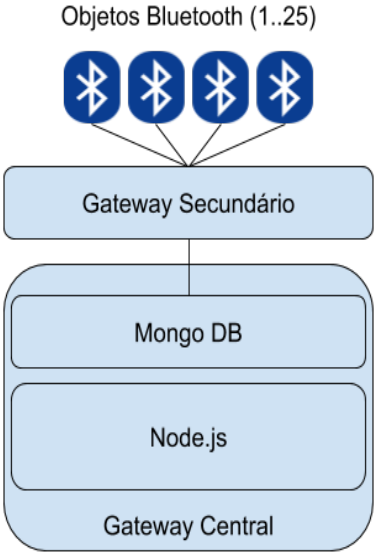

Figura 10 – Topologia Bluetooth.

Para a comunicação com os dispositivos BLE, foi utilizada a biblioteca Node Beacon Scanner, que manteve a comunicação em aberto com uma chamada de callback a cada vez que um novo dispositivo era identificado. O microsserviço desenvolvido foi cadastrado junto com uma lista de identificadores de dispositivos que fazem parte do escopo daquele serviço. A cada novo serviço, a lista de dispositivos de interesse era cadastrada, assim como os serviços que devem ser chamados caso identificados. Para cada objeto descoberto, o middleware verifica se o identificador está na lista e para cada serviço cadastrado para esse objeto o Smart Comm faz uma requisição assíncrona encaminhando os dados.

Os testes foram realizados em 5 baterias de testes com duração de 5 minutos cada. Os dispositivos BLE foram simulados por smartphones para emitirem o sinal de um Apple iBeacon a cada 100ms. Cada smartphone passou por uma verificação do próprio aplicativo de simulação para analisar o numero máximo de dispositivos em paralelo que o aparelho era capaz de simular sem afetar o desempenho.

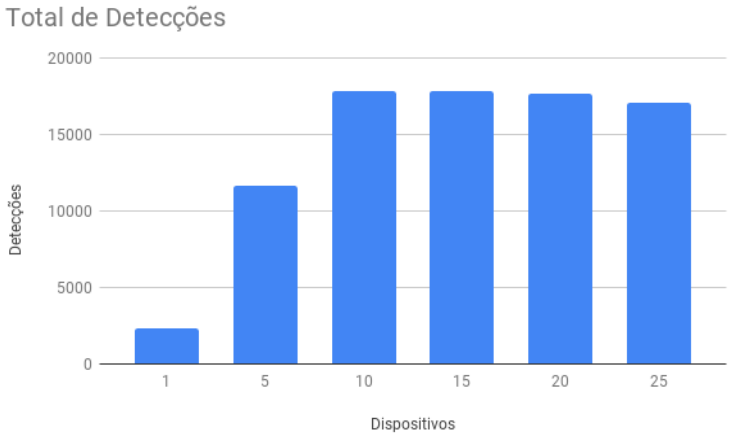

Figura 11 – Bluetooth - Total de Detecções.

A Figura 11 mostra a evolução do número total de detecções de dispositivos na medida que foram adicionados mais dispositivos. Foram consideradas como detecções, toda interação realizada entre os dispositivos bluetooth, simulados pelos smartphones para enviar seu identificador de maneira cíclica, e o gateway secundário. Uma vez que foram configurados para enviarem o sinal 10 vezes por segundo, no final de uma bateria de de testes de 5 minutos, era esperado que cada dispositivo possuísse um total de 3.000 detecções. Iniciando em 3.000 com um único dispositivo, o valor esperado na ultima bateria, para os testes com 25 dispositivos, era de 75.000. O valor encontrado após os testes, para apenas um dispositivo, foi de 2.296. Á quantidade aumentou quase que proporcionalmente quando foram realizados com 5 dispositivos, chegando em 11.677 e se estabilizou perto 18 mil para as baterias seguintes com 10, 15 e 20 dispositivos, tendo uma leve queda para 17.000 nas baterias com 25 dispositivos.

Em relação a quantidade de vezes que cada dispositivo foi encontrado, a Figura 12 mostra os resultados obtidos nos testes realizados. Com uma leve subida, demonstrando uma certa estabilidade, entre as simulações com 1 e 5 dispositivos, a partir de 10 dispositivos a quantidade média de detecções foi caindo até o final dos testes. Iniciando em um total de 2.296, o valor encontrado nos testes com 25 dispositivos foi o de 682, sendo que o valor esperado para ambos era de 3.000. Assim como no gráfico de total de detecções, o que se observa é que a

partir dos testes dom 10 dispositivos o total de detecções chega a um limite, fazendo com que a média de detecções por dispositivos caia com o acréscimo de novos dispositivos.

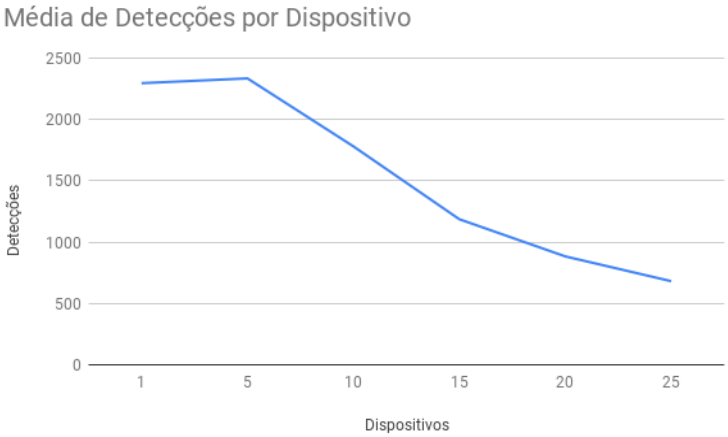

Figura 12 – Bluetooth - Média por Dispositivo.

Como é possível ver na Figura 13, a média de tempo de resposta nas requisições realizadas teve um crescimento em todas as configurações de testes, na medida em que mais dispositivos foram inseridos. O aumento mais expressivo ocorreu na mudança de 5 para 10 dispositivos onde a média de tempo de resposta teve um aumento de mais de 100%, pulando de 29.58ms para 67.51ms.

Figura 13 – Bluetooth - Média do Tempo de Resposta.

5.2.2 ZigBee

Para os testes do Smart Comm com objetos IoT com comunicação através do protocolo ZigBee foi utilizada uma arquitetura semelhante aos testes utilizados com BLE, utilizando apenas um gateway central e um gateway secundário equipado com um dongle USB para uma placa Xbee. Como objetos IoT, foram utilizadas 12 placas Xbee Serie 1. A Figura 14 mostra a topologia utilizada para esse caso de teste.

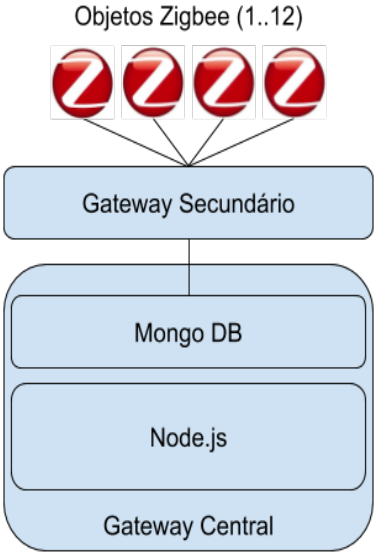

Figura 14 – Topologia ZigBee.

Nos mesmos moldes do serviço utilizado para comunicação com os objetos BLE, foi utilizada a biblioteca Xbee API para comunicação com as placas Xbee. O dongle USB utilizado no gateway secundário foi configurado para formar uma rede com as outras placas Xbee. Todas as requisições recebidas no gateway secundário fora redirecionadas para algum microsserviço desenvolvido. Assim como no caso do BLE, os microsserviços foram cadastrados junto com uma lista de identificadores que faziam parte do escopo. Para cada nova requisição, os dados enviados foram encaminhados pros serviços correspondentes. Os testes foram realizados em 5 baterias com duração de 5 minutos cada.

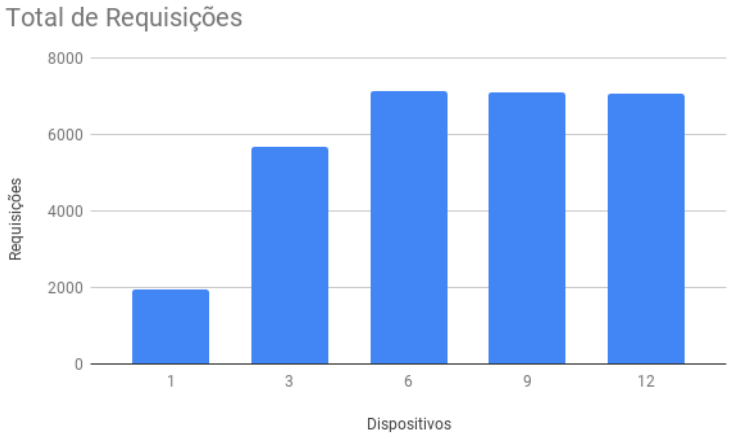

Figura 15 – ZigBee - Total de Requisições.

A Figura 15 mostra a evolução na quantidade de requisições obtidas com o aumento de dispositivos. Considera-se como requisição, toda mensagem recebida pelo gateway secundário enviada por uma placa XBee. Iniciando com o valor de 1.939 requisições para os teste com apenas 1 dispositivo, a quantidade cresceu quase que proporcionalmente (semelhante ao ocorrido com os dispositivos BLE), alcançando o valor de 5.673 requisições para 3 dispositivos. Assim como nos testes anteriores, os dispositivos ZigBee foram programados para enviar 10 requisições por segundo, sendo esperado 3000 requisições por placa no final de uma bateria de testes de 5 minutos. Para os testes com 1 e 3 dispositivos os valores encontrados correspondem a quase 2/3 do esperado. O número de requisições se estabilizou perto de 7.000 para os testes com 6, 9 e 12 dispositivos, alcançando aproximadamente 38%, 25% e 19% do valor esperado respectivamente.

A queda na porcentagem obtida em relação ao número de requisições esperado também pode ser observada na Figura 16, que mostra a média por dispositivo. Com uma pequena diferença entre os testes realizados com 1 e 3 dispositivos, a média teve sua queda mais acentuada entre os testes com 3 e 6 dispositivos. Ainda assim, os valores continuaram caindo na medida que mais dispositivos foram inseridos.

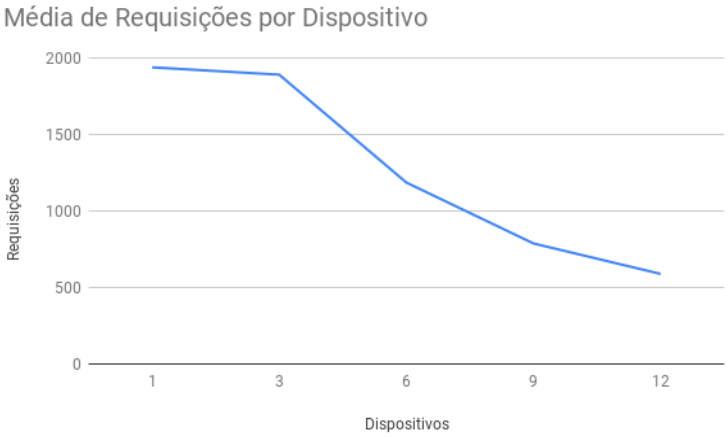

Figura 16 – ZigBee - Média por Dispositivo.

Em relação ao tempo de resposta, é possível ver na Figura 17 que a média de tempo de resposta obtida teve uma variação menor que a apresentada nos outros gráficos. Iniciando em 22.18 ms, o tempo de resposta média teve um aumento quando os testes foram feitos com 3 dispositivos, passando para 24.88 ms, a maior média encontrada para esse teste. A partir dos testes com 6 dispositivos a média foi caindo até alcançar 19.87 ms.

Com exceção do gráfico referente à média de tempo de resposta, o padrão de variação de quantidade de detecções e média encontrada por dispositiva ficou muito parecida com os dados encontrados para os dispositivos BLE. Já em relação ao tempo de resposta, enquanto para os dispositivos Bluetooth o tempo teve uma crescente constante e uma variação de mais de 200%, para os dispositivos a variação foi pequena e após uma leve subida começou a cair.

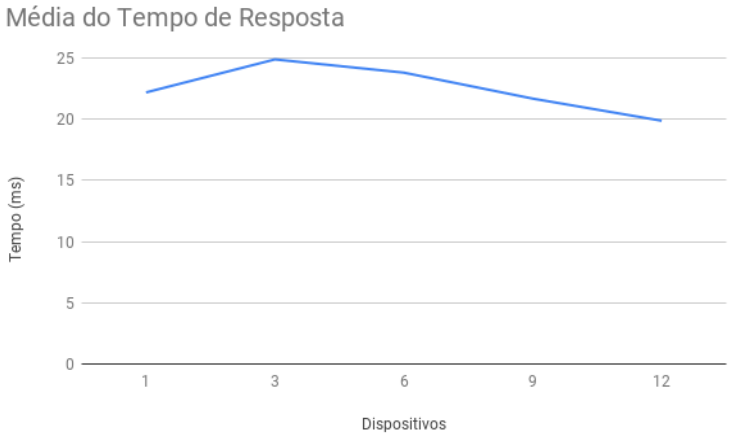

Figura 17 – ZigBee - Média do Tempo de Resposta.

5.2.3 Wi-Fi

Como é possível ver na Figura 18, a topologia utilizada para os testes com objetos Wi-Fi seguiu o mesmo padrão utilizado para os testes com BLE e ZigBee. Os gateways Central e secundário foram utilizados em conjunto com 12 placas Wemos em 5 baterias de teste de 5 minutos cada.

Os testes realizados pelos objetos Wi-Fi foram realizados através de chamadas diretamente ao microsserviço responsável. O serviço foi desenvolvido para receber chamadas HTTP das placas Wemos que foram programadas para realizarem requisições com intervalo de 100 ms e timeout máximo de 1 segundo. Como não foi possível realizar chamadas assíncronas por parte das placas, os valores esperados utilizados para os outros protocolos não puderam ser utilizados como referência nesse caso de teste, uma vez quem cada requisição podia ter que esperar até 1 segundo. A figura 19 mostra os resultados referentes a quantidade de requisições recebidas nas baterias de testes de 5 minutos.

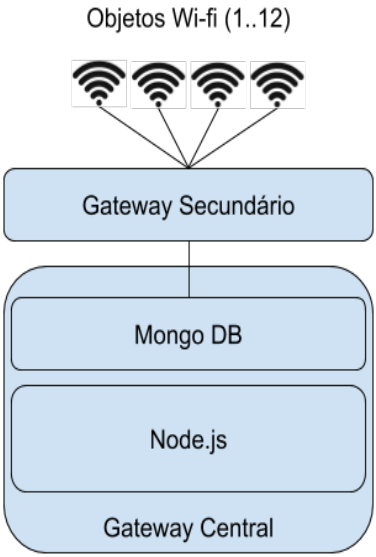

Figura 18 – Topologia Wi-Fi.

Diferente dos outros protocolos, o número de requisições de objetos Wi-Fi cresceu constantemente com a inserção de novos dispositivos. Iniciando em 1.018 requisições para o teste com 1 dispositivo, a quantidade foi crescendo quase que proporcionalmente até chegar ao valor de 11.389 para os teste com 12 dispositivos. Neste caso de teste, considerou-se como requisição o resultado de uma requisição HTTP realizada pelo objeto Wi-Fi.

Diferentemente dos casos anteriores, onde os objetos e o gateway secundário ficavam no mesmo ambiente e a comunicação era realizada de maneira direta, as requisições dos objetos Wi-Fi não foram feitas diretamente entre objeto e gateway secundário. Todos os objetos estavam em uma ambiente isolado, onde se conectavam a um roteador que fornecia o acesso até o gateway, que estava conectado a mesma rede através de uma sub-rede cabeada.

Em relação a média de requisições por dispositivo (Figura 20), o número de requisições se manteve quase que constante, ligeiramente superior a 1.000, com exceção dos testes com 12 dispositivos, onde a quantidade teve uma queda para 949.

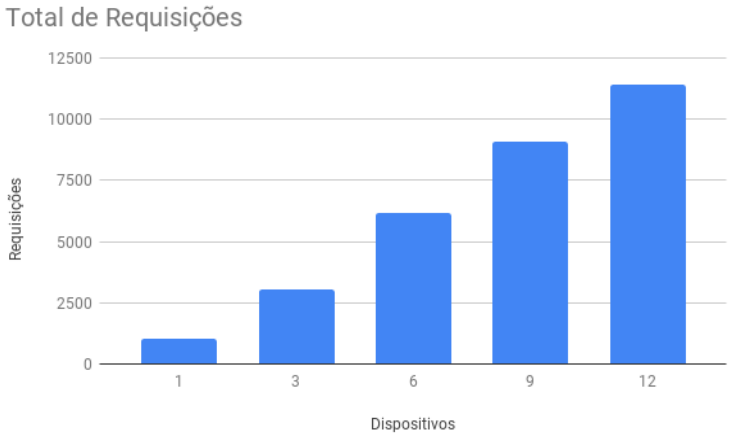

Figura 19 – Wi-Fi - Total de Requisições.

Figura 20 – Wi-Fi - Média por Dispositivo.

A Figura 21 mostra a evolução da média do tempo de resposta na medida que os novos dispositivos foram inseridos. Como é possível notar, a média de tempo de respostas para os testes de 1, 3 e 6 dispositivos foi muito próxima, se mantendo entre 62 e 64 milissegundos. A média teve um pequeno aumento no teste com 9 dispositivos, alcan-

çando o valor de 68 milissegundos e teve sua maior variação no teste com 12 dispositivos, onde alcançou o valor de 87 milissegundos.

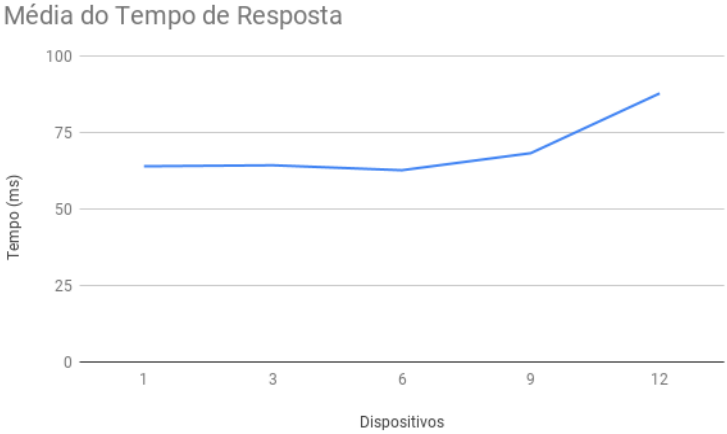

Figura 21 – Wi-Fi - Média do Tempo de Resposta.

5.3 INTEROPERABILIDADE

Os testes de interoperabilidade entre diferentes dispositivos, utilizando diferentes protocolos de comunicação, foram realizados para testar a interferência que esses dispositivos podem causar se forem utilizados dentro do mesmo ambiente. As topologias utilizadas nos testes anteriores foram migradas para um único gateway central, com 3 gateways secundários, onde cada um foi responsável por um tipo de dispositivo, como mostra a Figura 22.

Para simular uma utilização real, todos os objetos IoT foram utilizados dentro do mesmo ambiente. Os gateways secundários para dispositivos BLE e ZigBee foram colocados junto com os objetos IoT, enquanto o gateway secundário responsável pelos objetos Wi-Fi e o gateway central foram colocados em um ambiente separado, conectados a uma rede cabeada. O objetivo dos experimentos foi simular a utilização real em um cômodo de uma smart home. A Tabela 1 mostra a quantidade de cada tipo de dispositivo que foi utilizada em cada caso de testes. Para testar diretamente a interferência com o aumento na quantidade de outros dispositivos, decidiu-se por repetir a quantidade

Tabela 1 – Configuração dos Testes de Interoperabilidade

	Bluetoot LE	ZigBee	Wi-Fi
Caso de Teste 1	1	1	1
Caso de Teste 2	5	3	3
Caso de Teste 3	10	6	6
Caso de Teste 4	15	9	9
Caso de Teste 5	15	12	12

de 15 de dispositivos BLE ao invés de realizar o ultimo teste com uma quantidade de 20, que seria a sequência esperada considerando os testes isolados. Para os experimentos realizados, utilizou-se o termo *interoperabilidade* para denominar os testes realizados com os dispositivos BLE, ZigBee e Wi-Fi em conjunto. No gráficos apresentados, os resultados em azul são correspondentes aos dados obtidos com os testes realizados utilizando apenas o protocolo citado, enquanto os resultados em vermelho são referentes aos dados obtidos nos testes utilizando o protocolo conjunto com dispositivos de outros protocolos.

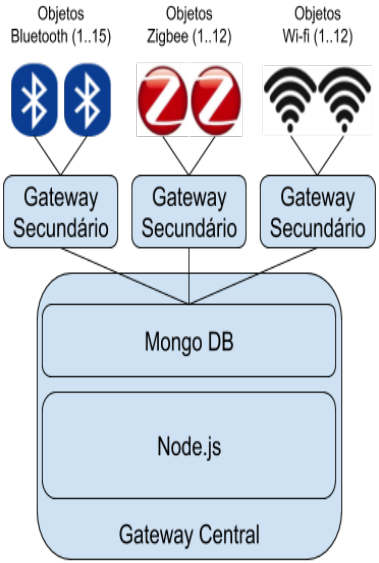

Figura 22 – Topologia Interoperabilidade.

5.3.1 Bluetooth LE

Os testes realizados para os objetos BLE utilizaram o mesmo microsserviço utilizado nos testes isolados. A Figura 23 mostra os resultados da quantidade de detecções nos testes em conjunto com os outros tipos de dispositivos em comparação com os testes isolados.

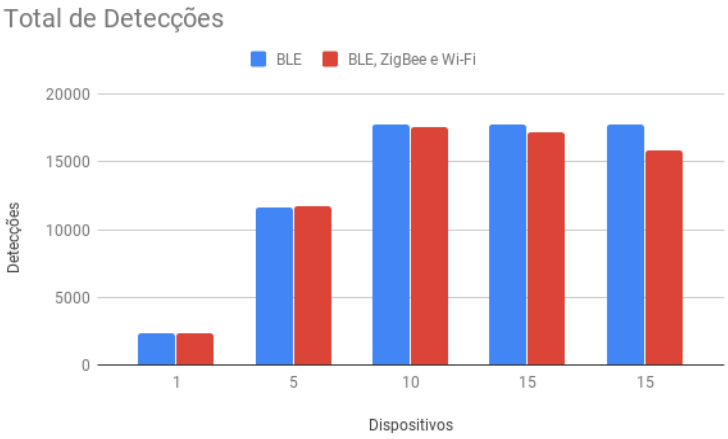

Figura 23 – Resultado Bluetooth - Total de Detecções.

É possível perceber no gráfico, que as quantidades de detecções nos testes isolados e de interoperabilidade se mantém muito próximas para os testes com 1, 5 e 10 dispositivos. A diferença aumentou levemente para os testes com 15 dispositivos, onde os outros gateways secundários possuíam 9 dispositivos ativos para cada protocolo (BLE e ZigBee). Por fim, a diferença mais acentuada pode ser vista na ultima bateria de testes, onde todos os dispositivos foram utilizados.

A Figura 24 mostra a comparação entre as quantidade média por dispositivos que foram encontrados nos testes isolados em comparação com os testes de interoperabilidade. Os valores de ambos os testes se mantiveram próximos em todas as quantidades de dispositivos. A diferença mais acentuada ocorreu na configuração com 15 objetos BLE, 12 ZigBee e 12 Wi-Fi. Nesse caso, a diferença ficou em 131, pouco mais de 10% to total.

Em relação a média do tempo de resposta, é possível perceber na Figura 25 que para os testes com 1 e 5 dispositivos, as médias se

mantiveram muito próximas. A diferença aumentou um pouco para os testes com 10 e 15 dispositivos, onde os testes de interoperabilidade ficaram aproximadamente 10% maior. A maior diferença ocorreu no ultimo teste, onde a média de tempo de resposta dos testes de interoperabilidade ficou quase 20% maior.

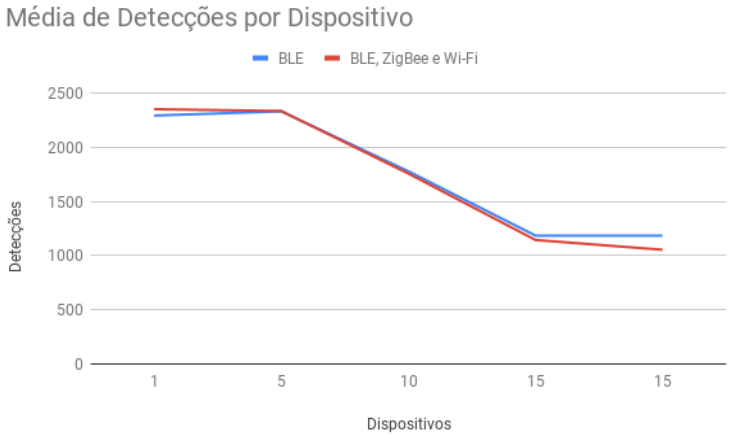

Figura 24 – Resultado Bluetooth - Média por Dispositivo.

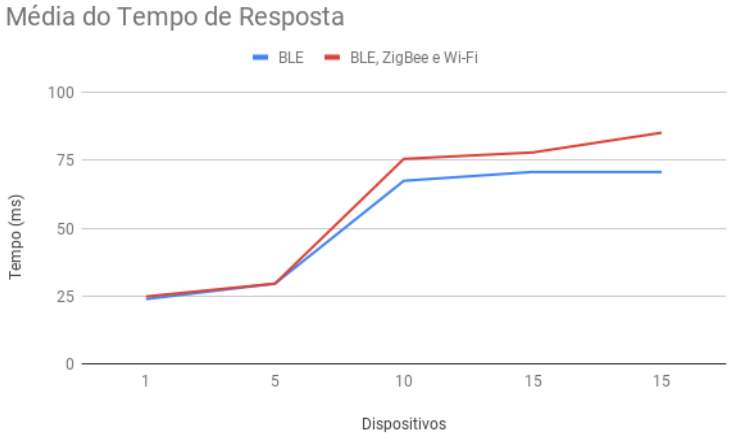

Figura 25 – Resultado Bluetooth - Média do Tempo de Resposta.

5.3.2 ZigBee

Para o gateway secundário que ficou responsável pelos dispositivos ZigBee, também foi utilizado o mesmo microsserviço dos testes isolados. A Figura 26 mostra os resultados da comparação entre o total de detecções nos testes isolados e nos testes de interoperabilidade.

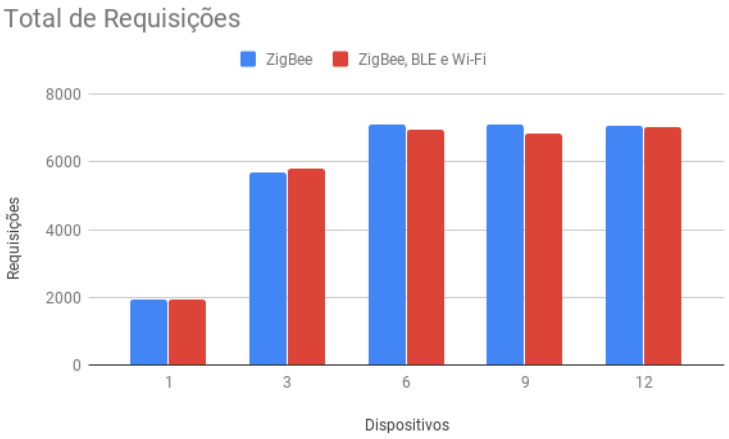

Figura 26 – Resultado ZigBee - Total de Detecções.

A quantidade de detecções dos testes se manteve muito próxima em todas as quantidades de objetos IoT testados. A diferença máxima observada foi de 2% a mais para os testes de interoperabilidade com 3 dispositivos e 3% a mais para os testes isolados, quando foram utilizados 6 objetos.

Assim como no gráfico anterior, é possível ver na Figura 27 que a média de detecções por dispositivo se manteve próxima em todas as quantidade de dispositivos testados. A diferença máxima entre os testes ocorreu na quantidade de 9 dispositivos, onde os testes isolados obtiveram uma média aproximadamente 3% maior.

Em relação à média de tempo de resposta, é possível ver na Figura 28 que a diferença obtida entre os testes isolados e de interoperabilidade foi um pouco maior que nos gráficos anteriores. Com uma média de tempo menor na configuração com 1 e 12 dispositivos, os testes de interoperabilidade apresentaram uma média maior nas outras 3 configurações. A maior diferença ocorreu no teste com 6 dispositivos,

onde os testes de interoperabilidade ficaram com uma média aproximadamente 7% maior.

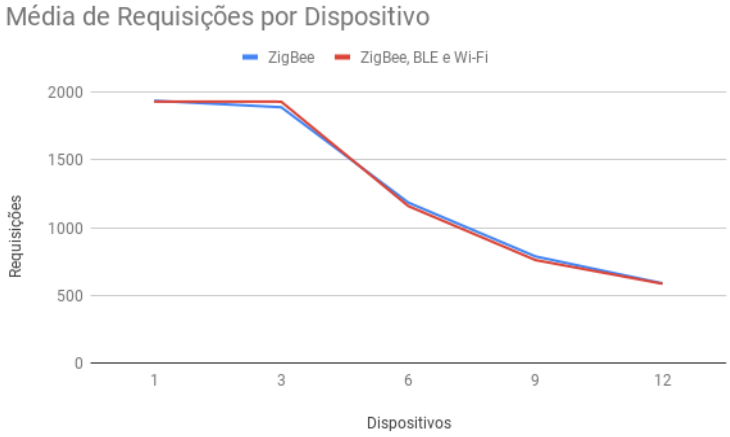

Figura 27 – Resultado ZigBee - Média por Dispositivo.

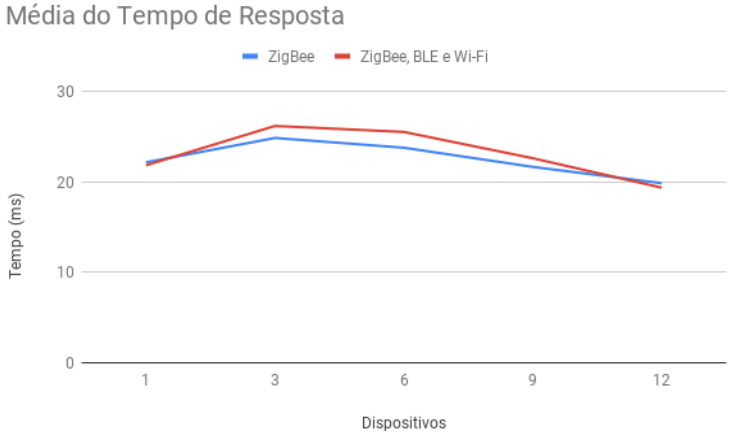

Figura 28 – Resultado ZigBee - Média do Tempo de Resposta.

5.3.3 Wi-Fi

Assim como com os outros dispositivos, os testes realizados com os objetos Wi-Fi utilizaram o mesmo microsserviço dos testes isolados. A Figura 29 mostra o resultado da comparação entre os testes isolados e de interoperabilidade.

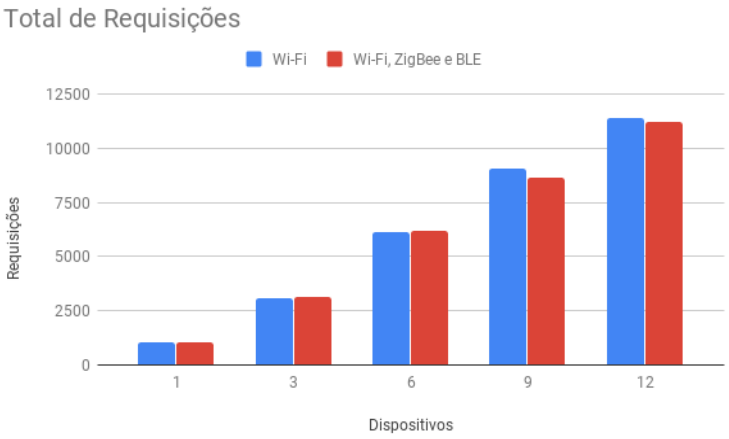

Figura 29 – Resultado Wi-Fi - Total de Detecções.

Como é possível ver no gráfico, a quantidade de detecções nos testes de interoperabilidade se mostraram um pouco superiores para as quantidades de 1, 3 e 6 dispositivos. Para os dois últimos testes (9 e 12 dispositivos) a quantidade de detecções dos testes isolados foi maior, encontrado seu ponto máximo de diferença com 9 dispositivos, onde chegou a ser quase 5% maior.

No gráfico referente a média de detecções por dispositivos (Figura 30), é possível perceber as médias se mantiveram próximas em quase todas as configurações de testes, tendo sua diferença máxima alcançada nos testes com 9 dispositivos, onde a média de detecção dos testes isolados chegou a alcançar uma diferença de 6% em relação aos testes de interoperabilidade. É possível notar uma relação com o gráfico anterior, uma vez que a quantidade de detecções cresceu quase que proporcionalmente, mantendo a média sempre próxima.

A Figura 31 mostra a comparação da evolução da média de tempo de resposta nos testes isolados e de interoperabilidade. A média

encontrada nos testes isolados permaneceu maior durante as configurações com 1, 3 e 6 dispositivos. Nos testes com 9 e 12 dispositivos as médias dos testes de interoperabilidade foram maiores, chegando a alcançar uma diferença aproximada de 26% com 9 dispositivos.

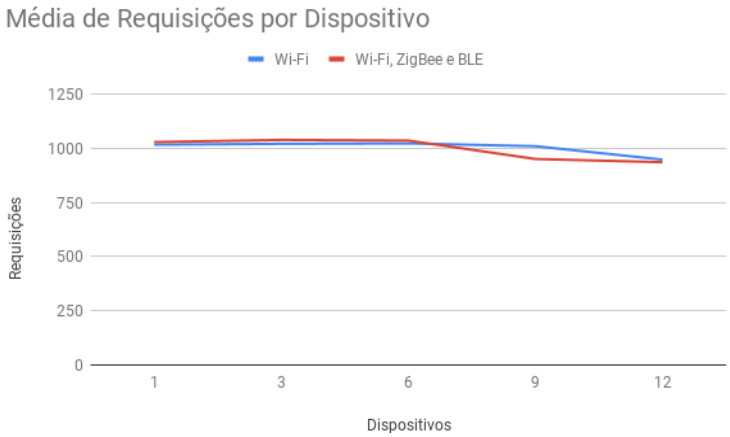

Figura 30 – Resultado Wi-Fi - Média por Dispositivo.

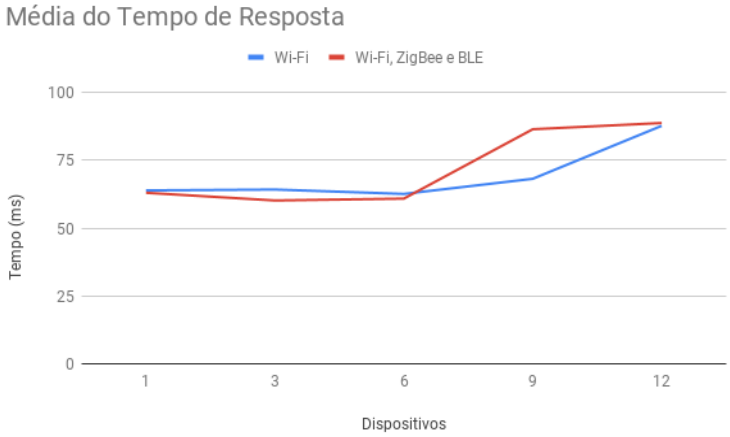

Figura 31 – Resultado Wi-Fi - Média do Tempo de Resposta.

5.4 CONSIDERAÇÕES

Neste capítulo foram apresentados os resultados dos testes realizados no Smart Comm para validação de algumas funcionalidades de administração do middleware, assim como a simulação de sua utilização em um ambiente real. Os testes foram planejados para verificar a interferência que a utilização de múltiplos protocolos de comunicação poderiam causar, uma vez que o Smart Comm foi proposto, entre outras coisas, para tentar viabilizar a utilização de dispositivos heterogêneos em um ambiente de Smart Home.

Para que fosse possível verificar a interferência, primeiramente foram realizados testes individuais com objetos de cada protocolo sugerido dentro do escopo deste trabalho, o BLE, ZigBee e Wi-Fi. Pensando em uma utilização em ambiente real, as topologias montadas foram pensadas de uma forma que os testes de interoperabilidade seguissem uma das topologias sugeridas no capitulo anterior, com um gateway para cada tipo de dispositivos em cada cômodo, podendo ter um gateway central por cômodo ou para a casa inteira. Uma vez que a simulação foi realizada em cima de um único cômodo, a utilização de apenas um ou de múltiplos gateways centrais não entrou no escopo.

Os testes foram realizados isoladamente para cada protocolo e após isso o mesmo teste foi realizado com todos os dispositivos conectados ao Smart Comm ao mesmo tempo. A quantidade de dispositivos de cada tipo que foi sendo adicionada a cada bateria de testes foi a mesma utilizada nos testes isolados, com exceção dos objetos bluetooth, onde se optou por repetir a configuração com 15 dispositivos em dois testes a fim de medir a interferência direta com a inserção de dispositivos de outros protocolos.

Em relação aos resultados obtidos, grande parte dos dados encontrados ficaram com valores muito próximos em ambos os testes. As diferenças mais significativas encontradas foram nas médias de tempo de resposta dos testes de interoperabilidade em relação aos testes isolados. Nos testes com Bluetooth essa diferença chegou a 20% enquanto nos testes com Wi-Fi encontraram seu maior valor, chegando a uma diferença de 26%. Foram ainda medidos os timeouts encontrados nas requisições realizadas aos microsserviços em cada bateria de testes. O valor mais expressivo encontrado foi de aproximadamente 0,02% de timeouts nos dispositivos bluetooth nos testes de interoperabilidade. Também foram medidos os pacotes inválidos gerados pelas placas Xbee, no protocolo ZigBee. Ocorreram apenas 2 erros em baterias distintas nos testes de interoperabilidade com 12 dispositivos, alcançando 0,014%.

6 CONCLUSÕES E TRABALHOS FUTUROS

Neste trabalho foi apresentado o middleware Smart Comm, que teve como objetivo viabilizar a interoperabilidade de dispositivos heterogêneos e a utilização de múltiplos protocolos dentro de um ambiente IoT, com uma aplicação focada para Smart Homes. A arquitetura proposta utilizou gateways secundários para comunicação com tipos específicos de dispositivos, e um gateway central para atividades relacionadas ao gerenciamento do middleware e para a centralização dos dados recebidos.

O gateway secundário foi desenvolvido pensando na flexibilidade que se faz necessária em um ambiente onde existe uma variedade muito grande de tipos de dispositivos e protocolos. Para isso, decidiu-se a utilização de uma abordagem utilizando microsserviços, garantindo que os componentes tenham uma certa independência entre si. Por lidar com dispositivos de baixo custo, foi decidido que para os dispositivos que necessitam armazenamento de dados, os mesmos ficariam centralizados. Embora essa abordagem contrarie parte dos princípios de separação em uma arquitetura de microsserviços, a utilização de múltiplos banco de dados poderia acarretar em problemas de performance.

Para o gateway central, foi desenvolvida uma aplicação voltada para o cadastro dos gateways secundários e objetos IoT disponíveis. As funcionalidades de criação de usuário, login e geração de tokens de autenticação para os gateways secundários também ficaram localizadas no gateway central. Para o cadastro desses dados, foi utilizada uma instância do banco de dados Mongo DB. Este, também ficou acessível aos microsserviços, centralizando os dados armazenados.

Foram desenvolvidos microsserviços para validação do Smart Comm na utilização de dispositivos de 3 protocolos: ZigBee, Bluetooth LE e Wi-Fi. Para os objetos Wi-Fi, foram utilizadas placas Wemos, e os serviços desenvolvidos foram instâncias para requisições HTTP. Para a utilização do BLE e ZigBee, dongles USB tiveram que ser instalados em seus respectivos gateways. Os microsserviços foram cadastrados junto com uma lista de identificadores de dispositivos que faziam parte de seu escopo. Em ambos os casos, a cada detecção realizada, os dados eram encaminhados para todos os serviços vinculados ao identificador daquele objeto. Foram realizados testes aumentando a quantidade de dispositivos de cada tipo visando gerar dados para comparação com os testes de interoperabilidade.

Para validação do uso dos dispositivos em conjunto, foi montada

uma arquitetura utilizando 1 gateway central e 3 gateways secundários, cada um responsável por um tipo de dispositivo. A quantidade de objetos IoT foi sendo incrementada até chegar a configuração final, utilizando 12 objetos ZigBee, 12 Wi-Fi e 15 Bluetooth. Embora na maior parte do experimento os resultados tenham ficado próximos dos resultados isolados, os dados coletados mostraram diferenças na quantidade de detecções totais, média de detecção por dispositivo e média no tempo de resposta. As maiores diferenças foram encontradas na média do tempo de resposta, alcançando um percentual de 26%. Para o total de detecções e média de detecções por dispositivos, a maior diferença encontrada chegou ao percentual de 12%.

Embora tenha havido uma diferença considerável no aumento do tempo de resposta em alguns casos, a quantidade de timeouts e pacotes inválidos, 0,02% e 0,014% respectivamente, leva a crer que a aplicação de uma arquitetura similar à utilizada nos testes, em um ambiente real, pode ser viabilizada. Uma vez que os testes foram programados para simular uma quantidade de requisições que em um cenário real ficariam muito acima do normal para um cômodo de uma smart home, há uma tendência de queda na diferença dos resultados apresentados.

Como trabalhos futuros, fica a necessidade de testes para ambientes completos, a fim de validar qual a melhor arquitetura de utilização do Smart Comm. A utilização de um único gateway central para a casa inteira ou um por cômodo deve ser testada para medir o tamanho da interferência que múltiplos cômodos podem causar. Além disso, podem ser testados layouts alternativos, com um gateway secundário sendo utilizado para se comunicar com mais de um tipo de dispositivo.

Também se faz necessário um aprofundamento na utilização dos recursos do Smart Comm. Conforme dito anteriormente, um dos diferenciais propostos foi a utilização de sua propria estrutura como um ambiente Fog Computing, ao invés de enviar as informações para um ambiente em nuvem. Embora o Smart Comm seja voltado para utilização de dispositivos de baixo custo, deve ser analisado o uso dos gateways em conjunto para realizar tarefas que seriam inviáveis individualmente. Dessa forma todo processamento necessário para algumas tarefas poderiam ser realizados localmente.

REFERÊNCIAS

AGOSTINHO, B. M. et al. Smart comm: A smart home middleware supporting information exchange. *Accepted on 44th Annual Conference of the IEEE Industrial Electronics Society, IECON.*, 2018.

AL-FUQAHA, A. et al. Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE Communication Surveys & Tutorials.*, v. 17, n. 4, 2015.

ALAM, M. R.; REAZ, M. B. I.; ALI, M. A. M. A review of smart homes: Past, present, and future. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, v. 42, n. 6, p. 1190–1203, Nov 2012.

ATZORI, L.; IERA, A.; MORABITO, G. The internet of things: A survey. *Comput. Netw.*, Elsevier North-Holland, Inc., New York, NY, USA, v. 54, n. 15, p. 2787–2805, out. 2010. ISSN 1389-1286. <http://dx.doi.org/10.1016/j.comnet.2010.05.010>.

AUGUSTIN, A. et al. A study of lora: Long range amp; low power networks for the internet of things. *Sensors*, v. 16, n. 9, 2016. ISSN 1424-8220. <http://www.mdpi.com/1424-8220/16/9/1466>.

BARONTI, P. et al. Wireless sensor networks: A survey on the state of the art and the 802.15.4 and zigbee standards. *Computer Communications*, v. 30, n. 7, p. 1655 – 1695, 2007. ISSN 0140-3664. Wired/Wireless Internet Communications. <http://www.sciencedirect.com/science/article/pii/S0140366406004749>.

BEDHIEF, I.; KASSAR, M.; AGUILI, T. Sdn-based architecture challenging the iot heterogeneity. In: *2016 3rd Smart Cloud Networks Systems (SCNS)*. [S.l.: s.n.], 2016. p. 1–3.

BELLAVISTA, P. et al. Convergence of manet and wsn in iot urban scenarios. *IEEE Sensors Journal*, v. 13, n. 10, p. 3558–3567, Oct 2013. ISSN 1530-437X.

BONOMI, F. et al. Fog computing and its role in the internet of things. In: *Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing*. New York, NY, USA: ACM, 2012. (MCC '12), p. 13–16. ISBN 978-1-4503-1519-7. <http://doi.acm.org/10.1145/2342509.2342513>.

BOTTARO, A.; GÉRODOLLE, A. Home soa -: Facing protocol heterogeneity in pervasive applications. In: *Proceedings of the 5th International Conference on Pervasive Services*. New York, NY, USA: ACM, 2008. (ICPS '08), p. 73–80. ISBN 978-1-60558-135-4. <http://doi.acm.org/10.1145/1387269.1387284>.

DASTJERDI, A. V.; BUYYA, R. Fog computing: Helping the internet of things realize its potential. *Computer*, v. 49, n. 8, p. 112–116, Aug 2016. ISSN 0018-9162.

DRAGONI, N. et al. Microservices: Yesterday, today, and tomorrow. In: _____. *Present and Ulterior Software Engineering*. Cham: Springer International Publishing, 2017. p. 195–216. ISBN 978-3-319-67425-4. <https://doi.org/10.1007/978-3-319-67425-4$_1$2>.

DUBEY, H. et al. Fog data: Enhancing telehealth big data through fog computing. In: *Proceedings of the ASE BigData & SocialInformatics 2015*. New York, NY, USA: ACM, 2015. (ASE BD&SI '15), p. 14:1–14:6. ISBN 978-1-4503-3735-9. <http://doi.acm.org/10.1145/2818869.2818889>.

FOWLER, M. *Microservices: a definition of this new architectural term*. 2014. Https://martinfowler.com/articles/microservices.html. Acessado: 2019-01-11.

GAIKWAD, P. P.; GABHANE, J. P.; GOLAIT, S. S. A survey based on smart homes system using internet-of-things. In: *2015 International Conference on Computation of Power, Energy, Information and Communication (ICCPEIC)*. [S.l.: s.n.], 2015. p. 0330–0335.

GIA, T. N. et al. Fog computing in healthcare internet of things: A case study on ecg feature extraction. In: *2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing*. [S.l.: s.n.], 2015. p. 356–363.

GOMEZ, C.; OLLER, J.; PARADELLS, J. Overview and evaluation of bluetooth low energy: An emerging low-power wireless technology. *Sensors*, v. 12, n. 9, p. 11734–11753, 2012. ISSN 1424-8220. <http://www.mdpi.com/1424-8220/12/9/11734>.

HELAL, A.; COOK, D. J.; SCHMALZ, M. Smart home-based health platform for behavioral monitoring and alteration of diabetes patients.

Journal of Diabetes Science and Technology, v. 3, n. 1, p. 141–148, 2009. PMID: 20046657. <https://doi.org/10.1177/193229680900300115>.

HOU, X. et al. Vehicular fog computing: A viewpoint of vehicles as the infrastructures. *IEEE Transactions on Vehicular Technology*, v. 65, n. 6, p. 3860–3873, June 2016. ISSN 0018-9545.

Jara, A. J.; Martinez-Julia, P.; Skarmeta, A. Light-weight multicast dns and dns-sd (lmdns-sd): Ipv6-based resource and service discovery for the web of things. In: *2012 Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*. [S.l.: s.n.], 2012. p. 731–738.

LIU, L. et al. Smart homes and home health monitoring technologies for older adults: A systematic review. *International Journal of Medical Informatics*, v. 91, p. 44 – 59, 2016. ISSN 1386-5056. <http://www.sciencedirect.com/science/article/pii/S1386505616300648>.

Luzuriaga, J. E. et al. A comparative evaluation of amqp and mqtt protocols over unstable and mobile networks. In: *2015 12th Annual IEEE Consumer Communications and Networking Conference (CCNC)*. [S.l.: s.n.], 2015. p. 931–936. ISSN 2331-9852.

MOAZZAMI, M. et al. Spot: A smartphone-based platform to tackle heterogeneity in smart-home iot systems. In: *2016 IEEE 3rd World Forum on Internet of Things (WF-IoT)*. [S.l.: s.n.], 2016. p. 514–519.

NGU, A. H. et al. Iot middleware: A survey on issues and enabling technologies. *IEEE Internet of Things Journal.*, v. 4, n. 1, February 2017.

PERUMAL, T.; RAMLI, A. R.; LEONG, C. Y. Interoperability framework for smart home systems. *IEEE Transactions on Consumer Electronics*, v. 57, n. 4, p. 1607–1611, November 2011. ISSN 0098-3063.

RAHMAN, T.; CHAKRABORTY, S. K. Provisioning technical interoperability within zigbee and ble in iot environment. In: *2018 2nd International Conference on Electronics, Materials Engineering Nano-Technology (IEMENTech)*. [S.l.: s.n.], 2018. p. 1–4.

ROTTA, G.; DANTAS, M. A. R. Um estudo sobre protocolos de comunicação para ambientes de internet das coisas. *Escola Regional de Alto Desempenho.*, 2017.

SAINT-ANDRE, P. et al. *XMPP: The Definitive Guide*. O'Reilly Media, 2009. (O'Reilly Series). ISBN 9780596521264. <https://books.google.com.br/books?id=ChTCQYLIDfoC>.

SARKAR, C. et al. A scalable distributed architecture towards unifying iot applications. In: *2014 IEEE World Forum on Internet of Things (WF-IoT)*. [S.l.: s.n.], 2014. p. 508–513.

SIEKKINEN, M. et al. How low energy is bluetooth low energy? comparative measurements with zigbee/802.15.4. In: *2012 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*. [S.l.: s.n.], 2012. p. 232–237.

SOUZA, A. M. C.; AMAZONAS, J. R. A. A novel smart home application using an internet of things middleware. *Smart SysTech.*, June 2013.

TANG, B. et al. A hierarchical distributed fog computing architecture for big data analysis in smart cities. In: *Proceedings of the ASE BigData & SocialInformatics 2015*. New York, NY, USA: ACM, 2015. (ASE BD&SI '15), p. 28:1–28:6. ISBN 978-1-4503-3735-9. <http://doi.acm.org/10.1145/2818869.2818898>.

THöNES, J. Microservices. *IEEE Software*, v. 32, n. 1, p. 116–116, Jan 2015. ISSN 0740-7459.

VAQUERO, L. M.; RODERO-MERINO, L. Finding your way in the fog: Towards a comprehensive definition of fog computing. *SIGCOMM Comput. Commun. Rev.*, ACM, New York, NY, USA, v. 44, n. 5, p. 27–32, out. 2014. ISSN 0146-4833. <http://doi.acm.org/10.1145/2677046.2677052>.

XIAO, G. et al. User interoperability with heterogeneous iot devices through transformation. *IEEE Transactions on Industrial Informatics*, v. 10, n. 2, p. 1486–1496, May 2014. ISSN 1551-3203.

XU, L. D.; HE, W.; LI, S. Internet of things in industries: A survey. *IEEE Transactions on Industrial Informatics*, v. 10, n. 4, p. 2233–2243, Nov 2014. ISSN 1551-3203.

ZHILIANG, W. et al. A soa based iot communication middleware. In: *2011 International Conference on Mechatronic Science, Electric Engineering and Computer (MEC)*. [S.l.: s.n.], 2011. p. 2555–2558.

APÊNDICE A – Artigo Publicado

Smart Comm: A Smart Home Middleware Supporting Information Exchange

44th Annual Conference of the IEEE Industrial Electronics Society

Qualis: B1

Abstract: In the research it has been noticed a large interest in academia and industry related to the adoption of Internet of Things (IoT) and Smart Homes. Although there are several approaches utilizing those environments, no common consensus exists on technology and no standards for devices communication, data collection and processing exist. Therefore, in this work is conceived to tackle these challenges, which was coined as the Smart Comm middleware. This software approach was designed for use in an IoT environment, with special emphasis on smart homes. Thus, serving as a basis to support information exchange. The Smart Comm was developed in a scenario architecture adopting microservices. As a result, the communication between the gateway and IoT devices can be done considering a peer communication. This procedure can also be observed in the exchange information between Smart Comm. Therefore, bringing security, scalability and the flexibility to handle the variety of devices. Targeting to demonstrate the feasibility and scalability of the proposed middleware, an environment was developed to serve as a testbed for this research work.

Index Terms: Smart Home, Internet of Things, Middleware, Gateways, Microservices