UNIVERSIDADE DE PASSO FUNDO
Programa de Pós-Graduação em Computação Aplicada

Dissertação de Mestrado

IOT NA SEGURANÇA PÚBLICA COMO TÉCNICA PARA IDENTIFICAR FURTO E ROUBO DE VEÍCULOS EM AMBIENTE SMART CAMPUS

FERNANDO POSSER PINHEIRO

UPF

UNIVERSIDADE DE PASSO FUNDO
INSTITUTO DE CIÊNCIAS EXATAS E GEOCIÊNCIAS
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO APLICADA

IOT NA SEGURANÇA PÚBLICA COMO TÉCNICA PARA IDENTIFICAR FURTO E ROUBO DE VEÍCULOS EM AMBIENTE SMART CAMPUS

Fernando Posser Pinheiro

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre em Computação Aplicada na Universidade de Passo Fundo.

Orientador: Prof. Dr. Roberto Rabello

Passo Fundo
2021

CIP – Catalogação na Publicação

P654i Pinheiro, Fernando Posser
IoT na segurança pública como técnica para identificar furto e roubo de veículos em ambiente smart campus / Fernando Posser Pinheiro. – 2021.
104 f. : il. ; 30 cm.

Orientador: Prof. Dr. Roberto Rabello.
Dissertação (Mestre em Computação Aplicada) – Universidade de Passo Fundo, 2021.

1. Internet das coisas. 2. Segurança pública. 3. SMART (Sistema de recuperação da informação). I. Rabello, Roberto, orientador. II. Título.

CDU: 004.7

Catalogação: Bibliotecário Luís Diego Dias de S. da Silva – CRB 10/2241

PPGCA
Programa de Pós-Graduação
em Computação Aplicada
Instituto de Ciências Exatas e Geociências | ICEG

ATA DE DEFESA DO TRABALHO DE CONCLUSÃO DE CURSO DO ACADÊMICO

FERNANDO POSSER PINHEIRO

Aos vinte e seis dias do mês de março do ano de dois mil e vinte e um, às 14 horas, realizou-se, de forma on-line, por meio de videoconferência, a sessão pública de defesa do Trabalho de Conclusão de Curso "IoT na Segurança Pública como Técnica para Identificar Furto e Roubo de Veículos em Ambiente Smart Campus", de autoria de Fernando Posser Pinheiro, acadêmico do Curso de Mestrado em Computação Aplicada do Programa de Pós-Graduação em Computação Aplicada – PPGCA. Segundo as informações prestadas pelo Conselho de Pós-Graduação e constantes nos arquivos da Secretaria do PPGCA, o aluno preencheu os requisitos necessários para submeter seu trabalho à avaliação. A banca examinadora foi composta pelos doutores Roberto dos Santos Rabello (Orientador), Marcelo Trindade Rebonatto (UPF), Paulo Antonio Leal Rego (UFC) e Adenauer Corrêa Yamin (UFPel). Concluídos os trabalhos de apresentação e arguição, a banca examinadora considerou o candidato **APROVADO**. Foi concedido o prazo de até quarenta e cinco (45) dias, conforme Regimento do PPGCA, para a acadêmica apresentar ao Conselho de Pós-Graduação o trabalho em sua redação definitiva, a fim de que sejam feitos os encaminhamentos necessários à emissão do Diploma de Mestre em Computação Aplicada. Para constar, foi lavrada a presente ata, que vai assinada pelos membros da banca examinadora e pela Coordenação do PPGCA.

Prof. Dr. Roberto dos Santos Rabello – UPF
Presidente da Banca Examinadora
(Orientador)

Prof. Dr. Marcelo Trindade Rebonatto – UPF
(Avaliador Interno)

p.p.
Prof. Dr. Paulo Antonio Leal Rego – UFC
(Avaliador Externo)

p.p.
Prof. Dr. Adenauer Corrêa Yamin – UFPel
(Avaliador Externo)

Prof. Dr. Carlos Amaral Hölbig
Coordenador do PPGCA

Dedico meus esforços e resultados alcançados ao longo da vivência no curso de mestrado à minha falecida mãe, que partiu cedo, mas que sozinha me criou e educou diante todas as adversidades; e que também me inspirou a refletir sobre tantas coisas, incluindo a segurança da vida, que ao longo do tempo me abriu portas para utilizar a tecnologia, área que tanto amo, para pensar em soluções que possam um dia levar um pouco de paz e tranquilidade ao povo brasileiro.

AGRADECIMENTOS

Agradeço ao Parque Tecnológico da UPF, por ceder prontamente, inclusive durante o período de quarentena provocado pela pandemia do COVID-19, os equipamentos necessários para a realização deste trabalho. Aos meus amigos e familiares, por todo o suporte dado desde o começo da minha trajetória no ensino superior. À Yasmin Damiani, por todo o apoio incondicional nos momentos mais difíceis provocados não apenas pela carga de demanda do curso de mestrado, mas também por todas as minhas tentativas de abraçar o mundo. Aos professores do PPGCA e ex-professores do bacharel, por incentivarem constantemente a realização do curso de mestrado. Ao professor Roberto Rabello, por me orientar, tirar as centenas de dúvidas que tive durante a realização deste trabalho e pelas ótimas conversas sobre ambientes inteligentes e empreendedorismo. Por fim, agradeço aos mestres, instrutores e colegas de tatame da Alliance Jiu-jitsu Passo Fundo, por manterem minha mente plena durante os momentos de inquietação. Oss!

“Observa o teu interior. A fonte do bem está em teu interior, fonte capaz de jorrar sempre desde que tu cavez sempre.”
(Imperador Marco Aurélio. 121 DC - 180 DC)

IOT NA SEGURANÇA PÚBLICA COMO TÉCNICA PARA IDENTIFICAR FURTO E ROUBO DE VEÍCULOS EM AMBIENTE SMART CAMPUS

RESUMO

Juntamente com as áreas de saúde e educação, a segurança pública é uma das principais preocupações sociais da população brasileira. Nos últimos anos, o Brasil tem se tornado um país com altos índices de criminalidade, o que afeta diretamente o comportamento socioeconômico dentro do território nacional. Em contrapartida ao cenário dos índices de segurança, as *smart cities* proporcionam cada vez melhores condições de vida e acesso à cultura para a população. Seguindo este mesmo modelo estão os *smart campus*, que possuem necessidades similares às encontradas nas cidades e aproveitam do potencial da comunidade acadêmica para o desenvolvimento de soluções aos problemas locais mais relevantes. Em ambientes inteligentes, o Bluetooth é uma das principais tecnologias que contribui na criação de aplicações IoT, principalmente após a versão 4 que introduziu o BLE (Bluetooth *Low Energie*). Foi após essa versão que surgiram dispositivos BLE beacons, o que possibilitou uso do chamado *indoor location*. Sua principal característica é a capacidade de prover dados sobre a aproximação entre dispositivos por meio de emissão de sinais, estes que podem ser gerados e também identificados por qualquer programa executado em um aparelho que possua tecnologia BLE. A maior parte das aplicações encontradas em bibliografia que empregam aproximação apontam para uso a curtas distâncias e em objetos fixos, não aproveitando a total capacidade dos dispositivos. Além disso, destacam problemas de oscilação destes sinais, dificultando a implementação de determinadas funcionalidades. Neste contexto, foi realizada uma pesquisa de abordagem experimental buscando viabilizar o uso de IoT com o objetivo de identificar furto e roubo de veículos em ambiente *smart campus* através de BLE beacons. Foram realizados dois experimentos em ambiente aberto, o primeiro busca identificar a área de atuação dos dispositivos; e observar o comportamento de oscilação de sinais a fim de traçar uma estratégia para o desenvolvimento de uma aplicação que possa identificar a movimentação de um beacon. O segundo experimento foi realizado com uma aplicação que gerencia o monitoramento de um BLE beacon instalado no interior de um veículo, que ao movimentar-se no período de vigilância faz com que o sistema identifique seu deslocamento e emita um alerta de risco indicando que o veículo está em situação de risco.

Palavras-Chave: internet das coisas, ble beacons, segurança pública, smart cities, smart campus.

IOT IN PUBLIC SAFETY AS TECHNIQUE FOR THEFT AND ROBBERY IDENTIFY OF VEHICLES IN SMART CAMPUS ENVIRONMENT

ABSTRACT

Along with health and education, public safety is one of the main brazilian social concerns. In the last years, Brazil has being a country with high level criminal indices, which directly affects the socioeconomic behavior inside national territory. In other hand along this scenario of safety's indication, the smart cities provide more and more better life conditions and access to culture for the citizens. Alongside with that there are the smart campus, having the same necessities find on the cities and taking the academic community potential to develop solutions to those more relevants local problems. In smart environments, bluetooth is one of the main technologies that contrib on creation of IoT applications, mostly after version 4 that introduces BLE (Bluetooth Low Energie). It was after that version that BLE beacons devices appear, what enable the use of indoor location. Your main property is the capacity to provide data about devices aproximation through signal emission, that can be generated and detected by any executed program in an equipment having BLE technology. The applications majoroty part find in bibliografhy using aproximation, aim to the implement this characteristic in short distances and in fixed objects, not taking the full devices advantage. Futhermore, shows oscillations problems, making it dificult to implement some specific features. In this context, it was performed a research using experimental approach looking for make feasible the use of IoT in objective to identify vehicles theaft and robbery in smart campus enviroment by using BLE beacons. Were fulfilled two expriments in open enviroment, first seeking for identify the devices operational areas and observe the oscillation behavior of signals to taking a develope strategy for an application that could identity the beacon moviment. Second experiment it was made with a program that manages BLE beacon monitoring inside a vehicle, that on moving on surveillance period makes the system recognize it and sending alert registering the risk situation.

Keywords: internet of the things, ble beacons, public safety, smart cities, smart campus.

LISTA DE FIGURAS

Figura 1. Funil de seleção de trabalhos 33
Figura 2. Relação de trabalhos encontrados 33
Figura 3. Relação dos trabalhos selecionados 34
Figura 4. Exemplo de transmissão a partir de tecnologia BLE enquanto um smartphone a recebe 42
Figura 5. Caso de uso dos identificadores do protocolo iBeacon 43
Figura 6. PubSub *workfkow* 49
Figura 7. Fluxograma Metodológico 53
Figura 8. Fluxograma do sistema 55
Figura 9. Fluxograma do sistema 55
Figura 10. Diagrama Entidade Relacionamento do Veacon Web 60
Figura 11. Beacon Manager 62
Figura 12. Funcionamento BeaconManager 63
Figura 13. Representação do objeto Watchpost 65
Figura 14. Watchpost Manager 66
Figura 15. Watchpost Manager - Funcionamento 68
Figura 16. Regra de alertas e atualizações 69
Figura 17. Cadastro de monitoramento no Veacon Web 71
Figura 18. Configuração do monitoramento 72
Figura 19. RSSI próximo, RSSI distante e mediana de RSSIs em situação normal 72
Figura 20. Identificação de anomalia 73
Figura 21. Identificação de deslocamento e envio de alerta 74
Figura 22. Tela de alertas 75
Figura 23. Distância alcançada pelo kit de BLE Beacons utilizados nos testes 78
Figura 24. Checklist do teste #1 79
Figura 25. Configuração do BLE beacon Estimote 81
Figura 26. Resultado de obtenção de dados do teste 1 (ordenados) 83
Figura 27. Resultado de obtenção de dados do teste 1 (desordenados) 84
Figura 28. Demonstração dos dados que podem causar falsos positivos 85
Figura 29. Intervalo vs. intervalo de transmissão 87
Figura 30. BLE beacon fixado no interior do veículo 88

Figura 31. Descrição visual do teste 2 88
Figura 32. Posição inicial do veículo 89
Figura 33. Distanciamento do veículo 90
Figura 34. Aproximação do veículo 91

LISTA DE TABELAS

Tabela 1. Palavras chave 31
Tabela 2. Divisão dos trabalhos encontrados por fonte de pesquisa 32
Tabela 3. Ambientes *smart* 35
Tabela 4. Aplicações IoT e caso de uso em ambientes *smart* 38
Tabela 5. Identificadores do protocolo iBeacon 42
Tabela 6. Especificação do protocolo Eddystone [53] 44
Tabela 7. Divisão de testes 77

LISTA DE ABREVIATURAS

pub/sub. – *Publisher Subscriber*

LISTA DE SIGLAS

API – *Application Programming Interface*

BLE – Bluetooth *Low Energie*

CRISP – Centro de Estudos de Criminalidade e Segurança Pública

IOT – *Internet of Things* - Internet das Coisas

LBS – Serviço Baseado em Localização

MVC – *Model View Controller*

MVP – *Minimum Viable Product - Produto Mínimo Viável*

REST – *Representational State Transfer*

RSS – *Received Strength Signal*

RSSI – *Received Strength Signal Indication*

SDK – *Software Development Kit* - Kit de Desenvolvimento de Software

SENASP – Secretaria Nacional de Segurança Pública

SIG – *Special Interest Group*

SSP – Secretaria de Segurança Pública

TI – Tecnologia da Informação

SUMÁRIO

1 INTRODUÇÃO 25
1.1 OBJETIVOS 30
1.2 OBJETIVO GERAL 30

2 FUNDAMENTAÇÃO TEÓRICA E REVISÃO SISTEMÁTICA 31
2.1 REVISÃO SISTEMÁTICA 31
2.2 MÉTODO 31
2.3 PRINCIPAIS TRABALHOS 33
2.3.1 ***Smart Campus*** 34
2.3.2 ***Internet of Things*** **(Internet das Coisas - IoT)** 38
2.3.3 **Bluetooth *Low Energy* e BLE Beacons** 39
2.3.3.1 iBeacon 41
2.3.3.2 Eddystone 43
2.3.3.3 RSSI e Tipos de Aplicação 45
2.3.3.4 *Deploy* de BLE Beacons 46
2.3.3.5 Vulnerabilidades de Segurança em BLE Beacons 47
2.4 TECNOLOGIAS INDICADAS PARA O DESENVOLVIMENTO DA BASE DE UM SISTEMA IOT 48
2.4.1 ***Publish Subscribe*** 48
2.4.2 **Django Rest e Django Web Framework** 50
2.5 RESULTADOS E DISCUSSÕES 51

3 MATERIAIS E MÉTODOS 53
3.1 METODOLOGIA 54
3.2 FLUXO E FUNCIONAMENTO DO SISTEMA 54
3.3 BLE BEACONS E AQUISIÇÃO DE DADOS 56
3.4 *PUBLISHER SUBSCRIBER* 57
3.5 SISTEMA WEB DE GERENCIAMENTO E ENVIO DE ALERTAS 57
3.5.1 **BLE Beacons** 58
3.5.2 ***Gateways*** **de monitoramento** 58
3.5.3 **Usuários** 59
3.5.4 **Veículos** 59

3.5.5 **Monitoramentos** 59
3.6 SISTEMA EMBARCADO DE MONITORAMENTO DE BLE BEACONS 59
3.6.1 ***Server Request*** 61
3.6.2 **PubSub** 61
3.6.3 **Beacons** 61
3.6.4 **Watchpost** 64
3.6.5 **Core** 69
3.7 FUNCIONAMENTO DO SISTEMA 69
3.7.1 **Cadastro de monitoramento no sistema Veacon Web** 70
3.7.2 **Cadastro de monitoramento no Veacon Rasp** 70
3.7.3 **Método de identificação e formação de alertas** 71
3.8 TESTES 75

4 RESULTADOS E DISCUSSÕES **81**
4.1 CONFIGURAÇÃO DOS EQUIPAMENTOS 81
4.2 CENÁRIO E COLETA DE DADOS DO TESTE #1 82
4.2.1 **Resultados obtidos durante o teste #1** 83
4.3 CENÁRIO DE APLICAÇÃO DO TESTE 2 86
4.3.1 **Resultados obtidos no teste 2** 89

5 CONCLUSÃO **93**

6 TRABALHOS FUTUROS **95**

REFERÊNCIAS **99**

1. INTRODUÇÃO

No Brasil, junto à questões como saúde, educação e combate à corrupção, um dos assuntos populares mais importantes dos últimos anos foi a segurança pública. Em 2018, foi o assunto mais comentado no país na plataforma Facebook [1], maior rede social do mundo. Ainda em 2018, durante as eleições, o tema ganhou grande repercussão nos debates e nas entrevistas pré-eleitorais e eleitorais dos candidatos à presidência [2]. A situação de insegurança do país preocupa o povo independentemente de classe e status social, credo ou nível de escolaridade. O alto índice de homicídios intencionais chama atenção e é comparável à países em situação de guerra [3].

Além da preocupação, a sociedade brasileira é uma das que mais sofre com a violência no mundo. Em 2018, a edição do Anuário Brasileiro de Segurança Pública demonstrou que em 2017 houveram 63.895 mortes violentas intencionais, sendo mais de 55 mil homicídios dolosos [3]. Este número mostra um aumento de mais de 2.500 mortes desde 2016, onde houveram 61.286 mortes violentas em território nacional [4], um aumento de quase 3%.

A segurança é uma das principais necessidades do indivíduo, como demonstrado por Maslow em seu estudo do comportamento humano [5] [1]. Através da hoje conhecida Pirâmide de Maslow, o autor ranqueou, de baixo para cima, as principais necessidades do ser humano. Após a primeira posição, que é denominada como necessidades fisiológicas básicas, está a necessidade de segurança, e acima desta, respectivamente, estão outras três necessidades de questões sociais de interação humana, auto-estima e realizações pessoais.

O que rege a ideia por trás da pirâmide de Maslow é que o ser humano busca satisfazer as necessidades acima apenas quando as necessidades abaixo forem atendidas. Seguindo este pensamento, o ser humano apenas terá auto-estima plena após as necessidades de interação humana serem saciadas, e as realizações pessoais serão alcançadas apenas quando a auto-estima for obtida [6]. Em 2013 uma pesquisa da Secretaria Nacional de Segurança Pública (SENASP), DATAFOLHA e o Centro de Estudos de Criminalidade e Segurança Pública (CRISP) demonstrou que “Questionados se deixam de fazer certas atividades por causa da violência, percebe-se que o sentimento de insegurança e o medo de vitimização trazem grandes impactos para a vida dos entrevistados” [7].

Dois anos depois, em 2015, uma matéria que explora um estudo da Organização para Cooperação e Desenvolvimento Econômico (OCDE), veiculada pela Folha de São Paulo, mostra mais de perto como a sensação de insegurança afeta a sociedade brasileira.

[1] Em termos, a Pirâmide de Maslow é geralmente utilizada para explicar o comportamento do consumidor em estudos de áreas de negócios. Neste trabalho, está sendo abordada a perspectiva de comportamento da sociedade e a influência que isso acarreta na qualidade de vida e desenvolvimento econômico do país.

> “Em uma comparação com 36 países, os brasileiros são os que se sentem menos seguros ao caminhar sozinhos à noite na cidade em que vivem, segundo um relatório divulgado nesta terça-feira pela OCDE (Organização para a Cooperação e Desenvolvimento Econômico, grupo que reúne majoritariamente países ricos)“. [8]

Em 2016, o Brasil já acumulava 30 vezes mais homicídios que toda a Europa [9]. A taxa de homicídios não é o único dado para analisar a segurança de um país, mas é o índice que mais se destaca ao analisar os números de estudos sobre segurança pública. Levando em consideração a análise sobre a Pirâmide de Maslow e os números apresentados até agora, pode-se imaginar que os reflexos da insegurança no âmbito social são profundos. O impacto da insegurança, representado pela segunda posição da pirâmide, não se reflete negativamente apenas no âmbito individual, uma vez que cada indivíduo impacta diretamente no padrão de consumo da massa da sociedade, e por consequência a indústria e a economia também são afetados. Comparando a taxa de homicídios brasileira de 2016 com a Europa, China, Estados Unidos, Indonésia, Oceania e África do Sul, Gesner Oliveira demonstra como em 2018 a falta de segurança já atingia a economia nacional e prejudicava empresas e instituições [10], diminuindo investimentos no país e aumentando a quantidade de despesas passivas na economia da máquina pública.

Como destacado neste capítulo, de 2016 a 2018 houve um aumento nas taxas de homicídio. Tendo isso em vista, podemos considerar que a sensação de insegurança da população também aumentara. Não se sabe exatamente o quanto estes números influenciam no arrastar do desenvolvimento socioeconômico do Brasil, mas, como visto, a sensação de segurança afeta diretamente o avanço do primeiro e segundo setor. Isso pode significar que o ritmo em que o país se desenvolve poderia ser positivamente diferente se a situação da segurança pública estivesse estabilizada. Por consequência, haveria melhores retornos de serviços que influenciam na qualidade de vida da população, gerando um ciclo virtuoso. Neste contexto, métodos modernos de apoio vêm sendo difundidos desde cedo.

Segurança é uma das principais demandas das grandes cidades. Além dela, existem outras, como: coleta e descarte de lixo, saneamento, eficiência energética, poluição, iluminação pública, transporte público, gerenciamento de desastres naturais, trânsito, saúde, educação, entre outras. Cada cidade possui suas demandas que podem variar dependendo de seu histórico de investimento, posição geográfica e outras variáveis que influenciam no cotidiano da localidade.

Desde antes da 2ª revolução industrial, um comportamento observado na sociedade e abordado nas introduções de inúmeras obras que tratam sobre o tema *Smart Cities* (Cidades Inteligentes), é a migração populacional dos campos e interior dos estados para as grandes metrópoles. Segundo levantamento da ONU, atualmente, cerca de 55% da população mundial vive nas grandes cidades, e informações de pesquisas feitas através de imagens de satélite dizem que esse número pode ser de até 85% em países em desenvolvimento e subdesenvolvidos, e de até 90% em países desenvolvidos [11][p.13].

O crescimento populacional das cidades acarreta na elevação da complexidade de manutenção dos serviços públicos. Um exemplo disso é o trânsito observado na cidade de São Paulo, que gera grandes transtornos para trabalhadores, moradores e turistas. Parte deste problema acontece pelo rápido desenvolvimento da cidade [12], ocasionado pelo alto crescimento de habitantes que migraram em busca de novas oportunidades. Tendo em vista as complexas adversidades ocasionadas pelas grandes migrações, Kon analisa a contribuição da tecnologia para a solução destes problemas no contexto de cidades inteligentes:

> "A Tecnologia da Informação permite hoje a coleta de milhões de dados sobre o que acontece na cidade a cada minuto. Esses dados podem ser armazenados, processados e analisados gerando um novo conhecimento que permite melhorar o funcionamento dos sistemas urbanos. Ainda estamos engatinhando nessa área, mas o potencial para melhor gerir a infraestrutura urbana economizando recursos, gerando menos poluição e menor impacto ambiental e oferecendo serviços de melhor qualidade para a população é enorme." [11] [p.14]

A tecnologia da informação (TI) tem sido o grande pilar no avanço de diversas áreas de atuação. Kon menciona que "ainda estamos engatinhando nesta área" quando trata-se da aplicação de TI em sistemas urbanos, estes que compreendem políticas públicas, planejamento urbano, apoio à tomada de decisão e diversos outros aspectos de governança e manutenção de cidades. O processo de aplicação tecnológica nas políticas públicas em cidades brasileiras pode ser lento, mas os primeiros passos já estão sendo dados. Um exemplo disso é o aplicativo para smartphone desenvolvido por demanda do Ministério da Justiça do Brasil, o Sinesp Cidadão. Em 2015 o aplicativo contribuiu para a recuperação de mais de 100 mil carros furtados ou roubados e na época registrava mais de 3,6 milhões de downloads [13], e em junho de 2019 já passava de 10 milhões de downloads no sistema operacional Android [14].

Na segurança pública essa potencialização também acontece visto que a informatização já faz parte dos órgãos públicos. Entretanto, esta é apenas uma parte do caminho para a transformação digital que oportuniza a eficiência na resolução de problemas. A inovação em processos é necessária para levar novas perspectivas aos problemas já conhecidos. Diversos projetos utilizando inovação tecnológica já fazem parte do cotidiano de órgãos de segurança no Brasil e no mundo [15][p.40-56] [16], enquanto os debates sobre o desenvolvimento e compra de novas tecnologias continuam em alta [17] [18] [19] [20].

Em exemplo mais recente, a Polícia Militar de Araçatuba demonstrou uma queda da criminalidade ao investir em tecnologia de análise de dados georreferenciados [21]. Assim como ocorre com o setor privado onde os sistemas de informação apoiam diversos segmentos das atividades, no setor público a soma de aplicações de TI junto às políticas focadas no enfrentamento de problemas de segurança pública mostra resultados positivos. A utilização de tecnologia como apoio na tomada de decisão para resolver problemas focados em questões sociais pode incluir um grande ativo estratégico no processo de entendimento

e combate destas adversidades, o que remete justamente a processos utilizados em *smart cities* (Cidades Inteligentes).

Conforme mencionado, em termos, cidades com densa população possuem diversos problemas, bem como, por via de regra, mesmo as pequenas que não se prepararam para o crescimento e enfrentam desafios ocasionados pela falta de estratégia e manutenção também possuem adversidades que podem ser solucionadas com aplicações de TI. O conceito de cidade inteligente abrange diversos pontos como: tecnologia, processos, aprendizado contínuo, inovação, comunicação e uma série de outros elementos, como destacado por Engel:

> "... o conceito de cidades inteligentes se vale do uso de projetos, nos quais um determinado espaço urbano é palco de experiências de uso intensivo de tecnologias sensíveis ao contexto da gestão urbana e ação social. Porém, o conceito de cidades inteligentes não é universal, mas múltiplo, sendo pesquisado em diversas áreas com vários olhares, contendo inúmeros outros atravessamentos. . . . Por inteligente, considera-se o uso inteligente dos recursos ambientais no que tange à sustentabilidade da cidade, bem como inteligente pode se referir ao uso de alta tecnologia no contexto urbano, dentre outros." [11] [p.19]

Percebe-se que uma cidade inteligente é o resultado de uma visão de longo ou médio prazo que estabeleceu um planejamento a fim de aprimorar a qualidade dos serviços prestados. Ainda, cidades que implementam ou melhoram processos para resolução de problemas pontuais, devem ser avaliadas antes de serem consideradas cidades inteligentes, uma vez que, este reconhecimento se dá quando há vários projetos que contribuem com a melhoria da eficiência da máquina pública e aumento da qualidade de vida, provindos tanto do primeiro quanto do segundo setor. Uma das formas de avaliar se a cidade se caracteriza no conceito de cidade inteligente é através da NBR ISO 37122 de 2019 [22], que estabelece padrões e indicadores de referência para cidades inteligentes e sustentáveis.

Cada cidade possui características particulares que podem ser trabalhadas utilizando conceito de *smart city*, portanto, elas devem possuir ambientes *smart* incorporados, como o ambiente de estudo proposto por Mishra *et al.* O local onde o projeto foi desenvolvido enfrenta problemas culturais de saneamento e higiene, e para isso foi implementado um ambiente *smart toilet* (banheiro inteligente). Através de gamificação, o sistema recompensa a repetição de bons hábitos comportamentais, neste caso o uso do banheiro público, ao mesmo tempo que gerencia a limpeza do local indicando os momentos de necessidade de manutenção. O sistema desenvolvido utiliza tecnologia IoT de BLE beacons de forma não intrusiva para identificar os usuários e pontuá-los de acordo com a prática do hábito [23].

O *smart toilet* é apenas um dos ambientes possíveis dentro de uma cidade inteligente. Mesmo este ambiente *smart* pode estar presente dentro outros locais como os *smart campus*. Em muitos casos, os campus universitários são vastos e frequentados por uma significativa quantidade de estudantes, professores, colaboradores do quadro administrativos e comunidade vizinha, como é o caso da Universidade de São Paulo (USP) [24]. Estas

peculiaridades aproximam as características de um campi às características de uma cidade, tanto que os *smart campus* são fortemente inspirados em soluções e métodos implementados nas *smart cities* [25]. Os problemas de infraestrutura, tráfego de veículos, pedestres, eficiência energética, acessibilidade entre tantos outros, tornam-se presentes em ambos territórios. No caso da USP, onde infraestrutura e segurança são duas demandas importantes, uma solução focada nestas necessidades dentro do campi foi desenvolvida para que os frequentadores auxiliem a identificar necessidades de infraestrutura e gestão de segurança [24]. Em campus maiores, soluções ainda mais extensas precisam ser criadas para atuar em educação, mobilidade urbana, serviços de saúde, funções administrativas e redução da poluição [25].

Uma forte característica dos campus é a presença da comunidade acadêmica, que tem o intuito de produzir propostas e novas soluções para problemas presentes na sociedade. Da mesma forma como fazem as cidades, os campus também precisam trabalhar além de problemas visíveis à maioria e buscar soluções para promover inclusão. O trabalho de [26] demonstra o uso de IoT para criação de rotas para pessoas com limitações visuais se locomoverem melhor dentro das dependências do campus. Com o auxílio da característica de *indoor* e *outdoor location* possibilitada pelos BLE beacons, explorado com mais profundidade nos capítulos de revisão sistemática e materiais e métodos, soluções como esta podem levar acessibilidade a qualquer indivíduo que tenha acesso ao aplicativo dentro do campus.

Os *smart campus* possuem um papel importante no desenvolvimento de novas soluções para as demandas da sociedade e andam junto com a evolução proporcionada por sistemas IoT. Embora a insegurança seja um tema importante, poucas alternativas com o uso deste paradigma estão sendo pensadas para resolver os problemas da área. Na série histórica de 2005 a 2014, o Rio Grande do Sul era o terceiro estado da federação com o maior índice de roubo de veículos [15][p.26]. Já nos anos de 2017 e 2018 era o quarto estado com maior número de veículos roubados e furtados do país, ficando atrás apenas de São Paulo, Rio de Janeiro e Minas Gerais, respectivamente [3]. De acordo com os dados da Secretaria de Segurança Pública do RS [27], a subtração de veículos vem caindo nos últimos cinco anos. Porém, os números da insegurança do Brasil permanecem altos em comparação com outros países, como aponta os dados da ONU [28] [29].

Este não é um problema que atinge apenas o proprietário que foi lesado no furto e ameaçado fisicamente no roubo, gerando problemas financeiros e psicológicos ao indivíduo, já que a ação dos assaltantes gera um problema social. Uma considerável porção destes veículos subtraídos em território nacional brasileiro são trocados por drogas, armas de fogo e dinheiro nas fronteiras, recursos estes que retroalimentam o problema da insegurança. Ademais, os veículos também podem ser utilizados para realizar outros crimes seguidos ao furto ou roubo, sendo geralmente violentos, o que pode acabar vitimizando outros indivíduos e também retroalimentando o problema da insegurança [30][31][32][33][34].

Tecnologias emergentes que utilizam IoT vem contribuindo cada vez mais com a tomada de decisão dentro de ambientes *smart*. Neste cenário, uma das tecnologias de comunicação que possibilitaram a expansão do paradigma foi o bluetooth, que incentivou o uso de dispositivos smartphones que se conectam com os mais diversos sensores e atuadores dos sistemas IoT. Dentro dessa tecnologia, o BLE (Bluetooth *Low Energie*) ganhou destaque por conta do baixo consumo de energia e de seu uso em soluções criativas. Ainda, tal tecnologia mostra-se, em certas circunstâncias abordadas no decorrer deste trabalho, mais adequada do que outras com propósitos similares como o GPS, RFID, LPWAN e WIFI, por disponibilizar características que beneficiam o uso de métodos *indoor* e *outdoor location*.

Todavia, dentro dos principais problemas da sociedade, como a subtração de bens materiais, verifica-se ausência de trabalhos acadêmicos focados em IoT. Esta pesquisa busca explorar a possibilidade de uso de BLE beacons para identificar furto e roubos de veículos em ambiente *smart campus*, um dos crimes de maior expressão no território brasileiro, enquanto eles ocorrem para que sejam possibilitadas contramedidas imediatas a estes atos. Ainda, a pesquisa leva como exclusão específica fontes de softwares proprietários de monitoramento de veículo desenvolvidos por fabricantes de automóveis como Mercedes Benz, Audi, Tesla, entre outros. Além de que estes sistemas fazem uso de GPS e comunicação via satélite, entre outras especificações indisponíveis por conta da baixa divulgação dos detalhes de implementação para manter baixos riscos de ataque cibernético e violação de propriedade intelectual, também são realidade para um baixo número proprietários em território nacional, oportunizando a pesquisa com foco de atuação sobre veículos populares.

1.1 OBJETIVOS

1.2 OBJETIVO GERAL

Analisar a aplicabilidade do uso de tecnologia IoT BLE beacon na segurança pública para identificar furto e roubo de veículos.

1. Explorar as tecnologias IoT aplicadas à segurança pública
2. Explorar as tecnologias emergentes que utilizam IoT em ambiente *smart campus* e *smart city*
3. Implementar um sistema utilizando IoT para identificar furto e roubo de veículos em ambiente *smart campus*
4. Identificar a viabilidade do uso de BLE beacons como ferramenta para identificação de furto e roubo de veículos

2. FUNDAMENTAÇÃO TEÓRICA E REVISÃO SISTEMÁTICA

Diversas tecnologias são utilizadas em sistemas de segurança pública no Brasil e no mundo, principalmente no que tange as *smart cities* e os *smart campus*. Neste trabalho busca-se entender quais os principais métodos utilizados pelos sistemas IoT no que se refere às aplicações de segurança pública para ambientes *smart*.

2.1 REVISÃO SISTEMÁTICA

A revisão sistemática é um método de estudo utilizado para levantar trabalhos e investigações úteis a um determinado tema, neste caso os trabalhos relevantes à segurança pública que utilizam tecnologia IoT.

2.2 MÉTODO

A fim de explorar possibilidades usando a revisão sistemática, foram formuladas 3 questões centrais para nortear a pesquisa:

1. É possível analisar furto e roubo de veículos com o uso de BLE Beacons?
2. Como as smart cities e os *smart campus* lidam com a segurança pública?
3. Quais tecnologias IoT estão sendo usadas para auxiliar indivíduos e instituições na segurança pública?

Estas três perguntas serviram para guiar os termos das palavras-chave usadas na busca de trabalhos similares. Foram definidas seis palavras-chave que filtram os temas centrais abordados. Elas são vistas abaixo, separadas por vírgulas:

Public security, *Public safety*, *Smart Cities*, *Smart Campus*, IoT, BLE beacons.

Utilizando as palavras mais pertinentes a esta investigação, foi formulado um método de busca que envolve a pesquisa para trabalhos em que há a ocorrência delas em títulos e palavras-chave, utilizando uma busca binária “(A0 OR A1) AND (B0 OR B1) AND (C0 OR C1) AND NOT “*cybersecurity*” ”. A tabela 1 mostra os respectivos códigos e suas palavras.

Tabela 1. Palavras chave

A0. *Public security*	B0. *Smart cities*	C0. IoT
A1. *Public safety*	B1. *Smart Campus*	C1. BLE beacons

Um problema identificado utilizando os termos “*security*” e “*safety*” (A1 e A0) em conjunto com “IoT” (C0) foi que muitos trabalhos focados em cibersegurança e proteção de dados são sugeridos pelas ferramentas de busca das bases de pesquisa. Como cibersegurança não é o foco do projeto, foi adicionado o termo de negação em conjunto com “*cybersecurity*” na busca binária para que trabalhos focados neste tema não fossem inseridos na pesquisa. Mesmo com essa regra sendo aplicada na pesquisa binária, alguns trabalhos na área de segurança de dados surgem mesmo sem utilizar a palavra “*cybersecurity*”. Portanto, algumas regras foram aplicadas para que os trabalhos mais adequados pudessem ser selecionados:

1. Não ter como foco a segurança e proteção de dados;
2. Ser focado em segurança da vida, ambientes smart IoT, BLE beacons ou tecnologias similares

A busca foi realizada nos seguintes periódicos e fontes de periódicos: Association for Computing Machinery (ACM), CiteSeer, Google Scholar, Institute of Electrical and Electronic Engineers (IEEE), Portal de Periódicos da CAPES, Scielo, Springer e dblp. Totalizando 8 fontes de informação diferentes e 89 trabalhos encontrados, dividido conforme mostra a tabela 2.

Tabela 2. Divisão dos trabalhos encontrados por fonte de pesquisa

Quantidade	**Fonte**
21	ACM
0	CiteSeer
4	dblp
31	Google Schoolar
15	IEEE
1	Portal de Periódicos da Capes
0	Scielo
17	Springer
89	**8**

Para reduzir a quantidade de trabalhos a um número expressivo, os trabalhos passaram por um funil disponível na Figura 1. A primeira parte aplica as regras descritas até então, e as partes seguintes segmentam os trabalhos.

O gráfico da Figura 2 mostra o resultado inicial da pesquisa e o gráfico da Figura 3 o resultado final da segmentação empregue pelo lógica do funil, que preservou a essência da metodologia aplicada, apenas removendo os trabalhos sem relação direta com a proposta do estudo, duplicatas idênticas e duplicatas de mesmos autores que publicaram trabalhos similares em periódicos diferentes apenas com textos distintos.

Ao todo 70 trabalhos foram descartados da amostra inicial, restando apenas os 19 com mais afinidade às regras elencadas nos itens de seleção do começo do capítulo.

Figura 1. Funil de seleção de trabalhos

Figura 2. Relação de trabalhos encontrados

2.3 PRINCIPAIS TRABALHOS

Abaixo seguem as tabelas divididas por áreas de estudo com o objetivo de elucidar os principais trabalhos encontrados que embasam o desenvolvimento técnico da pesquisa e que respondem às questões levantadas no método da Revisão Sistemática da Literatura. A primeira coluna corresponde ao identificador do trabalho, a segunda é referente ao título e autores, enquanto na terceira coluna as áreas de estudo e tecnologias utilizadas em cada

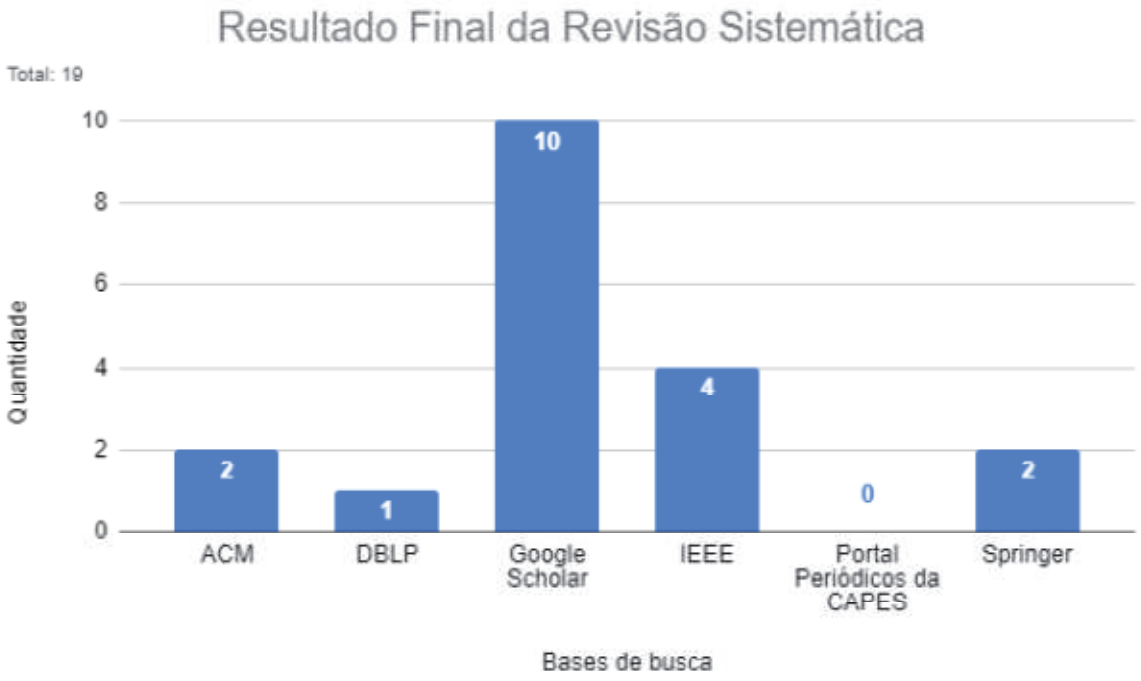

Figura 3. Relação dos trabalhos selecionados

uma das amostras que aqui refletem os temas centrais de cada pesquisa. Por fim, a última coluna refere-se ao ano de publicação da obra.

Os dados da Tabela 3 incluem trabalhos onde o ambiente *smart* é o principal tema abordado. A orientação da revisão sistemática busca temas ligados à *smart cities* e *smart campus*, que estão intrinsecamente conectados. Como elucidado na introdução deste trabalho, apesar de não haver um único conceito sobre as *smart cities*, elas são formadas pelo conjunto de diversas soluções tecnológicas e metodologias de inovação. Algumas dessas metodologias estão presentes em paradigmas como: *smart parking, smart building, smart campus, smart libraries, smart toilets,* entre outros ambientes; sendo locais aonde há presença de novas abordagens para resolver problemas com uso de tecnologia.

2.3.1 *Smart Campus*

KON [35] (trabalho #8) demonstra como a evolução tecnológica afetou e vem cada vez mais afetando positivamente o gerenciamento de recursos das cidades, principalmente no que tange tecnologias IoT. Não havendo conceitos enraizados sobre cidades inteligentes, todos eles convergem, em algum ponto, na entrega de qualidade de vida para os cidadãos locais e visitantes. Em paralelo às cidades inteligentes, os ambientes smart de menores proporções também surgem como conceitos focados na qualidade de vida do ser humano e na resolução de problemas recorrentes.

Tabela 3. Ambientes *smart*

#	Título - Autores	Área de estudo - Tecnologia	Ano
1	*Cloud-Based Parametrized Publish/Subscribe System for Public Safety Applications in Smarter Cities* - Aljawharah Al-muaythir *et al*.	*Smart cities* - Padrão Pub/Sub	2016
2	*DevOps for the Urban* IoT - John Moore *et al*.	*Smart cities - Dev Ops*	2016
3	*Smart Services: A Case Study on Smarter Public Safety by a Mobile App for University of* São Paulo - João Eduardo Ferreira *et al*..	*Smart Campus* - Web Mobile	2017
4	*UUID Beacon Advertisements For Lecture Schedule Information* - Wiwin Agus Kristiana *et al*.	*Smart Campus* - BLE e Web *Services*	2018
5	*A hybrid BLE and Wi-Fi localization system for the creation of study groups in smart libraries* - Kiril Antevski *et al*.	*Smart Library - Indoor Location*	2016
6	*AN IoT BASED SMART CAMPUS ARCHITECTURE FOR INSTITUTIONS IN DEVELOPING COUNTRIES* - BARROON ISMA'EEL AHMAD *et al*.	*Smart Campus - Framework*	2019
7	*Better campus life for visually impaired University students: intelligent social walking system with beacon and assistive technologies* - Utku Kose *et al*.	*Smart Campus* - Beacons e IA	2018
8	XXXVI Congresso da Sociedade Brasileira de Computação - Cap. 1. Cidades Inteligentes: Conceitos, plataformas e desafios - Fabio Kon *et al*.	*Smart Cities*	2016
9	*Smart Toilets using BLE Beacon Technology* - Nidhi R Mishra *et al*.	*Smart Toilet* - BLE Beacons e gamificação	2018
10	*Smart University, a new concept in the Internet of Things* - Marian Cata	*Smart campus*	2015
11	*A Survey on Internet of Things Enabled Smart Campus Applications* - Abdelrahman Abuarqoub *et al*.	*Smart campus* - IoT	2017

O conceito de smart campus é baseado diretamente do conceito de smart cities, como aponta Abuarqoub [36] (trabalho #11). Em termos, um smart campus possui o mesmo conceito que uma *smart city*, porém, difere em seu espaço geográfico e método de desenvolvimento de soluções. Enquanto uma *smart city* compreende todas as necessidades de uma cidade, o smart campus permeia apenas as situações adentro um campus univer-

sitário/estudantil. Portanto, os recursos de orçamento, tempo, complexidade tecnológica, número de usuários, manutenção, entre outros, são reduzidos a uma menor escala.

Em um primeiro momento, no ponto de vista de abrangência, um *smart campus* pode oferecer pouca escala para uma aplicação. Os dispositivos usados em um *smart campus* não atingem toda a população da cidade em que está sediado, apenas a comunidade acadêmica que frequenta o campus. Porém, a grande vantagem deste ambiente é justamente sua pequena abrangência comparada a uma cidade. Muito utilizado no modelo de gestão 4.0 [37], um MVP (*minimum viable product* - mínimo produto viável) busca validar hipóteses testando funcionalidades com o máximo de valor e o mínimo de esforço possível, gerando aprendizados a cada iteração [38]. Tendo isto em vista, um *smart campus* torna-se o local ideal para a implementação de novas tecnologias para *smart cities* utilizando metodologias modernas de mercado. Uma vez que as hipóteses estejam validadas e moldadas à resolução do problema, o próximo passo é ajustar a escalabilidade, reproduzindo o experimento para uma cidade.

Os campus universitários são locais extensos e frequentados, muitas vezes, por grandes grupos, o que reproduz os mesmos problemas vivenciados nas metrópoles, tais como nas áreas de trânsito, energia elétrica, inclusão, infraestrutura, comunicação, saúde, meio ambiente, segurança, entre outros; como descreve CATA [39] (#10), sobre o ecossistema dos *smart campus* frequentados pela comunidade acadêmica. Através de experimentos *in loco*, essas instituições elevam sua relevância na contribuição para soluções de problemas sociais da comunidade local. Uma vez que a proposta mostrar-se eficiente, o próximo passo é escalar para o uso em ambientes mais complexos (cidade), firmando o *smart campus* como um ator de relevância no desenvolvimento tecnológico do ecossistema.

Uma amostra disso pode ser vista no estudo de AHMAD *et al*. [40] (trabalho #6), onde o autor cita como a universidade utiliza um framework de gerenciamento de *smart campus* para administrar recursos de educação, serviços de saúde, mobilidade, *smart building* e controle administrativo. Tais necessidades estão presentes no campus e também nas cidades, e em diversos casos a colaboração entre os atores pode colher resultados promissores. Já o trabalho de ANTEVSKI *et al*. [41] (trabalho #5) faz uso de BLE beacons em ambiente *smart library* (biblioteca) que está incubado em um pequeno espaço em um *smart campus*; onde o intuito é facilitar a interação humana presencial através de grupos de estudo. No trabalho, percebe-se como o IoT pode ser usado em bibliotecas através de uma aplicação de *indoor location*. Essa aplicação direciona os estudantes aos pontos de interesse aonde devem encontrar grupos de estudo utilizando como ferramenta um aplicativo mobile. Com pequenos ajustes, esse tipo de sistema pode ser utilizado em bibliotecas municipais.

O ensaio #13 da Tabela 4 demonstra como a computação moderna se tornou uma das principais ferramentas de apoio para a segurança pública. O autor chama esta colaboração de "*Computational Public Safety*" (tradução: Segurança Pública Computacional) e a

define como “sistemas digitais e algoritmos que podem impactar no bem-estar da população”, algo que se assemelha muito com os conceitos de *smart cities*, que inclusive é citado pelo autor em seu capítulo 4.1. Desde a criação de simulações de desarme de objetos explosivos até a busca de vítimas de desastres naturais, os sistemas digitais dão suporte a diversos canais de segurança da vida, e mais recentemente os dispositivos IoT também vêm ganhando espaço nessa empreitada. O autor cita o uso dos dispositivos de detecção de disparo de arma de fogo, que identificam os locais de ocorrência de disparo e emitem alertas para que as autoridades competentes possam atuar. Essa mesma informação também é usada em sistemas de análise de dados que possibilitam identificar padrões criminais para a atuação de policiamento ostensivo preventivo, ou ainda, policiamento preditivo.

O caso de uso exposto no projeto #3 [42] demonstra uma aplicação focada na segurança pública dentro de um ambiente *smart campus*. O estudo levanta o caso da USP, no Brasil, conhecido por se tratar de uma região de insegurança. A USP possui uma densidade comparável com a de uma cidade, comportando mais de 90 mil discentes, 6 mil docentes e 14 mil funcionários administrativos em mais de 76 milhões de m². Para combater a insegurança foi desenvolvido um aplicativo mobile, que além de expor as demandas de infraestrutura da universidade também serve como entrada de informações para criar *hotspots* de vigilância.

Também em ambiente *smart campus*, os ensaios #4 [43] e #7 [26] demonstram como o IoT pode facilitar a produtividade de alunos em eventos e incentivar a inclusão de outros com limitações visuais dentro do ambiente estudantil com o uso de BLE Beacons. O trabalho #4 utiliza um *broadcast* beacon embarcado em um Raspberry Pi que envia dados relacionados às palestras e agendas das salas da instituição em tempo real para todos os alunos no alcance do dispositivo que estejam utilizando o aplicativo do *smart campus*. Os sinais constantes emitidos pelo Raspberry utilizando a tecnologia BLE podem ser percebidos por todos os smartphones com versão compatível à 4ª geração de sensores bluetooth. A partir daí o aplicativo usa o Bluetooth *Smart Ready* do smartphone para analisar os dados do sinal e realizar uma requisição em um servidor, que busca as informações pertinentes àquela sala no banco de dados e responde ao aplicativo. Ainda, usando a mesma tecnologia de atuadores, através do BLE, é possível o uso de metodologias *Indoor Location* que permitem o desenvolvimento de aplicativos e sistemas de localização *indoor* e *outdoor*. Um exemplo disso é o trabalho #7, que utiliza esta metodologia para criar rotas *indoor* e *outdoor* para alunos com necessidades especiais, da mesma forma como acontece no estudo #17 [44] e #18 [45] da Tabela 4. Através de beacons instalados dentro dos prédios e nas ruas dos campus, os sistemas são capazes de criar rotas personalizadas para deficientes visuais que acessam o serviço através de aplicativos com funcionalidades adaptativas de acessibilidade, gerando inclusão enquanto desenvolve novas tecnologias que podem ser escaladas para outros ambientes.

Tabela 4. Aplicações IoT e caso de uso em ambientes *smart*

#	Título - Autores	Área de estudo - Tecnologia	Ano
11	Location Fingerprinting With Bluetooth Low Energy Beacons - Ramsey Faragher *et al*.	Indoor Location - BLE Beacons	2015
12	ROTA: A Smart City Platform to Improve Public Safety - Jazon Coelho *et al*.	Segurança Pública - Web mobile	2016
13	Computational Public Safety: The Evolution to Public Safety Research - Nhan Tran *et al*.	Segurança Pública Computacional	2018
14	BLE Beacons for Internet of Things Applications: Survey, Challenges and Opportunities - Kang Eun Jeon *et al*.	Análise de tecnologia - Beacons	2018
15	IoT-based System for Indoor Location using Bluetooth Low Energy - Marco Terán *et al*.	Arquitetura Indoor Location - Machine Learning	2017
16	Bluetooth 5: An Augmented Technology for IoT - Sachin R. Ponde	Bluetooth 5.0	2019
17	NavCog: A Navigational Cognitive Assistant for the Blind - Dragan Ahmetovic *et al*.	Bluetooth Low Energie - Beacons e sistemas de navegação	2016
18	Ray: Smart indoor/outdoor routes for the blind using Bluetooth 4.0 BLE - Manuel Castillo-Cara *et al*.	Bluetooth Low Energie - Beacons e sistemas de navegação	2016
19	State of the Art, Trends and Future of Bluetooth Low Energy, Near Field Communication and Visible Light Communication in the Development of Smart Cities - Gonzalo Cerruela García *et al*.	IoT - BLE, NFC, VLC	2016

A Tabela 4 exibe os trabalhos encontrados na revisão sistemática que focam em tecnologia IoT.

2.3.2 *Internet of Things* (Internet das Coisas - IoT)

Atualmente, *Internet of Things* (Internet das Coisas - IoT) é um dos conceitos mais explorados nas ciências tecnológicas. O desenvolvimento científico proporcionou que cada vez mais dispositivos pudessem se conectar e transmitir informações. Nas *smart cities* e também nos *smart campus*, tecnologias IoT permitem o monitoramento de ambientes e posteriormente os dados gerados pelos dispositivos são analisados; e decisões estratégicas são tomadas com base nas informações descobertas [36].

O principal objetivo do uso de tecnologias IoT é criar uma ponte entre o mundo físico e o mundo digital. Os dispositivos conectados a um sistema IoT podem ser smartphones, sensores especialistas que possuem uma única função. Exemplo: sensor de temperatura) ou generalistas (como um *hub* ou *gateway* de sensores) e estarem presentes em geladeiras, lâmpadas led, portas, janelas e fechaduras eletrônicas, veículos, postes, paredes, livros, entre diversos outros. Existem aplicações que vão da análise de qualidade do ar ao monitoramento de estrutura de edifícios. Para criar essa interação são usados essencialmente três elementos [35]:

1. Hardware - utilizado para coletar informações do ambiente ou realizar interações. Para isso são utilizados sensores e atuadores, respectivamente.
2. *Middleware* - cria a ponte entre os dados gerados pelo hardware e o sistema que receberá a informação do dispositivo para que seja feito o processamento adequado.
3. Interface de aplicação - é através dela que um ser humano pode interpretar as informações provindas do sistema. Pode ser uma aplicação web, um aplicativo móvel ou até mesmo um relatório.

Um dos fatores mais relevantes para o crescimento do IoT é a comunicação wireless (que não depende de cabos e fios). Não à toa que a grande maioria dos estudos envolvendo IoT, bem como grande parte dos conceitos encontrados na literatura, mencionam a comunicação de dispositivos. Existem diversas formas de um hardware enviar e receber informações. Uma das formas mais comuns é através do Bluetooth, sendo esta uma das tecnologias que mais intensificaram o crescimento dos sistemas IoT [46].

2.3.3 Bluetooth *Low Energy* e BLE Beacons

A partir de sua criação, o bluetooth continua sendo uma tecnologia implementada na camada física de aplicações wireless utilizada para transferir dados de um dispositivo para outro. Nos anos 90, com estudos da empresa Ericsson Mobile Communications, as primeiras ideações do que se tornaria o bluetooth foram criadas. Através de tecnologias de baixo consumo de energia, a companhia buscava a substituição de cabos de dados por tecnologia sem fio. Nos anos seguintes, foi formado a Bluetooth *Special Interest Group* (SIG), e em meados dos anos 2000 o primeiro protocolo bluetooth foi criado [47].

Com a popularização do bluetooth, a tecnologia foi amplamente usada em telefones celulares com o propósito de superar a tecnologia anterior que tinha a mesma finalidade: o infravermelho. Além de celulares, o bluetooth também foi projetado para que dispositivos de impressão, mouses, teclados e aparelhos multimídia pudessem substituir cabos de tráfego de dados pela tecnologia sem fio [47]. Desde então, o bluetooth evoluiu e, no momento que este trabalho está sendo produzido, está em sua 4ª versão.

Criado em 2010, o Bluetooth 4.0, também chamado de *Smart* Bluetooth ou Bluetooth *Low Energy* (BLE), tem a capacidade de emitir dados a distâncias mais longas consumindo menos bateria que seus antecessores [48]. Outra novidade que veio com a versão 4 é a tecnologia BLE embutida em seu *chipset*. Ela utiliza os canais 37, 38 e 39 dos 40 canais disponíveis no bluetooth para emitir dados de propaganda [46]. Baseado nisso, foram criados dispositivos chamados BLE beacons que utilizam a tecnologia BLE. No presente, os beacons já são considerados dispositivos proeminentes dentro do mundo do IoT. Pela facilidade de *deploy* e transmissão de dados através de pequenos dispositivos, eles já são usados em diversas frentes onde o IoT é pioneiro, tais como: automação residencial (*smart houses*), indústria 4.0, economia de energia em residências e indústrias, identificação de indivíduos dentro de ambientes, entre outros [46].

Fazendo uso do BLE, os beacons podem emitir dados a variadas distâncias (de 1 a 200 metros) dependendo da necessidade da aplicação e modelo de BLE beacon utilizado. Também é reconhecido pelo baixo consumo de bateria e qualquer outro dispositivo com capacidade de comunicação na mesma frequência pode captar os dados emitidos. Porém, quanto maior a distância, mais energia é consumida e menos tempo a bateria do beacon permanece carregada. Em contrapartida, assim como o BLE, os beacons também têm baixo consumo de carga, o que pode elevar a longevidade da bateria para anos tendo baixo custo de compra e manutenção [48].

Os BLE beacons já são utilizados dentro de sistemas IoT com o mesmo propósito ao qual sua tecnologia base surgiu: emitir dados. É importante destacar que, salvo modelos fabricados para teste e desenvolvimento, os beacons não trocam informações de forma bidirecional com outros dispositivos capazes de identificar sinais da mesma frequência. Eles produzem comunicação unidirecional, apenas enviam sinais. De acordo com Jeon [46] e García [49], a tecnologia de comunicação BLE se destaca dentre as de outros tipos como GPS, RFID e roteadores WIFI quando se trata de *Indoor Location* (termo utilizado para localização em ambientes fechados ou interiores de estruturas). O GPS não possui efetividade nestes ambientes; enquanto o RFID possui baixo alcance para trabalhar com posicionamento; roteadores WIFI possuem alto custo para instalação e não são instalados para razões de localização e sim para cobrir áreas para servir conexão com a internet. Em virtude disso, os BLE beacons já possuem grande importância entre as tecnologias implementadas nos ambientes *smart*. Além de serem utilizados como forma de identificação de objetos e locais, também já estão presentes em empresas do varejo para que aplicativos possam triangular a posição do usuário e enviar notificações com promoções e produtos próximos [48].

O objetivo do uso desses equipamentos em certas aplicações vêm servindo como apoio aos objetivos das tecnologias IoT: aproximar o mundo virtual e o mundo real de forma automatizada. Uma das maneiras de se fazer isso através dos beacons é obtendo a infor-

mação do quanto um determinado aparelho, como um smartphone, que lê o sinal de um beacon está próximo do mesmo beacon.

Um caso de uso de beacons que executa essa abordagem foi implementado no Museu de *Guggenheim*, em Nova Iorque. Utilizando tecnologia Estimote, empresa especializada em BLE beacons, foi desenvolvido um sistema para que os visitantes possam facilmente ter acesso às informações das obras de arte expostas. O usuário faz a instalação do aplicativo mobile do museu em seu próprio smartphone e a partir daí estará apto para utilizar suas funcionalidades. Cada BLE beacon posicionado no museu emite um identificador que pode ser lido pelo aplicativo e então o sistema identifica a qual obra o beacon faz referência. Pode-se obter informações sobre autor, história, curiosidades, entre outras [50] [11], além de possuir opções de acessibilidade.

Utilizando componentes tecnológicos similares, mas com funcionalidades distintas, Kose [51] demonstra como os BLE beacons podem ser utilizados como dispositivos centrais no desenvolvimento de um sistema de navegação para deficientes visuais se locomoverem em um ambiente *smart campus*. Utilizando uma estrutura IoT de baixo custo juntamente com técnicas de Big Data e Inteligência Artificial, um aplicativo de smartphone lê os sinais dos BLE beacons e determina a posição atual do estudante por meio da contínua posição rastreada. A partir disso, o sistema de navegação indica qual a melhor rota a ser seguida até o local determinado pelo usuário usando de áudios para indicar pontos de atenção a fim de oferecer uma experiência de mobilidade aos estudantes que possuem limitações visuais.

Ambos os casos de uso mencionados nos parágrafos anteriores processam os sinais emitidos pelos BLE beacons. Um identifica a aproximação e o outro identifica a posição; para isso, cada um utiliza um método diferente, porém, o que ambos têm em comum é o processo de recepção dos sinais. Esse processo é favorecido pelos protocolos utilizados pelos BLE beacons na emissão de dados, o que possibilita que desenvolvedores criem softwares padronizados e capacitados a receber os sinais e explorar as informações geradas através dessa interação. Os protocolos mais conhecidos e usados no mercado são o iBeacon e o Eddystone.

2.3.3.1 iBeacon

O pioneiro protocolo iBeacon foi apresentado pela Apple como um componente integrado ao iOS 7. Originalmente foi criado para que desenvolvedores de apps pudessem reconhecer os sinais emitidos por dispositivos com tecnologia BLE e disparar gatilhos quando o smartphone com iOS 7 entrasse em um determinado espaço. Este protocolo permitia que fosse possível reconhecer o dispositivo BLE beacon com implementação iBeacon através de um identificador e também determinar a distância que o smartphone estava da

origem do sinal. Ilustrada na Figura 4, a distância pode ser determinada medindo a força com que o sinal chega ao dispositivo [52].

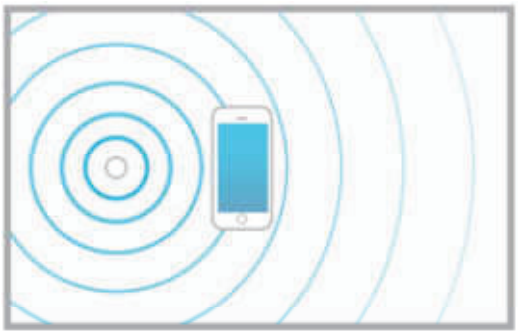

Figura 4. Transmissão e recepção de sinais BLE
Fonte: [52]

Descrito na Tabela 5, a tecnologia desenvolvida pela Apple emite essencialmente três tipos de dados em seu sinal. Estes dados podem ser utilizados para entender em qual área o usuário está entrando e posteriormente realizar uma ação a partir deste gatilho.

Tabela 5. Identificadores do protocolo iBeacon

Campo	Descrição
UUID	Serve para identificar grupos de beacons. Pode servir para identificar uma empresa, aplicativo, região, etc.
Major	Especifica uma sub-região do UUID
Minor	Especifica um sub-região do identificador Major

Os três atributos descritos na tabela funcionam de forma hierárquica. Enquanto o UUID (identificador universal) representa um grande conjunto para que seja possível saber a quem os beacons com esse identificador pertencem, os dados de *Major* e *Minor* criam subdivisões para que seja viável o desenvolvimento de eventos específicos caso o smartphone do usuário entre em uma determinada área. Obtido da documentação oficial da Apple, a Figura 5 ilustra um caso de uso de uma rede de lojas para explicar o conceito dos identificadores.

No caso de uso da Figura 5, é definido um UUID global que serve para reconhecer a qual empresa o beacon pertence; O dado de *Major* define a qual cidade a loja da rede está localizada; por fim, o identificador *Minor* define a qual categoria de seção o dispositivo está alocado (automotivo, utensílios domésticos, roupas e etc.). Desta forma é possível padronizar, por exemplo, o número 30 a todas as seções de produtos automotivos das lojas da rede, independente de qual local do mundo ela esteja. Em conjunto, é definido no identificador Major a qual loja se trata a localização do dispositivo. Com estes elementos é possível um aplicativo diferenciar em qual local de uma determinada loja o smartphone do usuário está. Para que isso seja viável, os identificadores do dispositivo BLE devem ser definidos pelo desenvolvedor da aplicação [52]. A partir dessa identificação, as regras de

Store Location		San Francisco	Paris	London
UUID		D9B9EC1F-3925-43D0-80A9-1E39D4CEA95C		
Major		1	2	3
Minor	Clothing	10	10	10
	Housewares	20	20	20
	Automotive	30	30	30

Figura 5. Caso de uso dos identificadores do protocolo iBeacon
Fonte: [52]

negócio do aplicativo podem entrar em ação e executar *n* tarefas a fim de entregar uma experiência de uso imersiva e que, dependendo do caso, não necessita de conexão com a internet para serem executadas.

2.3.3.2 Eddystone

O Eddystone surgiu com o mesmo propósito do protocolo iBeacon: possibilitar dispositivos BLE emitir sinais e padronizar sdk's (*Software Development Kit* - Kit de Desenvolvimento de Software) com o intuito de auxiliar desenvolvedores a criar dispositivos e softwares capazes de interpretar os sinais recebidos. Ele é um protocolo aberto criado pelo Google, e diferente do iBeacon, que inicialmente possuía intensão de funcionar apenas em dispositivos iOS[2], o Eddystone é totalmente *open source*, podendo ser embutido e interpretado por qualquer tipo de dispositivo, seja ele um BLE, aplicativo de smartphone ou outro dispositivo que capta as frequências emitidas usando Eddystone [53].

Na visão do autor deste trabalho, o Eddystone se mostra um protocolo mais diversificado que o iBeacon. Além de ser aberto e qualquer pessoa com conhecimento sobre desenvolvimento de software poder estudar e utilizar, o Eddystone também emite uma gama maior número de informações, o que possibilita diversas opções de implementação por parte de desenvolvedores. Seu formato comporta quatro tipos diferentes de estruturas, elucidados na tabela abaixo:

Conforme a Tabela 6, pode-se notar como o Eddystone comporta mais informações do que o protocolo iBeacon. O UID, similar ao UUID do iBeacon, é um identificador de 16 bytes formados por um grande identificador e duas subdivisões, possibilitando assim a identificação do dispositivo que está transmitindo. O campo de URL, como o próprio nome diz, transmite uma URL que pode ser usada por navegadores, ou, ainda, ser utilizada

[2]existem ferramentas desenvolvidas pela comunidade que possibilitam trabalhar com o iBeacon fora do sistema operacional iOS, entretanto não são oficiais ou padronizadas

Tabela 6. Especificação do protocolo Eddystone [53]

Campo	Descrição
Eddystone-UID	Similar ao iBeacon, esta estrutura emite um identificador para que o dispositivo emissor possa ser reconhecido.
Eddystone-URL	Emite uma URL.
Eddystone-TLM	Transmite a telemetria referente ao beacon e dispositivos acoplados a ele.
Eddystone-EID	É similar ao UID, porém, transmite um identificador ajustável e encriptado.

para realização de chamada em API por parte do sistema que a recebeu. TLM refere-se à telemetria do beacon. Com ela é possível obter informações vinculadas ao hardware do dispositivo emissor, como nível de bateria, sensores acoplados e pacotes de transmissão (utilizados para identificar a força do sinal). Por fim, o EID, similar ao UID, emite um identificador encriptado. A utilização desta estrutura possibilita uma série de vantagens a nível de segurança. O BLE que implementa o EID muda seu identificador após um determinado período de tempo. Esse identificador pode ser tratado e identificado por um serviço que implementa as especificações do protocolo Eddystone. Para fazer uso desta infraestrutura, o software que identificou o EID deve fazer uma solicitação juntamente com uma chave de permissão para validar a requisição. Um exemplo deste uso é através da Google Proximity Beacon API ou do Google Beacon Plataform[3], que implementam as especificações do protocolo Eddystone. O sistema se encarrega de identificar o EID e responder com dados referentes a solicitação, podendo dar sequência às regras de negócio do desenvolvedor da aplicação que leu os dados do beacon [54].

O Eddystone-URL foi uma ferramenta desenvolvida com o objetivo de explorar experiências de uso em locais públicos por meio do Android Nearby [55]. Regularmente chamada de “Web Física”, o objetivo era que o Nearby - app embutido no SO Android - fosse capaz de notificar o usuário com informações dos dispositivos Eddystone próximos emitindo URLs e com isso aumentar a experiência de uso. Essa funcionalidade foi descontinuada no Android em 2018 devido ao alto número de notificações spam e outras informações irrelevantes que prejudicavam a usabilidade e experiência de consumo deste recurso por parte dos clientes Android [56]. O protocolo Eddystone ainda permite o uso de URLs, porém, é necessário que no smartphone os desenvolvedores implementem suas próprias regras de negócio para interagir com as URLs; ou, ainda, podem utilizar APIs disponibilizadas pelo Google para gerenciar as notificações, como a Nearby Messages do Proximity Beacons API.

[3]Na data de desenvolvimento do presente trabalho, o Google Proximity Beacon API e Google Beacon Plataform serão descontinuados em 1 de abril de 2021.

2.3.3.3 RSSI e Tipos de Aplicação

Em suma, o foco de grande parte das aplicações IoT que envolvem os BLE beacons gira em torno do sinal emitido pelos dispositivos. Essas aplicações identificam o que se chama de Força do Sinal Recebido (*Received Strength Signal* - RSS) [48] ou Indicação de Força do Sinal Recebido (*Received Strength Signal Indication* - RSSI) para processá-lo e posteriormente executar suas regras de negócio, estas que podem variar de acordo com a funcionalidade exigida pelo sistema.

O RSS é um dado que varia de acordo com dois fatores: fonte e receptor. Além de um BLE beacon, a fonte pode ser qualquer dispositivo emissor que esteja agindo conforme BLE beacon, sendo o receptor qualquer dispositivo que esteja captando os sinais. Como a base desta tecnologia é BLE (disponível a partir do Bluetooth v4.0), qualquer dispositivo que embarque essa funcionalidade poderia agir tanto como fonte quanto como receptor, por exemplo: smartphones, notebooks, módulos bluetooth para Arduino, Raspberry Pi, entre outros.

De acordo com Jeon [48], o RSS é estipulado através de uma medida em dBm (*decibel miliwatt*) que depende diretamente da capacidade de transmissão do dispositivo beacon. Quando um BLE beacon é instalado em uma área ou objeto, ele age como uma fonte emissora enviando seus sinais para todas as direções, sem um foco específico. A responsabilidade do desenvolvedor da aplicação é identificar esses sinais e processá-los da melhor forma para reproduzir o efeito esperado no seu sistema.

Os resultados da Revisão Sistemática da Literatura deste trabalho indicam que existem dois grupos com tipos diferentes de implementação, ambos utilizando o dado de RSS, são eles: aproximação e detecção.

O primeiro tipo diz respeito à aproximação entre clientes e objetos ou locais marcados utilizando BLE beacons. Neste caso, as aplicações trabalham através da perspectiva de área de atuação, como o case do Smart Museum de *Guggenheim* [50], em que o usuário recebe informações sobre a obra exposta através da proximidade com ela. Da mesma forma, o case de *Smart Campus* da Universidade de Narotama utiliza deste mesmo método aqui chamado de *Location Based Service* (Serviço Baseado em Localização - LBS). Através do uso de dispositivos Raspberry Pi simulando o comportamento de beacons, os autores criaram um sistema que distribui avisos de status e agendas de palestras para os estudantes da universidade baseado na proximidade do receptor (smartphones) e das fontes (Raspberry Pi) [43].

O segundo item (detecção) foca em identificar a posição atual de um dispositivo. No geral, essa localização pode ser encontrada através de métodos de triangulação usando três ou mais dispositivos, como Faragher [57] demonstra no processo de *fingerprinting*, que visa estabelecer a posição do dispositivo de um usuário dentro de um determinado local. Apesar de ser a tecnologia com mais benefícios para detecção de posicionamento

em ambientes fechados, os BLE beacons apresentam problemas que podem dificultar esse processo.

Embora existam algumas complicações enfrentadas apenas pelos métodos de detecção, os testes aplicados por Faragher [57] indicam que um dos problemas mais evidentes, também enfrentados por aplicações de aproximação, é a taxa de decaimento da propagação do sinal emitido pelos beacons. De acordo com Jeon [48], essa característica torna o RSSI pouco confiável e quanto maior a distância menor será o índice de confiabilidade. Outro elemento que influencia essa percepção é o alto número de arquiteturas e tecnologias que são usadas em aparelhos diferentes que são capazes de detectar os sinais dos beacons. Por cada tecnologia ser diferente, o valor processado pode conter variações para cada uma delas. Ainda, outro fator que influencia a confiabilidade do RSSI é a densidade do local. Obstáculos como paredes interferem diretamente na propagação dos sinais dos dispositivos. Apesar dos agravantes, com o uso de alguns algoritmos é possível aperfeiçoar a estimativa do RSSI para obter graus regulares de confiabilidade.

Explorado no trabalho #19 [49], normalmente algoritmos de modelo de regressão polinomial, *channel-separate fingerprinting*, detecção de anomalia e *extended Kalman filtering* são usados para mensurar o RSS. Quando algo no ambiente onde o beacon está inserido pode causar problemas de ruídos e obstrução, algumas alternativas podem ser utilizadas, como o uso de diferentes métodos para analisar o RSS. Alguns deles são a *fuzzy decision tree* e o *k-nearest neighbors*.

2.3.3.4 *Deploy* de BLE Beacons

Diferente de sistemas de informação convencionais, o *deploy*[4] de BLE beacons é simples e pode ser conduzido por qualquer pessoa, mesmo uma que não tenha intimidade com desenvolvimento de tecnologia. Por exemplo: digamos que esteja havendo uma conferência profissional em um grande escritório e os participantes são guiados através da aproximação de seu smartphone com um beacon. O beacon relativo à palestra pode ser deixado em cima de uma bancada e estará transmitindo seu sinal à todos os dispositivos capazes de recebê-lo.

Em casos mais complexos, o beacon pode ser fixado em uma parede ou objeto através de uma fita dupla face. O foco central para o *deploy* de um BLE beacon deve ser posicioná-lo em um local adequado para que obstáculos não interfiram na propagação do sinal [48]. Os pontos mais apropriados variam de acordo com a necessidade da aplicação. Quando se está criando uma funcionalidade de detecção, locais altos são os mais indicados para que o sinal se propague com facilidade. Já em aplicações de aproximação, muitas vezes o objeto ou local de interesse pode estar próximo ao chão, mas, dentro do possível,

[4] Fase em que um sistema está sendo disponibilizado. Pode ser para testes ou para implantação, que é quando o sistema já possui maturidade para ser utilizado por usuários.

para obter um melhor desempenho do beacon, locais altos devem ser priorizados. Outro fator que influencia a preferência destes locais é para evitar ataques físicos, que é uma das vulnerabilidades dessa tecnologia.

2.3.3.5 Vulnerabilidades de Segurança em BLE Beacons

Embora a facilidade de implantação de um BLE beacon seja um forte aliado da tecnologia, um dos principais fatores de vulnerabilidade está atrelado a este processo. Como apontado no capítulo sobre *deploy*, locais altos são indicados para evitar ataques físicos ao dispositivo. Como demonstrado por Jeon [48], em um local de fácil alcance o beacon pode ser furtado e até mesmo danificado propositalmente para que pare de enviar sinais. Em casos mais incomuns o beacon pode até mesmo ser trocado por um falsificado que emita os mesmos sinais, porém, que se comporte de forma planejada pelo invasor.

Apesar da vulnerabilidade física ser uma das principais preocupações na criação de aplicações, não são apenas ataques físicos que podem afligir um beacon. Ataques cibernéticos podem gerar danos colaterais até mais severos do que os ataques físicos, uma vez que são realizados por especialistas, no geral com objetivos pontuais. Um desses ataques é o Beacon *Spoofing*, que implementa um método de clonagem de beacon. Um dispositivo capaz de captar os dados propagados pode armazená-los para que posteriormente um BLE beacon falsificado seja produzido. Isso faz com que o beacon possa ser utilizado fora de sua área de atuação [48]. Uma forma de evitar este tipo de ataque é através do Eddystone EID que torna o dispositivo capaz de emitir sinais criptografados. O problema dessa abordagem é que o dispositivo receptor requer uma conexão com a internet, uma vez que o pacote recebido deve ser validado em um servidor que implemente o protocolo Eddystone. Essa prática pode ser impossibilitada caso interações online não sejam possíveis por alguma razão.

Além de clonagem, outro ciberataque possível é contra a infraestrutura beacon como um todo através do *Packet Injection*. Similar ao *Spoofing*, o *Packet Injection* clona um ou mais beacons da infraestrutura e ao invés de usá-los fora da área de atuação, usa dentro desta mesma área [48]. Por exemplo: um aplicativo libera uma determinada funcionalidade apenas quando está captando sinais de um beacon instalado em uma área estabelecida; se este beacon for clonado e alocado em outro lugar, poderá causar uma desorientação ao usuário, e dependendo do caso, poderá causar falhas ainda não previstas no aplicativo.

Ambos os dois ataques descritos por último podem ser realizados mesmo sem acesso físico ao beacon, já que dependem apenas do processo de clonagem; este que decorre do sinal recebido por um beacon da infraestrutura, o que irá ocorrer mesmo que ele esteja instalado em um local alto. Uma vulnerabilidade que não depende de nenhum processo malicioso de clonagem é o *Piggybacking* (português: pegando carona), onde o serviço de um terceiro utilizará a rede de beacons que não o pertence [48]. Como qualquer

dispositivo pode ler os dados de um BLE beacon, aplicações podem ser desenvolvidas para monitorar sinais recebidos por bluetooth e gerar interações em seus serviços usando uma infraestrutura de beacons sem permissão. Assim como nos outros casos, o uso do EID pode dificultar esse processo uma vez que apenas o dono da rede de beacons terá acesso à como descriptografar o dado emitido.

2.4 TECNOLOGIAS INDICADAS PARA O DESENVOLVIMENTO DA BASE DE UM SISTEMA IOT

Ao utilizar uma infraestrutura de BLE beacons, é necessário que alguma aplicação atue em torno dos eventos possibilitados por ela. Explorado no subcapítulo sobre Internet das Coisas, os sistemas IoT são constituídos, de forma geral, por três camadas: hardware, middleware e interface de aplicação. Considerando que a camada de hardware atua sobre o ambiente ou, como acontece com os BLE beacons, gera sinais de entrada para outro sistema, as camadas de middleware e aplicação são responsáveis por realizar tomadas de decisão que resultarão em mudanças nas interfaces digitais e ou ações na camada de hardware. Este subcapítulo explora as principais tecnologias de implementação de software utilizadas em ambas as camadas de middleware e aplicação.

2.4.1 *Publish Subscribe*

O *design pattern Publisher Subscriber* (pub/sub) é utilizado principalmente em sistemas distribuídos, como microsserviços e dispositivos IoT, para garantir a troca de informações em tempo real entre os subsistemas envolvidos.

O pub/sub é, de forma resumida, uma rede de comunicação em tempo real baseada em eventos, pois envia mensagens em uma plataforma que centraliza canais, transmitindo dados dos *Publishers* (publicadores - pubs) para o *Subscribers* (inscritos - subs). Os pubs enviam uma mensagem em um canal da rede sem saber quem o está escutando. Dependendo da tecnologia pub/sub utilizada, essa mensagem pode ser um objeto ou uma *string*. No outro lado existem os subs, que ouvem um ou mais canais da rede e captam as mensagens recebidas pelos pubs [58].

O padrão pub/sub traz diversas vantagens aos sistemas IoT. Pela ampla variedade de sensores e atuadores que podem ser utilizados neste tipo de projeto, a padronização de comunicação pode tornar-se uma inimiga da performance e boas práticas de desenvolvimento. Observado na Figura 6, utilizando padrão pub/sub é possível enviar uma mensagem sem se preocupar de qual forma ela está sendo entregue, da mesma forma que o sistema receptor receberá a mensagem sem precisar se preocupar de que forma ela está sendo transmitida. Os pubs apenas se preocupam em enviar a mensagem e os subs em

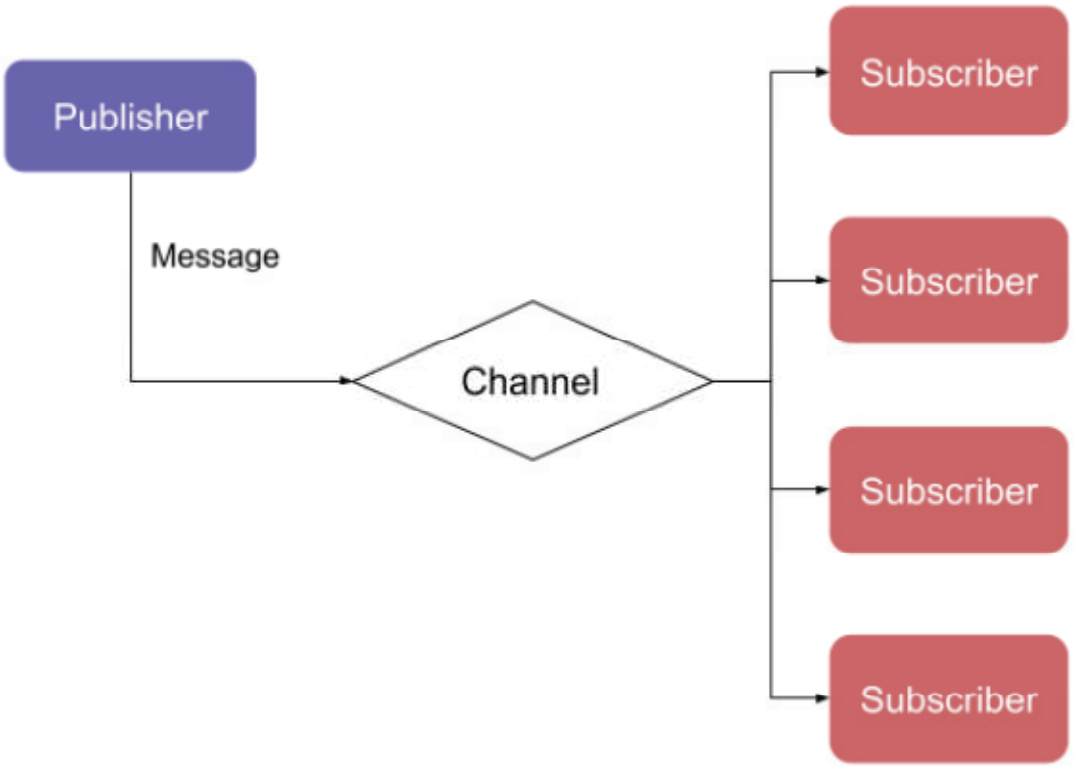

Figura 6. PubSub *workfkow*
Fonte: autor

recebê-las. Em termos, é muito útil quando se está trabalhando com uma grande variedade de dispositivos e sistemas, agregando velocidade de implementação e padronização da qualidade; assim como no envio de dados de um ou mais softwares integrados a uma *smart city* e ao desempenho do sistema.

O trabalho #1 [58] demonstra o uso desse método para contribuir com as boas práticas de arquitetura de sistemas IoT. Uma vez que um sistema IoT possui diversos sensores e atuadores descentralizados, por exemplo em um *smart building* que é composto por diversas dependências (sistemas de câmeras, sensores de fumaça, temperatura, luminosidade, presença, portão eletrônico, vibração, etc.), dispondo de variadas funcionalidades, é de vital importância que o desempenho e a comunicação dos dados gerados ou ações de atuadores seja confiável. No estilo de arquitetura distribuído diversos problemas podem surgir no desenvolvimento e execução.

No projeto #2 [59] o autor foca seus esforços na busca das melhores práticas para o desenvolvimento e manutenção de sistemas IoT, descrevendo o uso de três pilares. O primeiro pilar é o *design pattern* pub/sub, explorado profundamente em #1 que demonstra a capacidade de agregar valor a sistemas embarcados. Os outros dois pilares são a conteinerização e *logs* unificados. A conteinerização busca criar uma infraestrutura de proporção horizontal que beneficia a escalabilidade do *backend* da regra de negócios. Os *logs* unificados dizem respeito a análise de diversos softwares e outros sistemas embarcados

diferentes em um mesmo local em ordem cronológica, o que possibilita compreender os processos desempenhados pelo sistema e uma versão única da verdade, visto que a manutenção de sistemas com tempo de amadurecimento tendem a se tornar complexos em ambiente de produção.

2.4.2 Django Rest e Django Web Framework

O Django Web é um *framework* Python para desenvolvimento de *web services* que utiliza licença BSD. Originalmente desenhado para implementar sites que desempenham funções jornalísticas, é um dos *frameworks* mais conhecidos na linguagem de programação Python e também conta com uma das maiores comunidades desta linguagem. É um *framework web full stack*, ou seja, possui ferramentas que implementam as principais funcionalidades de um software web: *frontend*, *backend* e gerenciamento de banco de dados. Para implementar essas funcionalidades o *framework* usa o padrão MVC, que cuida do gerenciamento do banco de dados, renderização das páginas web e interpretação das solicitações, respectivamente [60].

O Django Web também é conhecido por possuir um bom desempenho para um *framework* de linguagem interpretada que passa por um processo de compilação. Além disso, é verticalmente escalável e implementa métodos de segurança e utilidades de programação como o Django *Signals*, que utilizando o *design pattern Observer*, é responsável por identificar ações realizadas com os *Models* e executar *scripts* de acordo com as regras de negócio da aplicação. Outra funcionalidade atrativa é a sua área administrativa, que é gerada automaticamente a partir da configuração dos *Models* [60].

Aliado ao Django Web também há o Django Rest. Um *framework* para construção de APIs Rest (*Application Programming Interface*. Rest: *Representational State Transfer*) e trabalha em conjunto com o Django Web. Nos últimos anos, a arquitetura Rest foi quem permitiu que integrações entre sistemas distribuídos pudessem se tornar mais ágeis e simplificadas, criando um padrão de desenvolvimento seguido por programadores a fim de expor suas soluções a terceiros. O Django Rest é um *framework* que auxilia a criação de APIs para tecnologias que foram implementadas usando Django Web [61]. Quando uma solicitação é feita ao Django buscando recursos de API, o Django Rest cuida das autenticações, permissões, interpreta a requisição, utiliza seus próprios recursos para interagir com o *Model* do Django e por fim se encarrega da resposta para a requisição.

Além das características presentes no *framework*, o uso da linguagem Python simplifica o desenvolvimento por possuir uma curva de aprendizagem curta que propicia a criação de protótipos e MVPs. Os sistemas desenvolvidos com essas características podem implementar qualquer tipo de funcionalidade, incluindo BLE beacons, como é o caso de [23] que utiliza beacons juntamente com regras de negócio envolvendo gamificação em

smart toilets; e [43] onde é implementado um sistema de notificações para ambiente *smart campus*.

2.5 RESULTADOS E DISCUSSÕES

Este capítulo desempenhou o trabalho de levantar os principais questionamentos para a busca de soluções presentes no que tange segurança pública (com foco em furto e roubo de veículos), ambientes *smart*, IoT e BLE beacons. Para isso, foram elaboradas 3 questões:

1. É possível analisar furto e roubo de veículos com o uso de BLE Beacons?
2. Como as smart cities e os *smart campus* lidam com a segurança pública?
3. Quais tecnologias IoT estão sendo usadas para auxiliar indivíduos e instituições na segurança pública?

As discussões abertas pela Revisão Sistemática da Literatura demonstraram que o uso de BLE beacons já vem sendo adotado nos mais variados cenários e cada vez mais ganham espaço nos ambientes inteligentes. Uma das características presentes nos tipos de aplicação possibilitados por este dispositivo é a identificação de sua presença dentro de um escopo (triangulação ou aproximação), o que abre possibilidades para a implementação de vigilância de veículos em ambientes *smart*. Ainda, o uso de BLE beacons se justifica à frente de outras tecnologias similares por mostrar-se mais adequada. Em comparação, de forma geral, os sinais do RFID possuem um curto alcance, enquanto os sinais de GPS possuem um longo alcance, porém com alto consumo de energia, WIFI é uma tecnologia com foco em cobertura de serviço de rede, enquanto o BLE beacon é pouco explorado e agrega atributos que exploram métodos de *indoor* e *outdoor location*.

As palavras-chave utilizadas para a elaboração da pesquisa tiveram grande foco em identificar as tecnologias IoT envolvendo BLE beacons no âmbito da segurança pública em contexto de cidades e campus inteligentes. Os trabalhos selecionados demonstraram as principais abordagens utilizadas nestes ambientes, que por sua vez ressaltaram a criatividade dos pesquisadores e desenvolvedores ao utilizar BLE beacons como uma tecnologia de suporte em suas soluções; uma vez que a camada de hardware funciona em conjunto com as outras camadas e não desfruta de isolado protagonismo.

Quanto aos métodos utilizados frente à insegurança através de IoT, poucos estudos abordaram o uso desse paradigma para propósitos de segurança da vida e propriedade. De forma geral, os estudos encontrados são dependentes de câmeras de vigilância e alarmes contra invasão e que foram descartados durante a pesquisa de RSL por não fazerem uso prático de tecnologias IoT com uso de BLE beacons. Tendo isso em vista, estes fatores mostraram um espaço pouco explorado e passivo de investigação.

Gradualmente o IoT vem ganhando espaço nos ambientes *smart*. Tradicionalmente, tecnologias de IoT são conhecidas pelo uso de diversos sensores e atuadores, principalmente no que diz respeito ao monitoramento da qualidade do ar, gases, movimento, presença, entre outros. A capacidade de monitoramento de sensores aliados à tecnologias de transmissão, processamento e análise de dados, criam diversos caminhos que podem ser empregados para entender o funcionamento das cidades e melhorar a tomada de decisão. No IoT as propostas que utilizam BLE beacons vem se moldando de acordo com a criatividade e necessidade do ambiente, seja um banheiro, sala de aula, biblioteca, um campus ou uma cidade inteira. A sua característica em sistemas *indoor* abre diversas possibilidades uma vez que se trata de uma tecnologia que vem ganhando recente notoriedade e ainda não foi explorada nas mais variadas situações.

Por fim, pode-se reconhecer que há espaço para novas propostas voltadas a ambientes inteligentes em benefício da segurança pública por ser um contexto pouco explorado. Na revisão sistemática não foram identificados trabalhos com BLE beacons focados em segurança da vida e segurança patrimonial. Apesar dos problemas discutidas no capítulo 2.3.3.3 (RSSI e Tipos de Aplicação), a proposta central desta pesquisa pode ser trabalhada visando diminuir o impacto negativo gerado pela oscilação dos RSSIs. Usando destes sinais é possível identificar a movimentação dos dispositivos, que alocados dentro de veículos podem indicar uma movimentação suspeita através da leitura e processamento adequado destas informações.

3. MATERIAIS E MÉTODOS

Este capítulo aborda o método utilizado para executar os experimentos da pesquisa. Entre as etapas estão os testes para análise de RSSIs dos BLE beacons e a aplicação desenvolvida para validar o objetivo proposto de investigar a viabilidade do uso de beacons para identificar furto e roubo de veículos. A aplicação implementada foi chamada de **Veacon** (Vehicle + Beacon).

O fluxo do projeto é apresentado na Figura 7, constituído de 3 macroatividades: dissertação, desenvolvimento tecnológico e testes e validação.

Figura 7. Fluxo Metodológico
Fonte: autor

A **Dissertação** é o componente que compreende toda a revisão literária que guia o desenvolvimento do projeto, bem como descreve os objetivos, justificativas e metodologia

aplicada. O **Desenvolvimento Veacon** contém as atividades necessárias para o desenvolvimento do sistema utilizado para testar a proposta do trabalho. Já os microprocessos de **Teste e validação** compõem o embasamento para a tarefa de testar as tecnologias e o resultado da proposta.

3.1 METODOLOGIA

Para atingir os objetivos propostos foi utilizada a abordagem metodológica experimental. O método experimental expõe determinado objeto estudado à variáveis controladas a fim de analisar o comportamento observado. Estas variáveis são estimuladas e manipuladas propositalmente para que alguma condição específica seja acionada, ou seja, não são variáveis com comportamento natural e sim controladas pelo pesquisador. Este método permite que haja uma melhor assimilação sobre os comportamentos do objeto de estudo, proporcionando uma análise mais aprofundada [62], que no caso deste trabalho conduzido é a solução desenvolvida para a identificação de furto e roubo de veículos.

3.2 FLUXO E FUNCIONAMENTO DO SISTEMA

O processo para a identificação de situações de furto e roubo de veículos ocorre através de um sistema que atua em 3 canais: hardware, *middleware* e camada de aplicação. Cada canal possui componentes que individualmente implementam funcionalidades que promovem a utilização de BLE beacons para identificar situações suspeitas em relação à movimentação de um veículo que pode estar sendo alvo de furto ou roubo, de acordo com o fluxograma da Figura 8.

Na camada de Hardware estão presentes dois componentes (assinalados com as cores azul e roxo) que trabalham em conjunto para o funcionamento do **sistema embarcado de monitoramento de BLE Beacons**. Já na camada de *Middleware* (em vermelho e marrom) estão os componentes responsáveis pelo transporte de informações entre os sistemas presentes na camada de Hardware e Aplicação. Os últimos componentes estão na Camada de Aplicação, em marrom e verde, representando o **sistema de gerenciamento de dados e alertas** e os *stakeholders* que interagem com o sistema, respectivamente.

A figura 9 demonstra, de forma macro, o fluxo da aplicação para a identificação de um possível incidente de furto e roubo desde o componente do BLE beacon até a Camada de Aplicação. Ao utilizar um BLE beacon em um veículo e inserir um novo monitoramento no sistema de gerenciamento de dados (etapa abordada com mais detalhes nos próximos capítulos), o sistema de gerenciamento de beacons monitora as alterações de estado do dispositivo utilizado no monitoramento. Ao detectar uma alteração, é utilizada a camada de *Middleware* para informar o sistema de gerenciamento de dados que um incidente está

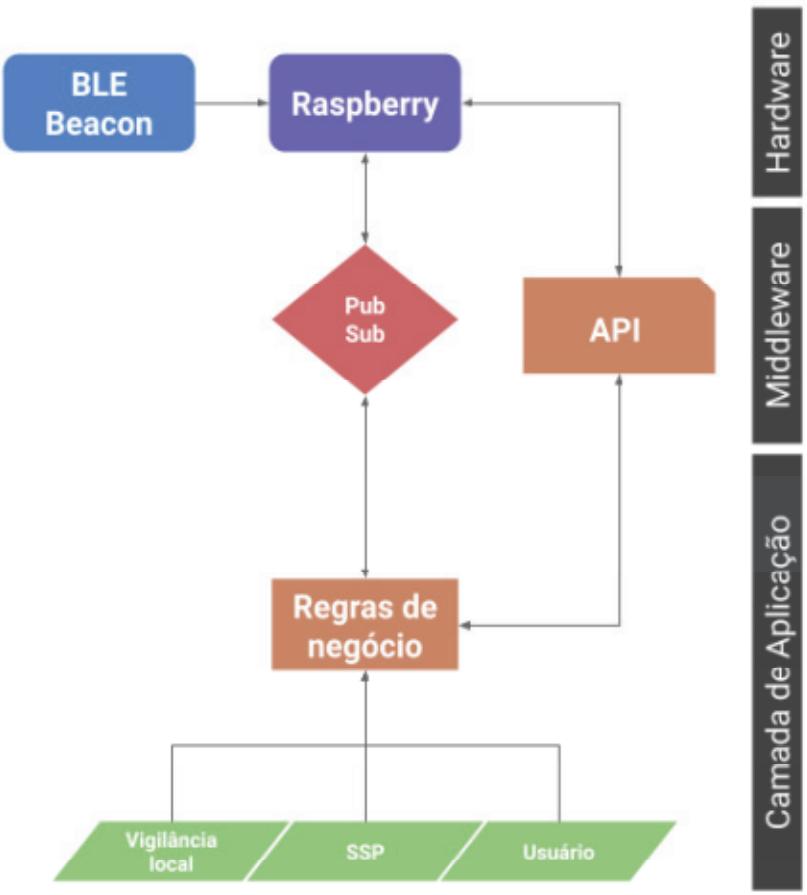

Figura 8. Fluxograma do sistema
Fonte: autor

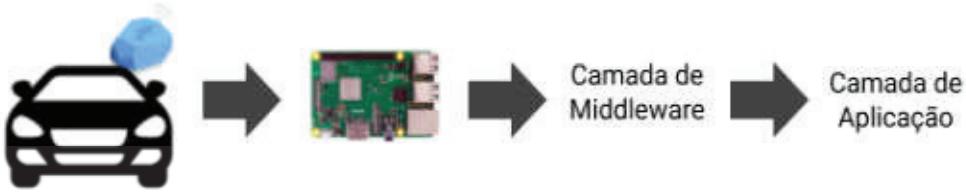

Figura 9. Fluxograma do sistema
Fonte: autor

ocorrendo. Nesta última camada o *stakeholder* de vigilância local do *smart campus* interage com o sistema para receber o alerta sobre o incidente.

De forma geral, o sistema funciona com os seguintes componentes: **BLE beacons e aquisição de dados** como tecnologia central que proporciona o uso do método aplicado pelo sistema para identificar situações de furto e roubo de veículos. O **sistema embarcado de monitoramento de BLE beacons** é a parte utilizada para gerenciar e monitorar os **BLE beacons** atuando no entorno do sistema. É através dele que toda a lógica para identificação de situações que representem possíveis ameaças aos veículos é implementada. Já o **sistema web de gerenciamento de dados e envio de alertas** é o centralizador de todos os elementos. É nele que as informações referentes aos BLE beacons, veículos e moni-

toramentos são gerenciados, e também é o responsável por enviar alertas quando algum incidente de furto ou roubo estiver ocorrendo. O sistema Veacon está presente no sistema embarcado e web, sendo chamado de Veacon Rasp e Veacon Web, respectivamente. Por fim, os *stakeholders* são os atores que representam as principais partes interessadas no projeto. Neste caso o órgão da Secretaria de Segurança Pública (SSP) responsável pelo policiamento ostensivo, representando uma organização que poderia se beneficiar da aplicação em um possível futuro, a força de Vigilância Local do *smart campus* e o Usuário (proprietário do veículo).

3.3 BLE BEACONS E AQUISIÇÃO DE DADOS

Uma característica presente nos protocolos de comunicação dos beacons é a capacidade de medir a força do sinal emitido do beacon até um dispositivo receptor. Essa funcionalidade é útil quando se está buscando a origem do sinal, ou, ainda, para saber o quão perto um dispositivo (receptor) está do beacon emissor (fonte do sinal). Isso pode ser utilizado para acionar eventos registrados nas regras de negócio da aplicação a fim de gerar uma experiência de uso que integre interações do mundo digital ao mundo físico e vice-versa.

Ao instalar um BLE beacon no meio de transporte e registrar um novo monitoramento, o subsistema de monitoramento verifica as mudanças de estado do beacon, neste caso a mudança de local. Caso o veículo movimente-se dentro do período de vigilância, o BLE beacon irá emitir um sinal de um ponto distinto do que estava originalmente, o que gera uma mudança de intensidade do RSSI, que ao ser recebido pelo sistema de monitoramento é processado e verificado o deslocamento. Portanto, os BLE beacons possuem o comportamento que permite utilizar as funcionalidades dos subsistemas para identificar o possível furto e roubo de veículo.

Neste trabalho estão sendo utilizados BLE beacons de um kit de desenvolvimento *location* beacons da marca Estimote, cedidos pelo parque tecnológico UPF Parque. Os dispositivos do kit de desenvolvimento permitem a mudança de diversas configurações, tais como: escolha e configuração de protocolo (Eddystone, iBeacon e Estimote), alcance de 1 a 200 metros e intervalo de envio de sinais de 100 ms a 5000 ms. Além disso, também possuem resistência à água, baterias com 4000 mAh capazes de oferecer cerca de 5 anos de funcionamento ao dispositivo, camada de adesivo para fixação, SDK para múltiplas plataformas e serviço de API e *dashboard* em nuvem.

3.4 *PUBLISHER SUBSCRIBER*

O Veacon, tanto a versão Web quanto Rasp, faz uso do módulo PubSub implementando um *design pattern* de mesmo nome: *Publish Subscribe* (pub/sub). No projeto, este padrão é implementado através do serviço de mensageria PubNub, utilizando o SDK Python PubNub 4 disponibilizado pelo provedor do serviço. O pub/sub facilita a comunicação entre sistemas distribuídos e possibilita a transferência de dados usando eventos e canais de comunicação.

Um sistema utilizando pub/sub pode enviar uma mensagem com informações para um ou mais canais. Nestes canais, outros sistemas autenticados podem estar inscritos para receber notificações de eventos com transferência de dados. A partir daí qualquer um deles que esteja ouvindo pode capturar os dados enviados ao canal. O processo de envio é conhecido como publicação (*publish*) e o de inscrição e posterior recebimento é possível devido à inscrição (*subscribe*). Neste formato, o *publisher* preocupa-se apenas em identificar o evento gatilho que aciona o envio de dados e encaminhar as informações para os canais necessários. Enquanto isso, o *subscriber* se preocupa apenas em inscrever-se no canal e aguardar as informações publicadas, se preocupando apenas em como irá tratá-las após o recebimento.

No projeto, este *design pattern* é utilizado para trocar dados entre o sistema web e embarcado a fim de transmitir as mudanças de estado de registros de monitoramento cadastrados na interface web. As características apresentadas por esta estratégia de *design* de implementação contribuem para as boas práticas de comunicação multi-sistema, como é o caso do Veacon que possui uma versão Web e uma versão Rasp, justificando o uso da metodologia e do serviço de mensageria. Outros serviços com o mesmo propósito podem ser utilizados, porém, o PubNub foi escolhido pela facilidade e rapidez de implantação.

3.5 SISTEMA WEB DE GERENCIAMENTO E ENVIO DE ALERTAS

O sistema Veacon Web de gerenciamento de envios e alertas é o responsável por gerenciar todos os dados dos recursos utilizados pelo sistema. Entre estes recursos estão:

1. BLE beacons utilizados pelo projeto
2. *Gateways* de monitoramento (dispositivos Raspberry PI 3 Model B rodando o programa desenvolvido para este trabalho chamado de Veacon Rasp)
3. Usuários (pessoas físicas que registram monitoramento de veículos)
4. Veículos (propriedades de pessoas físicas)
5. Monitoramentos (executados por *gateways*)

6. Alertas (disparados quando a regra de monitoramento é acionada no *gateway*)
7. API de acesso (utilizada pelos *gateways* para recuperar dados registrados no Veacon Web e enviar alertas)
8. Serviço de mensageria PubSub (serviço de troca de mensagens através do *design pattern* PubSub implementado no lado do servidor)

Essa aplicação foi implementada utilizando a linguagem Python 3.7, o *framework* para desenvolvimento Django Web Framework 3, Django Rest Framework e banco de dados SQLite3.

3.5.1 BLE Beacons

Este módulo do sistema gerencia as informações pertinentes aos beacons utilizados no projeto. Parte da regra de negócio para vigilância de veículos é o escaneamento de beacons instalados em veículos que estejam em monitoramento, porém, outros beacons que estejam agindo na região também podem ser percebidos durante o processo. Para evitar que sinais BLE de terceiros sejam percebidos durante essa etapa, é necessário cadastrar no sistema os dados de BLE beacons válidos para o uso em monitoramentos.

Durante testes realizados para explorar a tecnologia, o protocolo Eddystone mostrou-se o mais adequado para utilização durante o desenvolvimento do projeto. Parte da decisão foi baseada na simplicidade do uso do Eddystone UID, que envia sinais com um identificador universal, bem como a possibilidade de em trabalhos futuros utilizar o protocolo Eddystone EID, acrescentando opções de segurança ao sistema. Outro fator determinante é a performance apresentada pela biblioteca que realiza a leitura dos sinais em interpretar os *frames* Eddystone, que na versão utilizada mostrou-se superior à leitura de iBeacons (assunto abordado no capítulo 3.6.3). Para registro de um beacon com o UID é utilizado o nome do beacon (eddy_namespace). Este beacon é atrelado a um usuário ao qual pertence (diagrama ER Figura 10) e posteriormente é utilizado para cadastro de monitoramento de veículos (maiores detalhes nos próximos capítulos).

3.5.2 *Gateways* de monitoramento

Os *gateways* de monitoramento são os dispositivos que embarcam o sistema Veacon Rasp. No Veacon Web, esses dados são constituídos por um ID responsável por identificar cada *gateway* disponível para monitoramento, um nome (que em conjunto com o ID é utilizado para gerenciar as mensagens enviadas ao *gateway* pelo módulo PubSub), latitude e longitude utilizados para conhecer o posicionamento do dispositivo e observações (diagrama ER Figura 10).

3.5.3 Usuários

O módulo de usuários do sistema Veacon é baseado no sistema encapsulado pelo framework Django Web. A única alteração realizada para adicionar novos dados ao modelo de usuários do foi a adição de um número de telefone para em caso de furto ou roubo do veículo seja acionado o responsável pelo monitoramento.

3.5.4 Veículos

Além de realizar contato com o usuário responsável pelo monitoramento de um determinado veículo, o agente de segurança responsável pelo *smart campus* também precisa de acesso aos dados do veículo para identificá-lo em meio aos outros automóveis da via. Para isso é utilizado o cadastro de placa, cor, modelo e marca de cada veículo a ser monitorado (diagrama ER Figura 10). Ainda, o veículo pode possuir muitos proprietários (Usuários) e os usuários podem possuir muitos veículos, abrangendo uma forma flexível para o cadastro.

3.5.5 Monitoramentos

O cadastro de monitoramento, implementado como um módulo do sistema chamado de *Watchpost* (torre de vigia), é uma funcionalidade que engloba todos os outros itens. Para que o monitoramento seja realizado, ele depende do BLE beacon que será vigiado, do veículo ao qual ele foi instalado, do usuário que cadastrou o monitoramento e do *gateway* que irá monitorar o beacon.

Além destas dependências, o sistema implementa a informação de data e horário em que o monitoramento foi iniciado e finalizado, o valor de RSSI mais distante e mais próximo, observações adicionadas no cadastro realizado pelo usuário (observações que podem ser lidas pelo agente de segurança), status de monitoramento e data de última atualização.

3.6 SISTEMA EMBARCADO DE MONITORAMENTO DE BLE BEACONS

Por apenas transmitirem dados através de sinais propagados, os BLE beacons sozinhos não possuem a capacidade de implementar qualquer funcionalidade. A parte do sistema responsável por identificar possíveis alterações de estado que evidenciam situações de risco é o subsistema de monitoramento dos beacons que são utilizados no projeto. Esse sistema foi desenvolvido usando Python 3.7 e é executado em um Raspberry PI 3 Model B.

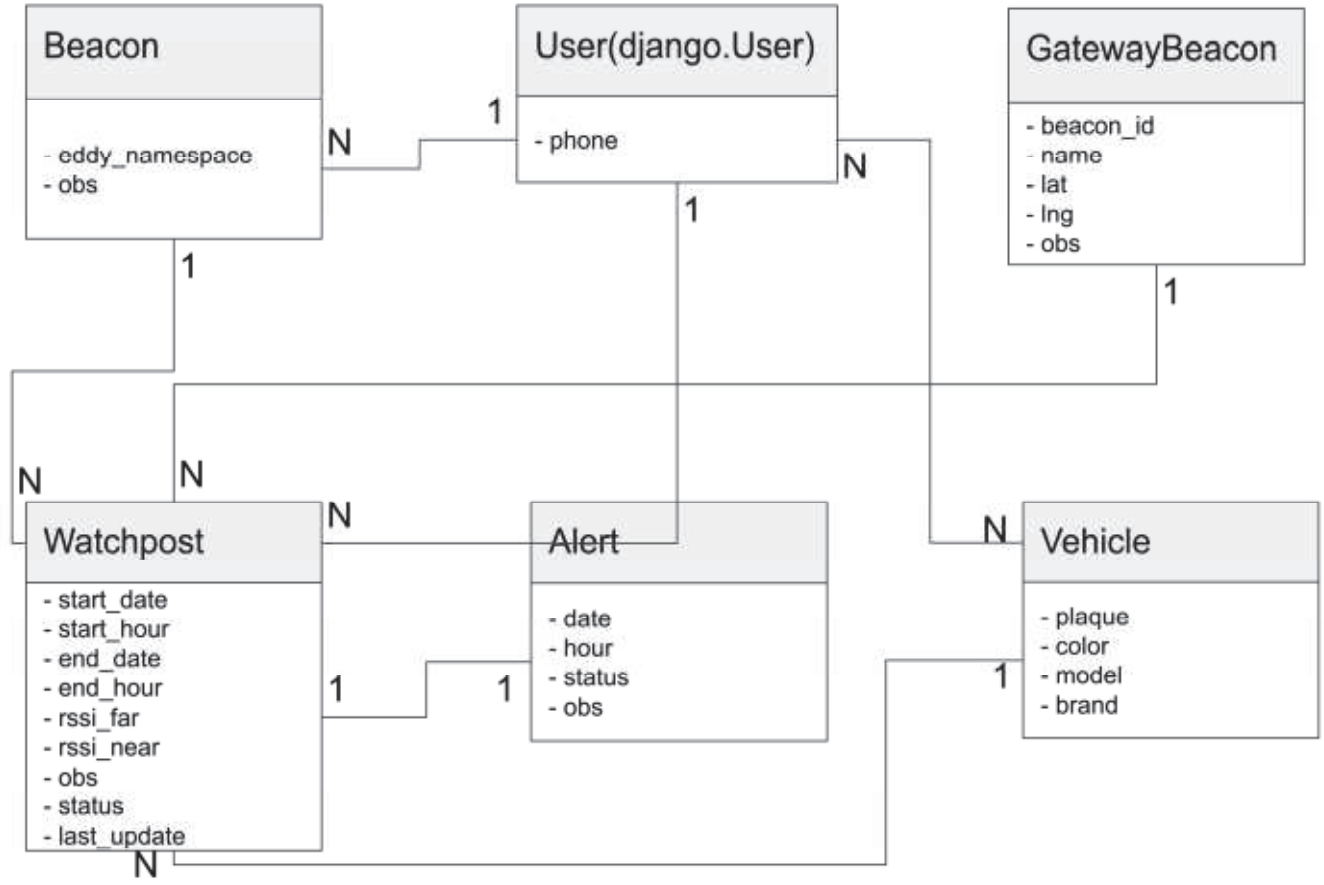

Figura 10. Diagrama Entidade Relacionamento do Veacon Web
Fonte: autor

Neste contexto, este subsistema faz uso de 4 módulos que implementam suas regras de negócio:

1. Beacons
2. *Watchpost*
3. PubSub
4. *Server request*

Todos os 4 módulos são implementações especialistas que gerenciam suas próprias atribuições, sendo conectadas pelo módulo *Core*, que trabalha como um *Facade*[5] para abstrair as complexidades de cada módulo e integrá-los em um único lugar.

[5] *Design Pattern Facade*. O *Facade* (português: Fachada) oferece uma interface de acesso simplificada de alto nível que abstrai as complexidades de outras implementações.

3.6.1 *Server Request*

O módulo *Server Request* implementa os recursos que possibilitam ao Veacon Rasp comunicar-se com o Veacon Web através de requisições HTTP[6]. Este módulo serve como base construtora de acesso para todos os outros módulos que requisitam informações do sistema web.

3.6.2 PubSub

No Veacon Rasp, o módulo PubSub implementa as especificações para recebimento das mensagens enviadas pelo Veacon Web. Este módulo é responsável por captar as mensagens e armazená-las adequadamente para serem posteriormente processadas pelo módulo *Watchpost*.

3.6.3 Beacons

Uma das principais funções desempenhadas pelo Veacon Rasp é a leitura de BLE beacons próximos. Neste processo é preciso identificar alterações de estado em beacons usados pelo sistema e desconsiderar outros que estejam emitindo dados na região. Para isso o módulo Beacons é implementado através de um "*Beacon Manager*" que gerencia todas as funcionalidades referentes aos BLE beacons que são utilizadas para monitoramento dos veículos. Entre essas funcionalidades estão: identificar beacons do sistema e ignorar beacons sobressalentes, processar atualizações do Veacon Web sobre beacons ativos e inativos (aqueles que devem e que não devem ser monitorados, respectivamente), adicionar e remover beacons ativos e inativos, escanear beacons próximos e gerenciar o formato de informações descobertas através do escaneamento dos beacons ativos da região.

Para realizar a leitura dos BLE beacons está sendo utilizada uma biblioteca *open source* Beacontools [63], que coleta os dados dos dispositivos atuantes na região. Posteriormente, a implementação presente no BeaconManager gerencia essas informações para que sejam consumidas pelo módulo de monitoramento.

A abstração do BeaconManager no sistema pode ser vista na Figura 11.

1. **ble_read_time**: tempo em segundos que será utilizado para escaneamento de beacons

2. **beacon_rssis**: estrutura chave valor onde a chave é o identificador do BLE beacon e o valor uma lista de RSSIs lidos deste mesmo beacon

[6]HTTP: *Hypertext Transfer Protocol* (português: Protocolo de Transferência de Hipertexto). É um protocolo de comunicação que utiliza a internet para transferir informações entre sistemas.

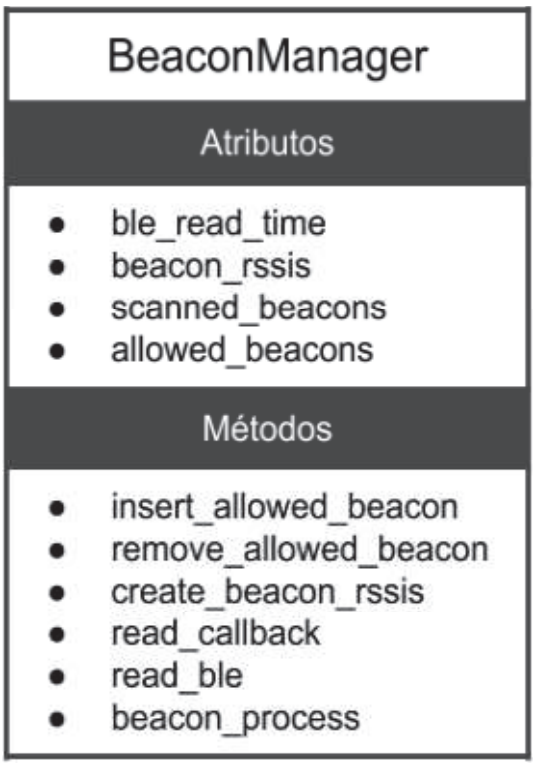

Figura 11. Beacon Manager
Fonte: autor

3. **scanned_beacons**: uma lista de tuplas, onde cada tupla possui dois valores: o identificador do beacon e uma leitura identificada pelo processo de escaneamento de sinal de beacon

4. **allowed_beacons**: lista com os beacons que podem ser examinados pelo processo de escaneamento

5. **insert_allowed_beacon**: adiciona novos beacons à lista **allowed_beacons**

6. **remove_allowed_beacon**: remove um beacon da lista **allowed_beacons**

7. **create_beacon_rssis**: retorna uma estrutura chave valor através da iteração sobre **scanned_beacons.** A chave é o identificador do beacon e o valor uma lista com os RSSIs do beacon lidos pelo processo de escaneamento

8. **read_callback**: verifica se um beacon escaneado está na lista **allowed_beacons.** Se estiver adiciona na lista **scanned_beacons** uma tupla com o identificador do beacon e o valor de RSSI identificado pelo Beacontools

9. **read_ble**: configura o Beacontools e inicia a leitura dos BLE beacons de protocolo Eddystone. Quando um beacon for lido, aciona o **read_callback** enviando os dados do escaneamento

10. **beacon_process**: limpa a lista **scanned_beacons**, aciona o **read_ble** e executa e retorna as informações de **create_beacon_rssis**

Este conjunto de atributos e comportamentos são utilizados para gerenciar todas as regras de negócio referentes à beacons no sistema Veacon Rasp. O fluxo de aplicações destas regras pode ser visto na Figura 12, que demonstra a forma com que o **BeaconManager** realiza o gerenciamento das informações quando solicitado pelo **WatchpostManager** (próximo capítulo).

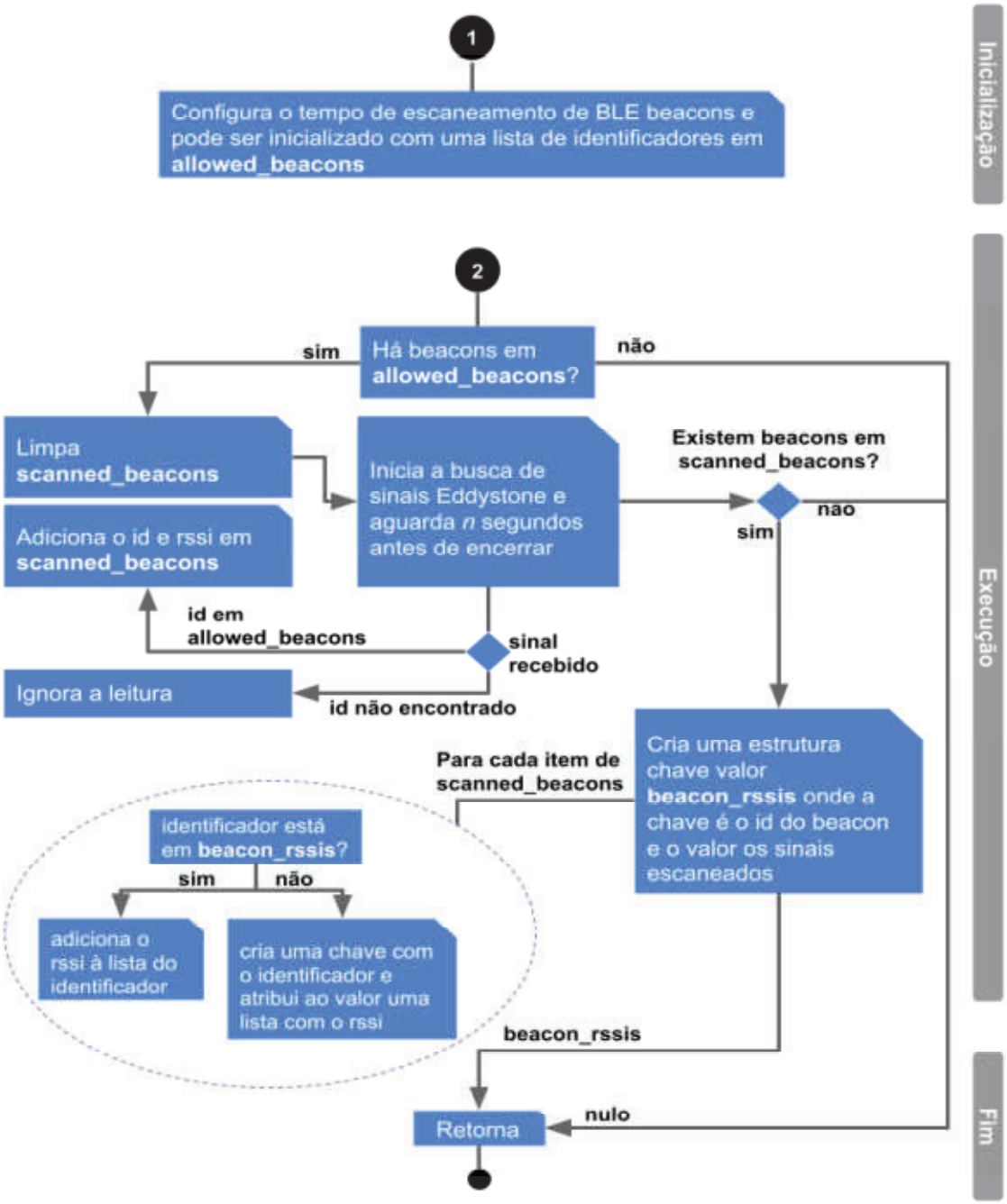

Figura 12. Funcionamento BeaconManager
Fonte: autor

A primeira etapa representa a inicialização do módulo. Sua principal função é ajustar o tempo de leitura que o Beacontools levará para realizar a leitura dos BLE beacons e inicializar variáveis utilizadas pelo próprio módulo. O valor de tempo de leitura deve estar em sincronia com a configuração dos beacons monitorados pelo sistema, tendo em vista que o intervalo de envio dos sinais dos BLE beacons são configuráveis. Quanto mais sinais por segundo o beacon envia dentro do seu campo de alcance, menos tempo o BeaconManager precisará para reunir a quantidade ideal de informações que serão usadas para identificar a movimentação dos veículos monitorados.

3.6.4 Watchpost

Uma das características observadas nos testes de leitura dos BLE beacons (subcapítulo 3.8) é a volatilidade das leituras recebidas pelo dispositivo receptor. Em virtude da natureza da tecnologia utilizada [57], a variabilidade do valor de RSSI lido é considerável. Isso impossibilita uma leitura sólida como um valor constante que não varie caso a distância entre o dispositivo emissor e receptor não seja alterada. Para contornar esse comportamento é necessário aplicar uma regra de negócio a fim de estabilizar as variáveis utilizadas para o processo de tomada de decisão, evitando assim alertas gerados por falsos positivos.

Watchpost é o principal módulo utilizado no sistema Veacon Rasp para consolidar os dados de leitura. Apesar do módulo Core (próximo subcapítulo) centralizar as funcionalidades implementadas pelos subsistemas e possibilitar que cada um coopere individualmente para complementar a funcionalidade central, é o módulo Watchpost que gerencia grande parte das regras de monitoramento através de um "**Watchpost Manager**".

Para desempenhar as regras de vigilância, o Watchpost Manager implementa dois principais itens: Beacon Manager e uma lista de monitoramentos. O Beacon Manager gerencia os beacons ativos e inativos que poderão ser vigiados pelo Watchpost Manager, e a lista de monitoramentos é uma estrutura chave valor em que a chave é a *String* do identificador de um BLE Beacon que esteja sendo monitorado e o valor é um objeto Watchpost que guarda informações sobre o alvo de interesse.

As especificações do objeto Watchpost podem ser vistas na Figura 13.

1. **id**: identificador do monitoramento no sistema Veacon Web
2. **eddy_namespace**: identificador do beacon dentro do veículo (UID)
3. **status**: status de monitoramento
4. **rssi_near**: valor de RSSI mais próximo identificado pelo Veacon Rasp
5. **rssi_far**: valor de RSSI mais distante identificado pelo Veacon Rasp

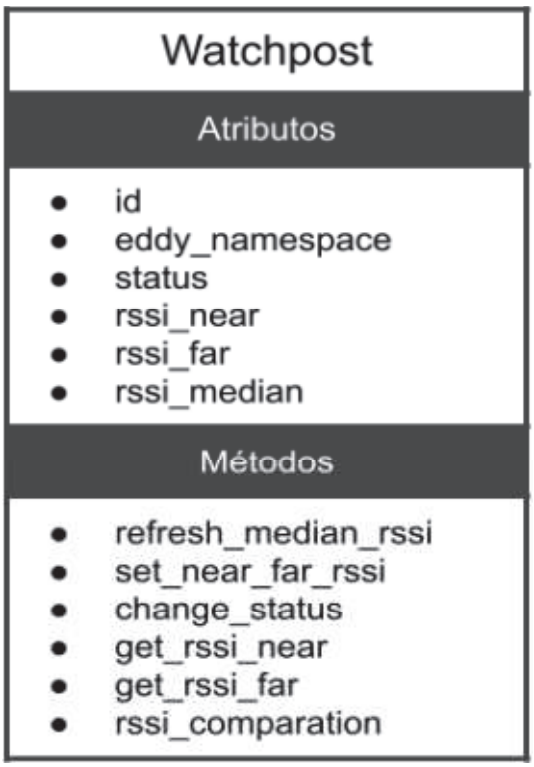

Figura 13. Representação do objeto Watchpost
Fonte: autor

6. **rssi_median**: valor de mediana identificado pelo WatchpostManager de acordo com as leituras do monitoramento
7. **refresh_median_rssi**: atualiza o valor da mediana do objeto através de uma lista de novas leituras do beacon referente ao monitoramento
8. **change_status**: atualiza o status de monitoramento do objeto
9. **rssi_comparation**: verifica o status do monitoramento, se estiver ativo analisa se o valor da mediana está entre o valor do **rssi_far** e **rssi_near**. Se estiver, indica que o veículo não mudou de lugar, do contrário, indica que o veículo moveu-se

Na Figura 14 Abaixo a descrição do WatchpostManager que implementa a lista de objetos Watchpost para gerenciar os monitoramentos do Veacon Rasp.

O **WatchpostManager** executa as regras de negócio utilizadas para gerenciar os monitoramentos do sistema. Os itens dessa abstração representam as seguintes funcionalidades:

1. **watchposts**: dicionário[7] de objetos Watchpost que representa os monitoramentos do gateway onde sua chave é o UID do beacon
2. **exists**: verifica se um determinado beacon já está sendo monitorado pelo sistema

[7] Estrutura chave-valor nativa da linguagem de programação Python.

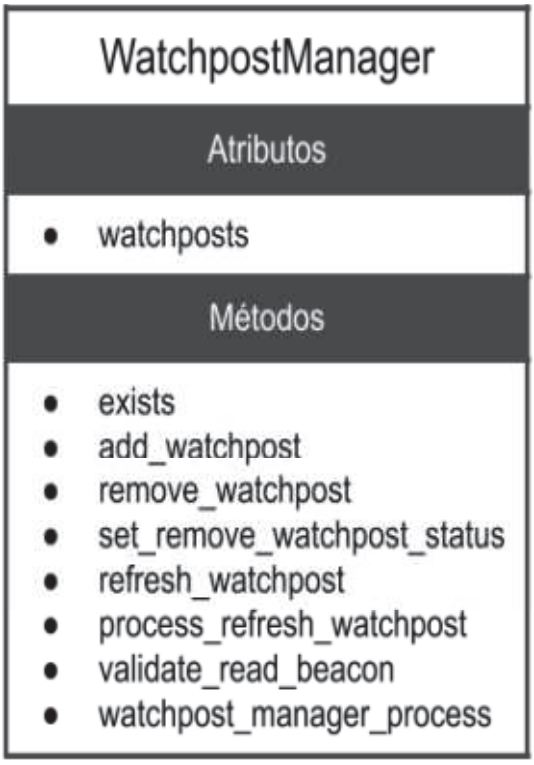

Figura 14. Watchpost Manager
Fonte: autor

3. **add_watchpost**: adiciona um novo objeto Watchpost (monitoramento) ao dicionário de monitoramentos watchposts

4. **remove_watchpost**: remove um objeto Watchpost do dicionário de monitoramentos de acordo com o identificador do beacon monitorado

5. **set_remove_watchpost_status**: atualiza o parâmetro de status do monitoramento para ser removido durante a etapa de atualização dos monitoramentos

6. **refresh_watchpost**: realiza o processo de atualização de um objeto Watchpost na lista de monitoramentos de acordo com seu status

7. **process_refresh_watchpost**: recebe uma estrutura de dados previamente processada resultante do escaneamento de BLE beacons utilizados para o monitoramento dos veículos. Itera sobre cada beacon escaneado utilizando o **refresh_watchpost** e ao fim da iteração atualiza o status de monitoramento

8. **validate_read_beacon**: verifica se houve a movimentação de um determinado BLE beacon através do **rssi_comparation** implementado pelo objeto Watchpost. Caso o dispositivo demonstre comportamento de movimentação envia um alerta através do módulo de requisições.

9. **process_validate_read_beacon**: itera sobre a lista de monitoramentos para executar o **validate_read_beacon**

10. **watchpost_manager_process**: é o processo que inicia a sequência de execução do **process_refresh_watchpost** para atualização do estado dos monitoramentos e o **process_validate_read_beacon** para analisar a situação de movimentação dos beacons monitorados

A Figura 15 demonstra o fluxograma que exibe a execução da abstração do Watchpost Manager buscando identificar a alteração de estado de um BLE beacon monitorado.

Ao iniciar o sistema, o primeiro processo inicializa o módulo injetando dados de monitoramentos cadastrados no Veacon Web. Para isso, realiza uma requisição que verifica a existência de monitoramentos cadastrados ativos para o presente *gateway*. Caso existam, irá inseri-los na estrutura de monitoramentos e adicionar os UIDs utilizados na lista de beacons permitidos pelo Beacon Manager.

Já a segunda etapa realiza o processo de atualização dos dados de monitoramento utilizando como base as informações do escaneamento dos beacons. Com a obtenção dos dados já realizada pelo Beacon Manager, o processo inicia uma iteração sobre a estrutura de monitoramentos. De acordo com o status de cada um, uma determinada ação será executada.

1. **A (Ativo)**: o monitoramento está ativo e os dados de leitura devem ser utilizados para atualizar a informação da mediana, que por sua vez será utilizada no próximo processo para descobrir se o beacon moveu-se

2. **I (Inativo)**: a monitoria foi cancelada pelo usuário. Com isso o monitoramento deve ser excluído e o UID do beacon deve ser removido da lista de leitura do Beacon Manager

3. **P (Processando)**: é o status utilizado quando o pedido de monitoria recém chegou ao Veacon Rasp. Isso indica que o objeto da lista de monitoramentos não está estruturado para passar pelo processo de validação (próxima etapa). Nesse status o sistema configura no objeto as informações de RSSI mais próximo e mais distante, mediana coletada pelo escaneamento e altera o status de 'P' para 'A'. Ao fim deste processo informa ao Veacon Web que o monitoramento está sendo realizado pelo *gateway*, detalhando os valores de leitura durante a requisição

A próxima e última etapa executada pelo Watchpost Manager valida o caso de movimentação dos BLE beacons. Iterando sobre cada objeto da lista de monitorias, o sistema verifica se o valor da mediana de RSSIs processada na etapa anterior está entre os valores de RSSI mais próximo e mais distante. Caso esteja, isso indica que o beacon permanece no mesmo local aproximado, caso contrário, aponta que uma movimentação do beacon ocorreu e um alerta deve ser enviado para o Veacon Web.

Esse fluxo implementa as regras utilizadas pelo Veacon para gerenciar os dados de monitoria e envio de alertas sobre situações de risco, como pode ser verificado no fluxo-

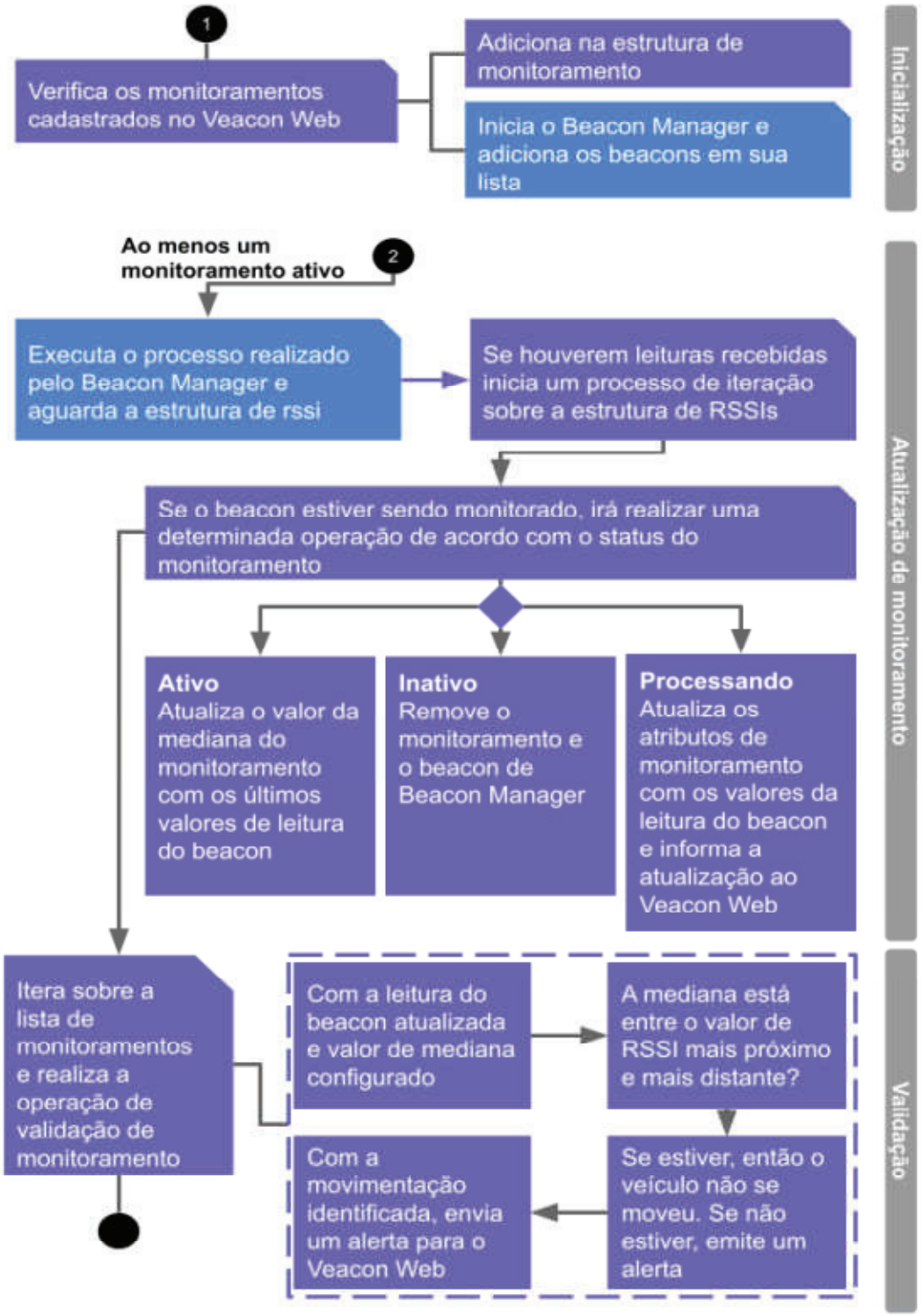

Figura 15. Watchpost Manager - Funcionamento
Fonte: autor

grama da Figura 16, que também aborda a criação, remoção e alteração de um monitoramento.

Apesar de o Watchpost Manager implementar as regras de adição, remoção e atualização, quem aciona estes eventos é o módulo Core do Veacon Rasp.

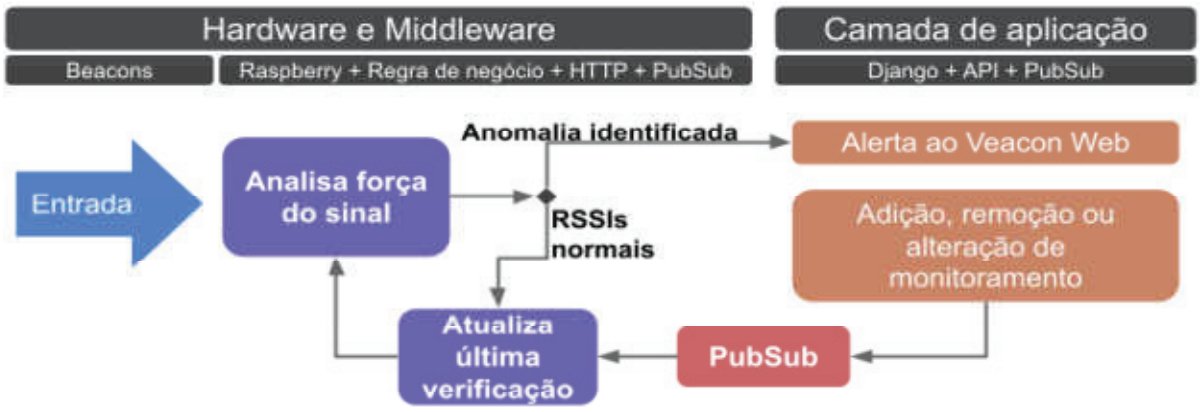

Figura 16. Regra de alertas e atualizações
Fonte: autor

3.6.5 Core

O módulo Core centraliza todas as operações do PubSub Manager e Watchpost Manager. Ele é o responsável por processar todas as mensagens recebidas e estabelecer o processo adequado para lidar com a requisição de cada uma delas. Por exemplo, é através dele que mensagens informando a criação de um monitoramento no Veacon Web são direcionadas como requisições de status P, bem como mensagens informando a remoção ou fim de um monitoramento são recebidas e também adequadamente encaminhadas para atualização dos objetos do sistema.

O módulo Core possui duas importantes etapas. A primeira é tratar os eventos do PubSub, que possui uma *thread* exclusiva para seu funcionamento, e assim que recebe alguma mensagem a armazena em uma lista. Posteriormente essa lista é verificada pelo Core e todas as mensagens recebidas são processadas. Após isso, a segunda etapa executa o método que dispara todo o processo de atualização e validação do Watchpost Manager.

3.7 FUNCIONAMENTO DO SISTEMA

O sistema Veacon funciona através da ação dos sistemas Veacon Web e Veacon Rasp, onde o primeiro armazena dados pertinentes sobre os dispositivos que serão utilizados para que o Veacon Rasp realize a vigilância dos beacons e identifique as situações de furto e roubo. Ao gerenciar os monitoramentos, o Veacon Web utiliza os recursos armazenados para enviar solicitações ao Veacon Rasp, que por sua vez processa as requisições e responde ao Veacon Web da forma apropriada. A qualquer sinal de uma anomalia na leitura dos RSSIs dos beacons que possa indicar uma movimentação dentro do período de monitoria, o Veacon Rasp aciona o Veacon Web para que este possa executar seus proce-

dimentos e avisar aos responsáveis pela segurança do *Smart Campus* que uma situação de irregularidade pode estar ocorrendo.

3.7.1 Cadastro de monitoramento no sistema Veacon Web

A Figura 17 demonstra a forma como um monitoramento é cadastrado no sistema. No primeiro campo o usuário informa qual é o *gateway* (Raspberry) de monitoramento. A arquitetura descentralizada permitiria que fossem operados *n gateways*, porém, para esta validação, está sendo utilizado apenas um. Essa mesma informação também é usada como o nome do canal de comunicação entre o Veacon Web e Veacon Rasp através do PubSub implementado usando o serviço PubNub. O segundo campo, *Vehicle*, faz referência ao veículo que está sendo monitorado e seus dados podem ser utilizados para informar aos responsáveis pela segurança do *Smart Campus* o veículo de interesse caso haja um alerta no sistema. O terceiro campo é referente ao BLE beacon que está sendo usado para monitorar o veículo. Este é um campo essencial que deve ser informado para que o sistema encaminhe o dado ao Veacon Rasp, que por sua vez a utiliza para identificar qual é o UID que deve ser gerenciado pelo Beacon Manager do sistema Veacon Rasp. O quarto campo é a informação de qual usuário é responsável pelo monitoramento. São os dados de contato deste indivíduo que serão utilizados caso aconteça alguma situação de alerta sobre o monitoramento. Os campos "*End date*, *End hour*, RSSI distante, RSSI próximo" e "Status" são gerenciados pelo próprio sistema. Destaque aos campos de RSSI que são informados após a primeira interação do Veacon Rasp com o beacon. O campo status é definido como "P" (processando) na ação de cadastro do monitoramento, para que a solicitação seja processada pelo Raspberry. Após a interação, o campo é alterado para "A" (ativo) e quando o monitoramento é removido, essa informação é alterada para "I" (inativo), e também é enviada para o Raspberry para que o monitoramento do BLE beacon referente seja encerrado.

Após o cadastro, uma mensagem é enviada ao canal do *gateway*, que a intercepta e processa a ação com base na requisição de status, como demonstrado no fluxograma de funcionamento do Veacon Rasp na Figura 15.

3.7.2 Cadastro de monitoramento no Veacon Rasp

Ao cadastrar um monitoramento no Veacon Web, o sistema publica uma mensagem no canal via PubSub informando ao *gateway* um monitoramento de status "P", que ainda não foi processado para iniciar a vigilância. Ao obter essa mensagem, o *gateway* inclui o UID do BLE beacon à lista de verificação de leitura de beacons do ambiente e um novo objeto watchpost sem leituras à lista de watchposts com o status "P".

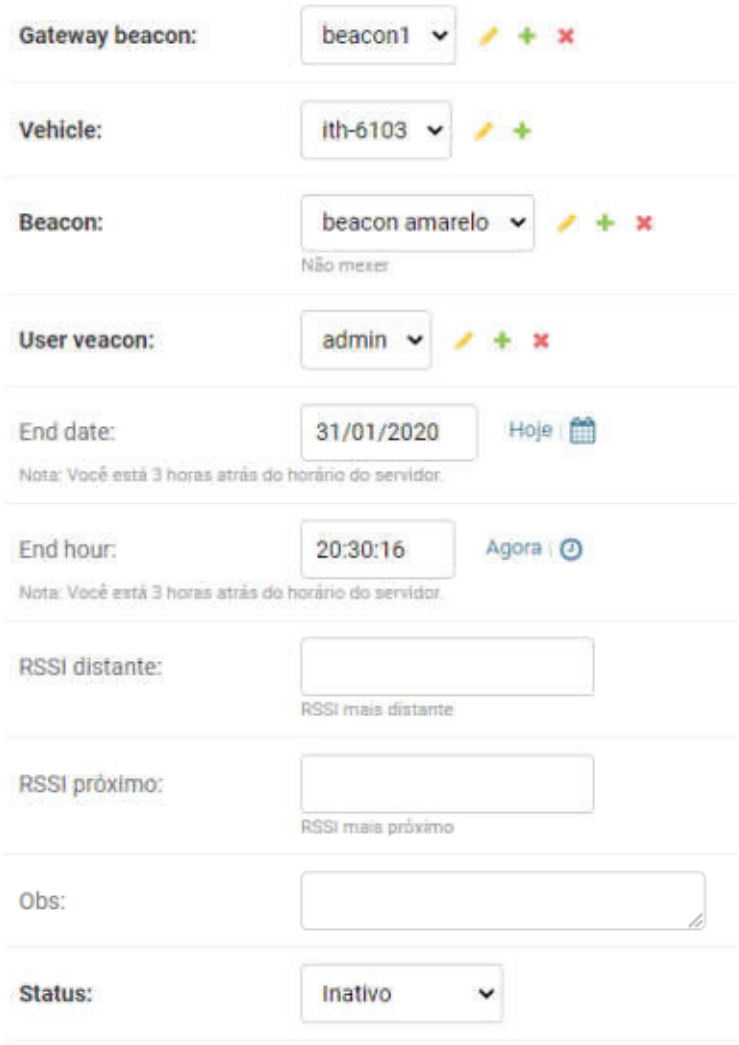

Figura 17. Cadastro de monitoramento no Veacon Web
Fonte: autor

Antes de encerrar a configuração do objeto na lista de monitoramento, o sistema realiza a leitura dos beacons por um tempo determinado via parâmetro (discutido no capítulo de resultados e discussões), e no processo de atualização de monitoramentos (*refresh_watchpost*), o objeto de status “P” recebe os dados de RSSI próximo e distante, a mediana dos dados lidos e status “A”. Após isso o Veacon Rasp informa o Veacon Web através de uma requisição à sua API, informando um *patch* para o monitoramento e encaminhando os dados de status e RSSIs (Figura 18).

Após isso, o sistema continua a realizar o *looping* de leitura e verificação de movimento dos BLE beacons monitorados enquanto ouve novas mensagens no canal PubSub.

3.7.3 Método de identificação e formação de alertas

Os alertas criados pelo deslocamento dos BLE beacons são identificados primeiro pelo Veacon Rasp, que age na borda da aplicação e analisa o movimento dos beacons

```
... Enviando patch id 4 para servidor: {'rssi_near': -90, 'rssi_far': -98, 'status': 'A'}


        ... Status enviado
        ...atualizado
        eddy_namespaceedd1ebeac04e5defa017
        status A
        ...Sem alerta de monitoramento para 'edd1ebeac04e5defa017'
Encerrando sequencia#57
sleep 10's
Iniciando sequencia #58
        ... Iniciando processo de monitoramento
reading ble for 15.0 s
RSSI List: {'edd1ebeac04e5defa017': [-97, -96, -96, -94, -95, -95, -94, -95, -94, -94, -93
...atuallizando watchpost: edd1ebeac04e5defa017
        ...atualizado
        eddy_namespaceedd1ebeac04e5defa017
        status A
        ...Sem alerta de monitoramento para 'edd1ebeac04e5defa017'
Encerrando sequencia#58
sleep 10's
Iniciando sequencia #59
        ... Iniciando processo de monitoramento
reading ble for 15.0 s
RSSI List: {'edd1ebeac04e5defa017': [-93, -93, -92, -91, -91, -95, -96, -95, -92, -94, -98
-93, -90, -98, -97, -95]}
...atualizando watchpost: edd1ebeac04e5defa017
        ...atualizado
        eddy_namespaceedd1ebeac04e5defa017
        status A
        ...Sem alerta de monitoramento para 'edd1ebeac04e5defa017'
Encerrando sequencia#59
```

Figura 18. Configuração do monitoramento
Fonte: autor

monitorados. Quando há movimentação brusca de um beacon cujo monitoramento não tenha sido interrompido por uma requisição vinda do Veacon Web, o Veacon Rasp interpreta como uma possível situação de risco, tendo em vista a diferença entre as leituras de RSSI originais e atuais, o que leva a um disparo de alerta para o Veacon Web.

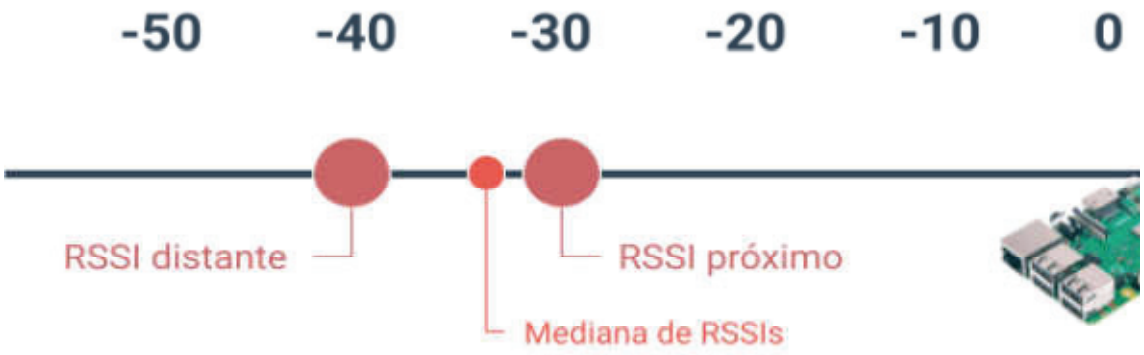

Figura 19. RSSI próximo, RSSI distante e mediana de RSSIs em situação normal
Fonte: autor

A Figura 19 reflete visualmente a forma com que a regra para identificação de um deslocamento foi desenvolvida pelo autor desta pesquisa. Durante a configuração de um monitoramento de status “P” feito pelo Veacon Rasp, é selecionado o dado de RSSI mais próximo e o mais distante, que serão usados como base para identificar o deslocamento. Os RSSIs são medidos através de dBm, ou seja, números negativos, então considere o RSSI

mais distante como -40 e o mais próximo como -30, descritos na figura. Enquanto houver monitoramentos ativos no Veacon Rasp, o sistema irá escanear os BLE beacons próximos a fim de identificar os sinais de RSSI recebidos. Encerrada essa etapa, a atualização dos monitoramentos verifica se o valor de mediana dos RSSIs recebidos de cada monitoramento está entre o valor de RSSI mais distante e mais próximo. Caso o valor seja, por exemplo, -34 (maior do que o RSSI mais distante e menor do que o mais próximo), isso indica que o BLE beacon está dentro da área de leitura inicial.

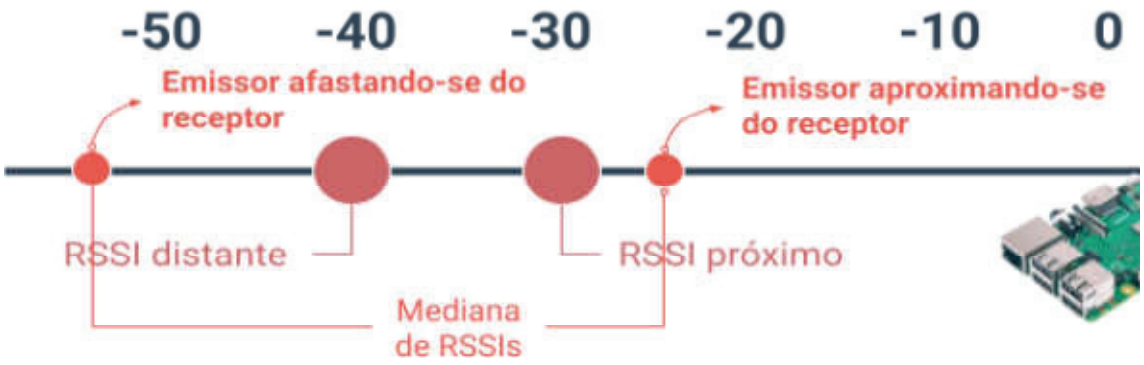

Figura 20. Identificação de anomalia
Fonte: autor

Por outro lado, caso este valor seja, por exemplo, como demonstrado na Figura 20, -55 (menor do que o primeiro RSSI reconhecido como mais distante) ou -25 (superior ao RSSI reconhecido como o mais próximo), isso caracterizará um afastamento ou uma aproximação, respectivamente, e o alerta é gerado.

A criação desta regra de negócio para indicar deslocamento do beacon vem de encontro aos tipos de aplicações identificados na Revisão Sistemática da Literatura: triangulação e aproximação. A triangulação busca estimar com precisão o local em que um determinado dispositivo encontra-se. Por exemplo: com o uso de três beacons em uma loja, o aplicativo de smartphone de um cliente pode ler os RSSIs de cada beacon e utilizar um serviço que através destas leituras identifica a posição do usuário. Assim, uma notificação, por exemplo, sobre a promoção de um produto próximo, pode ser disparada. Neste caso, a precisão sobre a estimativa deve ser acurada, e para isso, algoritmos que tratem a variação de RSSIs de BLE beacons devem ser utilizados, como o *K-NN* ou árvore de decisão. Já os métodos de aproximação não possuem essa mesma necessidade de acurácia. O objetivo é marcar um determinado local/objeto e emitir sinais dentro da sua região. Ao se aproximar dessa marcação, o smartphone de um usuário irá identificar o sinal e acionar algum evento descrito na programação. Para esse fim, não é necessário acurácia dos RSSIs, uma vez que identificar a entrada e saída da área marcada já é o suficiente. A metodologia proposta no Veacon é uma mescla de ambos os tipos de mecanismos.

A estratégia utilizada não tenta triangular a posição do BLE beacon e tampouco mensurar a distância entre os dispositivos emissor e o receptor. A aplicação busca identifi-

car o beacon e vigiar os sinais enviados por ele. Esta característica é predominantemente de aproximação, porém, para que a regra de negócio seja bem sucedida, ao realizar o monitoramento, a alteração da posição do beacon ocasionada pela movimentação do veículo deve ocorrer, o que pode ser percebido pela mudança brusca de RSSI apesar de o beacon permanecer atuante na região. Identificar essa ação antes do término da vigilância é o que pode indicar uma situação de furto ou roubo. Portanto, apesar das particularidades estarem fortemente atreladas à aproximação, o tratamento dos dados de RSSI, característico de aplicações de triangulação, deve estar presente.

Os programas que fazem uso de triangulação têm sua tomada de decisão influenciada diretamente pelo conjunto de RSSIs de três BLE beacons ou mais. No caso do sistema Veacon, essa decisão é tomada observando os dados gerados por cada beacon individualmente. Isso é possível através da leitura de uma determinada quantidade de dados de RSSI providos por cada beacon monitorado, aumentando sua representatividade, o que reduz o risco de falsos positivos causados pelos dados oscilantes. Outro fator causal do uso de algoritmos de tratamento de RSSIs é a busca por mensurar a posição e/ou distância do dispositivo (do usuário). Pelo Veacon não se tratar de uma aplicação que carece de tais características, foi determinado que não havia necessidade de aplicar algoritmos de tratamento, e foi implementada a estratégia utilizando os dados da primeira leitura que posteriormente serve como base para estabelecer a comparação entre os monitoramentos.

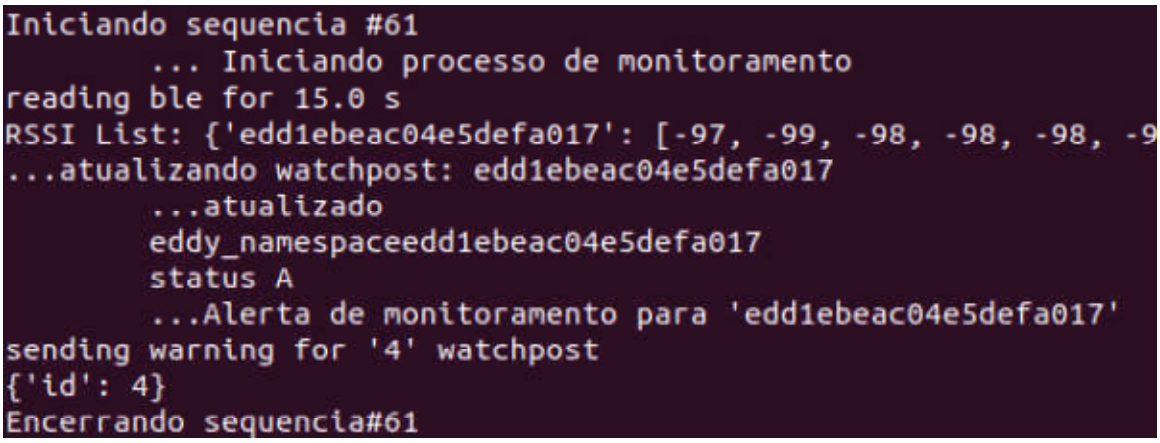

Figura 21. Identificação de deslocamento e envio de alerta
Fonte: autor

A Figura 21 exibe parte deste processo de identificação de deslocamento e envio de alerta, onde na primeira etapa reserva 15 segundos para ler sinais BLE de beacons cadastrados no Beacon Manager identificando e armazenando os dados de RSSI recebidos. Posteriormente inicia o processo de atualização dos monitoramentos e identifica o deslocamento do BLE beacon monitorado, enviando um alerta para o Veacon Rasp. Este alerta é recebido, armazenado no banco de dados e gera um alerta para que um responsável possa visualizar algumas das informações referentes à situação.

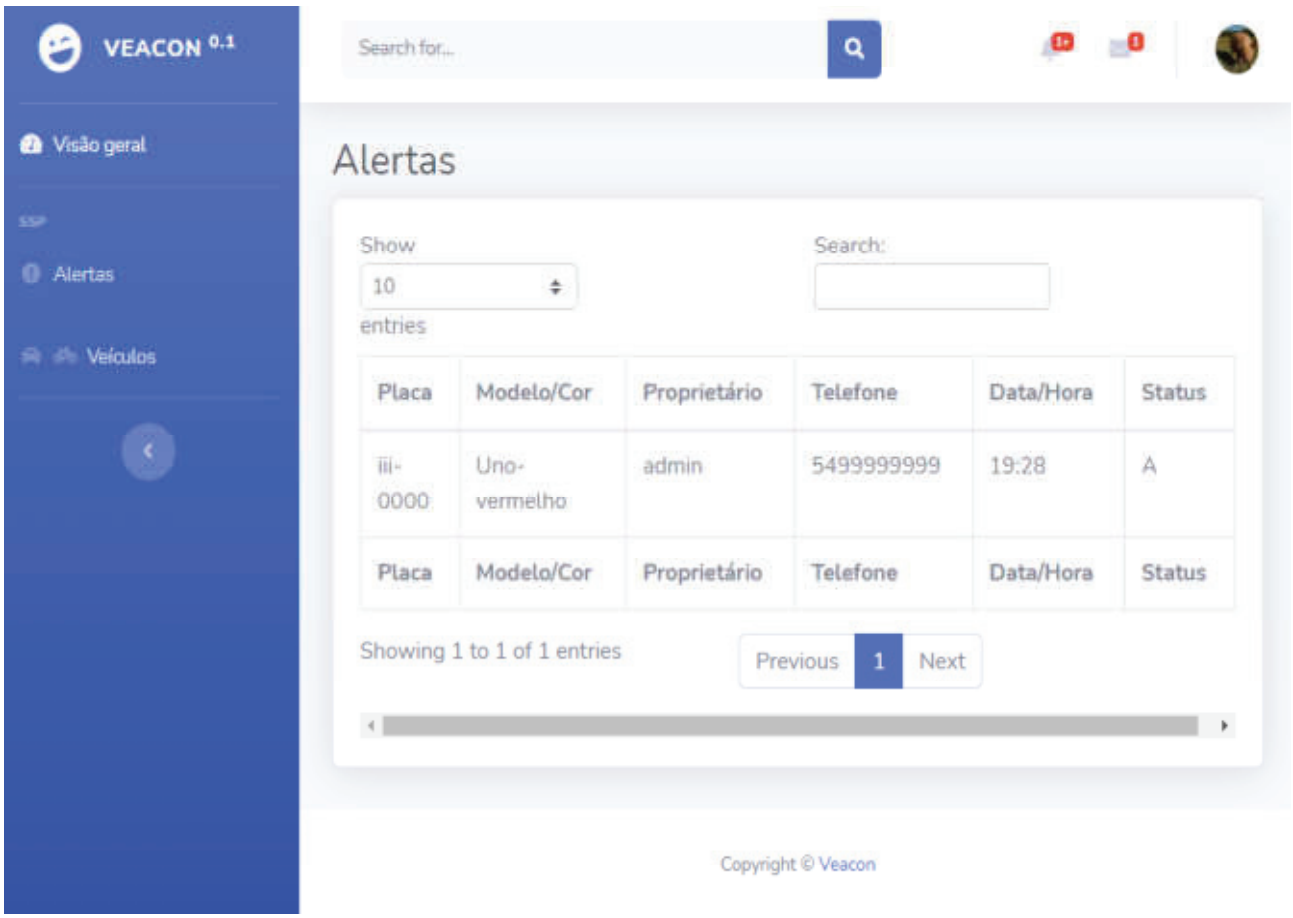

Figura 22. Tela de alertas
Fonte: autor

A Figura 22 exibe a tela padrão que pode ser acessada por um agente de segurança responsável pela vigilância e proteção do *Smart Campus*. A tela de alertas é atualizada e exibe as informações referentes às características do veículo e dados do proprietário que podem ser usados para atuar em uma abordagem de prontidão buscando minimizar as perdas da propriedade do usuário.

3.8 TESTES

Para alcançar os objetivos esperados foram aplicados dois tipos de testes com diferentes abordagens:

1. Testes com os BLE Beacons
2. Teste da aplicação

Os testes foram conduzidos através da abordagem metodológica experimental. O método do teste #1 consiste em expor dentro de um ambiente controlado o objeto emissor (BLE Beacon) e receptor (Raspberry PI 3 Model B) utilizando um software criado para

observar o comportamento do RSSI recebido. Já o teste #2, dirigido utilizando a mesma abordagem metodológica, utiliza o programa Veacon e testa a possibilidade de identificação do furto de um veículo também em ambiente controlado.

Os Testes com BLE beacons possuem foco em entender o comportamento dos dispositivos em diferentes situações, principalmente em virtude dos problemas de oscilação identificados durante o levantamento de trabalhos similares e desenvolvimento do programa Veacon. O entendimento do comportamento dos sinais emitidos pelos dispositivos beacon possui grande importância para a implementação da aplicação e para a parametrização do teste experimental utilizado para testar a hipótese do projeto, uma vez que são a base para a identificação da movimentação atípica dos veículos. Estes testes estão subdivididos na Tabela 7.

Enumerados na Tabela 7, os testes #1 e #2 são referentes aos beacons usados como hardware para monitoramento e *middleware* de envio de alertas do sistema. O teste #1 possui o objetivo de entender o funcionamento e fluxo do sinal emitido do beacon para fontes receptoras. A documentação o *Location Beacon* disponibilizada pelo fabricante exposta na Figura 23 menciona que o sinal pode alcançar até 200 metros de distância, porém, essa métrica precisa ser validada para que o teste #2 possa ser realizado em diferentes cenários a fim de entender o comportamento do sistema e mensurar o uso de BLE beacons para a identificação de furto e roubo de veículos. O objetivo deste teste é realizar um levantamento das distâncias em que o sinal permanece visível, forte e sem grandes oscilações para fontes receptoras que serão usadas para gerenciar os dispositivos monitorados.

O a segunda parte do teste #1 possui um objetivo similar à primeira: validar a qualidade do sinal emitido pelo BLE Beacon. Porém, com a diferença que este teste não tem o objetivo de medir a potência do sinal a diferentes distâncias. Nesta etapa, o objetivo específico visa entender o comportamento da entrega do sinal com a adição de obstáculos inseridos propositalmente entre a fonte e o receptor. Em um cenário de aplicação real, por exemplo na rua de um *Smart Campus* ou de uma cidade, haverão objetos em trânsito, tais como: veículos leves e pesados, pessoas, animais de pequeno e grande porte, árvores, paredes, monumentos, etc. Em virtude disso, este teste busca validar a viabilidade do uso de beacons em cenário de produção. As distâncias, tempo de leitura e obstáculos de ambas as partes do teste 1 estão elencados na Figura 24.

O teste #2 deve ser realizado no ambiente integrado entre todas as camadas do sistema Veacon. Nesta etapa o teste #1 já foi aplicado com a finalidade de identificar padrões de distâncias e obstrução de objetos, portanto, variáveis que usam estas métricas foram pré-definidas, tais como: local de teste, distância entre o beacon e o receptor de monitoramento e obstáculos que serão utilizados para simular o ambiente real de produção. Seu objetivo foca no software desenvolvido para este trabalho buscando manter a comunicação entre a camada de hardware e aplicação através do *middleware*, como foi demonstrado no fluxograma da Figura 9. Para que isso seja possível, o sistema deve receber

Tabela 7. Divisão de testes

#	Descrição	Objetivo	Método
1.1	Teste de recepção dos sinais emitidos pelos BLE beacons.	Identificar a qualidade e oscilação dos sinais a cada 30 metros. Iniciando em 30 metros e chegando a 180.	Manter um receptor realizando leituras de um BLE beacon e armazenar essas leituras para análise. O beacon será movido a cada 30 metros e um programa de monitoramento de BLE Beacons será executado no receptor para identificar a qualidade do sinal àquela distância e armazenando os dados em arquivos csv para posterior análise.
1.2	Teste de recepção dos sinais emitidos pelos BLE beacons com obstáculos.	Identificar a qualidade e oscilação dos sinais a cada 30 metros. Iniciando em 30 metros e chegando a 180 com obstáculos entre os pontos de observação. O foco é identificar como os obstáculos interferem na força do sinal recebido.	Este teste assemelha-se ao #1, porém, entre o receptor e o BLE beacon será inserido um obstáculo para dificultar a propagação do sinal do emissor e compreender o comportamento do sinal com obstruções entre emissor e receptor.
2.1	Teste de validação do sistema Veacon	Analisar o processo desempenhado por todos os sistemas distribuídos que compõem o sistema Veacon que busca identificar as situações de furto e roubo de veículos.	O receptor será inserido em um ponto fixo e um BLE beacon será fixado em um veículo. Em seguida um monitoramento será cadastrado para gerir a situação do beacon do veículo. Após alguns minutos, o veículo será movimentado para identificar a capacidade de gerar alertas pelo sistema Veacon.

e sincronizar os sinais emitidos pelos beacons. A distância será estipulada a um valor em que os sinais emitidos não possuam grandes oscilações e sejam recebidos sem perdas de pacote. Quando o sistema identificar a entrada de dados, deve aplicar as regras de negócio do Veacon Rasp para que os dispositivos sejam corretamente monitorados. Nesta etapa, uma simulação de furto e roubo de um veículo será realizada.

O veículo será movimentado para que o sistema possa identificar uma oscilação nos dados recebidos, assim estipulando quanto o BLE beacon deve se mover para gerar uma alteração significativa que possa ser percebida pelo sistema. Esse passo visa identifi-

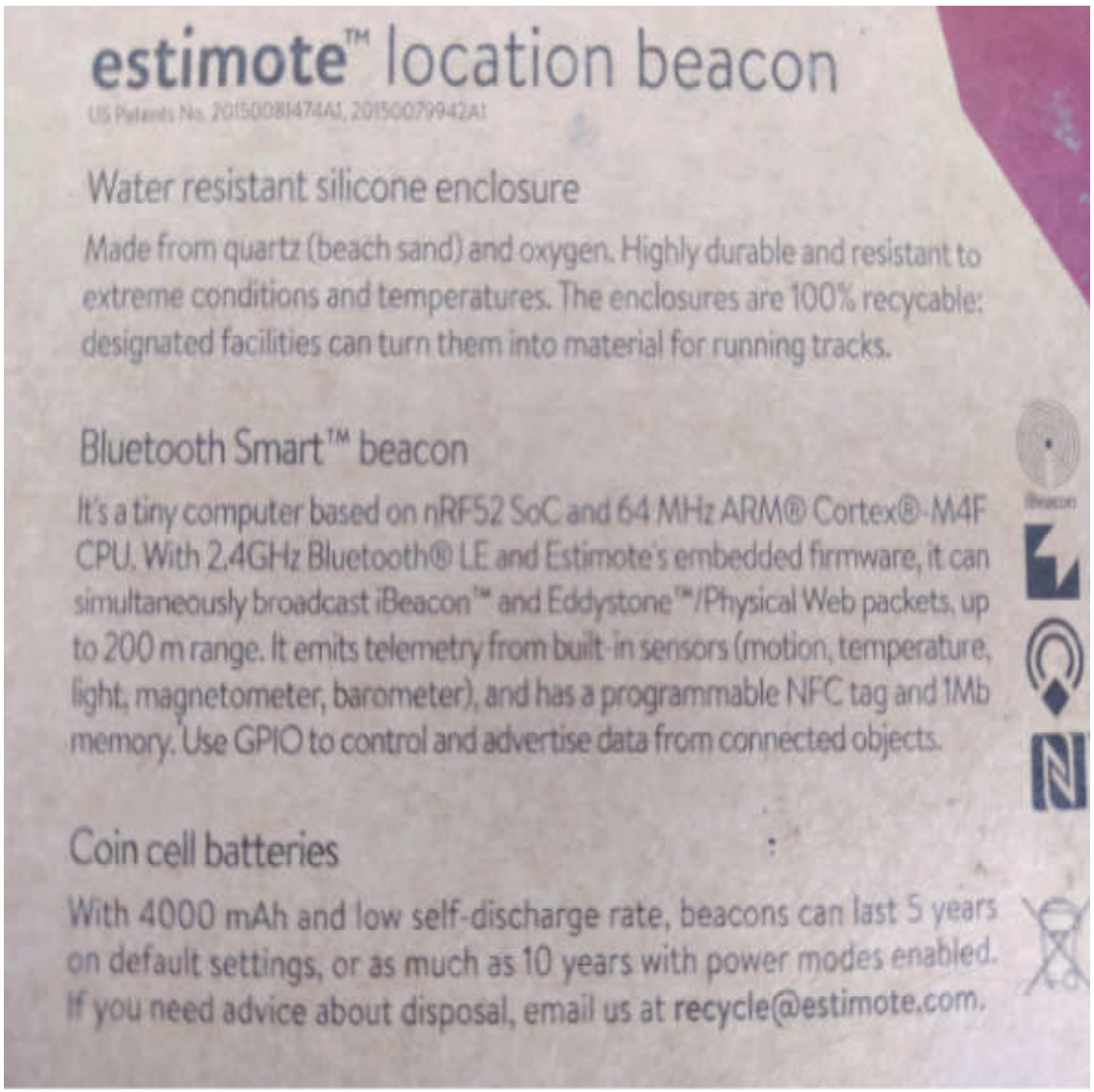

Figura 23. Distância alcançada pelo kit de BLE Beacons utilizados nos testes
Fonte: Estimote

car a aplicabilidade das regras de negócio a fim de atestar o funcionamento do método sem demandar por uma situação de perigo real.

Ao final, serão coletados dados referentes ao objetivo de analisar o uso de BLE beacons para identificação de furto e roubo de veículos:

1. Após quantos metros de movimentação o sistema identificou e gerou o alerta
2. Presença de falsos positivos

Através do levantamento destas informações, os dados gerados irão indicar a viabilidade do uso da tecnologia BLE beacon para possibilitar aos agentes responsáveis pela vigilância do *smart campus* a identificação de movimentações atípicas e oportunizar estratégias antecipadas de abordagem a veículos suspeitos.

#	Distância(m)	Tempo (s)	Obstáculo
1	30	30	Ar livre
2	30	60	Ar livre
3	60	30	Ar livre
4	60	60	Ar livre
5	90	30	Ar livre
6	90	60	Ar livre
7	120	30	Ar livre
8	120	60	Ar livre
9	150	30	Ar livre
10	150	60	Ar livre
11	180	30	Ar livre
12	180	60	Ar livre
13	30	30	Veículo
14	30	60	Veículo
15	60	30	Veículo
16	60	60	Veículo
17	90	30	Veículo
18	90	60	Veículo
19	120	30	Veículo
20	120	60	Veículo
21	150	30	Veículo
22	150	60	Veículo
23	180	30	Veículo
24	180	60	Veículo

Figura 24. Checklist do teste #1
Fonte: Autor

4. RESULTADOS E DISCUSSÕES

Este capítulo aborda os resultados dos testes experimentais realizados para avaliar a aplicabilidade do uso de tecnologias IoT utilizando BLE beacons como ferramenta central para identificar o furto e roubo de veículos em um ambiente *smart campus*.

O teste #1, aplicado para identificar as características dos dados de RSSI recebidos em distâncias e cenários diferentes, demonstrou o que já havia sido corroborado por outros pesquisadores: os sinais podem variar de acordo com as circunstâncias. O teste #2, destinado a identificar a valia dos BLE beacons na identificação de furto e roubo de veículos, apresentou sinais positivos do uso desta tecnologia para os fins da exploração deste trabalho.

4.1 CONFIGURAÇÃO DOS EQUIPAMENTOS

Para a execução de ambos os testes foi utilizado o mesmo equipamento demonstrado até então e a configuração do BLE beacon foi ajustada para adequar alcance e economia de energia (Figura 25).

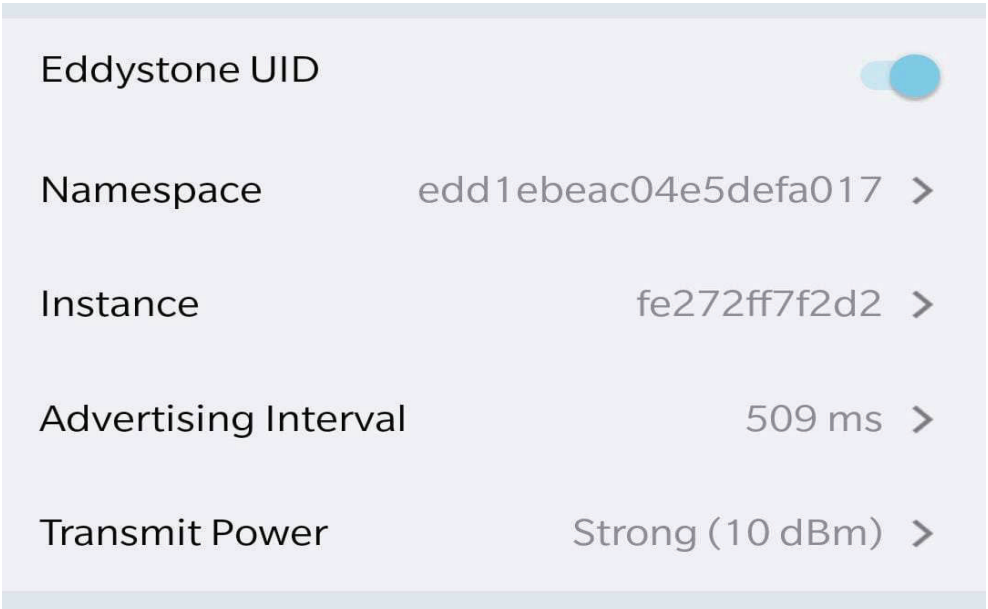

Figura 25. Configuração do BLE beacon Estimote
Fonte: autor

As principais configurações que interferem na economia de energia do dispositivo e que estão presentes em cada protocolo são: intervalo e alcance do sinal. O intervalo, representado na configuração do dispositivo Estimote como "*advertising interval*", refere-se ao tempo de espera entre os sinais enviados. Já o valor de alcance do sinal, correspondido

pelo campo “*transmit power*” da configuração, refere-se à energia que será utilizada para transmitir o sinal. Quanto menor o intervalo maior é o consumo de bateria por parte do dispositivo. Em relação ao alcance, quanto maior a área de abrangência, mais energia deve ser utilizada para cobri-la.

Devido ao baixo consumo de energia utilizada pelos BLE, mesmo estabelecendo configurações que demandam alto consumo de bateria, o BLE beacon pode manter-se funcionando por um longo período de tempo, podendo ultrapassar 2 anos de vida útil. Porém, uma das premissas do IoT, e também do uso de BLE beacons, é a possibilidade de utilizar diversos dispositivos para alcançar os melhores resultados de um serviço. Em virtude disso, manter as configurações demandando alto consumo de energia pode gerar problemas logísticos a longo prazo no momento em que as baterias começarem a falhar, o que demandaria a troca de pilhas de *n* BLE beacons utilizados pelos usuários do serviço.

Para gerar um aproveitamento eficiente das baterias, foi aplicada uma configuração com o objetivo de abranger o alcance de área máximo do dispositivo, balanceando um intervalo moderado com cerca de um sinal a cada 500 milissegundos. Isso faz com que a área de cobertura possa ser abrangente, reduzindo a quantidade de *gateways* a serem utilizados para cobrir uma determinada área; e possibilita o processamento adequado dos RSSIs que necessitam ser lidos para a execução das regras de negócio, o que é abordado com maior profundidade durante os subcapítulos dos testes.

4.2 CENÁRIO E COLETA DE DADOS DO TESTE #1

Para a realização do teste #1, em que foi analisada a característica de entrega e recepção dos sinais emitidos pelos BLE beacons disponíveis para este trabalho, foi desenvolvido um programa adequado à grade de testes apresentada na Figura 24. Ao iniciar o programa *main* [64], o usuário deve inserir o número do teste (cada item da coluna se refere a este número) e após isso o tempo de leitura (referente à coluna tempo). Com estas informações, o programa irá armazenar os RSSIs lidos pelo Raspberry PI em um arquivo csv separando cada valor com uma vírgula. Posteriormente estes arquivos são utilizados pelo programa *main_visualization* para gerar os gráficos disponíveis nas Figura 26 e 27.

As Figuras 26 e 27 apresentam os resultados do teste #1, sendo o eito y o valor de RSSI calculado, possuindo, acima do gráfico, o número referente ao teste da Figura 24. A Figura 26 apresenta os resultados dos testes através gráficos com dados de RSSI ordenados do menor para o maior. Nesta etapa é possível observar um padrão já descrito por outros pesquisadores, onde é notório que a oscilação do RSSI ocorrer independente da distância de leitura. Ainda, após o teste de #9 (leitura a 150m por 30s), as leituras tornam-se debilitadas com a ausência de dados nos testes 10, 11, 12 e 17 adiante com exceção do teste 22 que obteve um único dado à distância de 150 m com obstáculos [64].

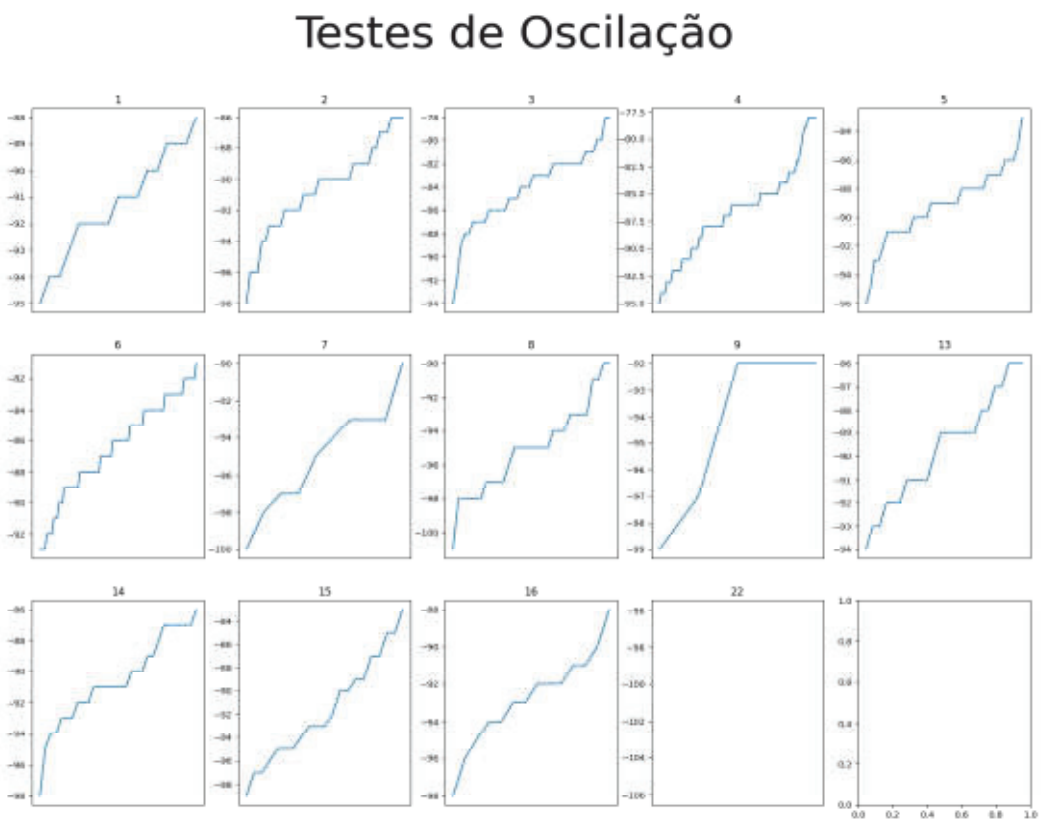

Figura 26. Resultado de obtenção de dados do teste #1 (ordenados)
Fonte: autor

A Figura 27 apresenta os mesmos dados, porém, sem ordenação, onde o eixo x é o tempo decorrido desde o início do escaneamento, demonstrando a intensidade de cada sinal à medida em que chegou até o dispositivo de leitura.

4.2.1 Resultados obtidos durante o teste #1

É possível observar que quanto maior a distância, mais fraco é a cobertura do sinal do BLE beacon, fazendo com que o Raspberry obtenha poucas leituras e com grandes oscilações devido à perda de pacotes.

Ainda, os números dos testes da Figura 24 ausentes nas Figuras 26 e 27 mostram que estes testes não obtiveram dados de RSSI durante o período de leitura. O teste 10 foi o primeiro sem obstáculos que não obteve resultados a 150 m, já o teste anterior, que foi conduzido à mesma distância e por menos tempo, captou dados de RSSI, entretanto, recebeu apenas 4 sinais. O restante dos testes sem obstáculos não receberam nenhum sinal. Já nos testes com obstáculos, o primeiro a não receber sinais foi o 17, que a 90 m já não captava nenhum sinal. Nenhum teste subsequente recebeu sinais com exceção do 22 que foi realizado a 150 m por 1 minuto, porém, recebeu apenas 1 dado do BLE beacon.

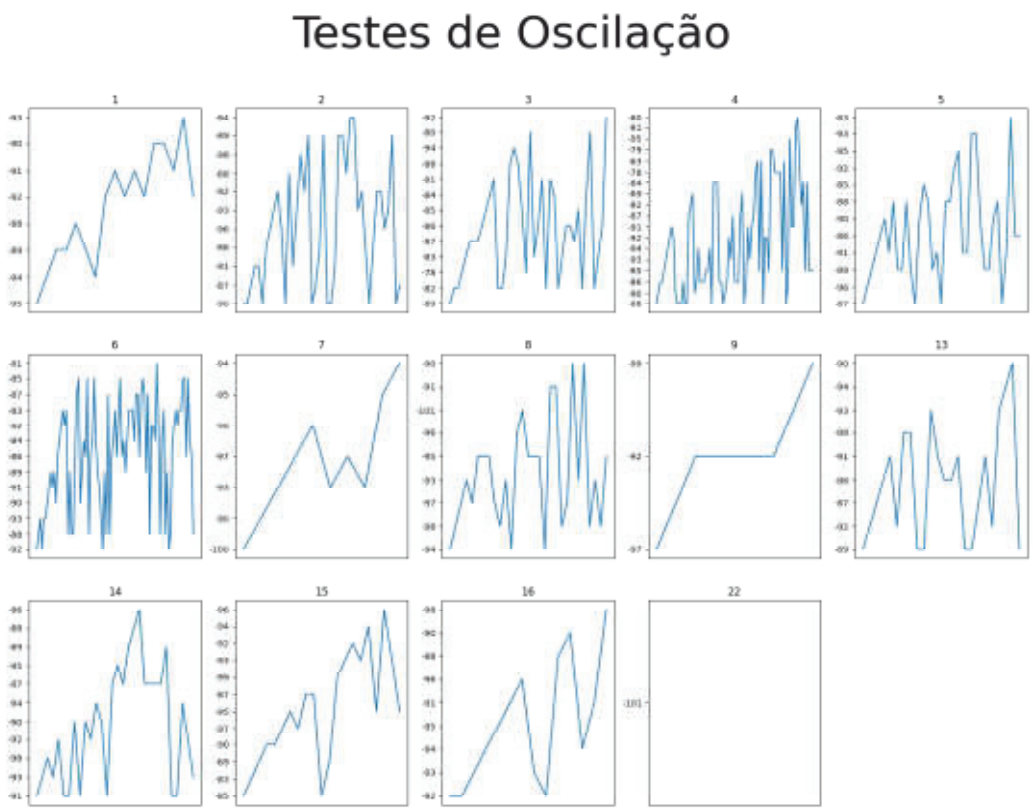

Figura 27. Resultado de obtenção de dados do teste #1 (desordenados)
Fonte: autor

Por meio destes resultados é possível verificar que a distância e obstáculos são fatores que influenciam diretamente a perda de pacotes. A distâncias superiores a 90 metros a recepção dos pacotes não têm a mesma qualidade como as mais próximas, e em casos em que há recepção os dados têm um baixo número de pacotes e uma grande oscilação.

Os sinais emitidos pelos dispositivos BLE sofrem constantes variações e os dados podem criar distorções no resultado esperado pelo programa que o consome. Este comportamento foi observado tanto durante o processo de revisão sistemática da literatura através da descrição de outros pesquisadores; quanto nas primeiras interações para entendimento do uso de BLE beacons desta pesquisa.

Para viabilizar a aplicação do teste experimental, estabeleceu-se que a volatilidade do sinal poderia não ser um problema caso houvesse dados suficientes que pudessem ser processados pelo sistema. Essa observação resultou no método de configuração que o Veacon Rasp utiliza na primeira interação com o monitoramento, estabelecendo os números de RSSI (próximo e distante) que são comparados com a mediana de cada pacote com leituras feitas durante as iterações. Para tal, é necessário um balanceamento entre o intervalo de sinais do BLE beacon e a quantidade de sinais recebidos pelo Raspberry. Se

a quantidade de sinais recebidos for baixa, pode haver uma representatividade de RSSIs distorcidos com potencial de gerar um alerta falso positivo.

A Figura 28 é um gráfico elaborado com a leitura de um BLE beacon a uma distância apropriada e sem obstáculos. Ela destaca de maneira nítida os dados que possuem o potencial de gerar falsos positivos quando o sistema deparar-se com uma situação de caso péssimo. Tendo em vista que o método do Veacon compara a mediana de RSSIs provindas da leitura com os valores de RSSI mais próximo e mais distante identificados na configuração inicial, caso a representatividade de dados seja abaixo da ideal, os valores destacados poderiam influenciar o resultado da mediana. Alternativas à possíveis problemas de oscilação são propostos no capítulo de trabalhos futuros.

Figura 28. Demonstração dos dados que podem causar falsos positivos
Fonte: autor

Por outro lado, havendo uma quantidade significativa de dados de RSSI, a representatividade destes dados fazem com que a chance de gerar falsos positivos seja baixa, uma vez que os sinais dentro do alcance com boa qualidade de recepção possui poucos dados oscilantes comparados à distâncias limites do dispositivo BLE, como observado após 120 metros nas figuras 26 e 27. Em testes preliminares ao teste #2, para definir a quantidade de sinais, observou-se que aproximadamente 30 dados de RSSIs já são mais que capazes de tornar a aplicação resistente à falsos positivos, não havendo nenhum caso registrado após essa calibragem. Para que este cenário favorável seja alcançado, duas variáveis devem ser controladas: o intervalo entre a emissão de sinais do BLE beacon e o tempo de leitura do Veacon Rasp.

Quanto menor a frequência de emissão do BLE beacon, mais sinais são emitidos e mais dados chegam até o Raspberry para alimentar a regra de negócio da aplicação. Por outro lado, essa abordagem reduz a vida útil da bateria do beacon. Outra alternativa é fazer com que o Veacon Rasp aguarde mais tempo para realizar a leitura dos beacons. Porém, esse procedimento faz com que a aplicação demore mais para emitir um sinal de alerta ao sistema caso um veículo esteja em situação de risco.

Ambos os problemas ocorrem por limitações na tecnologia BLE do beacon e implementação da aplicação Veacon Rasp. Soluções adequadas para estes problemas são propostas no capítulo referente aos trabalhos futuros, porém, para este teste foi adotada a estratégia de balanceamento uniforme entre o tempo de leitura do Veacon Rasp e intervalo

de emissão de sinais do beacon com objetivo de mitigar os contras de ambas as situações que poderiam influenciar negativamente o processo da realização o teste #2.

4.3 CENÁRIO DE APLICAÇÃO DO TESTE 2

O teste #2 pode ser considerado a etapa de maior importância para esta pesquisa. Após a obtenção dos dados coletados na realização do teste #1, uma visão geral sobre o cenário de aplicação dos BLE beacons pôde ser formulada com o objetivo de avaliar o uso da tecnologia como método central para identificação de furto e roubo de veículos em um cenário *smart campus*.

Os resultados obtidos durante a exploração do uso de BLE beacons, juntamente com o apuramento das características de leitura de RSSIs do teste #1, apontaram a questão de distância entre o dispositivo emissor e receptor como uma das principais variáveis que contribuem com a volatidade do RSSI. Por este valor se tratar da base de regra de negócio para identificar movimentação no beacon, de acordo com o resultado do teste #1, considerou-se distância de aproximadamente 45 metros como a ideal para a realização do teste.

Além disso, para balancear as configurações de economia de energia do BLE beacon e tempo de resposta do Veacon Rasp, foi considerado o intervalo de emissão de aproximadamente 509 milissegundos e 15 segundos de leitura. Além de servir a propósitos de design e sustentabilidade, esta calibragem propicia a execução das regras de negócio do Veacon Rasp para não ocasionar alertas falsos positivos. Em termos de design, um furto ou roubo deve ser imediatamente comunicado para que uma decisão de intervenção seja tomada. Para isso, a sensibilidade do sistema em reconhecer as movimentações dos veículos deve ser alta e por consequência a entrada de informações (RSSI) deve ser igualmente alta para que hajam dados a serem processados a fim de identificar as anomalias. Isso pode ser obtido através de um curto tempo de leitura do Veacon Rasp para que o processo de avaliação das anomalias ocorra com maior frequência e em menor tempo, porém, a configuração do beacon deve seguir uma proporção de envio de sinais condizente. Para isso o beacon necessita de uma configuração onde o intervalo de emissão entre sinais seja curta, dessa forma, porém, a bateria seria rapidamente consumida. Inversamente proporcional, caso o intervalo de emissão de sinais seja muito longo, economizaria mais energia, porém, haveriam poucos envios, o que ocasionaria uma baixa de dados de RSSI e isso influenciaria negativamente a execução da regra de negócio, podendo, inclusive, causar alertas falsos positivos. Como é possível notar, o design da aplicação e a sustentabilidade no uso dos beacons estão diretamente ligados. A configuração foi determinada para balancear a proporção entre estes fatores, como é destacado na Figura 29.

Além da calibragem do BLE beacon e do Veacon Rasp, outro elemento também contribuem para a execução do teste: O local escolhido teve como base 2 fatores:

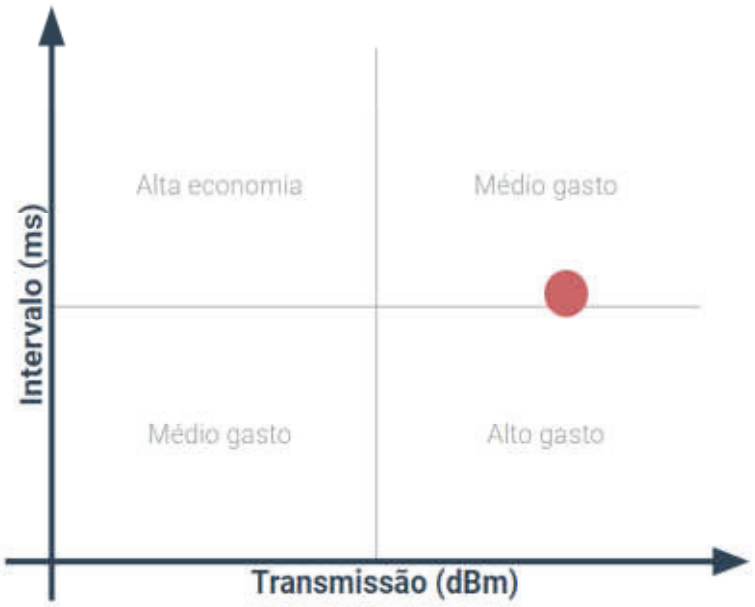

Figura 29. Demonstração dos dados que podem causar falsos positivos
Fonte: autor

1. Livre de obstáculos: para tornar as condições de leitura do Veacon Rasp a ideal para identificar e armazenar corretamente os RSSIs emitidos pelo BLE beacon
2. Conexão com a internet: para que a comunicação entre o Veacon Rasp seja efetuado com o Veacon Web disponibilizado em nuvem

Os testes foram realizados em frente ao bloco B5 da UPF. Para isso, um BLE beacon foi fixado dentro de um veículo (Figura 30) que movimentou-se a fim de tomar distância entre o emissor e o receptor, buscando gerar alertas de movimentação.

Após a fixação do beacon, o veículo é posicionado a 45 metros de distância do Raspberry. Um novo monitoramento é cadastrado no Veacon Web e o processo de comunicação com o Veacon Rasp para monitoramento é iniciado. Este teste é constituído de 4 partes:

1. O monitoramento é observado durante 60 sequências de interação
2. Movimenta-se 2 metros para o beacon ficar mais distante do Raspberry
3. Volta-se à posição inicial e inicia-se um novo monitoramento
4. Movimenta-se 2 metros para o beacon ficar mais próximo do Raspberry

Este processo pode ser observado na Figura 31.

Na primeira parte, 60 sequências de monitoramento são realizadas. Cada sequência possui 15 segundos de leitura, o que totaliza aproximadamente 15 minutos de monitoramento sem a ocorrência de falsos positivos. Após as iterações sem falsos positivos, o

Figura 30. BLE beacon fixado no interior do veículo
Fonte: autor

Figura 31. Descrição visual do teste 2
Fonte: autor

veículo é movido juntamente com o beacon à uma distância aproximada de 2 metros e é aguardado até o fim da interação (~15 segundos) para verificar a ocorrência de alertas enviados. Caso não haja o disparo de um alerta, o processo é realizado novamente até que o alerta seja gerado. O passo seguinte é repetir a mesma tarefa, entretanto, com a aproximação do veículo ao Raspberry para apurar o comportamento do teste enquanto o veículo move-se em direção ao Raspberry. Este processo pode ser observado nas Figuras 32, 33, 34.

Figura 32. Posição inicial do veículo
Fonte: autor

4.3.1 Resultados obtidos no teste 2

Após o início do monitoramento, não houveram falsos positivos durante as primeiras 60 iterações (~15 minutos), demonstrando que o balanceamento entre intervalo de

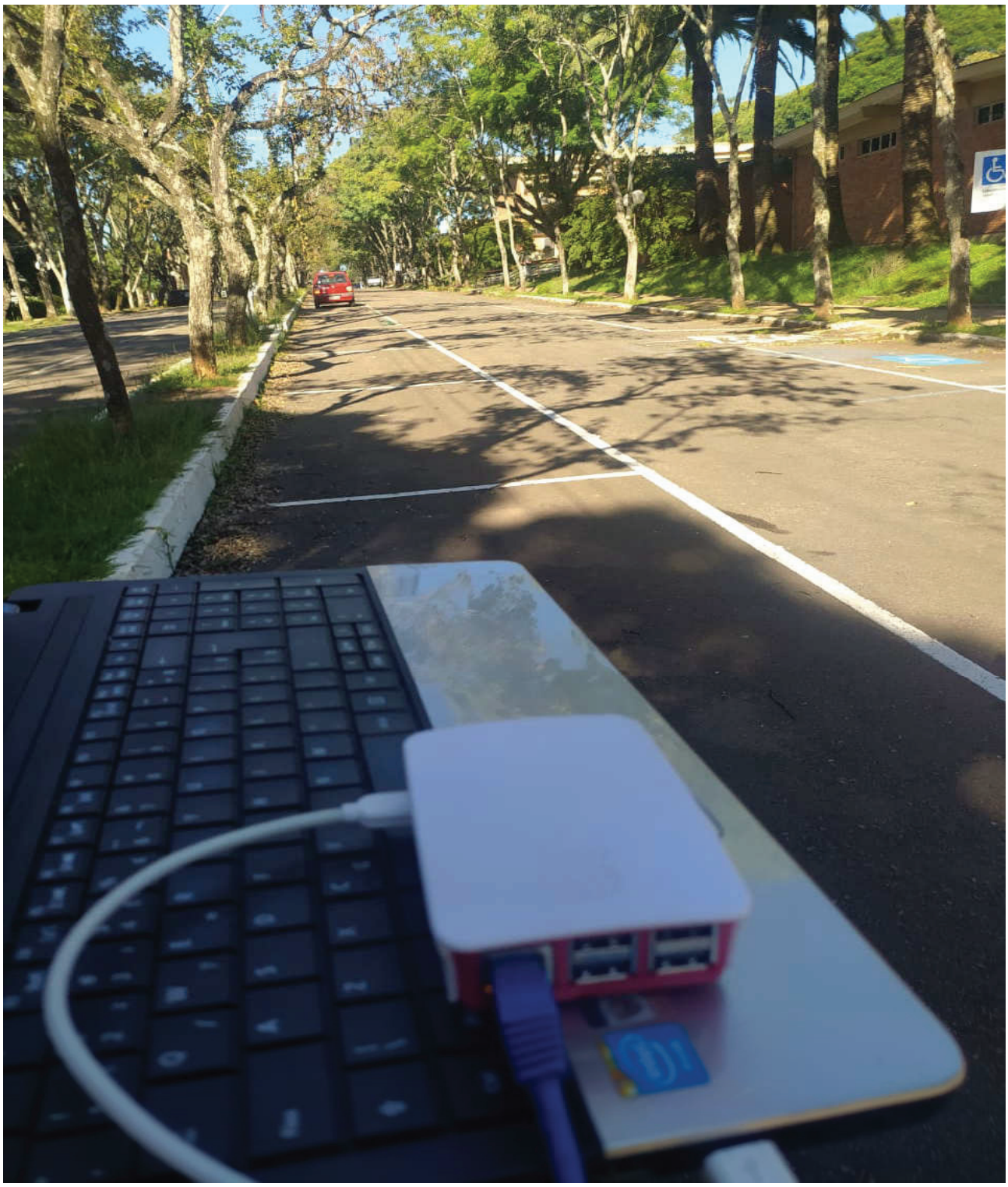

Figura 33. Distanciamento do veículo
Fonte: autor

emissão do beacon (509 ms) e tempo de leitura do Veacon Rasp (~15 s) a fim de obter 30 leituras, foi adequada para a execução da regra de negócio.

Figura 34. Aproximação do veículo
Fonte: autor

Além disso, em testes realizados previamente para verificar a possibilidade da hipótese, o sistema monitorou beacons por longos períodos de tempo. O maior período foi por 24 horas a uma distância de pouco menos de 30 m.

Ademais a ausência de falsos positivos, item que necessita de grande atenção para não gerar situações acidentais aonde o usuário da plataforma fosse levado a acreditar que um veículo esteja sendo furtado ou roubado, o sistema demonstrou acurada sensibilidade às modificações de estado do monitoramento por efeito do deslocamento do veículo tanto na aproximação quanto no afastamento. Na primeira mudança de local, o alerta é gerado baseado na alteração do valor de mediana de RSSIs do beacon monitorado (Figura 21), e o alerta é recebido e processado no sistema web dentro de um tempo previsto à uma aplicação IoT (Figura 22), levando pouco mais de 1 segundo para persistir os dados referentes ao alerta.

5. CONCLUSÃO

Em virtude da segurança prostrada, o Brasil não apenas priva-se das oportunidades de negócios que abatem intensamente o desenvolvimento econômico, mas também perde milhares de vidas que afetam a moral da sociedade como um todo. Além da consternação social provocada pela insegurança da vida, o sentimento de perda patrimonial causado pelo alto índice de furto e roubo de propriedade privada também influencia esse sentimento negativo do povo brasileiro. Nesse aspecto, os crimes envolvendo veículos ocupam uma porção considerável das ilegalidades cometidas nas cidades. Através da informatização, as *smart cities* vem buscando melhorar os processos de tomada de decisão como uma forma de aprimorar os serviços prestados à população, o que inclui suas demandas mais relevantes, como a segurança pública. As propostas de solução muitas vezes são iniciativas procedentes da comunidade acadêmica, que busca oferecer soluções para os problemas latentes da sociedade. No contexto de ambientes inteligentes, os *smart campus* ocupam um papel importante no desenvolvimento de projetos relevantes às cidades, uma vez que possuem similaridades com os reveses enfrentados e seguem um caminho semelhante na busca do aprimoramento.

Tendo em vista os objetivos específicos declarados no capítulo de introdução, este trabalho apresentou o uso de IoT para identificar furto e roubo de veículos em um ambiente *smart campus*. Para atingir esse objetivo, foram utilizados os recursos de aplicação disponibilizados por dispositivos BLE beacons, Raspberry PI e software de gerenciamento em nuvem.

Os objetivos 1 e 2 com foco na descoberta de tecnologias IoT aplicadas à segurança pública em ambientes inteligentes foram abordados de acordo com os resultados da Revisão Sistemática realizada nesta pesquisa. A maioria dos trabalhos envolvendo BLE beacons e IoT não evidenciaram o uso deste paradigma na segurança pública. Além do uso de câmeras de vigilância como dispositivos de hardware conectados a uma camada de comunicação, poucos trabalhos utilizam IoT ou BLE beacons no apoio da segurança patrimonial e da vida. Portanto, esta pesquisa desempenhou o trabalho de estender o diálogo do uso destas tecnologias para outras áreas de atuação dos ambientes inteligentes além das aplicações habituais.

Os objetivos de verificar e implementar o uso de IoT para identificar furto e roubo de veículos através de BLE beacons mostrou-se positivo mesmo frente às indicações de variabilidade de sinais encontrados em bibliografias anteriores. Este foi um obstáculo tratado durante a validação de regras de negócio do teste experimental proposto; e demonstrou que o uso de beacons é promissor em ambiente *smart campus* com a finalidade de evitar situações de furto e roubo. Este resultado foi alcançado por meio de dois testes: o primeiro que identificou o comportamento de oscilação dos dispositivos BLE beacons, e abriu margem

para buscar uma estratégia de regra de negócio viável ao segundo teste, que implementou um software de monitoramento de beacons instalados dentro de veículos. Por sua vez, este software faz uso de uma aplicação em nuvem e outra executada dentro de um dispositivo Raspberry PI para reconhecer a mudança de estado de posição dos beacons através da variação no RSSI de cada beacon vigiado.

Devido à resposta positiva, obtida por uma regra de negócio simples que foi executada dentro de um sistema sem foco em desempenho computacional, o uso de BLE beacons pode ser aproveitado nos ambientes *smart campus* com os objetivos abordados nessa pesquisa.

Para que isso seja possível, é necessário abertura do campus a ser utilizado como laboratório de cidades inteligentes, sendo possível aplicar as tecnologias propostas pelo corpo acadêmico a fim de tratar os problemas enfrentados pela comunidade local. Dessa maneira, necessitando do aval de atores que preenchem os postos de tomada de decisão da instituição.

Outro ponto importante é a participação da organização responsável pela segurança e vigilância da instituição, esta que possui a qualificação e experiência para agir em situações que podem envolver segurança patrimonial e da vida.

Por fim, o uso da tecnologia BLE beacon mostra-se próspero, não apenas no campo de ambientes *smart*, mas também no campo de produtos e serviços como um todo, uma vez que grandes *players* do mercado vêm contribuindo com a comunidade no desenvolvimento de aplicações e boas práticas de implementação.

6. TRABALHOS FUTUROS

Este capítulo tem por objetivo elucidar os próximos passos que podem ser adotados para ampliar o uso de IoT e BLE beacons como tecnologia a ser utilizada na segurança pública do *smart campus*, bem como explorar as técnicas e novas abordagens tecnológicas em torno do sistema desenvolvido.

No ano de 2020 a pandemia do COVID-19 afetou diversas atividades da sociedade, o que inclui este trabalho. Por conta da necessidade de isolamento social, o Teste #2 não teve o devido tempo necessário para aprofundar análises de diferentes variáveis. Apesar do teste concluir que a tecnologia possui viabilidade para o objetivo proposto, medir as especificações do cenário poderiam contribuir com mais dados para a pesquisa. Identificar com mais precisão a distância que o BLE beacon precisa mover-se para gerar um alerta é importante para a documentação da aplicação, por isso, mover o veículo a curtas distâncias em cada ciclo de atualização de monitoramento pode estimar esta variável com mais precisão. Além disso, realizar este mesmo procedimento a distancias diferentes e comparar os resultados pode elucidar o comportamento de monitoramentos executados em maiores e menores distâncias além da inferida (45 m).

A comunidade acadêmica é um importante recurso no desenvolvimento de tecnologias em ambientes inteligentes, porém, a abertura do campus como um ambiente de testes e laboratório de prototipagem é de suma importância para a exploração das possibilidades. Diálogos, protocolos e testes em ambientes de produção devem ser realizados antes de levar tecnologias como a apresentada neste trabalho para um ambiente real.

Uma alteração simples que pode gerar efeito positivo sobre a redução de obstáculos entre o dispositivo emissor e o receptor é a alocação do Raspberry em um local alto. Fixá-lo no alto de um poste/baliza/pilar pode eliminar boa parte dos obstáculos da via do estacionamento. Entretanto, essa é uma mudança que deve ser testada, uma vez que a gravidade exerce influência sobre as ondas eletromagnéticas utilizadas por dispositivos sem fio. O subcapítulo sobre o *deploy* dos dispositivos aborda a disposição do beacon em um local alto, porém, a bibliografia base utilizada menciona o principal motivo desta escolha como uma forma de evitar ataques físicos contra a infraestrutura de BLE beacons. Outras bibliografias apontam a perda de sinal ao longo do trajeto de longas distâncias, provocada pela gravidade que afeta o decaimento dos sinais bluetooth.

A nível da aplicação desenvolvida para este trabalho, uma das principais alterações que deve ser executada é o desacoplamento entre a camada de gerenciamento de BLE beacons (*Beacon Manager*) e gerenciamento de monitoramento (*Watchpost Manager*). Por conta do curto tempo disponível, o sistema não implementou as melhores práticas de desenvolvimento de software. Este projeto é open source e seu código fonte pode ser

encontrado no repositório do seguinte link: https://github.com/feerposser/ppgca-dissertacao , que disponibiliza os códigos aplicados à este trabalho e links atualizados para ambos os gerenciadores com códigos atualizados.

Ainda, melhorias podem ser feitas a respeito do Raspberry utilizado para a aplicação. O modelo disponibilizado para esta pesquisa foi o Raspberry PI 3 Model B, porém, novos modelos foram lançados no mercado durante o desenvolvimento deste trabalho. Os recursos de processamento adicionais destes modelos podem ampliar a carga de quantidade e complexidade de monitoramento utilizados no sistema. Uma destas complexidades pode ser o uso de um algoritmo de detecção de anomalias para os valores de RSSIs. Identificando os números que dizem respeito à variabilidade, os recursos necessários para identificar o deslocamento do BLE beacon podem ser reduzidos e/ou simplificados. Além disso, o uso mais intensivo de *threads* pode promover o aproveitamento de recursos de processamento e tempo de espera dos monitoramentos, tendo em vista que o sistema se caracteriza como um processo CPU/I-O que não seria influenciado negativamente pelo Python *Global Interpreter Lock*. O sistema experimental desenvolvido para este trabalho possui uma *thread* principal que gerencia os beacons e monitoramentos, e uma *thread* secundária que recebe as mensagens do canal pub/sub. Um melhor aproveitamento das *threads* poderia separar o processamento dos recursos em *n threads* para monitoramentos, leitura de BLE beacons e envio de alertas para o sistema web, podendo assim reduzir o tempo de resposta de cada um destes módulos, mas principalmente desacoplando funções que podem ser executadas em *threads* separadas. O último ponto de melhoria envolvendo o Veacon Rasp é uma metodologia de correção dos dados de RSSI próximo e distante. Mesmo com o dispositivo Raspberry acoplado em um local alto, obstáculos podem surgir, alterando assim a mediana de RSSIs. Para evitar falsos positivos decorrentes deste tipo de situação, o sistema deve alterar o RSSI próximo e distante derivado de pequenas alterações. Durante o processo de atualização e verificação de monitoramento, ao identificar uma mudança leve de sinal, o sistema deve abrir uma nova linha de processo para analisar se os valores possuem uma mudança acentuada. Caso essas alterações de sinal não sejam graves, podem ter acontecido em decorrência de um novo obstáculo entre o emissor e o receptor, como um automóvel alto ou uma pessoa parada em frente ao veículo. Ao corrigir essa mudança automaticamente, o sistema estará evitando o envio de falsos positivos.

Com relação ao Veacon Web, algumas melhorias possuem espaço para serem aplicadas. O primeiro ponto é em relação à forma de envio de alertas. O método escolhido para este trabalho tem como premissa o teste experimental que ocorre em ambiente controlado. Por isso, foi implementada uma API que recebe os alertas como requisições POST informando o ID do monitoramento, que após recebidos criam um novo recurso no banco de dados e respondem ao Veacon Rasp com um status referente à requisição. Porém, ao utilizar este tipo de software em ambiente de produção, a melhor prática para garantir que problemas decorrentes de possíveis indisponibilidades do Veacon Web ocorram é o uso de

filas, similar ao módulo PubSub já presente na aplicação. Em segundo ponto, no decorrer do trabalho os dados de latitude e longitude do *gateway* não puderam ser utilizados. Ao inicio o projeto, esses dados tinham como objetivo prover uma interface visual para mostrar aos agentes de vigilância qual era o ponto em um mapa aonde um determinado veículo estava sendo monitorado ou um alerta havia sido disparado. Outra possibilidade de uso é em ambiente real, onde *n gateways* podem ser utilizados, e o usuário deve cadastrar o monitoramento através de um app de smartphone. Para encontrar os *gateways* mais apropriados para o monitoramento, a localização do smartphone em comparação com a localização dos *gateways* poderiam definir os dispositivos mais adequados para a tarefa. O último ponto referente à melhorias no Veacon Web é sobre a programação de término de monitoramento, que também não foi implementado para este trabalho experimental devido ao tamanho de escopo e tempo de desenvolvimento. Os dados de horário e data final presente em cada monitoramento podem ser utilizados por bibliotecas de agendamento para encerrar as vigilâncias de forma automatizada, fazendo com que o usuário não precise se preocupar em encerrar a monitoria.

Além das melhorias a nível de gerência de campus e aplicação de software, uma melhoria considerável nos BLE beacons pode ser efetuada. Durante o desenvolvimento deste trabalho a versão 5 do Bluetooth foi desenvolvida. Segundo os primeiros experimentos [65], os BLE beacons podem emitir sinais a até 800 metros de distância utilizando a mesma quantidade de bateria da versão 4 e com o mesmo tamanho físico. No fim de 2020 alguns BLE beacons com a versão 5 do bluetooth já começaram a ser comercializados em mercados internacionais com descrições conferindo autonomia de até 500 metros. A ampliação da emissão de sinais pode reduzir a quantidade de Raspberrys a serem utilizados para cobrir uma grande área de atuação, também reduzindo o orçamento necessário para implementar o método no *smart campus*.

REFERÊNCIAS BIBLIOGRÁFICAS

[1] GLOBO, O. *Em abril, segurança pública dominou debate no Facebook*. 2018. Disponível em: <https://blogs.oglobo.globo.com/lauro-jardim/post/em-abril-seguranca-publica-dominou-debate-no-facebook.html>. Acesso em: Out. 15, 2019.

[2] NE10. *CNI vai cobrar cuidado com segurança pública aos candidatos a presidente*. 2018. Disponível em: <https://blogs.ne10.uol.com.br/jamildo/2018/07/02/cni-vai-cobrar-cuidado-com-seguranca-publica-aos-candidatos-a-presidente/>. Acesso em: Out. 15, 2019.

[3] PúBLICA, F. B. de S. *Anuário Brasileiro de Segurança Pública 2018*. Rua Amália de Noronha, 151. São Paulo. SP - Brasil, 2018. 89 p. Disponível em: <http://www.forumseguranca.org.br/publicacoes/anuario-brasileiro-de-seguranca-publica-2018/>. Acesso em: Jun. 14, 2019.

[4] PúBLICA, F. B. de S. *Anuário Brasileiro de Segurança Pública 2018*. Rua Amália de Noronha, 151. São Paulo. SP - Brasi, 2017. 108 p. Disponível em: <http://www.forumseguranca.org.br/publicacoes/11o-anuario-brasileiro-de-seguranca-publica/>. Acesso em: Jun. 14, 2019.

[5] MASLOW, A. Motivation and personality. *New York: Harper Row*, 1970.

[6] MARQUES, J. Comparação entre as teorias de freud e maslow sobre os estímulos para o consumo. p. 1–9, 2011.

[7] SENASP; DATAFOLHA; CRISP. *Pesquisa Nacional de Vitimização*. [S.l.], 2013. 230–231 p. Disponível em: <http://www.crisp.ufmg.br/wp-content/uploads/2013/10/Relatório-PNV-Senasp_final.pdf>. Acesso em: Jun. 14, 2019.

[8] PAULO, F. de S. *Brasileiro é o que mais teme andar sozinho a noite, aponta pesquisa*. 2015. Disponível em: <https://www1.folha.uol.com.br/bbc/2015/10/1693406-brasileiro-e-o-que-mais-teme-andar-na-rua-a-noite-aponta-pesquisa.shtml>. Acesso em: Jun. 14, 2019.

[9] IPEA; FBSP. *Atlas da Violência 2018*. [S.l.], 2018. 3–4 p. Disponível em: <http://www.ipea.gov.br/portal/images/stories/PDFs/relatorio_institucional/180604_atlas_da_violencia_2018.pdf>. Acesso em: Jun. 15, 2019.

[10] OLIVEIRA, G. *A falta de segurança castiga a economia*. 2018. Disponível em: <https://gesneroliveira.blogosfera.uol.com.br/2018/02/19/a-falta-de-seguranca-castiga-a-economia/>. Acesso em: Jun. 15, 2019.

[11] ALMEIDA, G. G. F. de; ENGEL, V. *Inovação e Cidades Inteligentes: desafios e oportunidades nas cidades no séc XXI*. Santa Cruz do Sul, RS, Brasil: Editora The Help, 2019. 307 p.

[12] ESTADãO. *Em horários de pico, tempo que paulistanos ficam no trânsito é 69% maior*. 2018. Disponível em: <https://exame.abril.com.br/brasil/em-horarios-de-pico-tempo-que-paulistanos-ficam-no-transito-e-69-maior/>. Acesso em: Jun. 20, 2019.

[13] HOJE, J. *Aplicativo do Ministério da Justiça ajuda a recuperar carros roubados*. 2015. Disponível em: <http://g1.globo.com/jornal-hoje/noticia/2015/02/aplicativo-do-ministerio-da-justica-ajuda-recuperar-carros-roubados.html>. Acesso em: Jun. 20, 2019.

[14] PLAY, G. *Sinesp Cidadão*. 2019. Disponível em: <https://play.google.com/store/apps/details?id=br.gov.sinesp.cidadao.android&hl=pt_BR>. Acesso em: Jun. 20, 2019.

[15] PINHEIRO, F. P. *Aumento da Eficiência na Prevenção de Incidentes de Segurança Pública com o Auxílio da Tecnologia*. Senador Pinheiro 304, Passo Fundo, Brasil, 2017. Disponível em: <https://www.imed.edu.br/sobre-a-biblioteca-1/banco-de-tcc-s/sistemas-de-informacao>.

[16] ESTADO. *Pelo menos 36 órgãos de segurança pública já usam drones no Brasil*. 2018. Disponível em: <https://brasil.estadao.com.br/noticias/geral,pelo-menos-36-orgaos-de-seguranca-publica-ja-usam-drones-no-brasil,70002297742>. Acesso em: Out. 16, 2019.

[17] ESTADO, C. do. *"A segurança pública ganha se o Estado investir em tecnologia", diz Bluma*. 2018. Disponível em: <https://www.correiodoestado.com.br/eleicoes-2018/a-seguranca-publica-ganha-se-o-estado-investir-em-tecnologia/337031/>. Acesso em: Out. 16, 2019.

[18] PúBLICO. *PSP pede mais tecnologia para prevenir criminalidade*. 2018. Disponível em: <https://www.publico.pt/2018/07/24/sociedade/noticia/psp-pede-mais-tecnologia-para-prevenir-criminalidade-1839002>. Acesso em: Out. 16, 2019.

[19] MIDIAMAX. *Polícia Militar e CDL buscam em aplicativo de celular ajuda para solucionar crimes*. 2018. Disponível em: <https://www.midiamax.com.br/policia/2018/policia-militar-e-cdl-buscam-em-aplicativo-de-celular-ajuda-para-solucionar-crimes>. Acesso em: Out. 16, 2019.

[20] SUL, E. do Rio Grande do. *Fórum nacional debate uso da tecnologia para qualificar Segurança Pública*. 2018. Disponível em: <https://estado.rs.gov.br/forum-nacional-debate-o-uso-da-tecnologia-na-seguranca-publica>. Acesso em: Out. 16, 2019.

[21] REGIÃO, F. da. *Tecnologia ajuda a PM a reduzir crimes em Aracçatuba*. 2018. Disponível em: <http://www.folhadaregiao.com.br/2018/08/05/tecnologia-ajuda-a-pm-a-reduzir-crimes-em-aracatuba>. Acesso em: Jun. 20, 2019.

[22] STANDARDIZATION, I. O. for. *ISO 37122:2019, Sustainable cities and communities – Indicators for smart cities*. [S.l.], 2019.

[23] MISHRA, N. R.; SURI, P. M.; CHOPRA, S. Smart toilets using ble beacon technology. *IEEE*, 2018.

[24] FERREIRA, J. E. et al. Smart services: A case study on smarter public safety by a mobile app for university of são paulo. *IEEE*, 2017.

[25] AHMAD, B. I. et al. An iot based smart campus architecture for institutions in developing countries. *i-manager's Journal on Embedded Systems.*, v. 7, n. 1, p. 18–25, 2018.

[26] KOSE, U.; VASANT, P. Better campus life for visually impaired university students: intelligent social walking system with beacon and assistive technologies. *Springer*, 2018.

[27] SUL, S. de Segurança Pública do Estado do Rio Grande do. *Estatísticas Criminais*. 2019. Disponível em: <https://ssp.rs.gov.br/estatisticas>. Acesso em: Out. 16, 2019.

[28] G1. *Brasil é o segundo país mais violento da América do Sul, aponta ONU*. 2019. Disponível em: <https://g1.globo.com/mundo/noticia/2019/07/08/brasil-e-o-segundo-pais-mais-violento-da-america-do-sul-aponta-onu.ghtml>. Acesso em: Out. 16, 2019.

[29] R7. *Brasil é o 9º país mais violento do mundo, segundo a OMS*. 2018. Disponível em: <https://noticias.r7.com/cidades/brasil-e-o-9-pais-mais-violento-do-mundo-segundo-a-oms-17052018>. Acesso em: Out. 16, 2019.

[30] PARANá, G. do Estado do. *Polícia Civil desarticula quadrilha de roubo e adulteração de veículos*. 2015. Disponível em: <http://www.aen.pr.gov.br/modules/noticias/article.php?storyid=87145tit=Policia-Civil-desarticula-quadrilha-de-roubo-e-adulteracao-de-veiculos>. Acesso em: Abr. 14, 2021.

[31] PAULO, F. de S. *Em SP, um carro some a cada cinco minutos*. 2000. Disponível em: <https://www1.folha.uol.com.br/fsp/veiculos/cv3004200009.htm>. Acesso em: Abr. 14, 2021.

[32] GOMES, L. F. *Roubos e furtos de veículos no Brasil*. 2015. Disponível em: <https://professorlfg.jusbrasil.com.br/artigos/159559237/roubos-e-furtos-de-veiculos-no-brasil>. Acesso em: Abr. 14, 2021.

[33] CLICRBS. *Operação ataca esquema de troca de carros roubados no RS por armas e drogas no Uruguai*. 2020. Disponível em:

<https://gauchazh.clicrbs.com.br/seguranca/noticia/2020/02/operacao-ataca-esquema-de-troca-de-carros-roubados-no-rs-por-armas-e-drogas-no-uruguai-ck6vzp8le0kzq01qd5749dlej.html>. Acesso em: Abr. 14, 2021.

[34] CLICRBS. *ÁUDIOS: diálogos revelam planos de roubos e trocas de veículos por drogas no Uruguai*. 2020. Disponível em: <https://gauchazh.clicrbs.com.br/seguranca/noticia/2020/02/audios-dialogos-revelam-planos-de-roubos-e-trocas-de-veiculos-por-drogas-no-uruguai-ck6wcctbb0l3701qduxy9imsw.html>. Acesso em: Abr. 14, 2021.

[35] KON, F.; SANTANA, E. F. Z. Xxxvi congresso da sociedade brasileira de computação. In: SOCIEDADE BRASILEIRA DE COMPUTAçãO. Porto Alegre, RS, Brasil: SBC, 2016. p. 2–49.

[36] ABUARQOUB, A. et al. A survey on internet of things enabled smart campus applications. *Association for Computing Machinery*, 2017.

[37] MAGALDI, S.; NETO, J. S. *Gestão do Amanhã*. [S.l.]: Editora Gente, 2018. 256 p.

[38] CAROLI, P. *Lean Inception*. [S.l.]: Editora Caroli, 2018. 160 p.

[39] CATA, M. Smart university, a new concept in theinternet of things. *IEEE*, 2015.

[40] AHMAD, B. I. et al. An iot based smart campus architecture for institutions in developing countries. *i-manager's Journal on Embedded Systems.*, v. 7, n. 1, p. 18–25, 2018.

[41] ANTEVSKI, K.; REDONDI, A. E. C.; PITIC, R. A hybrid ble and wi-fi localization system for the creation of study groups in smart libraries. In: *9th IFIP Wireless and Mobile Networking Conference (WMNC)*. [S.l.]: IEEE, 2016.

[42] FERREIRA, J. E. et al. Smart services: A case study on smarter public safety by a mobile app for university of são paulo. *IEEE*, 2017.

[43] KRISTIANA, W. A. et al. Uuid beacon advertisements for lecture schedule information. *Proceeding of EECSI*, IEEE, Malang, Indonésia, 2018.

[44] AHMETOVIC, D. et al. Navcog: A navigational cognitive assistant for the blind. *MobileHCI*, ACM, Florença, Itália, n. 16, 2016.

[45] CASTILLO-CARAA, M. et al. Ray: Smart indoor/outdoor routes for the blind using bluetooth 4.0 ble. In: *The 7th International Conference on Ambient Systems, Networks and Technologies (ANT 2016)*. [S.l.]: Procedia Computer Science 83, 2016. p. 690–694.

[46] JEON, K. E. et al. Ble beacons for internet of things applications: Survey, challenges and opportunities. *IEEE INTERNET OF THINGS JOURNAL*, IEEE, A, 2018.

[47] FERRO, E.; POTORTI, F. Bluetooth and wi-fi wireless protocols: A survey and a comparison. *IEEE Wireless Communications*, IEEE, p. 12–26, 2005.

[48] BOUKHECHBAA, M. et al. A novel bluetooth low energy based system for spatial exploration in smart cities. *Science Direct*, 2017.

[49] GARCíA, G. C.; RUIZ, I. L.; GóMEZ-NIETO, M. Ángel. State of the art, trends and future of bluetooth low energy, near field communication and visible light communication in the development of smart cities. *Sensors*, MDPI, University of Córdoba. Córdoba, Espanha, 2016.

[50] ESTIMOTE. *The icon of modern art puts Estimote beacons on display*. 2017. Disponível em: <https://blog.estimote.com/post/157200820650/the-icon-of-modern-art-puts-estimote-beacons-on>. Acesso em: Out. 13, 2019.

[51] KOSE, U.; VASANT, P. Better campus life for visually impaired university students: intelligent social walking system with beacon and assistive technologies. *Wireless Networks*, Springer, 2018.

[52] APPLE. *iBeacon*. Disponível em: <https://developer.apple.com/ibeacon/>. Acesso em: Out. 14, 2019.

[53] GOOGLE. *Eddystone Documentation*. Disponível em: <https://github.com/google/eddystone>. Acesso em: Out. 14, 2019.

[54] INC., G. *Google Beacon Platform*. Disponível em: <https://developers.google.com/beacons/>. Acesso em: 15 de dezembro de 2019.

[55] GOOGLE. *Nearby*. Disponível em: <https://developers.google.com/nearby>. Acesso em: 15 de julho de 2020.

[56] NAYAK, R. *Discontinuing support for Android Nearby Notifications*. 2018. Disponível em: <https://android-developers.googleblog.com/2018/10/discontinuing-support-for-android.html>. Acesso em: 15 de julho de 2020.

[57] FARAGHER, R.; HARLE, R. Location fingerprinting with bluetooth low energy beacons. *IEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS*, IEE, v. 33, n. 11, p. 2418–2428, 2015.

[58] AL-MUAYTHIR, A.; HOSSAIN, M. A. Cloud-based parametrized publish/subscribe system for public safety applications in smarter cities. *Association for Computing Machinery*, 2016.

[59] MOORE, J. et al. Devops for the urban iot. *Urb-IoT*, ACM, Tokyo, Japão, n. 16, 2016.

[60] COMMUNITY, D. *Django Web Framework Project*. Disponível em: <https://www.djangoproject.com/>. Acesso em: Out. 13, 2019.

[61] COMMUNITY, D. *Django Rest Framework Project*. Disponível em: <https://www.django-rest-framework.org/>. Acesso em: Out. 13, 2019.

[62] GIL, A. C. *Como Elaborar Projetos de Pesquisa*. São Paulo: Editora Atlas, 2002. 176 p.

[63] SEELE, F. *Beacontools*. 2017. Disponível em: <https://pypi.org/project/beacontools/>. Acesso em: 12 de dezembro de 2019.

[64] PINHEIRO, F. P. *veacon_testes*. 2021. Disponível em: <https://github.com/feerposser/veacon_testes>. Acesso em: 10 de fevereiro de 2021.

[65] PONDE, S. R. Bluetooth 5: An augmented technology for iot. *National Conference On Technology Innovation: Disrupting Businesses, Transforming Market*, International Journal of Research in Engineering, IT and Social Sciences, v. 9, n. 6, p. 10–20, 2019.

UPF
UNIVERSIDADE
DE PASSO FUNDO

UPF Campus I - BR 285, São José
Passo Fundo - RS - CEP: 99052-900
(54) 3316 7000 - www.upf.br