



UNIVERSIDADE FEDERAL DE RORAIMA
PRÓ-REITORIA DE ENSINO E EXTENSÃO
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO

LUARKIAN KAYPE DE SOUSA

INEXT: Um Sistema Computacional para Localização Indoor de Objeto com RFID

Boa Vista - RR

2019

LUARKIAN KAYPE DE SOUSA

**INEXT: Um Sistema Computacional para Localização Indoor de Objeto com
RFID**

Monografia de Graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Roraima como requisito para a obtenção do grau de Bacharel em Ciência da Computação.

Orientador: Dr. Herbert Oliveira Rocha

Boa Vista - RR

2019

Dados Internacionais de Catalogação na publicação (CIP)
Biblioteca Central da Universidade Federal de Roraima

S725i Sousa, Luarkian Kaype de.

Inext : um sistema computacional para localização de objeto com
RFID / Luarkian Kaype de Sousa. – Boa Vista, 2019.
62 f. : il.

Orientador: Prof. Dr. Herbert Oliveira Rocha.

Trabalho de Conclusão de Curso (graduação) - Universidade Federal
de Roraima, Curso de Ciência da Computação.

1 - Localização indoor. 2 - RFID. 3 - IoT. I - Título. II - Rocha,
Herbert Oliveira (orientador).

CDU - 681.3.01

Ficha Catalográfica elaborada pela Bibliotecária/Documentalista:
Maria de Fátima Andrade Costa - CRB-11/453-AM

LUARKIAN KAYPE DE SOUSA

INEXT: Um Sistema Computacional para Localização Indoor de Objeto com RFID

Monografia de Graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Roraima como requisito para a obtenção do grau de Bacharel em Ciência da Computação. Defendido em 10 de Julho de 2019 e aprovado pela seguinte banca examinadora:



Prof. Dr. Herbert Oliveira Rocha
Orientador / Curso de Ciência da
Computação - UFRR



Prof. MSc. Acauan Cardoso Ribeiro
Curso de Ciência da Computação - UFRR



Prof. Dr. Leandro Nelinho Balico
Curso de Ciência da Computação - UFRR

Dedico com muito amor àquela que luta todos os dias pela minha educação, chora as minhas lágrimas, sorri com as minhas alegrias. Minha mãe Maria Francisca.

AGRADECIMENTOS

Primeiramente, agradeço a minha família, por sempre estar ao meu lado nos momentos mais difíceis da faculdade, pelos conselhos e pelo incentivo em todos estes anos de faculdade.

Agradeço ao meu orientador, professor Herbert Oliveira, por todo apoio e por todas as orientações no desenvolvimento deste projeto, além dos incentivos para vida pós faculdade.

Agradeço aos professores do DCC que durante estes anos de curso foram fonte de aprendizado e incentivos durante aulas e que cada uma pode contribuir de alguma forma no meu desenvolvimento como aluno e futuro profissional.

”Uma flecha só pode ser lançada se for puxada para trás. Então, quando a vida te puxar para trás, significa que ela vai te lançar para algo grande. Apenas mantenha o foco e continue mirando.” - Autor Desconhecido

RESUMO

A ampla utilização da Internet das Coisas (IoT) tem estimulado uma crescente demanda por sistema de localização em ambiente fechados, tendo em vista que o sistema de posicionamento global (GPS) pode não ser eficiente ou mesmo não funcionar tão bem em tal ambiente. A tecnologia de identificação por radiofrequência (RFID) facilita a construção de sistemas com baixo custo para localização e identificação de objetos. Neste trabalho propomos um sistema computacional que utiliza leitores e etiquetas RFID passivas, para o gerenciamento e controle de ativos, sendo capaz de localizar e identificar objetos etiquetados em edifícios.

Palavras-chaves: Localização Indoor, RFID, IoT.

LISTA DE FIGURAS

Figura 1 – Sistemas Embarcados.	14
Figura 2 – Microcontrolador ATmega328P	15
Figura 3 – Arduíno Nano	16
Figura 4 – NodeMcu	16
Figura 5 – Internet of Things.	17
Figura 6 – Transmissão não-direcional.	18
Figura 7 – Transmissão direcional.	18
Figura 8 – Constelação de Satélites.	19
Figura 9 – Triangulação.	20
Figura 10 – Trilateração.	20
Figura 11 – Reconhecimento de padrões.	21
Figura 12 – LANDMARC.	22
Figura 13 – Rede Bayesiana	23
Figura 14 – Elementos UML	27
Figura 15 – Modelo Proposto	33
Figura 16 – Caso de uso	34
Figura 17 – Objeto com tag RFID	35
Figura 18 – Tela inicial	35
Figura 19 – Tela para editar objeto	36
Figura 20 – Fluxograma de cada sala	37
Figura 21 – Representação da rede	38
Figura 22 – Arquivo de log	39
Figura 23 – Restrição	39
Figura 24 – Leitor RFID RC522	41
Figura 25 – Esquema de Conexões	42
Figura 26 – Arquivos do Back-end	44
Figura 27 – Protótipo Arduíno Nano	48
Figura 28 – Protótipo NodeMcu	49
Figura 29 – Cenário de simulação	50
Figura 30 – Identificação dos objeto	51
Figura 31 – Transição de objeto	52
Figura 32 – Cadastro de usuários e Criação de Restrições de objeto	53
Figura 33 – Transições com Restrições	54
Figura 34 – Inventário	55

LISTA DE TABELAS

Tabela 1 – Resultados	55
Tabela 2 – Dados coletados do questionário realizado	56

SUMÁRIO

1	INTRODUÇÃO	11
1.1	Definição do Problema	12
1.2	Objetivos	13
1.3	Organização do trabalho	13
2	CONCEITOS E DEFINIÇÕES	14
2.1	Sistemas Embarcados	14
2.1.1	<i>Internet of Things (IoT)</i>	16
2.2	Comunicação Sem Fio	17
2.2.1	Transmissão por rádio	17
2.3	Localização	18
2.3.1	GPS	19
2.3.2	WLAN	19
2.3.3	RFID	21
2.4	Algoritmos de Localização indoor	22
2.4.1	Multilateração	22
2.4.2	Inferência Bayesiana	23
2.4.3	K-Nearest-Neighbor	24
2.4.4	Proximidade	25
2.5	Node.js	25
2.6	MongoDB	26
2.7	Modelagem de Sistemas	26
2.7.1	UML - <i>Unified Modeling Language</i>	26
3	TRABALHOS CORRELATOS	29
3.1	<i>An indoor localization mechanism using active RFID tag</i>	29
3.2	<i>Real-Time Locating Systems Using Active RFID for Internet of Things</i>	30
3.3	<i>Object localization using RFID</i>	30
3.4	<i>RFID localization algorithms and applications-A review</i>	31
4	MÉTODO PROPOSTO	32
4.1	Visão geral do método proposto	32
4.2	Identificação dos objetos rastreáveis	34
4.3	Monitoramento e Localização dos objetos rastreáveis	36
4.3.1	Fluxo de execução do método em cada sala	36

4.3.2	Localização dos objetos	37
4.4	Configuração de controle e monitoramento de transição dos objetos	38
4.5	Conexão dos dispositivos de porta com o servidor	40
4.6	Dispositivos de porta	40
4.7	Servidor Web	42
4.7.1	Implementação	43
4.7.1.1	<i>Back-End</i>	43
5	AVALIAÇÃO EXPERIMENTAL	47
5.1	Planejamento e projeto dos experimentos	47
5.2	Execução dos experimentos e Análise dos Resultados	50
5.2.1	Resultados	53
6	CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS	57
7	ANEXO A	58
7.1	Revisão Sistemática	58
7.1.1	Planejamento da Revisão Sistemática	58
7.1.2	Procedimento de Seleção de Critérios	59
	REFERÊNCIAS	61

1 INTRODUÇÃO

A localização é um tipo de informação situacional muito importante, no contexto geral temos aplicações que fornecem a localização de algo, por exemplo para casas nós temos ruas, número da casa e bairro. No sistema de posicionamento global (GPS) para um ponto temos latitude e longitude. Entretanto quando se restringe o ambiente considerando apenas edifícios e o alvo para itens de uso geral (cadeiras, mesas, notebooks, desktop, etc) seja para realizar monitoramento, acompanhamento do fluxo e/ou rastreamento, poucas ferramentas proporcionam isso com baixo custo (ZHOU, 2009).

As conexões entre computadores que chamavam de redes de computadores quando se referia a um conjunto de computadores autônomos interconectados (TANENBAUM, 2002), esse conceito está obsoleto pois não se tem apenas computadores conectados, mas diversos dispositivos que podem ser ou não semelhantes, podendo ser câmeras, sensores, *smartphones*, ou qualquer objeto que possua um hardware com capacidade de conectividade (SANTOS et al., 2016).

Com os avanços tecnológicos interações com objetos tornou-se cada vez mais frequente, isso se deu a um paradigma que visa uma interligação de objetos em rede para interação e cooperação podendo mudar ou não a forma como as atividades rotineiras serão realizadas, esse modelo é chamado de *Internet of Things (IoT)* (ZHANG et al., 2016).

A IoT que também está relacionada com computação ubíqua é chamada de tecnologia do futuro, a comunicação de diferentes dispositivos, edifícios ou qualquer equipamento que possua um microcontrolador ou microprocessador embutido e que podem se conectar a redes para assim ter uma comunicação com outros se encaixa nessa relação (JIN et al., 2006).

No trabalho de Leite et al. (2017) é realizado uma abordagem para *smart cities*, que são aplicações que visam a troca de informações entre veículos, *smartphones*, semáforos e qualquer outro dispositivo capaz de enviar e receber dados, com a troca de dados entre os dispositivos pretende-se ter um melhor fluxo de carros em uma cidade, gerenciamento de desperdícios e monitoramento de qualidade de vida na cidade.

Ni et al. (2003) apresenta um sistema de localização RFID, que foi aplicado para facilitar a gestão de hospitais e outras organizações, para melhorar os serviços, economizar custos e reduzir os riscos. O sistema também era utilizado para monitorar pacientes infecciosos e garantir resposta no tempo adequado para emergências.

A interligação dos objetos as redes possibilitou que o meio científico propusessem várias maneiras de localização em ambientes confinados, visto que o GPS não funciona tão bem para a solução proposta, tendo em vista que um edifício possui muitas salas ou/e andares, e com o GPS

não seria possível saber em qual sala por não ter uma planta baixa do prédio ou informar qual o andar o objeto se encontra pois o GPS retornaria para a aplicação as coordenadas de latitude e longitude (ZHOU, 2009).

Para solucionar essa incapacidade de localização do GPS, inúmeras aplicações foram propostas para a localização indoor. Aplicações que utilizam tecnologia infravermelho difuso para estimar a posição, sistemas que utilizam adaptadores de rede padrão IEEE 802.11 para rastrear os objetos no interior do prédio, sistemas baseados na tecnologia ultra-sônica e também sistemas que utilizam RFID para rastrear indoor foram propostos (JIN et al., 2006).

O RFID foi a tecnologia escolhida por proporcionar a utilização de etiquetas que não necessitam de uma fonte de alimentação para seu funcionamento, pois são alimentadas pelas ondas eletromagnéticas emitidas pela antena do leitor, sendo que isso não é possível em outras tecnologias. Uma outra funcionalidade do RFID é a possibilidade de identificação das etiquetas que possui uma ID única.

1.1 Definição do Problema

A dificuldade de localizar objetos em ambientes confinados ou fechados, cria uma necessidade de novas ferramentas para tal aplicação sendo que o GPS não se aplica a tal problema por na localização do GPS os prédios são apresentados como um edifício sem salas e divisórias, sem contar o custo benefício para anexar módulos de GPS aos objetos, isso iria deixar o protótipo caro. Contudo, o motivo desta pesquisa é auxiliar na localização de objetos permanentes (como computadores e móveis) em ambientes de uma forma que não tenha um custo elevado quanto a utilização do GPS (JIN et al., 2006).

Localizar e identificar objetos em certos ambientes têm grande importância quando se quer gerenciar e controlar ativos, por exemplo em comércios, indústrias ou instituições de ensino que possuem uma gama de bens que podem ser movimentados dentro do local, poder identificar o objeto que está sendo movido e saber em que local o objeto está, é de grande importância para ter um maior controle sobre os bens (ZHANG et al., 2016).

A tecnologia RFID é um meio de comunicação sem fio que utiliza campo eletromagnéticos de radiofrequência para identificar etiquetas. As etiquetas podem ser passivas, ativas e semi-passivas. As etiquetas semi-ativa e ativas possuem uma fonte de alimentação, a semi-ativa só emite sinais quando entra em contato com o leitor, já a ativa emite os sinais a todo momento, as etiquetas passivas não possuem fonte de alimentação entrando em funcionamento apenas quando as ondas eletromagnéticas emitidas pelo leitor os alimentam (ZHANG et al., 2016). As funcionalidade da tecnologia RFID fez com que muitas técnicas de localização fossem propostas.

O problema considerado neste trabalho é expresso na seguinte questão: **Como projetar um sistema computacional capaz de localizar e identificar objetos em tempo real com**

baixo custo, em ambientes confinados ou fechados, garantindo confiabilidade nas informações fornecidas e auxiliando usuários no gerenciamento de bens por meio da utilização de radiofrequência?

1.2 Objetivos

O objetivo principal deste trabalho é projetar e desenvolver um sistema computacional autônomo para o gerenciamento de objetos com RFID em ambientes confinados ou fechados, utilizando técnicas de localização e identificação de redes IoT.

Objetivos específicos:

1. Analisar soluções computacionais de baixo custo que são capazes de localizar objetos indoor;
2. Identificar algoritmos para localização de objetos indoor;
3. Projetar e desenvolver um protótipo de um sistema capaz de localizar e identificar objetos em tempo real, no âmbito confinado utilizando RFID; e
4. Validar o método proposto com o propósito de examinar sua eficácia e aplicabilidade.

1.3 Organização do trabalho

A introdução deste trabalho apresentou: o contexto, definição do problema, objetivos, metodologia e contribuições dessa pesquisa. Os capítulos restantes são organizados da seguinte forma:

No **Capítulo 2, Conceitos e Definições**, são apresentados fundamentos teóricos que abordam os seguintes assuntos: sistemas embarcados, sistemas de comunicação e localização, algoritmos utilizados em localização, e modelagem.

No **Capítulo 3, Trabalhos Correlatos**, são apresentados trabalhos correlatos a utilização de RFID e localização.

No **Capítulo 4, Método Proposto**, é descrito as etapas da solução proposta neste trabalho, de forma a mostrar a arquitetura do sistema proposto e componentes necessários para a aplicação.

No **Capítulo 5 Avaliação Experimental**, descreve-se a execução de uma avaliação experimental.

E por fim no **Capítulo 6, Considerações Finais e Trabalhos Futuros**, será apresentado as considerações finais e trabalhos futuros.

2 CONCEITOS E DEFINIÇÕES

O principal objetivo desse capítulo é apresentar conceitos necessários para o entendimento deste trabalho. Os assuntos abordados neste capítulo são: sistemas embarcados, sistemas de comunicação e localização, algoritmos utilizados em localização, e modelagem.

2.1 Sistemas Embarcados

Segundo [Almeida et al. \(2017\)](#) sistemas embarcados estão presente em quase todos os ambientes, tais sistemas possuem uma única função específica e que não pode ser alterada. Eles são controlados por microprocessadores ou microcontroladores de forma que possuem muitas restrições em relação a recursos computacionais.

Atualmente é possível encontrar sistemas embarcados em diversos dispositivos, por exemplo: televisores, micro-ondas, sistemas de gerenciamento de aviação, esteiras, etc. Os dispositivos que fazem uso de eletricidade para seu funcionamento, basicamente possuem um sistema embarcado para articular o seu funcionamento ([ALMEIDA et al., 2017](#)).

Na [Figura 1](#) estão alguns dos aparelhos em que é possível se encontrar sistemas embarcados. É possível notar que aparelhos antigos já utilizavam tais sistemas e com o passar dos anos cada vez mais estão sendo introduzidos nos eletrônicos.

Figura 1 – Sistemas Embarcados.



Fonte: ([LI; YAO, 2003](#))

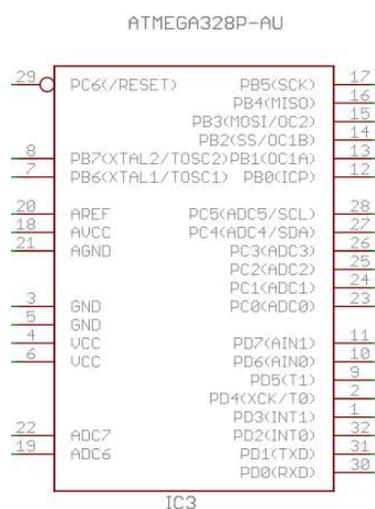
Uma definição geral para sistemas embarcados: são sistemas que realizam uma função dedicada e possuem a integração de hardware e software fortemente acoplados, geralmente são

uma parte específica de um sistema maior (LI; YAO, 2003).

Os microcontroladores e microprocessadores são dispositivos fundamentais nos sistemas embarcados, porém eles possuem algumas diferenças, entre as principais delas é que os microcontroladores já possuem todos os componentes necessários para seu funcionamento diferente dos microprocessadores que necessitam de outros componentes e periféricos para funcionar por exemplo memória, dispositivos de entrada/saída, conversores de sinais (AYALA, 1991).

Um exemplo de microcontrolador é o ATmega328P, que pode ser encontrado no Arduino Nano, esse microcontrolador conta com 32 Kilobytes para a gravação de programas, 2 Kilobytes de SRAM (*Static Random Access Memory*, Memória de acesso aleatório estático), 1 Kilobyte de EEPROM (*Electrically-Erasable Programmable Read-Only Memory*, Memória somente leitura programável apagável eletricamente) e 2 Kilobytes para o carregador de inicialização, ATmega328P pode ser visualizado na Figura 2 (ARDUINO, 2018).

Figura 2 – Microcontrolador ATmega328P

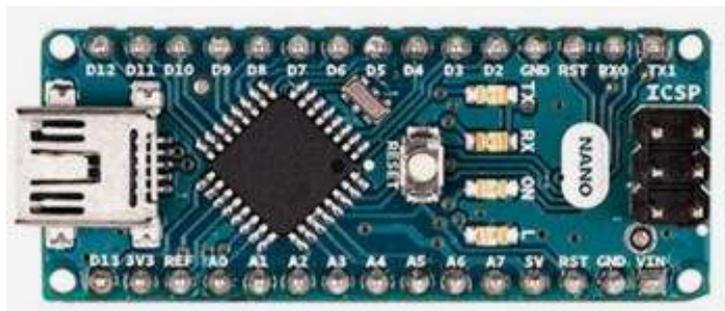


Fonte: (ARDUINO, 2018)

O Arduino é uma popular ferramenta para desenvolvimento de aplicações IoT (*Internet of things*, Internet das coisas), também sendo utilizado para ensino e seu hardware é *Open Source* possibilitando que a comunidade contribua com tutoriais e suporte. O Arduino Nano é mostrado na Figura 4, é uma placa difundida e designers, engenheiros, estudantes, desenvolvedores e fabricantes de todo o mundo estão usando o Arduino para inovar (ARDUINO, 2018).

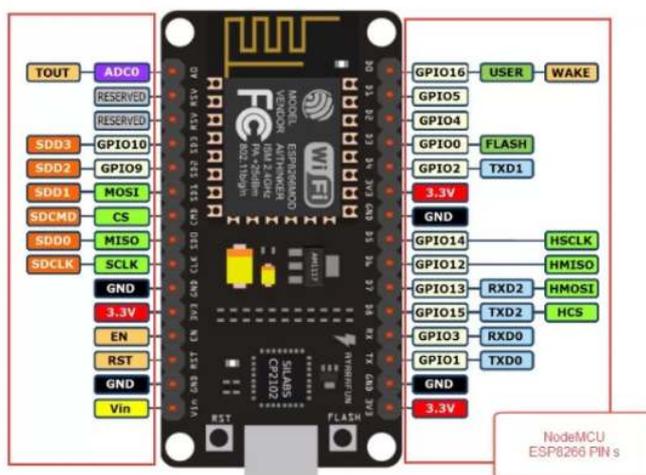
Uma outra ferramenta de desenvolvimento é o NodeMcu, uma placa *Open Source* que combina o chip ESP8266 (ESP-12E) com a pilha de protocolos TCP/IP integrada, essa combinação possibilita a conectividade com rede WiFi. O NodeMcu conta com um processador (Tensilica LX106) podendo atingir 160MHz, possui uma memória RAM de 20KB e memória flash de 4MB. A linha de programação pode ser realizada utilizando LUA ou a IDE do Arduino (NODEMCU, 2018).

Figura 3 – Arduino Nano



Fonte: (ARDUINO, 2018)

Figura 4 – NodeMcu



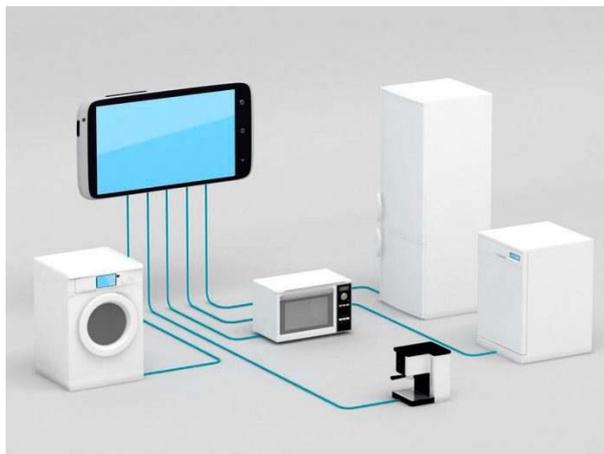
Fonte: (MEES, 2018)

2.1.1 Internet of Things (IoT)

A possibilidade de comunicação entre objetos de uso cotidiano do ambiente real com a internet referênci o termo IoT, quando um objeto está conectado a rede de computadores e passa a transmitir informações de seu funcionamento ou estado, tal objeto passa a ser denominado de objeto inteligente (SÔNEGO RODERVAL MARCELINO, 2016).

De acordo com Oliveira (2017), o conceito de IoT não é novo, pois desde os passos iniciais da internet já se pensava em formas de traçar a comunicação entre objetos do dia a dia com a internet. Com os avanços de sistemas embarcados o desenvolvimento de uma infinidades de padrões e protocolos para a integração de WSN (*Wireless sensor network*, rede de sensores sem fio) tornaram IoT uma realidade.

Figura 5 – Internet of Things.



Fonte: (MAGNUS, 2017)

A Figura 5 representa bem o cenário que IoT proporciona, na imagem é possível notar que todos os objetos estão conectados por uma espécie de cabo azul, mas isso é só uma representação, essa conexão também pode ser através de conectividade sem fio. Ainda sobre a imagem, o *smartphone* tem um papel interessante, pois ele está sendo o encarregado de mostrar as informações enviadas pelos objetos para o usuário.

Han e Lim (2010) em seu trabalho, propõem um sistema *smart home* que contém os serviços de automação residencial controle de ar-condicionados, umidificadores, filtros de ar e movimentos das cortinas baseados nas informações do clima, segurança doméstica para prevenir explosões e acidentes e o último serviço é o gerenciamento através da internet é um exemplo real de que os conceitos de IoT vem sendo estudados e implementados no dia-a-dia. Muitas aplicações vêm sendo desenvolvidas para o conforto, segurança ou monitoramento partindo dos conceitos de IoT.

2.2 Comunicação Sem Fio

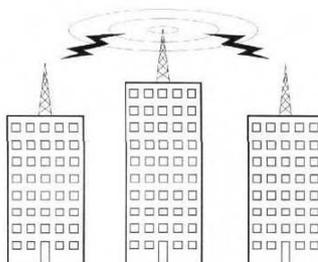
A medida que os elétrons se movimentam ondas eletromagnéticas são criada no espaço, essas ondas são medidas de acordo com suas oscilações e chamadas de frequência (Hertz), o comprimento dessa onda é medido pela distância de dois pontos máximos ou dois pontos mínimos seguidos (TANENBAUM, 2002). Segundo Tanenbaum (2002) ao colocar uma antena em um circuito elétrico apropriado pode-se transmitir e receber ondas eletromagnéticas, a comunicação sem fio é baseada nisso.

2.2.1 Transmissão por rádio

A transmissão de dados por rádio pode acontecer de duas maneiras: não-direcional e direcional (TORRES, 2001).

- Não-Direcional: Quando a transmissão é feita pela forma não-direcional, as frequências são emitidas em todas as direções e qualquer antena localizada na região de alcance das ondas de rádio podem captar os dados (TORRES, 2001), isso pode ser visto na Figura 6.

Figura 6 – Transmissão não-direcional.

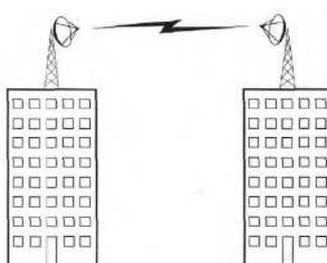


Fonte: (TORRES, 2001)

A Figura 6 mostra que as ondas de rádio são emitidas em todas as direções e que qualquer antena/receptor que estiver no alcance pode receber os dados do emissor.

- Direcional: A transmissão no sistema direcional necessita que os aparelhos transmissores e receptores estejam apontando um na direção do outro para que haja comunicação, sem contar que não pode ter obstáculos entre eles senão dificulta a transmissão (TORRES, 2001). A Figura 7 exemplifica o funcionamento dessa forma de transmissão.

Figura 7 – Transmissão direcional.



Fonte: (TORRES, 2001)

De outro modo a Figura 7 mostra que o primeiro transmissor deve estar direcionado para a direção do segundo transmissor, e assim deve acontecer com o segundo transmissor também.

2.3 Localização

Segundo Zhou (2009), as informações situacionais de pessoas ou objetos têm um papel muito importante nas aplicações, dessa forma é possível saber a posição dos objetos ou pessoas

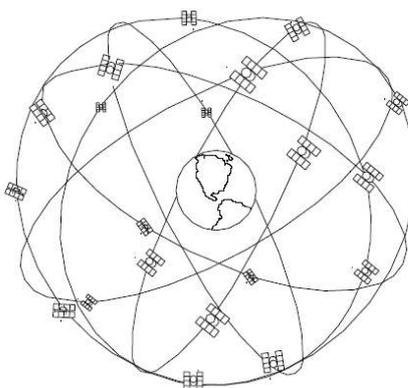
para assim fazer um monitoramento ou utilizar para várias outras aplicações. Essas informações podem ser obtida através de sistemas de posicionamento, que podem ser classificados dependendo do ambiente, podendo ser destinada a ambientes internos ou externo. Esta seção aborda alguns dos tipos de sistemas de localização.

2.3.1 GPS

GPS é um sistema de posicionamento global implementado pelo programa NAVSTAR (*Navigation System Timing and Ranging*), iniciado no ano de 1973. Era mantido pela divisão de sistema espacial dos Estados Unidos e era destinado apenas para uso militar (HUERTA ALDO MANGIATERRA, 2005).

O principal objetivo do uso de GPS é determinar as coordenadas espaciais de pontos referentes a um sistema mundial, para isso o sistema faz uso de distâncias entre quatro satélites, a posição do receptor é calculada a partir dos sinais recebido pelos satélites (HUERTA ALDO MANGIATERRA, 2005).

Figura 8 – Constelação de Satélites.



Fonte: (HUERTA ALDO MANGIATERRA, 2005)

A Figura 8 mostra a movimentação dos satélites em torno da Terra para assim enviar sinais para os receptores que por sua vez interpretam esses sinais resultando em seu posicionamento na Terra.

2.3.2 WLAN

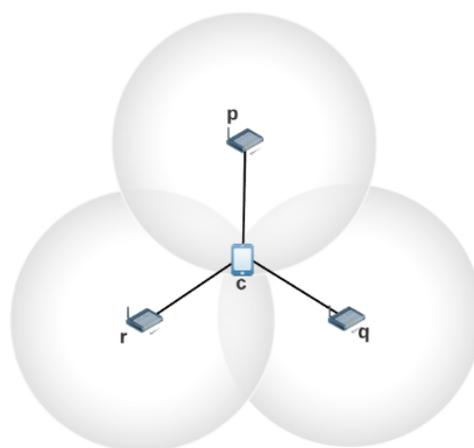
A localização em ambientes indoor utilizando WLANs pode ser feita com a RSSI, *Angle of Arrival* (AOA), ou *Time Difference of Arrival* (TDOA) (FERNANDES, 2012). Os dispositivos devem possuir conectividade sem fio para que seja possível saber seu posicionamento no ambiente. A função que permite a utilização de RSSI está disponível em todas as interfaces 802.11 (FANG; LIN, 2012).

Entre as inúmeras maneiras de localizar dispositivos em ambientes fechados utilizando WLANs, algumas são:

- Triangulação:

Essa forma de localizar faz uso de AOA (*angle of arrival*), que seria computação dos ângulos a múltiplos ponto de acesso, o resultado disso é uma interceptação que resulta na provável localização, isso pode ser visto na [Figura 9 \(FERNANDES, 2012\)](#).

Figura 9 – Triangulação.

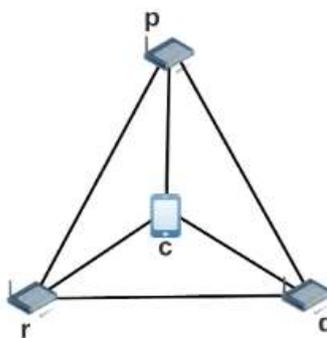


Fonte: (FERNANDES, 2012)

- Trilateração:

Utilizando propriedades geométricas essa técnica faz cálculos entre múltiplos pontos de acesso para assim obter a posição do dispositivo (FERNANDES, 2012).

Figura 10 – Trilateração.



Fonte: (FERNANDES, 2012)

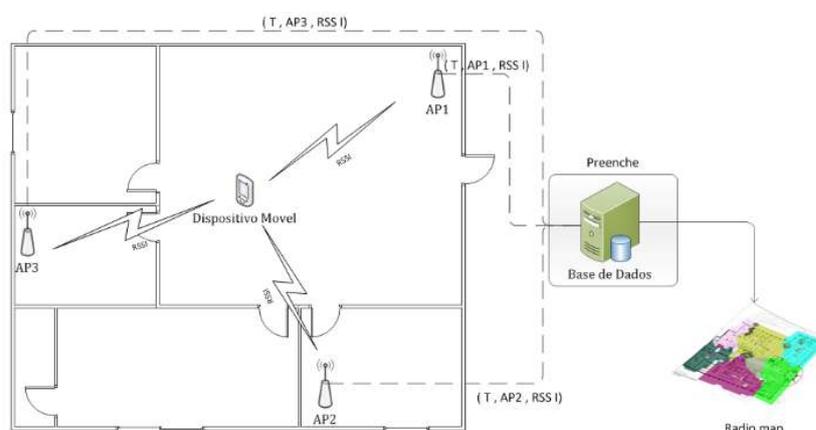
Na [Figura 10](#) é mostrado que há uma comunicação entre os pontos de acesso para assim poder efetuar os cálculos, esses cálculos são uma forma de saber a TDOA (*time difference of arrival*) para assim estimar a posição do objeto (FERNANDES, 2012).

- Reconhecimento de padrões:

O RSS (*received strength signal*) é o principal requisito dessa técnica, em que é feita medições prévias para fazer uma comparação com os dados do banco de dados. Inicialmente é necessário uma fase em que é feita a calibração para se obter o mapa de assinatura (FERNANDES, 2012).

O mapa de assinatura é basicamente dados do banco que representam a coleção das medidas de RSSI (*received strength signal indicator*) em diferentes locais para todos os pontos de acesso no ambiente (FERNANDES, 2012). Na Figura 11 é mostrado seu funcionamento, onde cada ponto de acesso se comunica com o servidor para assim fazer uma comparação com os dados do mapa de assinatura.

Figura 11 – Reconhecimento de padrões.



Fonte: (FERNANDES, 2012)

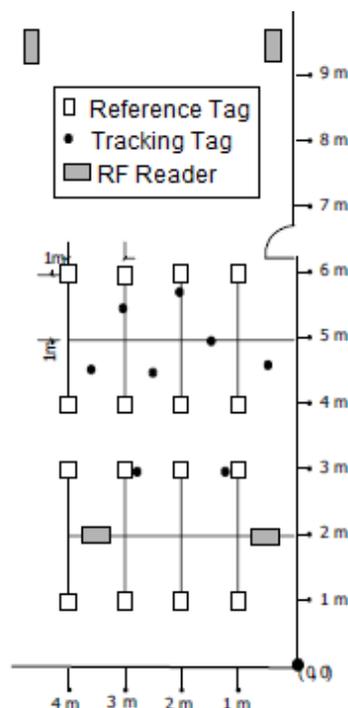
2.3.3 RFID

Nessa subseção estaremos mostrando a técnica de localização com RFID utilizada no sistema *LocAlizatioD iDentification based on dynaMic Active Rfid Calibration* (LANDMARC) proposto por Ni et al. (2003), visto que esse sistema é citado na grande maioria das fontes que retratam a utilização de RFID para localizar objetos em ambientes fechados.

O LANDMARC faz uso de tags RFID ativas para determinar o local das tags que serão localizadas, essas tags ativas são colocadas em pontos já conhecidos pelo sistema e servem como pontos de referência e assim diminuir o número de leitores e ter uma melhor precisão no ambiente (SANPECHUDA; KOVAVISARUCH, 2008).

Através das tags ativas é possível se obter informações com relação a intensidade do sinal, essa informação é utilizada para calibrar a distância para as tags de rastreamento por meio de uma soma com o peso atribuído às tags de referências mais próximas, é importante ressaltar que a precisão dos resultados depende da forma que as tags de referência são posicionadas (SANPECHUDA; KOVAVISARUCH, 2008).

Figura 12 – LANDMARC.



Fonte: (NI et al., 2003)

Na Figura 12 mostra a aplicação sendo utilizada em um ambiente real por Ni et al. (2003), em que os retângulos cinzas são leitores RFID, os retângulos brancos são tags de referências e os pontos pretos são tags a serem localizadas, e dessa forma o sistema calcula as coordenadas dos objetos.

2.4 Algoritmos de Localização indoor

Nesta seção serão abordados alguns dos algoritmos que já foram utilizados para localização em ambientes indoor.

2.4.1 Multilateração

A multilateração estima a coordenadas do nó de destino a partir das distâncias do nó de destino para o nó de referência que possui coordenadas conhecidas, é o mesmo que ocorre na trilateração, porém a multilateração pode-se utilizar 2,3 ou n nós de referências, é um método utilizado para suprir a trilateração. A inserção de mais nós tem por finalidade aumentar a precisão e diminuir a região de incerteza (ZHOU, 2009).

Segundo Zhou (2009) os cálculos são executados da seguinte forma, havendo n nós de referência, que serão representado por R_k , $k = [1, 2, \dots, n]$ com suas coordenadas já conhecidas

(x_k, y_k) , T representando o nó alvo com coordenadas desconhecidas (x, y) , podemos estimar utilizando a fórmula da distância da geometria analítica:

$$\begin{cases} r_1^2 = (x - x_1)^2 + (y - y_1)^2 \\ r_2^2 = (x - x_2)^2 + (y - y_2)^2 \\ \dots \\ r_n^2 = (x - x_n)^2 + (y - y_n)^2 \end{cases} \quad (2.1)$$

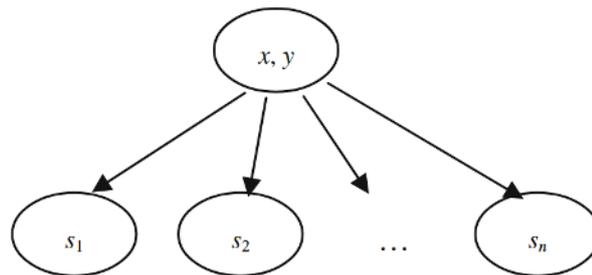
Depois subtraímos cada uma das equações a partir da primeira para denotar $b_{i1} = \frac{1}{2}(x_1^2 - x_i^2 + y_1^2 - y_i^2 + r_1^2 - r_i^2)$, sendo que $i = [2, 3, \dots, n]$ e em seguida linearizar o sistema (Equação 2.2) que também pode ser escrito no formato de matriz $b = AX$. Esse algoritmo requer pouca computação e está em uso em muitos sistemas de localização.

$$\begin{cases} b_{21} = x(x_1 - x_2) + y(y_1 - y_2) \\ b_{31} = x(x_1 - x_3) + y(y_1 - y_3) \\ \dots \\ b_{n1} = x(x_1 - x_n) + y(y_1 - y_n) \end{cases} \quad (2.2)$$

2.4.2 Inferência Bayesiana

Segundo [Castro et al. \(2001\)](#) a inferência bayesiana é dedução estatística através de evidências ou análise observatória para que a inferência de uma hipótese possa vir a ser verdadeira, e no caso da localização de um dispositivo isso pode ser realizado utilizando RSS entre o nó de destino e os pontos de acesso.

Figura 13 – Rede Bayesiana



Fonte: ([ZHOU, 2009](#))

Na [Figura 13](#) temos uma representação de uma rede bayesiana para localização estacionária, onde as coordenadas (x, y) representam o nó que será localizado e o $S_i, i = 1, \dots, n$ são uma série de intensidades de sinais transmitidas ou recebidas pelos pontos de referências do sistema, como nesse sistema as probabilidade de S_i são independentes e não interferem uma na outra dizemos que há uma satisfação em relação a condição de Markov ([ZHOU, 2009](#)).

Zhou (2009) afirma que a localização do dispositivo alvo pode ser obtida através da fórmula recursiva $P((x, y)|s_1, s_2, \dots, s_n) = \alpha P(s_n|(x, y)) \times P((x, y)|s_1, s_2, \dots, s_{n-1})$, nesta expressão α é um fator de normalização para que a soma da probabilidade posterior, no caso a primeira parte da expressão $P((x, y)|s_1, s_2, \dots, s_n)$ seja uma e $(s_n|(x, y))$ calcule a probabilidade da RSSI dada para a localização do dispositivo, é importante ressaltar que essa é uma expressão para localizar alvos estáticos.

2.4.3 K-Nearest-Neighbor

Os algoritmos de vizinhança ou no caso de *nearest-neighbor* tem uma abordagem bem simples que envolve utilizar os nó já classificados para classificar os novos nós a partir da medida de similaridade (XIE et al., 2016), de outro modo, utilizar-se dos seus vizinhos para assim obter suas coordenadas com os cálculos baseados em RSS (ZHOU, 2009). Algumas das métricas abordada por Xie et al. (2016) são:

- Distância Euclidiana

$$Dist(r, s) = \sqrt{(x_r - x_s)(x_r - x_s)'}$$

- Padronização da Distância Euclidiana

$$Dist(r, s) = \sqrt{(x_r - x_s)D^{-1}(x_r - x_s)'}$$

- Distância de Mahalanobis

$$Dist(r, s) = \sqrt{(x_r - x_s)V^{-1}(x_r - x_s)'}$$

- Distância de Manhattan

$$Dist(r, s) = \sum_{j=1}^n |x_{rj} - x_{sj}|$$

- Distancia de Minkowski

$$Dist(r, s) = \sqrt[p]{\sum_{j=1}^n |x_{rj} - x_{sj}|^p}$$

Zhou (2009) faz mostra uma outra abordagem para calcular as coordenadas do alvo obtidos a partir do sistema a seguir:

$$\begin{cases} x = \sum_{i=1}^k w_i x_i \\ y = \sum_{i=1}^k w_i y_i \end{cases} \quad (2.3)$$

Nesse sistema o k representa a quantidade de vizinhos, (x_i, y_i) são as coordenadas dos pontos de referências vizinhos e w_i os pesos desses pontos, o cálculo dos pesos é feito a partir da diferença de RSSI entre os pontos de referências e o alvo e as fórmulas para tal são:

- $w_i = \frac{1/\sum_{j=1}^m |s_{ij} - s_j|}{\sum_{j=1}^k (1/\sum_{j=1}^m |s_{ij} - s_j|)}$

$$\bullet w_i = \frac{1/\sqrt{\sum_{j=1}^m (s_{ij}-s_j)^2}}{\sum_{j=1}^k (1/\sqrt{\sum_{j=1}^m (s_{ij}-s_j)^2})}$$

nas expressões de w_i , dizemos que s_j e s_{ij} , $j = 1, \dots, m$ representam RSSI dos respectivos ponto de referências e nó alvo que será localizado.

2.4.4 Proximidade

A técnica de Proximidade, segundo [Bouet e Santos \(2008\)](#) baseia-se no princípio de que se o alvo está no alcance de uma antena ou leitor, sua localização será a mesma que a da antena e se o alvo está no alcance de duas ou mais antenas, a localização dada é aquela cujo possui a intensidade do sinal mais forte. Já para [Zhou \(2009\)](#) quando o alvo está no alcance de dois ou mais receptores, utiliza-se o cálculo de centroide para estimar a localização.

2.5 Node.js

De acordo com [Tilkov e Vinoski \(2010\)](#) Node.js é um ambiente para desenvolvimento *server-side* que utiliza JavaScript como linguagem de programação. É baseado no V8 um interpretador JavaScript desenvolvido pela Google e ambos são implementados em C e C++ para ter baixo consumo de memória e alta performa, isso só é possível pois antes de executar, o código é compilado para linguagem nativa para em seguida ser executado.

Node.js é uma plataforma escalável permitindo a programação direta com os protocolos de rede ou a utilização de bibliotecas para acessar os recursos do sistema operacional. Uma outra vantagem do Node.js é que ele possui seu próprio gerenciado de pacotes o NPM (*Node Package Manager*), através dele é possível instalar, remover, listar ou atualizar novos módulos ([PEREIRA, 2014](#)).

Alguns módulos do Node.js:

- Mongoose: é uma biblioteca que tem como finalidade a modelagem de objetos do MongoDB;
- Express: é uma biblioteca que fornece ferramentas e recursos para desenvolver aplicações web e sistema de roteamento;
- Body-parser: é um analisador de requisição sob a entrada *req.bory*;
- EJS: é uma linguagem de *templates* quer permite gerar marcações HTML com JavaScript;
- Moment: é uma biblioteca para manipulação, validação, analisa e formatar datas;
- Fs: é uma biblioteca para manipulação e criação de arquivos;

- Nodemailer: é uma biblioteca para envio de e-mail a partir da aplicação.

2.6 MongoDB

Segundo (ABRAMOVA; BERNARDINO, 2013) o MongoDB é um sistema de gestão de base de dados(SGBD) NoSQL (não relacional), *open source* e multiplataforma. Foi desenvolvido utilizando a linguagem de programação C++ em 2007. O MongoDB utiliza o formato BSON (Binary JSON) para armazenar os documentos, tais documentos são agrupados em coleções de acordo com sua estrutura e a identificação é feita por meio de ID, também podendo ser feita de maneira combinatória com ID e *timestamp*.

MongoDB é um SGBD ágil e muito escalável, o seu nome vem da palavra “**humongous**” que enfatiza a escalabilidade e alto desempenho que o MongoDB fornece. Como é baseado NoSQL descarta o uso de colunas e linhas como em banco de dados relacional e basicamente faz uso de objetos como modelo (DAYLEY, 2014).

2.7 Modelagem de Sistemas

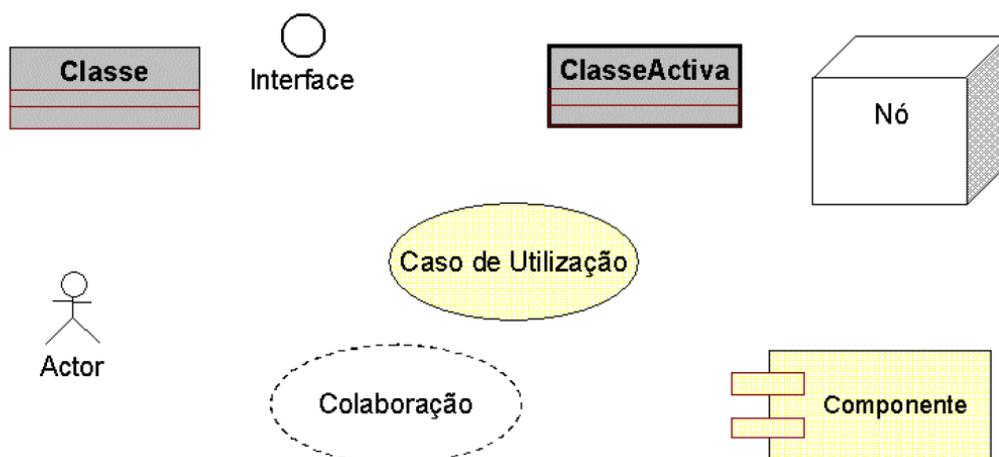
Esta seção tem por finalidade apresentar as forma cujo os sistemas computacionais são modelados de maneira abstrata, o processo de modelagem de sistema tem papel importante pois ajuda a entender e extrair os requisitos necessários para a construção do sistema.

2.7.1 UML - *Unified Modeling Language*

O UML (em português Linguagem de Modelagem Unificada) é uma linguagem cujo principal papel é a construção, visualização e documentação de projetos de sistemas. Surgiu na década de 90, tendo em vista que havia a necessidade de uma linguagem para modelagem que funcionasse como uma norma e fosse aceita e utilizada pela indústria, ambientes acadêmicos e de investigação (SILVA, 2001).

Na UML é possível agrupar elementos e relacioná-los de maneira lógica ou estrutural, isso é o conceito de diagramas. Os elementos têm um papel importante nos modelos, eles estarão organizados de acordos com a função desempenhada no sistema, podendo representar componentes do sistema, usuário, interfaces (SILVA, 2001).

Figura 14 – Elementos UML



Fonte: (SILVA, 2001)

Na [Figura 14](#) está um resumo de alguns elementos básicos que é utilizados em modelos de projetos de software, esses componentes por mais que sejam básicos, possibilitam a representação de vários componentes em um projeto. A seguir uma breve descrição dos diagramas comuns da UML (NUNES; O'NEILL, 2011; SILVA, 2001).

- **Diagrama de Caso de Uso** - Descreve as funcionalidades do sistema, ou seja nesse diagrama é apresentado o que o sistema deve fazer e os serviços que serão disponibilizados para os utilizadores.
- **Diagrama de Classe** - Descreve a estrutura do sistema, como o modelo geral de informações utilizando orientação a objetos.
- **Diagrama de Estados** - Descreve o comportamento dos objetos, as representação dos possíveis estados de um objeto e eventos que alteram valores de atributos dos objetos desencadeando as transições.
- **Diagrama de Sequência** - Descreve as interações dos objetos no sistema em determinado período de tempo. Essa interação é realizada por meio de troca de mensagens síncrona ou assíncrona e cada objeto possui uma linha temporal.
- **Diagrama de Componentes** - Descreve interações entre objetos dando ênfase maior as ligações entre objetos e numerando as mensagens dessa forma não apresenta o tempo como no diagrama de sequência.
- **Diagrama de Estados** - Descreve o comportamento dos objetos, as representação dos possíveis estados de um objeto e eventos que alteram valores de atributos dos objetos desencadeando as transições.

- **Diagrama de Atividade** - Descreve o fluxo de trabalho que classifica os estados de "estados de atividades", onde e realizados a transições quando há a conclusão das atividades dos estados anteriores.
- **Diagrama de Sequência** - Descreve as interações dos objetos no sistema em determinado período de tempo. Essa interação é realizada por meio de troca de mensagens síncrona ou assíncrona e cada objeto possui uma linha temporal.
- **Diagrama de Componentes** - Descreve interações entre objetos dando ênfase maior as ligações entre objetos e numerando as mensagens dessa forma não apresenta o tempo como no diagrama de sequência.

3 TRABALHOS CORRELATOS

Este capítulo tem como objetivo apresentar os principais trabalhos relacionados com o sistema proposto neste trabalho, exaltando as diferenças e contribuições para o método proposto. Os trabalhos em específico abordados utilizam *Radio-Frequency IDentification*(RFID) ou referenciam sistemas que fazem uso de tal ferramenta para localizar e identificar objetos em ambientes fechados.

3.1 *An indoor localization mechanism using active RFID tag*

O trabalho de [Jin et al. \(2006\)](#) faz uma análise do método utilizado no sistema LANDMARC que é um sistema de localização por meio de tag RFID. As tags de referência são tags ativas colocadas em pontos fixos no ambiente que servem como pontos de referência para o sistema e calibração da localização, esse sistema utiliza tags ativas visando diminuir a quantidade de leitores RFID para baratear o sistema.

Segundo [Jin et al. \(2006\)](#) o LANDMARC funciona da seguinte maneira: primeiro os leitores detectam as tags RFID que estão no seu alcance, após isso é concedido ponderação as tags de referências de acordo com RSS, sendo valor mais alto para as mais próximas. O passo seguinte é realizar o cálculo da distância euclidiana na RSS entre uma tag de rastreamento e uma tag de referência e quanto menor o resultado mais próximo da tag de referência. Por fim o sistema escolhe k tags com os menores valores no resultado do cálculo e obtém a localização da tag de rastreamento por meio da seguinte fórmula.

$$(x, y) = \sum_{i=1}^k w_i(x_i, y_i), w_j = \frac{\frac{1}{E_i^2}}{\sum_{i=1}^k \frac{1}{E_i^2}} \quad (3.1)$$

[Jin et al. \(2006\)](#) mostra que LANDMARC possui alguns problemas, sendo efetuando cálculos desnecessários durante a escolha das tags vizinhas e comparação de valores RSS de leitores que não estão no alcance da tag alvo. Para melhorar o sistema, [Jin et al. \(2006\)](#) propõe a utilização apenas dos leitores RFID e tags de referência que alcançam o alvo, sendo que o número de leitores e das tags de referência tem que ser maior que 3, pois depois o método consiste em utilizar a técnica de Triangulação.

O método utilizado por [Jin et al. \(2006\)](#) tem objetivo de ter alcançado grande precisão no posicionamento em relação ao ambiente, já o método proposto neste trabalho tem objetivo de saber exatamente a sala que os objetos estão, tendo em vista que o ambiente possui inúmeras salas. Entretanto essa aplicação para posicionamento na sala pode ser acrescentado e estudada

em trabalhos futuros, visto que neste primeiro momento do sistema proposto neste TCC visa-se um sistema de baixo custo com o uso de tags passivas.

3.2 *Real-Time Locating Systems Using Active RFID for Internet of Things*

O trabalho de [Zhang et al. \(2016\)](#) propõe um sistema de localização em tempo real, o iLocate que é mais um sistema que faz uso de tags ativas, e utilizar-se de ZigBee ([GISLASON, 2008](#)) para assim assegurar que a transmissão de dados a longa distância seja garantida.

O sistema proposto por [Zhang et al. \(2016\)](#) inicialmente aplica uma técnica de *fingerprinting* que seria uma técnica equivalente ao reconhecimento de padrões abordada em [subseção 2.3.2](#), iniciando com leituras breves de RSSI e implantação de tags de referências no mapa de interesse, seguindo para o armazenamento das leituras no banco de dados e construção de uma matriz com dados de RSSI.

O iLocate conta com um refinamento para ter uma localização mais precisa, que funciona criando tags de referência virtuais que também utiliza a técnica de *fingerprinting* para comparar com as reais. Outro recurso encontrado no iLocate é a comunicação tag-tag em que possibilita a troca de mensagens entre tags de referências que estão no mesmo intervalo de trabalho ([ZHANG et al., 2016](#)).

A utilização de uma base de dados por [Zhang et al. \(2016\)](#) é um recurso que também será utilizado no sistema proposto neste trabalho, por mais que os dados armazenados sejam diferentes o banco de dados terá função importante para controle e monitoramento dos objetos que irão utilizar as tags passivas de RFID.

3.3 *Object localization using RFID*

Na pesquisa de [Chawla et al. \(2010\)](#) é proposto um método de localização que permite estimar a posição de objetos com rapidez e velocidade utilizando a variação dos níveis de potência dos leitores, o método ainda faz uso de tags de referência (tags ativas) para auxiliar na localização das tags passivas que serão rastreadas.

O método proposto por [Chawla et al. \(2010\)](#) foi aplicado em uma sala cuja as dimensões são 2m x 3m. A região que assemelha a um retângulo foi dividida em oito sub-regiões de tamanho iguais e denominadas de setores, em seguida cada setor foi dividido em quatro regiões de tamanho iguais chamados de quadrantes.

Depois de fazer a divisão do ambiente é passado para inserção de tags de referências em cada quadrante, essas tags de referência são utilizadas inicialmente para calibrar os leitores RFID com uma relação de potência em contraste com distância. Por fim é utilizados algoritmos

que ficam variando o nível de potência do leitor, esses algoritmos podem iniciar a variação da potência do menor para maior até alcançar um nível mínimo para localizar o objeto ou podendo utilizar uma variação da potência do maior para o menor (CHAWLA et al., 2010).

O método Chawla et al. (2010) diferencia-se do método proposto no caso de buscar uma localização e posicionamento preciso em ambiente fechado, o que não se torna uma prioridade para o método proposto neste trabalho, sendo inicialmente suficiente para o sistema proposto a localização para controle de inventário.

3.4 *RFID localization algorithms and applications-A review*

O artigo de Zhou (2009) tem uma abordagem diferente dos trabalhos anteriores, não propondo nenhum método de localização mas mostrando as aplicações existentes, revisando algoritmos de localização indoor e potenciais da localização de RFID. RFID pode ser empregado em várias aplicações, de acordo com Zhou (2009) RFID é utilizado no porto de Cingapura para posicionar os contêiner, na saúde para localização de pacientes, suprimentos e equipamentos e também na gestão de material de construção permitindo monitoramento do andamento e status da construção.

Em Zhou (2009) é realizado uma revisão dos algoritmos de localização classificando-os em dois grupos, o grupo que calibra a distribuição do sinal de radiofrequência e depois estima a posição do objeto e o segundo grupo que calculam diretamente a posição do objeto utilizando dados de RSSI. Dentro do primeiro grupo encontramos os algoritmo de multilateração e inferência bayesiana, já no segundo grupo estão os algoritmos de aprendizados em proximidade (ZHOU, 2009). O artigo de Zhou (2009) serve de base para este trabalho de TCC em relação a definição de conceitos e estratégias para o posicionamento de RFID em sistemas de localização.

4 MÉTODO PROPOSTO

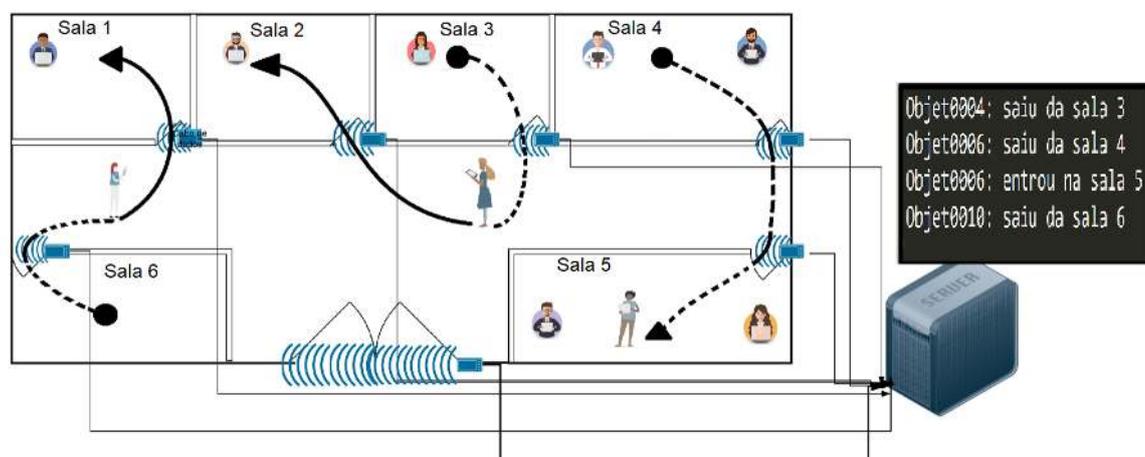
Neste capítulo descreve-se o projeto e fluxo de execução do sistema computacional, denominado de INEXT, que utiliza leitores e etiquetas RFID passivas, para o gerenciamento e controle de ativos, sendo capaz de localizar e identificar objetos etiquetados em edifícios. Esta solução computacional proposta foi pensada para ser aplicada para o controle e levantamento de patrimônio da Universidade Federal de Roraima e está disponível no seguinte repositório do GitHub <<https://github.com/luarkian/nodeArd-INEXT.git>>.

4.1 Visão geral do método proposto

O método proposto consiste em um sistema que localiza qualquer objeto, considerado como patrimônio, que contenha uma etiqueta RFID passiva fixada em seu corpo, as etiquetas são localizadas e identificadas a medida em que transitam de uma sala para outra no edifício. A localização e identificação é feita por meio de leitores RFID e micro-controladores colocados próximos às portas da sala de um prédio, assim os leitores devem ler as tags que passam pela porta, bem como, o módulo wireless irá propagar esta informação via comunicação com um servidor web.

Cada leitor RFID junto ao controlador é responsável por uma sala, portanto qualquer etiqueta lida por aquele leitor ocasionará nas ações referentes a mesma sala. As ações serão de atualização sobre a localização do objeto ou informando que o objeto está em transição para outra sala no edifício.

Figura 15 – Modelo Proposto



Fonte: Própria

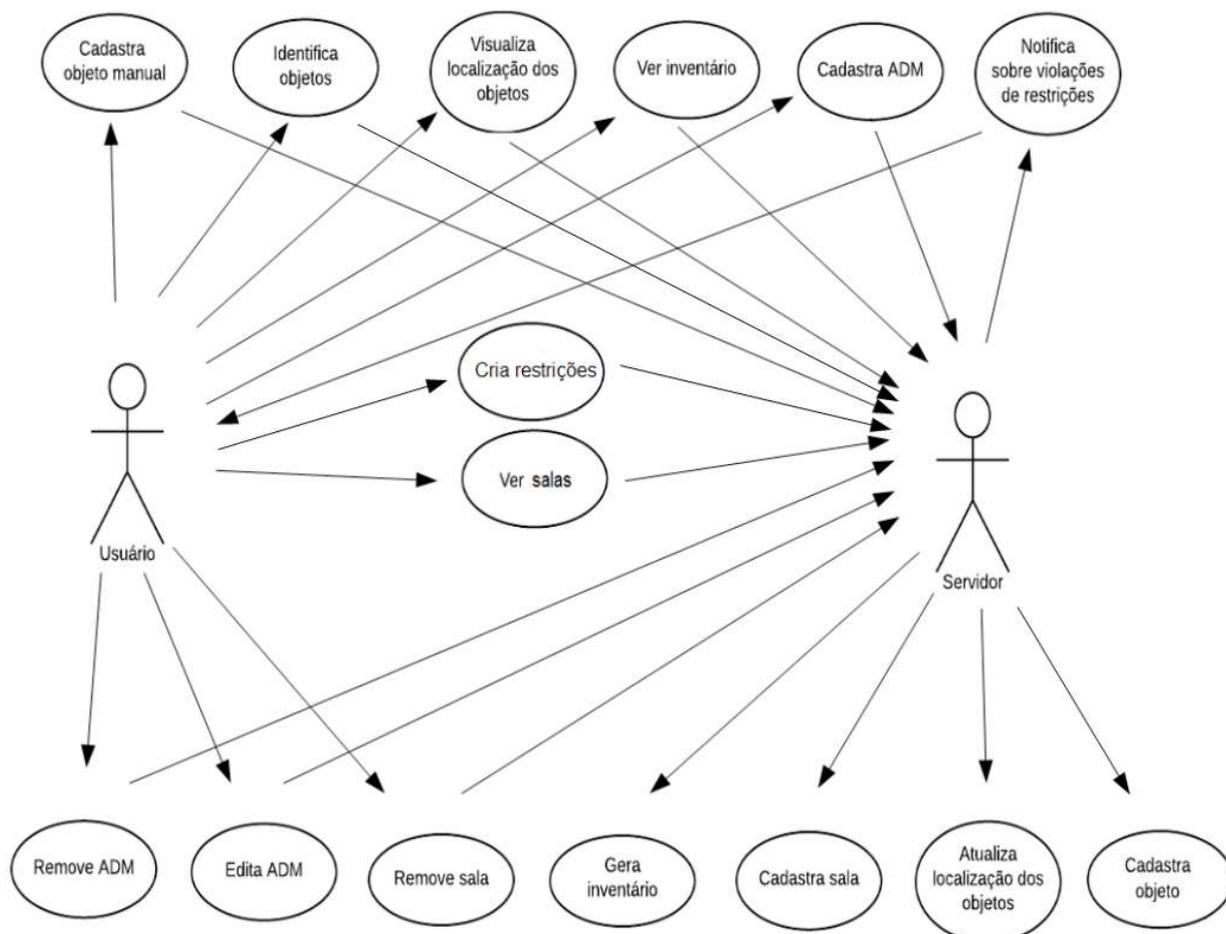
Na [Figura 15](#) é possível notar que há um dispositivo acoplado próximo a porta de cada sala, esse dispositivo contém o leitor RFID e será encarregado de consultar o servidor para assim tomar decisões do que será feito com o status do objeto, como cada objeto possui uma tag RFID fixada nele e quando o mesmo passar pela porta o leitor identifica o objeto e altera suas informações sobre sua localização na base de dados no servidor web.

O método proposto consiste nas seguintes etapas:

1. Identificação dos objetos que serão rastreados;
2. Monitoramento e localização dos objetos rastreáveis em ambiente indoor;
3. Configuração de controle e monitoramento de transição dos objetos; e
4. Levantamento de todos objetos etiquetados cadastrados no sistema.

As funcionalidades do sistema e a interação com o usuário pode ser visualizada na [Figura 16](#) e que seguem o seguinte fluxo: as setas partindo do ator usuário até o módulo representa funções que são realizadas pelo próprio usuário do sistema; as setas que saem dos módulos e chegam no servidor são as operações que são solicitadas e que o servidor executa; e as setas que saem do servidor e chegam nos módulos representa operações que o servidor executa de forma automática.

Figura 16 – Caso de uso



Fonte: Própria

4.2 Identificação dos objetos rastreáveis

Esta etapa consiste na identificação dos objetos. Primeiramente, as etiquetas RFID passiva são anexadas em cada objeto, essas etiquetas trabalham na frequência $13,56MHz$, um exemplo de objeto com uma etiqueta anexada em seu corpo pode ser visto na [Figura 17](#).

Figura 17 – Objeto com tag RFID



Fonte: Própria

O segundo passo é realizar o cadastramento dos objetos utilizando algum dispositivo de porta, fazemos com que o dispositivo faça a leitura da tag do objeto, dessa forma o sistema realizará o cadastro da etiqueta porém todas as informações referente aquela etiqueta irão constar no sistema como **Desconhecido**, assim como a [Figura 18](#) mostra. Vale ressaltar que para facilitar este processo de cadastro, as etiquetas RFID podem ser previamente cadastradas no sistema com o auxílio do dispositivo de porta, sem a necessidade de ter o objeto presente, para então em outro momento apenas efetuar a adição das etiquetas.

Figura 18 – Tela inicial



Nome	TagRFID	Descrição	Tombo	Localização (Sala)	Restrição	Ult. Alteração		
Desconhecido	56 6C 96 A5	Desconhecido	0	101			Editar	Remover

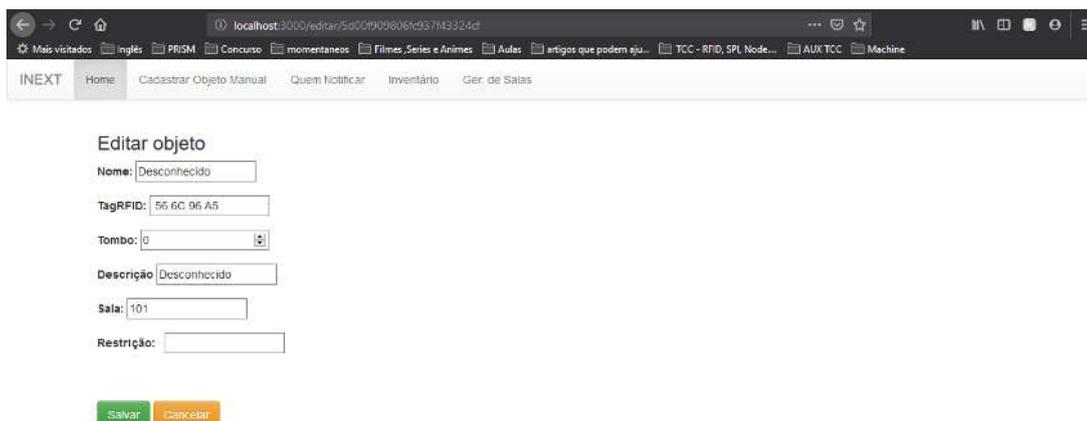
Objetos com restrições violadas:

Nome	TagRFID	Descrição	Tombo	Localização	Restrição
------	---------	-----------	-------	-------------	-----------

Fonte: Própria

Durante o cadastro dos objetos, o próximo passo é editar as informações de acordo com o objeto que constam como desconhecido no sistema. Na pagina inicial é listado todos objetos cadastrados e ao lado de cada objeto possui a opção **Editar** (ver [Figura 17](#)), localizamos o objeto que acabamos de realizar seu cadastro e selecionamos a opção de editar, logo seremos redirecionado para a pagina de edição de objeto e preenchemos o formulário de acordo com as características referente ao objeto desejado, após ter preenchido é necessário clicar no botão **Salvar** ao final do formulário, conforme [Figura 19](#).

Figura 19 – Tela para editar objeto



Editar objeto

Nome:

TagRFID:

Tombo:

Descrição:

Sala:

Restrição:

Fonte: Própria

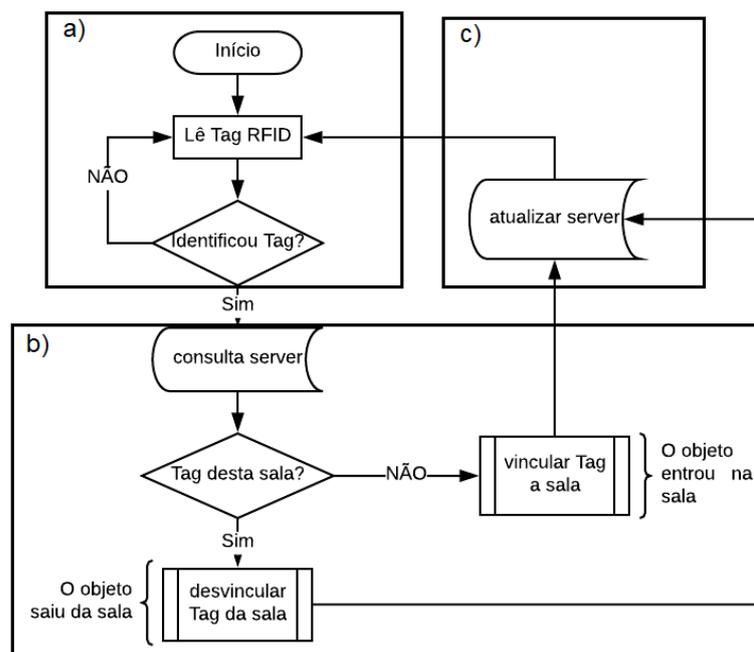
4.3 Monitoramento e Localização dos objetos rastreáveis

O monitoramento é realizado pelo módulo de porta contido em cada sala do edifício, tal módulo será descrito nas seções seguintes. No momento em que um objeto é identificado por um módulo de porta de uma sala e esse objeto tem suas informações atualizadas, significa que o objeto em questão está executando uma ação de entrada ou saída de alguma sala então o servidor deve atualizar seu status de localização.

4.3.1 Fluxo de execução do método em cada sala

O dispositivo de cada sala executa o seguinte fluxo de execução para monitoramento dos objetos da sala, conforme [Figura 20](#). No fluxograma da figura, os retângulos representam processos e os retângulos com bordas duplas processos pré-definidos, os losangos representam as tomadas de decisões, os ícones com bordas curvas para o mesmo lado representam operações nos dados do sistema e o ícone com bordas curvas opostas representam o início do fluxo.

Figura 20 – Fluxograma de cada sala



Fonte: Própria

O fluxograma é separado em três módulos: **a**, **b** e **c**. O módulo **a** é onde é realizado o monitoramento da da sala, realizando leituras dos objetos que entram e saem da sala. No módulo **b** é o onde são tomadas as decisões referente a etiqueta anexada ao objeto lida naquele momento, verificando se a etiqueta está ou não localizada naquela sala e processando as alterações necessárias. O módulo **c** é o último processo que atualiza as informações do servidor para que os dados consultados sejam salvos e criado os logs.

4.3.2 Localização dos objetos

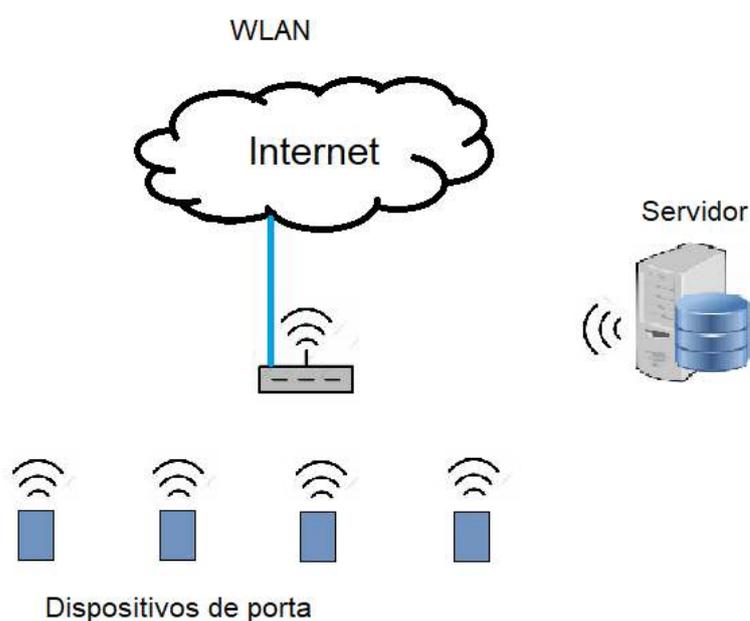
Os objetos são localizado no sistema baseado no algoritmo de proximidade (ZHOU, 2009), ou seja, a partir do momento em que o leitor RFID identifica uma etiqueta, esse objeto terá ou deixará de ter a localização referente àquele módulo de porta, cada módulo representa uma sala, portanto as informações referente a localização dos objetos são definidas de acordo com o dispositivo que executa a leitura da etiqueta anexada ao objeto.

O sistema possui um servidor web que para este trabalho foi desenvolvido utilizando o framework Node.js v10.15 disponível pra download em <<https://nodejs.org/en/download/>>, que deve está conectado à mesma rede LAN (*local area network*) que os dispositivos de porta. O servidor web também possui uma base de dados, sendo o sistema de gerenciamento de dados o MongoDB v3.6.5 disponível em <<https://www.mongodb.com/download-center/community>>, para armazenar informações e a descrição e informações de cada objeto. Logo, o servidor

web é responsável por tratar problemas relacionados as leituras das etiquetas, também sendo responsável por realizar a levantamento do inventário total e notificar os usuários sobre violações.

O Node.js e MongoDB foram as tecnologias escolhidas, pois fornece facilidades no desenvolvimento do *front-end*, *back-end* e operações com o banco de dados utilizando apenas uma linguagem de programação que é o JavaScript, o custo de hardware dessas tecnologias é baixo viabilizando ainda mais a implementação e pode ser realizado o *deploy* para outro sistema operacional com facilidade.

Figura 21 – Representação da rede



Fonte: Própria

4.4 Configuração de controle e monitoramento de transição dos objetos

Visando um melhor gerenciamento e rastreamento dos objetos no sistema proposto, todas as operações do sistema geram logs. Os arquivos contendo os logs são criados diariamente e os nomes desses arquivos são de acordo com a data, por exemplo, um log do dia 07 de junho de 2019 tem como seu nome a data 2019-06-07. Um arquivo de log pode ser visualizado na [Figura 22](#), no canto superior esquerdo da imagem tem o nome do arquivo e no seu corpo tem algumas das operações que ocorreram.

Figura 22 – Arquivo de log

```

2019-06-12.txt
1 Objeto criado, tagRFID: 56 6C 96 A5 time:08:59
2 Objeto id: 5d00f727d8faa62824eed487 foi excluído, time: 09:07
3 Objeto criado, tagRFID: 56 6C 96 A5 time:09:07
4 Objeto TagRFID: 56 6C 96 A5, id: 5d00f909806fc937f43324cf, nome: Desconhecido, descrição: undefined, tombo: 0,
localização (sala): 101, restrição: , foi modificado para: [nome, tagRFID, descrição, tombo, localização(sala),
restrição][Cadeira, 56 6C 96 A5, cadeira azul, 20180619, 101, ], time: 09:12
5 Objeto TagRFID: 56 6C 96 A5 id: 5d00f909806fc937f43324cf saiu da sala: 101 time: 09:20
6 Objeto TagRFID: 56 6C 96 A5 id: 5d00f909806fc937f43324cf entrou na sala: 101 time: 09:20
7 Objeto TagRFID: 56 6C 96 A5 id: 5d00f909806fc937f43324cf saiu da sala: 101 time: 09:22
8 Objeto TagRFID: 56 6C 96 A5 id: 5d00f909806fc937f43324cf entrou na sala: 101 time: 09:22
9 Objeto TagRFID: 56 6C 96 A5 id: 5d00f909806fc937f43324cf saiu da sala: 101 time: 09:23
10 Objeto TagRFID: 56 6C 96 A5 id: 5d00f909806fc937f43324cf entrou na sala: 101 time: 09:23
11
    
```

Fonte: Própria

O sistema também conta a opção de definir restrições aos objetos monitorados, exemplo, um objeto que no campo de restrição tem o nome de uma sala, ao sair dessa sala o sistema identifica esta violação de condição e comunica os administradores enviando um e-mail com os dados do objeto que violou a restrição. Na Figura 23 é mostrado como o objeto com restrição é mostrado no sistema e o e-mail enviado. Essa função é importante para objetos que não podem ser levados para outras salas, tendo que permanecer em uma única sala.

Figura 23 – Restrição

The screenshot shows a web application interface with a table of objects and a corresponding email notification. The table lists objects with columns for Name, TagRFID, Description, Tombo, Location (Room), Restriction, and Last Change. Below the table, there is a section for 'Objetos com restrições violadas:' which highlights a specific object. The email notification, received from cruiserweights@zoho.com, details the violation for the object 'Cadeira' (Chair) with TagRFID 56 6C 96 A5, located in room 101, which violated restriction 202.

Nome	TagRFID	Descricao	Tombo	Localização (Sala)	Restrição	Ult. Alteração
Cadeira	56 6C 96 A5	cadeira azul	20180619	101	202	2019-06-12

Nome	TagRFID	Descricao	Tombo	Localização	Restrição
Cadeira	56 6C 96 A5	cadeira azul	20180619	101	202

Fonte: Própria

4.5 Conexão dos dispositivos de porta com o servidor

A comunicação no sistema entre os dispositivos da porta com o servidor web acontece por meio de uma WLAN, ou seja todos devem estar em uma rede local e assim poderem se comunicar, entretanto a comunicação será diretamente entre um dispositivo e o servidor, não havendo comunicação entre dois dispositivos. Neste trabalho, foi utilizado a placa NodeMcu ESP-12E, *datasheet* disponível em <https://www.electrodragon.com/w/ESP-12F_ESP8266_Wifi_Board>, que possui um módulo ESP-8266, esse módulo possibilita a conexão com uma rede wireless 802.11 b/g/n, esse padrão possibilita conexões com velocidade de até 300Mbps em um espectro de 2.4GHz ou 5GHz.

Os dispositivos de porta, ao realizar a leitura de um etiqueta, geram um objeto JSON (*JavaScript Object Notation*), esse objeto contém o nome da sala e o identificado da tag RFID, depois de criado o objeto é enviado para o servidor via requisição POST para executar as operações e modificar o status do objeto que esta com a etiqueta anexada.

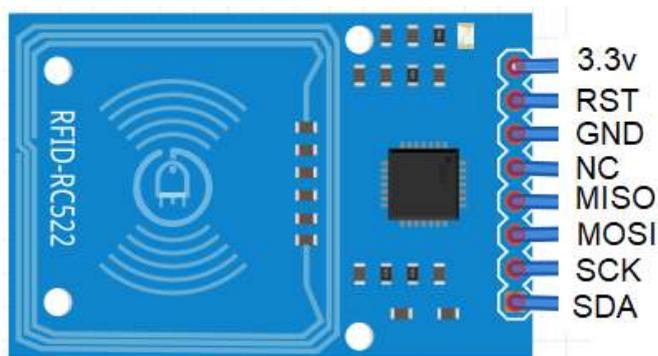
```
1 // Exemplo de um JSON gerado
2 {
3   "sala": "101",
4   "tagRFID": " 56 6C 96 A5"
5 }
```

O NodeMcu opera em uma voltagem de 5V de energia contínua, um consumo relativamente baixo em relação as outros dispositivos eletrônicos e as etiquetas passivas não possuem alimentação, elas são alimentadas apenas pela energia eletromagnética emitida pelos leitores RFID, esses dois quesitos tornam o protótipo ainda mais viável, pois isso é um fator positivo visto que o custo pós implantação não é alto em relação ao consumo de energia.

4.6 Dispositivos de porta

O protótipo desenvolvido neste trabalho comumente chamado de dispositivo da porta, é composto por uma placa NodeMcu e um leitor RFID RC522 Mifare, *datasheet* disponível em <<https://www.nxp.com/docs/en/data-sheet/MFRC522.pdf>>. O leitor RFID pode ser visto na **Figura 24**, esse leitor é capaz de ler etiquetas que operam em frequências de 13,56MHz. Vale ressaltar que o protótipo visa avaliar a aplicabilidade do sistema proposto, visto que o leitor RFID pode ser facilmente substituído por outro de maior alcance de leitura das etiquetas. O código para o NodeMcu se encontra no repositório do GitHub <<https://github.com/luarkian/nodeArd-INEXT.git>> dentro da pasta `prototype_nodemcu` e antes de embarcar o código deve ser colocado o endereço IP do servidor web na variável `url`.

Figura 24 – Leitor RFID RC522

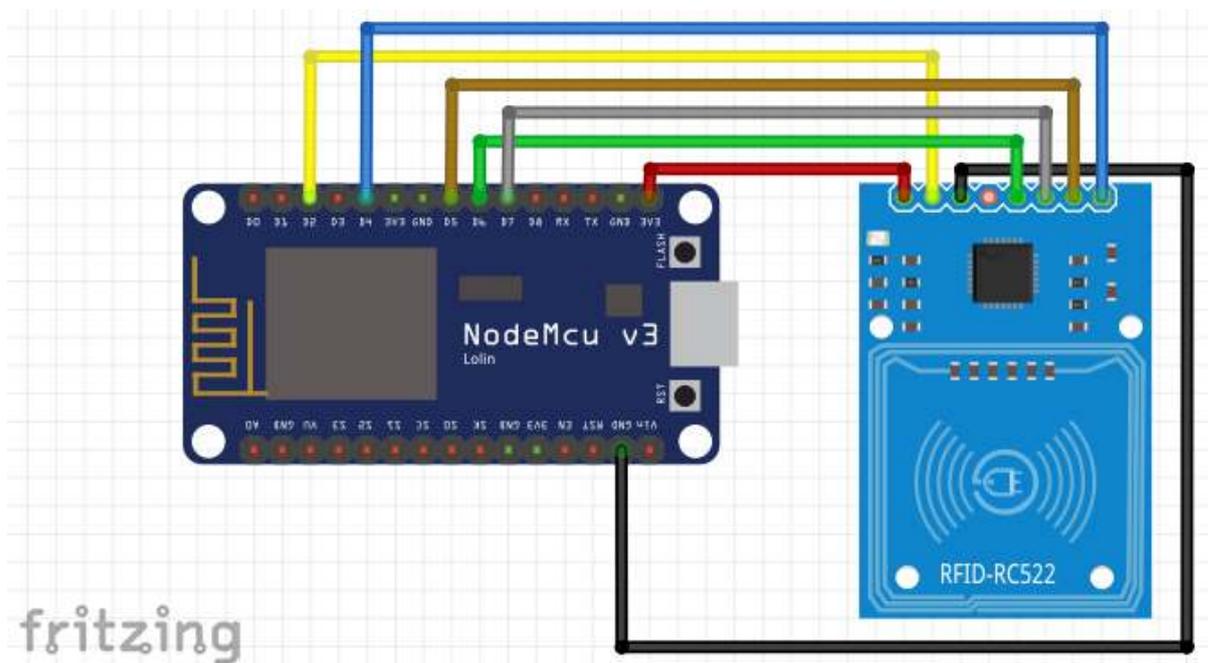


Fonte: Fritzing

O leitor RFID utiliza a interface SPI (*Serial Peripheral Interface*) para comunicação com o NodeMcu, essa conexão é síncrona e é realizada por meio dos pinos de **D2**, **D4**, **D5**, **D6** e **D7**. A prototipagem entre o NodeMcu e o leitor RFID segue o esquema na [Figura 25](#):

- 3.3V - conectado ao pino de 3.3v no NodeMcu, essa conexão faz a alimentação do leitor RFID;
- RST (*Reset*)- conectado ao pino D2 do Arduino;
- GND (*Graduated Neutral Density filter*) - conectado ao pino GND do NodeMcu;
- NC/IRQ (*Interrupt Request*)- não utilizado;
- MISO (*Master In Slave Out*) - conectado ao pino D6;
- MOSI (*Master Out Slave In*) - conectado ao pino D7;
- SCK (*Serial Clock*) - conectado ao pino D5; e
- SDA/SS (*Serial Data Line/ Select Slave*) - conectado ao pino D4.

Figura 25 – Esquema de Conexões



Fonte: Própria

4.7 Servidor Web

O servidor web foi desenvolvido utilizando a tecnologia Node.js *v10.15* utilizando a linguagem de programação JavaScript e o sistema de gerenciamento de banco de dados orientado a documento MongoDB *v3.6.5*. Através do Node.js e seu gerenciado de pacotes NPM foram instaladas as seguintes bibliotecas e suas respectivas versões. Todas as bibliotecas instaladas podem ser encontradas nas dependências no arquivo *package.json* que se encontra no repositório deste trabalho.

- Express *v4.16.4* - foi utilizada para criar as rotas e instanciar o serviço HTTP;
- Express-ejs-layouts *v2.5* - foi utilizada pra acessar as *views*;
- Mongoose *v3.1.13* - foi utilizada para modelar os objetos;
- Nodemailer *v6.1.1* - foi utilizado para o envio de e-mails a partir da aplicação;
- EJS *v2.6.1* - foi utilizada para a criação de *views* do *front-end*;
- Consign *v0.1.6* - foi utilizado para realizar o carregamento de *scripts* automaticamente;
- Moment *v2.24* - foi utilizada para a manipulação de datas e hora.

O sistema do servidor web possui as seguintes telas:

- Tela Inicial
- Tela de Cadastro de Objeto Manual
- Tela de Editar Objeto
- Tela de Quem Notificar
- Tela de Editar ADM
- Tela de Inventário
- Tela de Ger. Sala

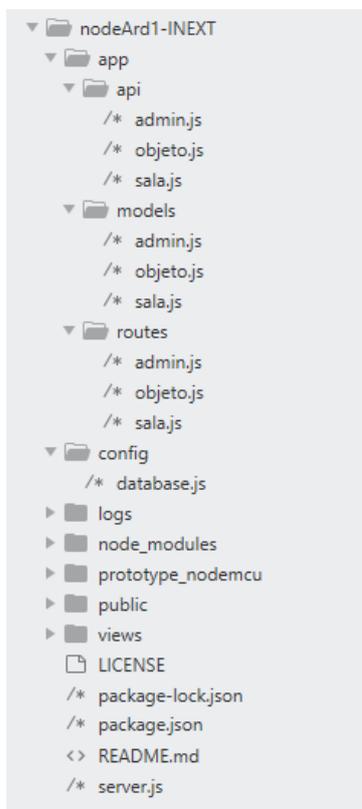
4.7.1 Implementação

Na codificação do sistema foi utilizada a linguagem de programação JavaScript no *back-end* e *front-end*, a linguagem de marcação HTML (*HyperText Markup Language*) para construção das telas e mecanismo CSS (*Cascading Style Sheets*) para modificar o estilo das páginas web, sendo que alguns estilos são fornecidos pelo framework *Bootstrap v3.3.4* disponível em <http://getbootstrap.com>.

4.7.1.1 Back-End

O *Back-End* deste sistema possui a estrutura mostrado na [Figura 26](#).

Figura 26 – Arquivos do Back-end



Fonte: Própria

Na pasta *config* possui o arquivo *database.js*, é neste arquivo em que estão as configurações do mongoose para realizar a conexão com o MongoDB. Dentro da pasta **app** há mais três pastas: **api**, **models** e **routes**. Os arquivos dentro da pasta **models** são para descrever os campos que os objetos JSON devem ter para serem salvos e manipulados no sistema. Na pasta **api** estão os arquivos com módulos que executam as operações do sistema e na pasta **routes** é onde são implementados os caminhos para acessar os módulos da pasta **api**. Assim como na pasta **models** as pastas **api** e **routes** também possuem cada uma os arquivos com os nomes *admin.js*, *objeto.js* e *sala.js*, porém os conteúdos dos arquivos contém a implementação dos módulos e dos caminhos.

Módulos dos arquivos *admin.js* contidos nas pastas **api** e **routes**.

- */admin/*, *api.lista*: rota e módulo que retornam todos os usuários cadastrados através de uma requisição GET;
- */admin_dados*, *api.criar*: rota e módulo para cadastrar administradores no sistema, deve ser enviado um JSON via requisição POST;
- */admin_edit/:id*, *api.buscarPorId*: rota e módulo que retorna ao administrador que for informado o ID na URL, é uma requisição GET;

- `/editarAdm/:id, api.atualizaDados`: rota e módulo para atualizar dados do administrador que for informado o ID na URL e um objeto JSON dever ser enviado com as novas informações via requisição POST; e
- `/removeAdm/:id, api.removePorId`: rota e módulo para exclusão do administrador que for informado o ID na URL, é uma requisição GET.

Módulos dos arquivos `sala.js` contidos nas pastas **api** e **routes**

- `/sala/, api.listar`: essa rota juntamente com esse modulo retornar as lista de todas as sala cadastradas, é uma requisição GET; e
- `/removeSala/:id, api.excluir`: essa rota e módulo excluem a sala cujo o ID é passado na URL, é uma requisição GET;

Módulos dos arquivos `objeto.js` contidos nas pastas **api** e **routes**

- `/:` essa rota não possui um módulo, seu módulo foi implementado no próprio arquivo de rotas, é uma requisição GET e retorna todos os objetos cadastrados no banco de dados;
- `/cadastrar, api.adiciona`: rota e módulo para cadastrar objeto no sistema, deve ser enviado um JSON via requisição POST;
- `/objetos, api.lista`: rota e módulo que retornam todos os objetos cadastrados através de uma requisição GET;
- `/verificacao, api.verificar`: rota e módulo que executa as operações de transições dos objetos, cria alertas e envia e-mail para os administradores, deve ser enviado um JSON via requisição POST;
- `/ver, api.ver`: rota e módulo destinada a teste para imprimir os JSON's recebidos no console, deve ser enviado um JSON via requisição POST;
- `/editar/:id, api.buscaId`: rota e módulo que retorna o objeto que for informado o ID na URL, é uma requisição GET;
- `/editar/:id, api.atualizar`: rota e módulo para atualizar dados do objeto que for informado o ID na URL e um objeto JSON dever ser enviado com as novas informações via requisição POST;
- `/remove/:id, api.removePorId`: essa rota e módulo excluem o objeto cujo o ID é passado na URL, é uma requisição GET; e
- `/inventario, api.inventario`: rota e módulo que retornam todos os objetos e salas cadastrados através de uma requisição GET.

O arquivo `server.js` na raiz do diretório é responsável por importar todos os arquivos mencionados anteriormente e iniciar o serviço *HTTP* na porta 3000. É importante ressaltar que antes de qualquer rota deve ser colocado o endereço IP que o servidor possui na rede WLAN.

5 AVALIAÇÃO EXPERIMENTAL

O presente capítulo consiste em apresentar o planejamento, execução, e análise dos resultados referente ao método proposto. Assim, visando avaliar de forma experimental o sistema computacional desenvolvido a partir do método proposto, denominado de INEXT, para identificação e localização de objeto em edifícios utilizando RFID.

5.1 Planejamento e projeto dos experimentos

Esta avaliação experimental consiste em avaliar o INEXT, tendo como questões principais a identificação e localização de objetos em ambientes confinados e então o gerenciamento dos objetos. Em virtude disso, foi definida as seguintes questões:

- QP1 : O sistema INEXT consegue identificar e rastrear os objetos em um âmbito confinado?
- QP2 : Quais vantagens o sistema INEXT prove em relação ao gerenciamento dos objetos em edifícios?
- QP3 : O sistema INEXT é capaz de gerar de forma automática o inventário de objetos em um âmbito confinado?

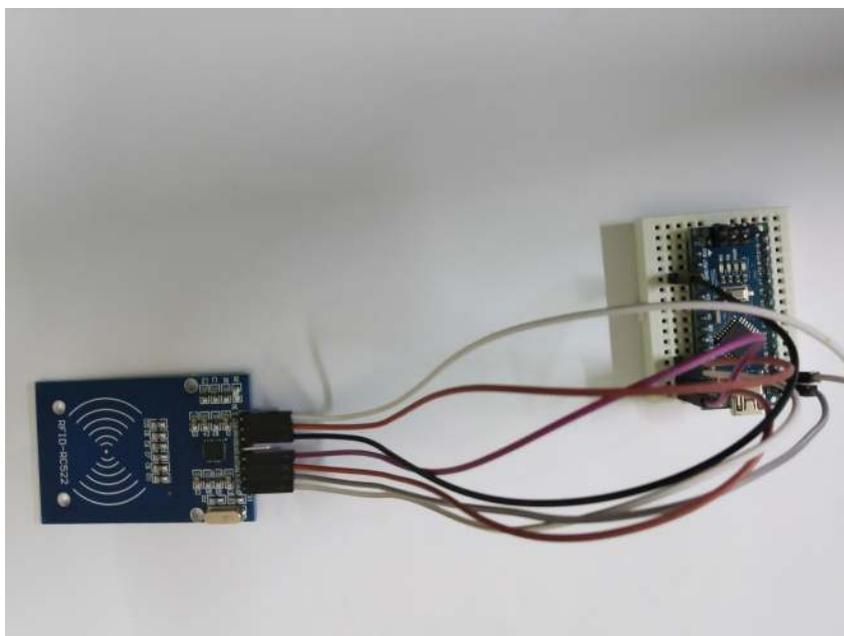
Objetivando responder tais questões de pesquisa, foram utilizados um Arduíno Nano V3.0 ATmega168 e um computador comunicando-se através da porta serial para simular a segunda sala, pois não dispomos de duas placas NodeMcu para a comunicação sem fio. Um script na linguagem Python v3.7.1 foi adotado para ler os dados da porta serial, criar um objeto JSON e enviar para o servidor, tais códigos estão disponíveis no repositório deste trabalho (endereço mencionado no [Capítulo 4](#)) dentro da pasta `prototype_arduino`. O computador utilizado possui 4GB de memória RAM, processador Intel Core i5 – 2500, HD de 500GB e sistema operacional Windows 7 Professional de 64Bits.

Os protótipos dos dispositivos de porta podem ser visualizados nas [Figura 27](#) e [Figura 28](#). A [Figura 27](#) mostra o Arduíno Nano juntamente com o leitor RFID RC522 conectados através de *jumpers*, já na [Figura 28](#) mostra o NodeMcu conectado com o leitor RFID RC522. A prototipação do Arduíno segue abaixo:

- 3.3V - conectado ao pino de 3.3v no Arduíno, essa conexão faz a alimentação do leitor RFID;
- RST (*Reset*)- conectado ao pino 9 do Arduíno;

- GND (*graduated neutral density filter*) - conectado ao pino GND do Arduino;
- NC/IRQ (*Interrupt Request*)- não utilizado;
- MISO (*Master In Slave Out*) - conectado ao pino 12;
- MOSI (*Master Out Slave In*) - conectado ao pino 11;
- SCK (*Serial Clock*) - conectado ao pino 13; e
- SDA/ SS (*Serial Data Line/ Select Slave*) - conectado ao pino 10.

Figura 27 – Protótipo Arduino Nano



Fonte: Própria

Figura 28 – Protótipo NodeMcu

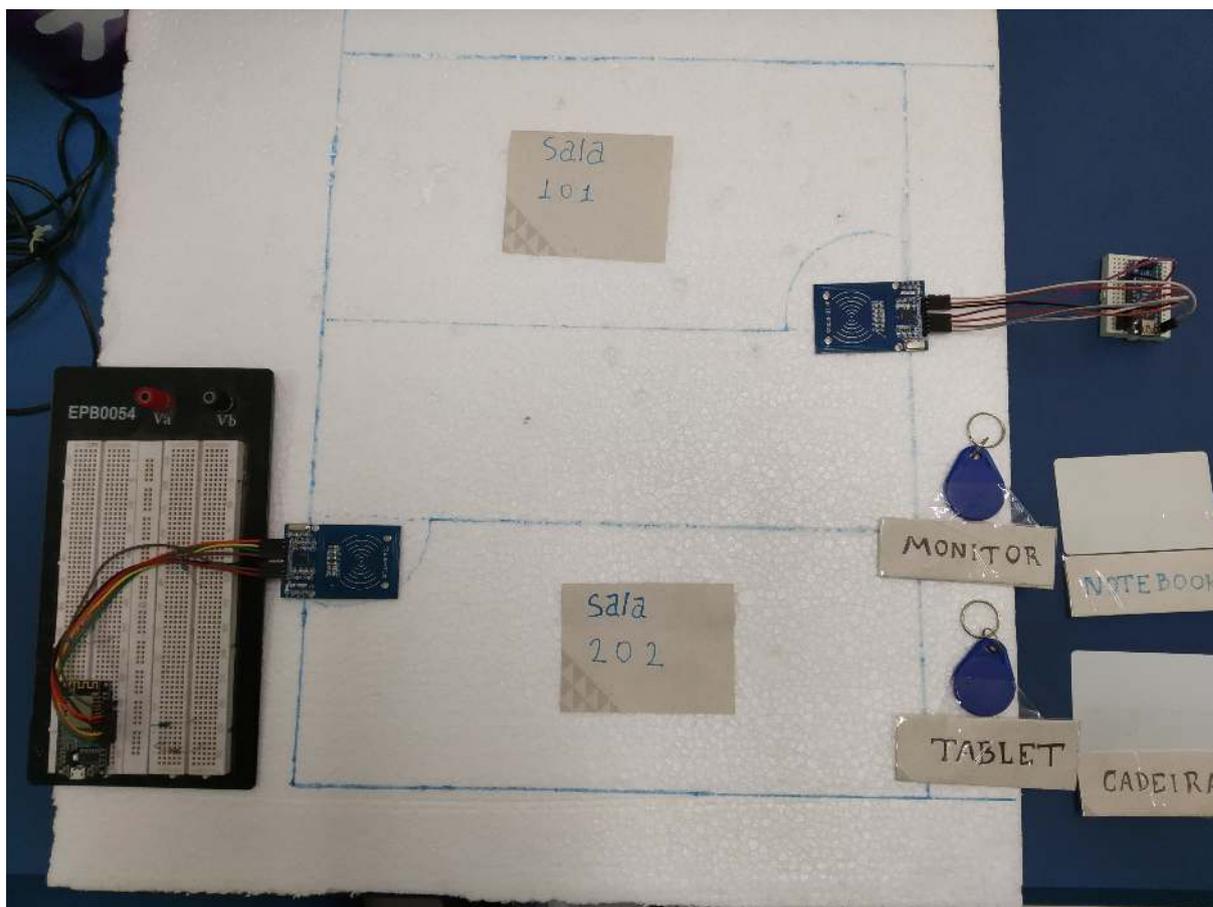


Fonte: Própria

Para a simulação do servidor, foi utilizado um notebook com 6GB de memória RAM, processador Intel Core i5 – 4200, HD de 750GB, placa de vídeo GeForce 720M de 2GB de memória dedicada e sistema operacional Windows 10 *Home Single Language*. A rede LAN utilizada foi criada através de um roteador TP-LINK modelo TL-WR720N. Também foram utilizadas quatro etiquetas RFID passivas, cada uma dessas etiquetas representa um objeto diferente para a simulação.

A execução do experimento aconteceu no cenário apresentado na [Figura 29](#), foi simulado duas salas e quatro objetos rastreáveis em um edifício. Visando responder as questões foram executados os seguintes testes: **(1)** localização e identificação dos objetos em ambientes confinados, no intuito de obter respostas para a primeira questão de pesquisa; **(2)** criação de restrição de movimentação sobre objetos, afim de verificar se as notificações ao violar restrição são uma forma de se ter melhor gerenciamento dos objetos; e **(3)** levantamento de todos os objetos em todos os ambientes monitorados pelo sistema INEXT.

Figura 29 – Cenário de simulação

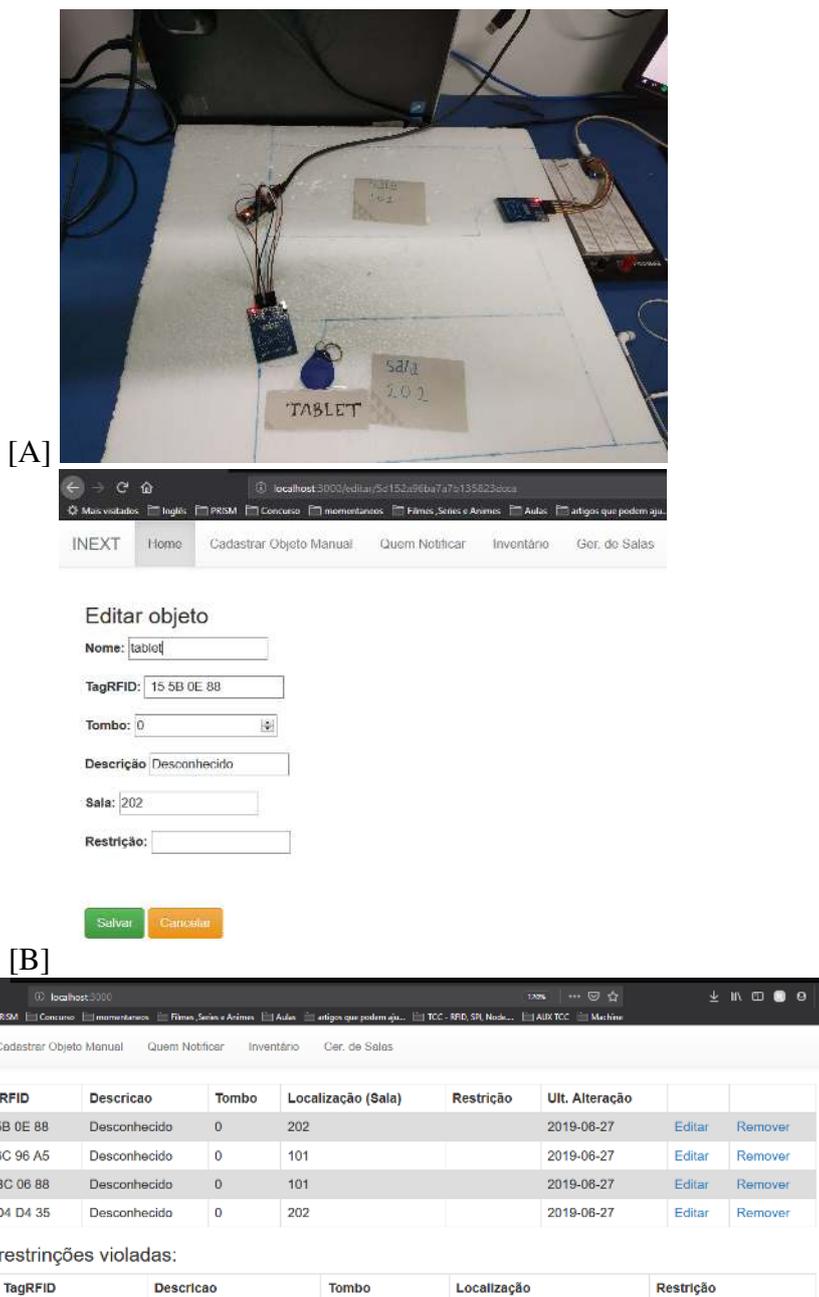


Fonte: Própria

5.2 Execução dos experimentos e Análise dos Resultados

Visando responder as questões de pesquisa no planejamento dos experimentos, o teste (1) no cenário do experimento foi executando cadastrando as quatro etiquetas no sistema e identificando-as cada uma como um objeto diferente, duas etiquetas foram lidas pelo dispositivo de porta da sala 101 e as outras duas foram lidas pelo dispositivo de porta da sala 202, o *pipeline* do processo pode ser visualizado na [Figura 30](#) de **A** e **B**, esse processo de cadastro foi repetido por mais três vezes e o resultado final desse processo com as quatro etiquetas é apresentado no relatório do sistema INEXT na [Figura 30 C](#).

Figura 30 – Identificação dos objeto

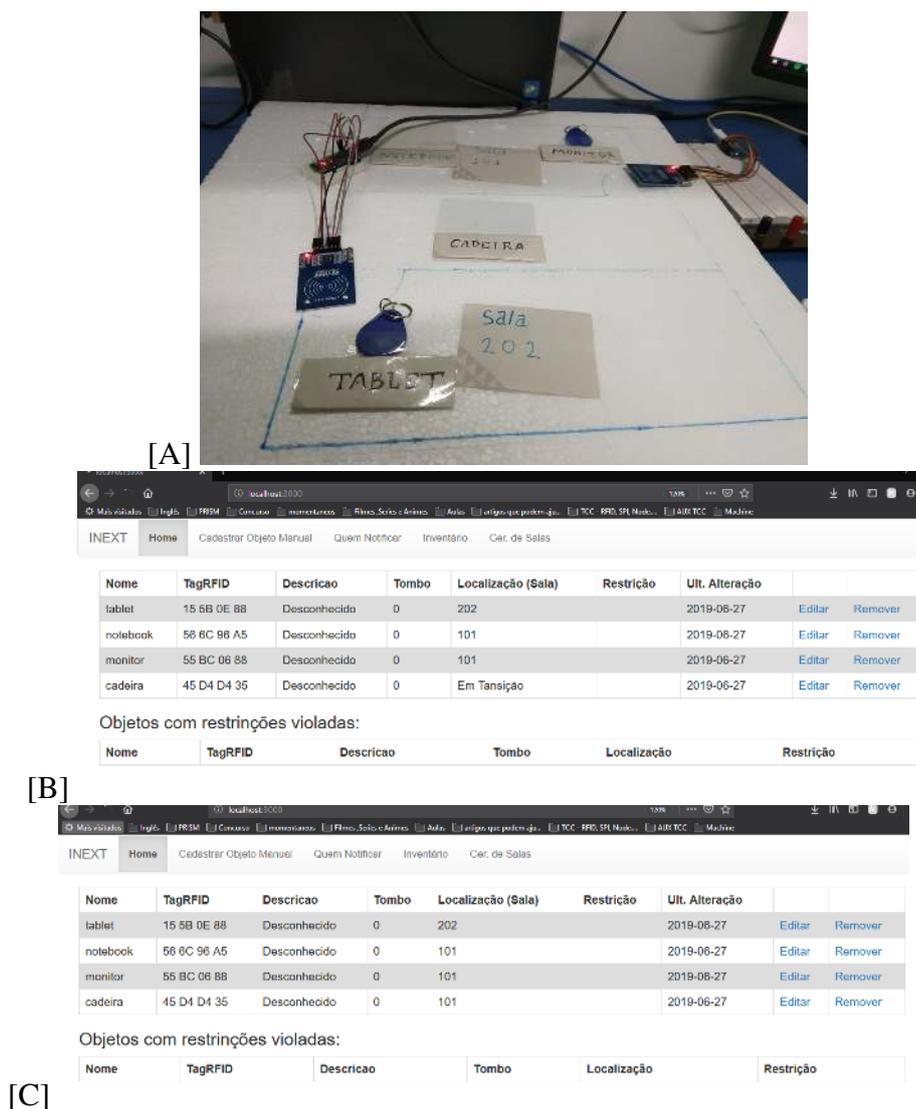


[C]

Fonte: Própria

Ainda realização do teste (1) no cenário, o processo de transição dos objetos foi executado observando quando uma etiqueta é lida, quando sai da sala e quando entra na outra sala, esse *pipeline* pode ser visto na Figura 31 de A até C. Foram executadas as transições para todos os objetos, onde a movimentação foi da sala 101 para a sala 202, e os da sala 202 foram para a sala 101.

Figura 31 – Transição de objeto



Fonte: Própria

No teste (2) do cenário experimental, foram criados dois usuário na tela de quem deve ser notificado a transição dos objetos para receber as notificações via e-mail. Para isso foi utilizado os e-mails do próprio autor deste trabalho, que utilizam os servidores de *webmail* Gmail e Outlook, em seguida foi criada restrição para cada objeto monitorado no sistema. Na tela principal do sistema INEXT, cada objeto possui uma opção para editar ao seu lado, foi editado um objeto por vez colocando no campo restrição o mesmo nome que ele possuía no campo de localização (sala). O cadastro dos usuários e a edição dos objetos pode ser visualizado no *pipeline* da Figura 32.

Figura 32 – Cadastro de usuários e Criação de Restrições de objeto

[A]



Cadastrar Adm

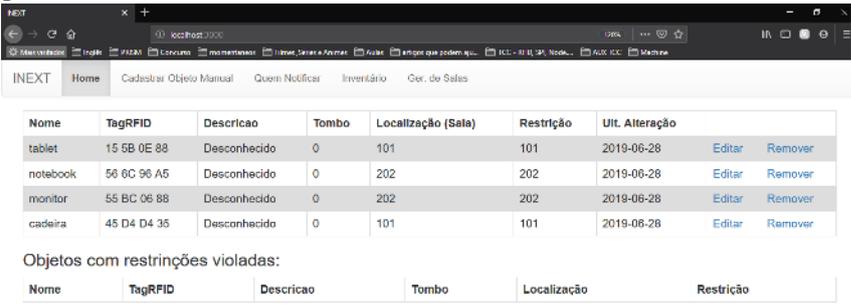
Nome: Luarkian
email: luarkian@gmail.com

Salvar Carregar

Tabela Adm

Nome	Email		
Luarkian	kaype_x@hotmail.com	Editar	Remover

[B]



Nome	TagRFID	Descricao	Tombo	Localização (Sala)	Restrição	Ult. Alteração		
tablet	15 5B 0E 88	Desconhecido	0	101	101	2019-06-28	Editar	Remover
notebook	56 6C 96 A5	Desconhecido	0	202	202	2019-06-28	Editar	Remover
monitor	55 BC 06 88	Desconhecido	0	202	202	2019-06-28	Editar	Remover
cadeira	45 D4 D4 35	Desconhecido	0	101	101	2019-06-28	Editar	Remover

Objetos com restrições violadas:

Nome	TagRFID	Descricao	Tombo	Localização	Restrição
------	---------	-----------	-------	-------------	-----------

Fonte: Própria

Ainda no teste (2) foi realizado processo de transição com os objetos, afim de verificar se o sistema notifica os usuários através do e-mail e na tela inicial quando o sistema proposto identifica que um dado objeto viola a sua restrição, ou seja, um dado objeto é identificado em uma sala para o qual não foi designado. A execução desta operação é apresentada na Figura 33. Foram realizadas transições para todos os objetos com restrições.

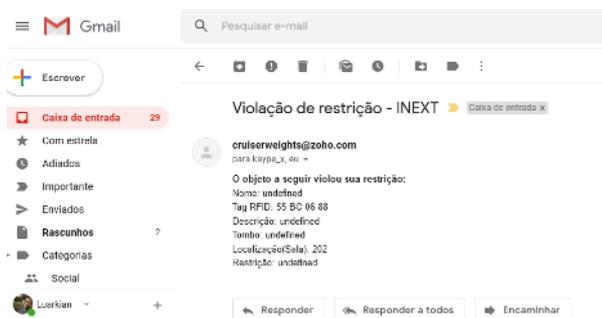
No teste (3) no cenário é executado o levantamento de todos os objetos monitorados (o inventário) pelo sistema INEXT em suas respectivas salas. Neste sentido, na tela web do sistema INEXT deve ser executado as seguintes ações: primeiramente é necessário ir até a tela Ger. de Salas para verificar as salas cadastradas no sistema e não há repetição de uma mesma sala; e em seguida ir para a tela Inventário, o inventário é gerado referente aquele momento, se ocorrer qualquer operação após a geração do inventário deverá ser realizado o processo novamente. O resultado da execução desse processo é apresentado na Figura 34.

5.2.1 Resultados

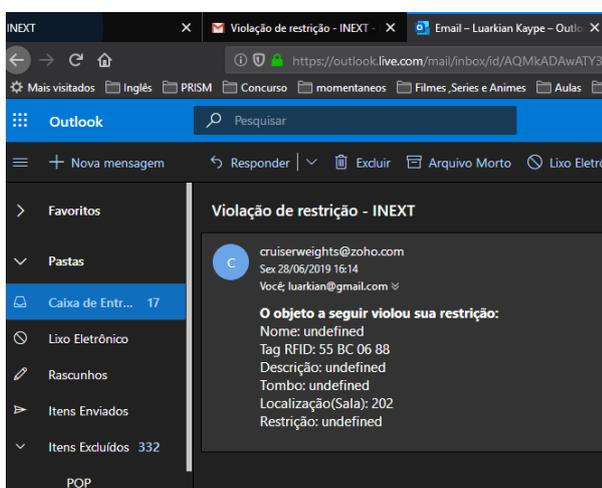
Após a execução dos testes, foram obtidos os resultados apresentados na Tabela 1, onde cada linha representa: um cenário de execução com o seu respectivo número de execuções; número de execuções em que o sistema INEXT funcionou de maneira correta (**Funcionou**); números de execuções em que o sistema INEXT por alguma questão não funcionou perfeitamente (**Parcialmente**); e o número de execuções em que o sistema falhou durante a suas execução, ou seja, não retornou o resultado esperado (**Falhou**).

O INEXT conseguiu localizar objetos rastreados por RFID em ambientes confinados, porém o sistema não proporciona a posição específica do objeto no ambiente, exemplo, no lado

Figura 33 – Transições com Restrições



[A]



[B]

Nome	TagRFID	Descricao	Tombo	Localização (Sala)	Restrição	Ult. Alteração		
tablet	15 5B 0E 88	Desconhecido	0	Em Transição	101	2019-06-28	Editar	Remover
notebook	56 6C 96 A5	Desconhecido	0	Em Transição	202	2019-06-28	Editar	Remover
monitor	55 BC 06 88	Desconhecido	0	Em Transição	202	2019-06-28	Editar	Remover
cadeira	45 D4 D4 35	Desconhecido	0	Em Transição	101	2019-06-28	Editar	Remover

Objetos com restrições violadas:

Nome	TagRFID	Descricao	Tombo	Localização	Restrição
tablet	15 5B 0E 88	Desconhecido	0	Em Transição	101
notebook	56 6C 96 A5	Desconhecido	0	Em Transição	202
monitor	55 BC 06 88	Desconhecido	0	Em Transição	202
cadeira	45 D4 D4 35	Desconhecido	0	Em Transição	101

[C]

Fonte: Própria

direito da sala. O sistema também foi capaz de identificar os objetos, mas essa etapa ainda ocorre de maneira manual necessitando ser realizada por usuários a edição dos objeto no sistema após ter o seu cadastro no sistema. Durante os teste houve uma execução em que os script em Python que é responsável por ler os dados da porta serial do Arduíno Nano, gerar um JSON e enviar para o servidor via POST, travou e não enviou os JSON's gerados para o servidor devido ao cabo USB que foi tocado no momento e se desconectou, sendo necessário parar o script e iniciar novamente e assim realizar a leitura novamente para que a etiqueta tivesse a localização correta.

Figura 34 – Inventário

Objetos na sala: 202

Nome	TagRFID	Descrição	Tombo	Restrição
notebook	56 BC 96 A5	Desconhecido	0	202
monitor	55 BC 06 88	Desconhecido	0	202

Objetos na sala: 101

Nome	TagRFID	Descrição	Tombo	Restrição
tablet	15 5B 0E 88	Desconhecido	0	101
cadeira	45 D4 D4 35	Desconhecido	0	101

Fonte: Própria

Tabela 1 – Resultados

	Nº de execuções	Funcionou	Parcialmente	Falhou
Teste 1	8	7	-	1
Teste 2	8	8	-	-
Teste 3	7	6	1	-

O sistema proporcionou uma boa forma de gerenciar os objetos, de acordo com os resultados do experimento. De forma que por meio do uso do sistema INEXT foi possível informar os usuários sobre as violação de restrições através do e-mail e também na tela principal do sistema. Durante a primeira tentativa de gerar o inventário dos objetos cadastrados no sistema, houve repetição de dados justamente porque foram criadas salas repetidas no banco de dados, esse problema pode ser solucionado acessando a tela Ger. de Salas e remover as salas repetidas, dessa forma a geração do inventário funcionou sem a repetição dos objetos e salas.

Afim de ter uma melhor avaliação do sistema foi elaborada um questionário contendo quatro questões, e aplicado a três pessoas que foram/são coordenadores/chefes de departamentos de cursos da UFRR. As as questões são listadas abaixo:

1. De 0 à 10, você acha que o sistema pode auxiliar o gerenciamento dos bens da universidade?
2. De 0 à 10, qual nota você daria para avaliar essa solução aplicada ao problema?
3. De 0 à 10, qual a probabilidade de indicar o sistema para os funcionários da UFRR que realizam a tarefa de gerenciamento e levantamento de patrimônio?

As perguntas 1, 2 e 3 possuem representatividade quantitativa em que: 0 e 1 são muito ruim, 2 e 3 ruim, 5 e 6 razoável, 7 e 8 bom, 9 e 10 muito bom. O resultado do questionário pode ser visto na [Tabela 2](#).

Tabela 2 – Dados coletados do questionário realizado

	Pessoa 1	Pessoa 2	Pessoa 3	Média
Pergunta 1	10	9	10	9.66
Pergunta 2	8	8	7	7.66
Pergunta 3	9	9	10	9.33

A média das respostas em relação a pergunta 1 é de 9.66, mostrando que o sistema pode auxiliar o gerenciamento e levantamento de bens da universidade, de forma que se implantando proporciona facilidade em relação como a maneira que o levantamento é realizado atualmente, em que é necessário a visita de um funcionário em todas as salas verificando-as e anotando todos as referencias dos objetos lá situados.

Na pergunta de número 2, as respostas obtiveram média igual a 7.66, o que mostra que os sistema é bom, porém o sistema não é capaz de resolver todos os problemas, por exemplo, quando o objeto está em transição, se por algum motivo não entrar em uma outra sala o objeto não possuirá localização no sistema. Por fim, a pergunta número 3, obteve média igual 9.33, que de modo geral mostra que o sistema possui grande potencial de aplicabilidade e grandes indícios que pode ser indicada para os funcionários da UFRR que executam essa função.

6 CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS

Este trabalho propôs um sistema, denominado de INEXT, de localização e identificação de objetos em edifícios por meio de radiofrequência, usando etiquetas passivas. O real objetivo consiste em facilitar e auxiliar o gerenciamento/controlado de bens em edifícios cujo possuem uma grande quantidade de salas.

Os experimentos realizados demonstraram que o sistema proporcionou uma boa forma de gerenciar os objetos, de acordo com os resultados do experimento. De forma que por meio do uso do sistema INEXT foi possível informar os usuários sobre as violação de restrições de movimentação dos objetos aos seus respectivos administradores.

O sistema ainda precisa de melhorias para que se torne um produto final, por outro lado a proposta tem pontos positivos e satisfatórios. Um dos pontos a ser melhorado é que o método até o momento possui limitações referente ao leitor RFID utilizado, o leitor possui o alcance baixo necessitando que as etiquetas sejam praticamente encostadas para que a leitura seja realizada. Neste sentido, como trabalho futuro será realizado um estudo com diferentes leitores RFID para que o sistema seja implantado em um prédio.

No questionário, aplicado durante a avaliação do sistema, com pessoas que já foram ou ainda possuem cargo de chefia ou coordenador de departamentos de cursos na universidade. As respostas indicaram que o sistema pode proporcionar mais facilidade nas tarefas de gerenciamento ou servir como ferramenta de auxílio, de forma que mostra a viabilidade do sistema.

Adicionalmente, visa-se estudar métodos para ter uma localização mais precisa sobre a localização dos objetos, por exemplo, pode-se utilizar etiquetas ativas para auxiliar e em seguida aplicar algoritmos de RSSI, dessa forma tende-se a ter uma localização com maior precisão no ambiente e uma maneira para executar o processo de identificação dos objetos de maneira automática.

7 ANEXO A

7.1 Revisão Sistemática

Nesta seção é apresentado o protocolo definido para a aplicação e execução da revisão sistemática que foi parcialmente executada neste trabalho, devido ao tempo e quantidade de publicações identificadas, no total foram 549. Todos os artigos citados na Seção 3 foram selecionados através da execução deste protocolo proposto.

Segundo [Kitchenham et al. \(2009\)](#) a revisão sistemática é uma seleção da literatura que segue um protocolo formal, sendo uma revisão metodologicamente rigorosa dos resultados e com o objetivo de agregar todas as evidências existentes em uma pesquisa. A sua aplicação consiste em uma sequência de passos definidos a ser seguidos e se comparado a revisão da literatura informal requer um grande esforço.

A revisão sistemática atua como um meio de identificação e avaliação para pesquisa conduzindo-o para tópicos relevantes sobre o tema de interesse, diferentemente da revisão da literatura informal em que a presença de variações entre os estudos representa um fator negativo ([MAFRA; TRAVASSOS, 2006](#)). De acordo com [Mafra e Travassos \(2006\)](#) o processo da revisão sistemática envolve três etapas:

1. **Planejamento da Revisão:** nesta etapa os objetivos são listados e é definido o protocolo de revisão;
2. **Condução da Revisão:** etapa onde as fontes para a revisão sistemática são selecionadas, identificação de estudos primários e avaliados com critérios de inclusão e exclusão;
3. **Publicação dos Resultados:** na terceira etapa os dados dos estudos são extraídos e sintetizados para serem publicados.

Conduzimos a revisão deste trabalho com a finalidade de realizar um estudo exploratório de caracterização da área, e assim podemos afirmar que essa revisão se caracteriza como uma quasi-sistemática pois está baseada nas três etapas citadas anteriormente ([MAFRA; TRAVASSOS, 2006](#)), seguindo o processo da revisão sistemática e preservando o rigor e o mesmo formalismo para as fases metodológicas de elaboração e execução do protocolo, faltando apenas uma meta-análise do que poderia ser aplicado futuramente.

7.1.1 Planejamento da Revisão Sistemática

Objetivo: O objetivo desta revisão é analisar publicações científicas com o propósito de identificar métodos para a localização de objetos com a utilização de tag RFID no contexto

acadêmico ou industrial.

Formulação das Perguntas.

- Quais os métodos para a localização de tags RFID?
- Está disponível alguma ferramenta para localização de tags RFID ?
- Quais as limitações em relação a identificação de mais de uma tag RFID simultaneamente?
- Quais os requisitos necessários para execução do método?
- Quais as limitações do método proposto ?
- Quais são as perspectivas futura para a melhoria do método e da aplicação?

- **População**

- **Palavras-Chave** "tag RFID"OR "RFID signaling scheme"OR "RFID reader"OR "Handheld RFID Reader"OR "RFID Label"OR "RFID Tag Reader"OR "Passive RFID Tags"OR "Passive RFID Readers"OR "Radio-Frequency IDentification" OR "Radio Frequency IDentification" OR "review of RFID" OR "RFID technology" OR "RFID localization" OR "method for RFID" OR "Active RFID"

- **Intervenção**

- **Palavras-Chave** "Positioning principles" OR "object localization indoor"OR "positioning algorithms"OR "indoor location technique"OR "indoor object management"OR "asset localization"OR "localization-based services"OR "facilities management"OR "pattern matching"OR "mapping"OR "indoor automatic identification"OR "indoor tag classification"OR "indoor classify RFID"OR "indoor classify tag"OR "indoor localization techniques"OR "Applications and techniques" OR "system for indoor" OR "real-time locating" OR "localization method" OR "tags for indoor" OR "Efficient object localization" OR "using RSS" OR "algorithms and applications" AND NOT "using UHF RFID"

7.1.2 Procedimento de Seleção de Critérios

- **CE1-01:** Não serão selecionadas publicações em que não contém as palavras-chave da string de busca no título, resumo, método e/ou resultados. Os campos de seções de agradecimentos, biografia dos autores, referências bibliográficas e anexos não serão incluídos.
- **CE1-02:** Publicações cujo se assemelham com tutoriais e cursos não serão selecionadas.

- **CE1-03:** Não serão selecionadas publicações em que técnicas e algoritmos para localização de objetos não trabalham com identificação por rádio frequência (RFID).
- **CE1-04:** Não serão selecionadas publicações em que façam utilização de frequência ultra-alta (UHF) RFID.
- **CE1-05:** Não serão selecionadas publicações que não fazem uma abordagem para utilização de RFID para localizar objetos com tags RFID.

Publicações que podem ser incluídas no conjunto.

- **CI1-01:** Podem ser selecionadas publicações em que o contexto das palavras-chave utilizadas no artigo leva a crer que faz uma citação para uma abordagem de utilização de RFID para localizar objetos utilizando tags RFID.
- **CI1-02** Podem ser selecionadas publicações em que o contexto das palavras-chave utilizadas no artigo leva a crer que a publicação cita recomendações de algoritmos, métodos e ferramentas que possam ser utilizados para localizar objetos com tag RFID.

REFERÊNCIAS

ABRAMOVA, V.; BERNARDINO, J. Nosql databases: Mongodb vs cassandra. In: *Proceedings of the International C* Conference on Computer Science and Software Engineering*. New York, NY, USA: ACM, 2013. (C3S2E '13), p. 14–22. ISBN 978-1-4503-1976-8. Disponível em: <http://doi.acm.org/10.1145/2494444.2494447>.

ALMEIDA, R. de; MORAES, C. de; SERAPHIM, T. de F. P. *Programação de Sistemas Embarcados: Desenvolvendo Software para Microcontroladores em Linguagem C*. Elsevier Editora Ltda., 2017. ISBN 9788535285192. Disponível em: <https://books.google.com.br/books?id=7LA4DwAAQBAJ>.

ARDUINO. *Arduino Uno*. Arduino, 2018. Disponível em: <https://store.arduino.cc/usa/arduino-nano>.

AYALA, K. J. *The 8051 microcontroller: architecture, programming, and applications*. [S.l.]: West Publishing Co., 1991.

BOUET, M.; SANTOS, A. L. dos. Rfid tags: Positioning principles and localization techniques. In: *2008 1st IFIP Wireless Days*. [S.l.: s.n.], 2008. p. 1–5. ISSN 2156-9711.

CASTRO, P. et al. A probabilistic room location service for wireless networked environments. In: *Proceedings of the 3rd International Conference on Ubiquitous Computing*. Berlin, Heidelberg: Springer-Verlag, 2001. (UbiComp '01), p. 18–34. ISBN 3-540-42614-0. Disponível em: <http://dl.acm.org/citation.cfm?id=647987.741335>.

CHAWLA, K.; ROBINS, G.; ZHANG, L. Object localization using rfid. In: *IEEE 5th International Symposium on Wireless Pervasive Computing 2010*. [S.l.: s.n.], 2010. p. 301–306.

DAYLEY, B. *Node.js, MongoDB, and AngularJS Web Development*. Pearson Education, 2014. (Developer's Library). ISBN 9780133844344. Disponível em: <https://books.google.com.br/books?id=8kTCAwAAQBAJ>.

FANG, S. H.; LIN, T. Principal component localization in indoor wlan environments. *IEEE Transactions on Mobile Computing*, v. 11, n. 1, p. 100–110, Jan 2012. ISSN 1536-1233.

FERNANDES, J. P. da F. *Localização em redes Wi-Fi*. Dissertação (Mestrado) — Universidade do Minho, 2012. [Online; accessed 22-Julho-2018]. Disponível em: <http://hdl.handle.net/1822/28011>.

GISLASON, D. *Zigbee Wireless Networking*. Pap/onl. Newton, MA, USA: Newnes, 2008. ISBN 0750685972, 9780750685979.

HAN, D.; LIM, J. Design and implementation of smart home energy management systems based on zigbee. *IEEE Transactions on Consumer Electronics*, v. 56, n. 3, p. 1417–1425, Aug 2010. ISSN 0098-3063.

HUERTA ALDO MANGIATERRA, G. N. E. *GPS posicionamiento satelital*. 1th. ed. [S.l.]: UNR - Universidade Nacional de Rosario, 2005. 148 p. ISBN 950-673-488-7.

- JIN, G. yao; LU, X. yi; PARK, M.-S. An indoor localization mechanism using active rfid tag. In: *IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC'06)*. [S.l.: s.n.], 2006. v. 1, p. 4 pp.—.
- KITCHENHAM, B. et al. Systematic literature reviews in software engineering - a systematic literature review. *Inf. Softw. Technol.*, Butterworth-Heinemann, Newton, MA, USA, v. 51, n. 1, p. 7–15, jan. 2009. ISSN 0950-5849. Disponível em: <<http://dx.doi.org/10.1016/j.infsof.2008.09.009>>.
- LEITE, J.; MARTINS, P.; URSINI, E. A internet das coisas (iot) : Tecnologias e aplicações (issn 2447-8326). 12 2017.
- LI, Q.; YAO, C. *Real-Time Concepts for Embedded Systems*. 1st. ed. Boca Raton, FL, USA: CRC Press, Inc., 2003. ISBN 1578201241, 9781578201242.
- MAFRA, S. N.; TRAVASSOS, G. H. Estudos primarios e secundarios apoiando a busca por evidencia em engenharia de software. *Relatorio Tecnico, RT-ES*, v. 687, n. 06, 2006.
- MAGNUS, T. *O que é IoT (Internet das Coisas)?* 2017. Acessado em: 04 de agosto de 2018. Disponível em: <<https://transformacaodigital.com/o-que-e-iot-internet-das-coisas/>>.
- MEES, G. *First Steps with the NodeMCU ESP8266 Wi-Fi Module | Maker, MakerED, Coding | Tips*. 2018. Acessado em: 07 de junho de 2019. Disponível em: <<https://gustmees.wordpress.com/2018/07/04/first-steps-with-the-nodemcu-esp8266-wi-fi-module-maker-makered-coding-tips/>>.
- NI, L. M. et al. Landmarc: indoor location sensing using active rfid. In: *Proceedings of the First IEEE International Conference on Pervasive Computing and Communications, 2003. (PerCom 2003)*. [S.l.: s.n.], 2003. p. 407–415.
- NODEMCU. *NodeMcu Connect Things EASY*. NodeMcu, 2018. Disponível em: <https://www.nodemcu.com/index_en.html>.
- NUNES, M.; O'NEILL, H. *Fundamental de UML*. [S.l.]: FCA (BRASIL), 2011. ISBN 9789727224814.
- OLIVEIRA, S. de. *Internet das Coisas com ESP8266, Arduino e Raspberry Pi*. [S.l.]: NOVATEC, 2017. ISBN 9788575225813.
- PEREIRA, C. *Aplicações web real-time com Node.js*. Casa do Código, 2014. ISBN 9788566250930. Disponível em: <<https://books.google.com.br/books?id=Wm-CCwAAQBAJ>>.
- SANPECHUDA, T.; KOVAVISARUCH, L. A review of rfid localization: Applications and techniques. In: *2008 5th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology*. [S.l.: s.n.], 2008. v. 2, p. 769–772.
- SANTOS, B. P. et al. Internet das coisas: da teoria à prática. In: *Livro de Minicursos SBRC 2016*. [S.l.: s.n.], 2016.
- SILVA, C. A. E. V. Alberto Manuel Rodrigues da. *UML, Metodologias e Ferramentas CASE*. 1st. ed. [S.l.]: Editora Centro Atlântico, 2001. ISBN 972-8426-36-4.

SÔNIGO RODERVAL MARCELINO, V. G. A. A. A internet das coisas aplicada ao conceito de eficiência energética: uma análise quantitativo-qualitativa do estado da arte da literatura. *AtoZ: novas práticas em informação e conhecimento*, v. 5, n. 2, p. 80–90, 2016. ISSN 2237-826X. Acessado em: 29 de maio de 2018. Disponível em: <<https://revistas.ufpr.br/atoz/article/view/47860>>.

TANENBAUM, A. *Computer Networks*. 4th. ed. [S.l.]: Prentice Hall Professional Technical Reference, 2002. 912 p. ISBN 0130661023.

Tilkov, S.; Vinoski, S. Node.js: Using javascript to build high-performance network programs. *IEEE Internet Computing*, v. 14, n. 6, p. 80–83, Nov 2010. ISSN 1089-7801.

TORRES, G. *Redes de Computadores*. [S.l.]: Axcel Books, 2001. 663 p. ISBN 9788573231441.

XIE, Y. et al. An improved k-nearest-neighbor indoor localization method based on spearman distance. *IEEE Signal Processing Letters*, v. 23, n. 3, p. 351–355, March 2016. ISSN 1070-9908.

ZHANG, D. et al. Real-time locating systems using active rfid for internet of things. *IEEE Systems Journal*, v. 10, n. 3, p. 1226–1235, Sept 2016. ISSN 1932-8184.

ZHOU, J. S. J. Rfid localization algorithms and applications - a review. *Journal of Intelligent Manufacturing*, Springer US, p. 695–707, 2009. ISSN 1572-8145. Disponível em: <<https://doi.org/10.1007/s10845-008-0158-5>>.