



**Universidade de Brasília
Faculdade de Tecnologia**

**Investigação, avaliação e implementação de
sistema de navegação autônomo, embarcado
em dispositivos móveis para atendimento de
alunos com deficiência visual**

João Lucas da Cruz

**TRABALHO DE GRADUAÇÃO
ENGENHARIA DE CONTROLE E AUTOMAÇÃO**

Brasília
2023

**Universidade de Brasília
Faculdade de Tecnologia**

**Investigação, avaliação e implementação de
sistema de navegação autônomo, embarcado
em dispositivos móveis para atendimento de
alunos com deficiência visual**

João Lucas da Cruz

Projeto final de curso submetido como requisito parcial para obtenção do grau de Engenheiro de Controle e automação.

Orientador: Prof. Dr. Carlos Humberto Llanos Quintero

Coorientador: Prof. Mayra Batista Correa

Brasília

2023

C957i Cruz, João Lucas da.
Investigação, avaliação e implementação de sistema de navegação autônomo, embarcado em dispositivos móveis para atendimento de alunos com deficiência visual / João Lucas da Cruz; orientador Carlos Humberto Llanos Quintero; coorientador Mayra Batista Correa. -- Brasília, 2023.
55 p.

Projeto final de curso (Engenharia de Controle e Automação)
-- Universidade de Brasília, 2023.

1. Fingerprint. 2. Wi-Fi. 3. Acessibilidade. 4. Filtro de Kalman. I. Llanos Quintero, Carlos Humberto, orient. II. Correa, Mayra Batista, coorient. III. Título

**Universidade de Brasília
Faculdade de Tecnologia**

Investigação, avaliação e implementação de sistema de navegação autônomo, embarcado em dispositivos móveis para atendimento de alunos com deficiência visual

João Lucas da Cruz

Projeto final de curso submetido como requisito parcial para obtenção do grau de Engenheiro de Controle e automação.

Trabalho aprovado. Brasília, 20 de dezembro de 2023:

**Prof. Dr. Carlos Humberto Llanos
Quintero, UnB/FT/ENM**
Orientador

Prof. Dr. Jones Yudi Mori Alves da Silva,
UnB/FT/ENM
Examinador interno

Prof. Dr. Henrique Cezar Ferreira,
UnB/FT/ENE
Examinador interno

Brasília
2023

*Este trabalho é dedicado às crianças adultas que,
quando pequenas, sonharam em se tornar cientistas.*

Agradecimentos

Agradeço primeiramente à minha mãe, que sempre cuidou de mim e mesmo agora continua a me trazer lições sobre paciência e gentileza. Agradeço também aos meus primos Lucas Henrique e Mariana Alves por sempre me escutarem e darem conselhos, aos meus amigos de longa data Júlia Caroline, Railton Magalhães e Eric Millani, que me proporcionaram muitos momentos de descontração e aprendizado que me ajudaram a não desanimar mesmo em situações difíceis.

*“If you find that you’re spending almost all your time on theory,
start turning some attention to practical things;
it will improve your theories.
If you find that you’re spending almost all your time on practice,
start turning some attention to theoretical things;
it will improve your practice.”*
(Donald Knuth)

Resumo

Um ambiente é considerado acessível quando este pode ser devidamente utilizado por diferentes usuários, incluindo pessoas com deficiência visual. Este público pode encontrar várias dificuldades no cotidiano, sendo a navegação em percursos desconhecidos ou especialmente complexos uma delas. Essa dificuldade torna-se ainda mais acentuada em ambientes *indoor*, que podem apresentar um maior número de obstáculos se comparados com ambientes *outdoor*. Os sistemas de posicionamento *indoor* (IPS) ainda apresentam desafios técnicos a serem explorados e superados devido às especificidades dos ambientes que dificultam a generalização de soluções. Além disso, tecnologias que utilizam estruturas que fazem parte do *Global Navigation Satellite Systems (GNSS)* não são adequadas para utilização em recintos fechados. A Universidade de Brasília (UnB), *locus* desse estudo, ainda não possui solução tecnológica para auxiliar na navegação *indoor* de seus usuários. Assim, este trabalho apresenta uma proposta de metodologia para o desenvolvimento de um IPS na forma de aplicativo (MovUnB) para *smartphones*. O aplicativo deve ser capaz de guiar usuários com ou sem deficiência visual dentro de edifícios, utilizando-se das técnicas de *Pedestrian Dead Reckoning (PDR)* e *fingerprint* de sinais Wi-Fi, amplamente difundidas na literatura, combinados a partir de um filtro de Kalman linear. O aplicativo proposto é capaz de estimar a posição do usuário com precisão de 1,73 m, o que tem potencial para auxiliar o usuário a localizar salas de aula e banheiros, aumentando a acessibilidade de localização, o que gera sentimento de pertencimento dos estudantes e consequente permanência dos estudantes em ambiente universitário.

Palavras-chave: *fingerprinting*, Wi-Fi, PDR, IPS, acessibilidade, filtro de Kalman.

Lista de ilustrações

Figura 1 – Elipsoide ilustrando coordenadas cartesianas e geodésicas. (Fonte: Jansen (2009))	26
Figura 2 – Coordenadas locais. (Fonte: Burkholder (2008))	27
Figura 3 – Grafo de termos relevantes tecnologias de poiiionamento indoor	29
Figura 4 – Diagrama de casos de uso	32
Figura 5 – Sistema de coordenadas do celular. (FONTE: (ANDROID, 2023))	35
Figura 6 – Sistema de coordenadas locais (FONTE: (ANDROID, 2023))	35
Figura 7 – Pontos selecionados para estimar o azimute	38
Figura 8 – Equipamento trimble para captura de coordenadas	39
Figura 9 – Pontos selecionados	40
Figura 10 – Tela de login e integração com conta google	44
Figura 11 – Telas de navegação e seleção de destino com entrada de áudio	44
Figura 12 – Manual do usuário, tela de configurações e caixa de saída	45
Figura 13 – Distribuição dos erros na estimativa da posição pelo fingerprint	48
Figura 14 – Ambiente de testes	49
Figura 15 – Trajetos estimados no pior e melhor cenário.	50

Lista de tabelas

Tabela 1 – Valores RSSI médios por ponto ponto	33
Tabela 2 – Coordenadas dos pontos coletados.	39
Tabela 3 – Coordenadas dos pontos coletados.	40
Tabela 4 – Frequência de acertos na ocorrência de passos	46
Tabela 5 – Erro angular médio do passo	47
Tabela 6 – Erro angular médio do passo	47
Tabela 7 – Distribuição de erros fingerprinting	48

Lista de abreviaturas e siglas

GNSS	Global Navigation Satellite system	13
IPS	Indoor Positioning System	13
PDR	Pedestrian Dead Reckoning	13
RFID	Radio Frequenct Identification	13
UnB	Universidade de Brasília	14
UWB	Ultra-wide Band	13

Sumário

1	INTRODUÇÃO	13
1.1	Contextualização	13
1.2	Motivação	14
1.3	Hipótese do trabalho	15
1.4	Objetivo Geral	15
1.4.1	Objetivos específicos	15
1.5	Aspectos metodológicos	16
1.6	Contribuições do trabalho de graduação	16
2	FUNDAMENTAÇÃO TEÓRICA	17
2.1	Princípios de acessibilidade	17
2.2	Interfaces acessíveis	17
2.3	Acessibilidade para dispositivos móveis	18
2.4	Localização indoor	19
2.5	<i>Pedestrian dead reckoning</i>	20
2.6	Wi-Fi Fingerprinting	21
2.7	Algoritmo K nearest neighbours	21
2.8	Filtros de Kalman	22
2.8.1	Predição	23
2.8.2	Correção	24
2.9	Sistemas de coordenadas	24
2.9.1	Coordenadas geodésicas	25
2.9.2	Coordenadas locais	27
2.10	Matrizes de rotação e azimutes	28
2.11	Estado da Arte	29
3	METODOLOGIA E IMPLEMENTAÇÃO	31
3.1	Criação do caso de uso do aplicativo	31
3.2	Seleção do ambiente de testes	31
3.3	Desenvolvimento dos algoritmos para cálculo de posição	32
3.3.1	Implementação da técnica de <i>fingerprinting</i> de sinais Wi-Fi	33
3.3.2	Implementação do algoritmo <i>PDR</i>	34
3.4	Determinação do ângulo de azimute	37
3.5	Implementação do filtro de Kalman	41
3.6	Cálculo de rotas	42
3.7	Desenvolvimento da interface	43

4	TESTES, RESULTADOS E DISCUSSÃO	46
4.1	Testes do PDR	46
4.2	Testes da implementação do fingerprinting	47
4.3	Teste da trajetória calculada final	49
5	CONCLUSÕES	51
5.0.1	Trabalhos futuros	51
	REFERÊNCIAS	52

1 Introdução

1.1 Contextualização

Para os indivíduos com deficiência visual, compreender e navegar pelos espaços físicos pode ser tarefa desafiadora, devido a ausência do sentido da visão e de tecnologias assistivas, que visam auxiliar na localização e no deslocamento. Segundo dados da Organização Mundial da Saúde (OMS¹), cerca de 2,2 bilhões de pessoas em todo o mundo tem alguma deficiência visual. No Brasil, este número chega a 35,7 milhões de pessoas (IBGE, 2010).

Existem alguns mecanismos e ferramentas para auxiliar as pessoas com deficiência visual em seu trajeto, tais como: bengalas, cão guia, mapas, pisos táteis, placas braile, aparatos tecnológicos, entre outros. Porém, mesmo com o uso dessas tecnologias assistivas, o usuário ainda pode necessitar de auxílio de outras pessoas, principalmente em ambientes internos.

A fim de auxiliar na localização de pessoas ou objetos no espaço, existem tecnologias, denominadas sistemas de posicionamento, projetadas para fornecer e disponibilizar orientações para o deslocamento de um ponto de partida até um destino desejado (KUNHOTH et al., 2020). Tais sistemas desempenham papel fundamental na qualidade de vida dos usuários com deficiência visual, e podem impactar positivamente na qualidade de vida, tanto por gerar maior autonomia aos usuários quanto por trazer independência na localização e deslocamento pelos ambientes (SRIDHARAN et al., 2019).

Os sistemas de posicionamento podem ser classificados como sistemas *indoor*, ou *outdoor*. Sistemas de navegação por satélite, como o *Global Navigation Satellite System* (GNSS), são sistemas para uso *outdoor*, considerados com precisão suficiente para muitas atividades ao ar livre e possuem assistência satisfatória à navegação dos indivíduos (GOMES et al., 2018). No entanto, as características de propagação dos sinais utilizados nesses sistemas tornam inviável a sua utilização em sistemas de posicionamento *indoor*, *Indoor Positioning Systems* (IPS), dadas as restrições físicas existentes, o que torna necessário o uso de outras tecnologias (KUNHOTH et al., 2020). Em levantamento realizado por Kunhoth et al. (2020) sobre as principais técnicas utilizadas para o desenvolvimento de IPSs, é destacado o uso das seguintes tecnologias: 1) visão computacional; 2) tecnologias de comunicação por ondas eletromagnéticas, dentre elas: Wi-Fi, *bluetooth*, *Radio Frequency Identification* (RFID), *Ultra-Wide Band* (UWB) e 3) *Pedestrian Dead Reckoning* (PDR). Projetar um modelo matemático de navegação com rotas para guiar deficientes visuais em ambiente *indoor* ainda é considerado uma tarefa complexa (PAUL; SATO, 2017)

A seleção das tecnologias a serem utilizadas no desenvolvimento de um IPS depende

¹ <https://www.who.int/publications-detail/world-report-on-vision>

da estrutura do ambiente para qual o sistema será construído e também de sua finalidade. Algumas das aplicações típicas de IPSs são: (a) auxílio de pessoas com deficiência visual, (b) posicionamento de equipamentos, (c) navegação dentro de edifícios, entre outras. Há muitas tecnologias capazes de localizar pessoas em lugares fechados, porém, poucas atendem as demandas de pessoas cegas (COLOMBO et al., 2013).

No Brasil há 2.608 instituições de ensino superior, das quais 302 são públicas (ESTUDOS E PESQUISAS EDUCACIONAIS ANÍSIO TEIXEIRA - INEP, 2019). A exemplo, a Universidade de Brasília (UnB), *locus* desta pesquisa é uma instituição pública federal que 90 edificações e aproximadamente 50 mil alunos regularmente registrados, desses, 0,12% tem deficiência visual e 1.16% outras deficiências (ANUÁRIO... , 2019). Atualmente, a universidade não possui uma solução IPS que auxiliem os alunos dentro do campus, principalmente aos que desconhecem o ambiente ou têm alguma especificidade.

1.2 Motivação

Segundo correa2016metodologia o respeito as pessoas com deficiência perpassa pela criação de mecanismos que possibilitem o amplo acesso aos espaços físicos. Apesar da legislação exigir que o meio urbano seja adaptado para a acessibilidade de deficientes físicos, a realidade vivida é completamente diferente. A impossibilidade de deslocar-se de forma autônoma provoca no sujeito prejuízos sociais, afetivos e intelectuais (BRASIL, 2004). A acessibilidade é um tema que merece destaque por ser fundamental à inclusão de pessoas com deficiência e com mobilidade reduzida, caracterizadas na Lei no. 10.098/2000, como tendo “temporária ou permanentemente [...] limitada sua capacidade de relacionar-se com o meio e de utilizá-lo” (BRASIL, 2000).

A partir da década de 1990 houve a oficialização da educação inclusiva e, para que esta seja inclusiva faz-se necessário, também, que os espaços oferecidos pelas instituições sejam igualmente inclusivos. De acordo com Sasaki (2009) "a preocupação das universidades em adaptar seus ambientes físicos tem sido tão tímida quanto à preocupação em adaptar o acesso ao currículo e em preparar os professores dos cursos superiores".

Dessa forma, a fim de colaborar com acessibilidade dos ambientes físicos da UnB, em pesquisa de doutorado no com foco em acessibilidade para pessoas com deficiência visual para o Programa de Pós-graduação em Engenharia Mecatrônica (PPMEC), a pesquisadora Mayra Batista Corrêa realizou pesquisa semi-estruturada a fim de catalogar os principais problemas que impedem o acesso pelo dos estudantes com e sem deficiência visual nas dependências da universidade.

Um desafio para a construção IPSs vem do fato de que técnicas baseadas no uso do GNSS não fornecem precisão adequada em recintos fechados, o que torna necessário o uso de outras tecnologias (KUNHOTH et al., 2020).

Os resultados da pesquisa apontam que os alunos tem dificuldade na localização e navegação dentro de ambientes *indoor*, principalmente ao tentar encontrar salas de aula e banheiros na universidade. Dessa forma, a fim de tentar sanar o problema de acessibilidade na localização dos usuários em ambientes internos, este trabalho propõe uma metodologia para o desenvolvimento de um IPS na forma de aplicativo (MovUnB) para *smartphones* capaz de guiar usuários com ou sem deficiência visual dentro de ambiente *indoor* na UnB.

1.3 Hipótese do trabalho

Este trabalho está norteado pela seguintes hipóteses:

1. É possível construir uma aplicação de localização *indoor* utilizando as técnicas de PDR e *fingerprinting* de sinais Wi-Fi, combinadas por meio do uso de filtro de Kalman linear.
2. A precisão desta aplicação é suficiente para que pessoas com deficiência visual localizem salas de aula e banheiros dentro de ambientes *indoor*.
3. É possível construir tal aplicação para atender à comunidade universitária sem adição de despesas para a universidade, garantindo uma solução de posicionamento de baixo custo com um erro compatível com as necessidades de usuários com deficiência visual.

1.4 Objetivo Geral

Tendo em conta as motivações deste trabalho, visamos desenvolver uma metodologia para implantação de IPS, sem adição de custos, com acessibilidade para pessoas com ou sem deficiência visual.

1.4.1 Obetivos específicos

1. Implantação de algoritmo de *fingerprint* capaz de determinar posição do usuário em ambiente *indoor* a partir de leituras de sinais *Wi-Fi*.
2. Utilização da técnica de PDR para determinar o deslocamento do usuário.
3. Realização do cálculo de rotas partindo da localização, em tempo real, do usuário até um destino selecionado.
4. Elaboração de interface acessível seguindo princípios do *Web Content Accessibility Guidelines* (WCAG).

1.5 Aspectos metodológicos

Para o desenvolvimento da metodologia proposta, foram escolhidas técnicas baseadas na utilização de sinais Wi-Fi e PDR amplamente difundidas na literatura (POULOSE; KIM; HAN, 2019; KUNHOTH et al., 2020; BOORANAWONG et al., 2021). Esta escolha foi feita pois permite o aproveitamento da estrutura de *internet* provida pela UnB e também das unidades de medição inercial *IMU* presentes nos *smartphones* dos usuários. Dessa forma, o aplicativo pôde ser implantado sem a adição de custos para a universidade. Assim, foram seguidos os seguintes passos:

1. Seleção de ambiente para validação do protótipo.
2. Obtenção da planta baixa do ambiente selecionado.
3. Obtenção do ângulo azimute do edifício em que se encontra o ambiente escolhido.
4. Georreferenciamento da planta baixa do ambiente escolhido.
5. Coleta de dados de intensidade de sinais Wi-Fi em múltiplas posições dentro do ambiente.
6. Desenvolvimento da funcionalidade de cálculo da posição por meio da realização do PDR, fingerprinting e multilateração de sinais Wi-Fi.
7. Implantação de filtro de Kalman utilizado para combinar as estimativas de posição providas pelos métodos acima.
8. Avaliação das incertezas na estimativa da posição.
9. Implantação do algoritmo A* para cálculo de rotas dentro do ambiente selecionado.
10. Realização de teste de rotas geradas.
11. Criação de casos de uso do aplicativo.
12. Desenvolvimento do aplicativo com interface acessível.

1.6 Contribuições do trabalho de graduação

- Validação de metodologia para desenvolvimento de sistemas de posicionamento *indoor*.
- Desenvolvimento de um sistema de localização *indoor*, MovUnb, com foco na acessibilidade para pessoas com deficiência visual.
- Desenvolvimento de algoritmos de cálculo de posição que não necessitam de adicionar custos à universidade para o funcionamento .

2 Fundamentação teórica

Neste capítulo, são apresentados os principais conceitos necessários para o desenvolvimento da metodologia proposta: acessibilidade, interfaces acessíveis, sistemas de referência de coordenadas e técnicas usadas para estimar a posição do usuário, sendo estas: 1) *finger-printing* de sinais Wi-Fi, 2) multilateração, 3) PDR e 4) filtros de Kalman.

2.1 Princípios de acessibilidade

O termo acessibilidade é definido como: “a possibilidade e condição de alcance para utilização, com segurança e autonomia, dos espaços,[...] e das edificações.” (BRASIL, 2004). A acessibilidade pode ser definida também como a facilidade no uso de um produto, ambiente ou serviço para pessoas com a mais ampla variedade de capacidades, não estando limitada a pessoas com deficiência (ISO, 2008). A lei nº 13.146, artigo 53º, assegura que acessibilidade é um direito que garante à pessoa com deficiência uma vida independente e a plena capacidade de exercer seus direitos de cidadania (BRASIL, 2015).

A acessibilidade é, também, um parâmetro da usabilidade, que por sua vez é a medida em que um produto ou sistema pode oferecer eficácia, eficiência e satisfação aos usuários na execução de suas tarefas (ISO, 2008). A usabilidade é composta pelas seguintes subcaracterísticas: reconhecimento de adequação; aprendizagem; operabilidade; proteção contra erros do usuário; estética da interface do usuário; e acessibilidade.

Para que uma aplicação seja considerada acessível, é necessário que suas interfaces sejam acessíveis, visto que em um ambiente informatizado, é por meio de interfaces que se apresentam painéis com informações, dados, controles, comandos e mensagens (ABREU CYBIS; BETIOL; FAUST, 2015).

2.2 Interfaces acessíveis

De acordo com Nascimento e Amaral (NASCIMENTO; AMARAL, 2010) (p. 14-15) “interface gráfica é a zona em que se realiza a interação entre usuário e programa”. Assim, construir interfaces com foco na usabilidade é fundamental para a fidelização, redução de gastos, geração de sentimento de pertencimento do usuário, além de contribuir com a interação entre usuário e sistema, principalmente quando o público possui alguma especificidade.

Para que as interfaces tenham usabilidade, estas devem ter cinco atributos: 1) facilidade de aprendizado; 2) eficiência durante o uso; 3) facilidade de compreensão; 4) baixo índice de erros e, 5) experiência satisfatória para usuários iniciantes e experientes (NIEL-

SEN, 1994). Dessa forma, cada componente utilizado na interface deve possuir parâmetros espaciais como tamanho, espaçamento em diferentes direções, geometria do componente, entre outros.

2.3 Acessibilidade para dispositivos móveis

A fim de garantir a acessibilidade em aplicativos para dispositivos móveis, são utilizados padrões como o SiDI¹ e o guia de desenvolvimento *Android*². O SiDI apresenta um guia com orientações de usabilidade aos desenvolvedores, *designers* e testadores de aplicativos que contemplam: 1) *design* minimalista, 2) fluxo natural, 3) coerência externa e interna, 4) evitar excesso de informações e 5) contraste de cores.

Assim como o SiDI, o guia de desenvolvimento *Android*, para garantir a usabilidade de aplicações utilizadas em dispositivos móveis, existe um tamanho mínimo recomendado para componentes clicáveis. Dessa forma, deve-se levar em conta que a aplicação será utilizada em múltiplos dispositivos, os quais podem possuir diferentes tamanhos de tela e densidades de *pixel*.

Para levar em consideração as dimensões de tela dos diferentes aparelhos em que a aplicação poderá ser utilizada, o *density independent pixel* (dp) é usado como referência para espaçamentos e tamanhos de componentes³. Um dp vale cerca de um *pixel* em uma tela de densidade média 160 *dots per inches* (dpi). O valor recomendado de tamanho mínimo para componentes clicáveis (em uma aplicação), tanto para a largura quanto para a altura, é de 48 dp⁴.

As orientações providas pela Google para o desenvolvimento de aplicativos com acessibilidade⁵ são: 1) *elementos de etiqueta* - oferecer aos usuários etiquetas descritivas e úteis para cada elemento de UI interativo do app; 2) *extensão de widgets (componentes) do sistema* - os *widgets* fornecidos pelo sistema já têm a maioria dos recursos de acessibilidade necessários e, por isso, recomenda-se estender esses componentes fornecidos pelo sistema ao invés de criar novos; 3) *múltiplos indicadores de elementos* - para auxiliar os usuários com deficiências visuais faz-se necessário o uso de outros tipos de indicadores além da cor para distinguir os elementos da UI nas telas. Essas técnicas podem incluir o uso de diferentes formas ou tamanhos, disponibilização de texto, padrões visuais ou adição de retorno ao clicar, tendo como mecanismo de resposta o áudio ou vibração, a fim de marcar as diferenças dos elementos e, 4) *conteúdo de mídia mais acessível* - incluir controles que permitam aos usuários pausar ou interromper a mídia, mudar o volume e alternar as legendas. Por exemplo,

¹ <https://www.sidi.org.br/guiadeacessibilidade/inicio>

² <https://developer.android.com/guide/topics/ui/accessibility/apps>

³ <https://developer.android.com/guide/topics/ui/accessibility/apps>

⁴ developer.android.com/guide/topics/ui/accessibility/apps

⁵ <https://developer.android.com/guide/topics/ui/accessibility/apps?hl=pt-br>

se um vídeo apresentar informações essenciais para a conclusão de um fluxo de trabalho, deve-se disponibilizar o mesmo conteúdo em um formato alternativo, como uma transcrição.

Outra forma de garantir a acessibilidade em dispositivos móveis consiste na utilização de ferramentas de teste de acessibilidade automatizadas, como o *Apple Accessibility Inspector*⁶ para aplicativos voltados para o sistema operacional (SO) *IOS* e o *Google Accessibility Scanner*⁷ para *Android*. Tais ferramentas auxiliam na identificação dos componentes sem a acessibilidade devida para contemplar diferentes públicos.

2.4 Localização indoor

Um IPS deve conseguir localizar o usuário no espaço (KUNHOTH et al., 2020). Para realizar esta tarefa em ambientes fechados foram desenvolvidas diversas técnicas com diferentes abordagens mais ou menos adequadas umas que as outras a depender da situação em que são aplicadas. (KUNHOTH et al., 2020).

Segundo Mao, Fidan e Anderson (2007) e Kunhoth et al. (2020), os princípios utilizados nas principais técnicas usadas para determinar a posição de um dispositivo em ambientes *indoor* são: a) trilateração (ou multilateração), b) triangulação, c) detecção de proximidade, d) análise de cenários e e) Dead Reckoning. A seguir será feita uma breve descrição de cada um deles:

a) A trilateração, ou multilateração, consiste em determinar a posição de um ponto de interesse com base nas distâncias entre este ponto e outros cujas posições são conhecidas previamente. Para isso, no mínimo três pontos de posições conhecidas são necessários (BOORANAWONG et al., 2021). Uma maneira de aplicar este princípio é a partir da utilização de um conjunto de sinais como Wi-Fi ou *bluetooth*. Para tanto, é utilizada a intensidade dos sinais emitidos por transmissores para obter as distâncias destes até um receptor, para então calcular sua posição (BOORANAWONG et al., 2021).

b) A triangulação, assim como a multilateração, utiliza pontos cujas posições são previamente conhecidas para estimar a posição de um novo ponto de interesse. Porém, na triangulação, a posição é determinada por um conjunto de ângulos e relações trigonométricas (NUAIMI; KAMEL, 2011). Para a implementação deste método podem ser utilizados os valores de *Angle Of Arrival (AOA)* de um conjunto de sinais vindos de transmissores em diferentes posições em conjunto com a orientação do receptor a fim de estimar sua posição (AMUNDSON; KOUTSOUKOS, 2009).

c) No princípio da proximidade, é empregada uma estrutura de antenas (transmissores) com posições fixas. Ao identificar o dispositivo de interesse (receptor), a antena mais próxima é utilizada no cálculo da posição do objeto. Caso o receptor seja identificado por

⁶ <https://www.apple.com/accessibility/vision/>

⁷ <https://encurtador.com.br/CGMSY>

múltiplas antenas, aquela que emite o sinal mais potente é então utilizada para determinar a posição (NUAIMI; KAMEL, 2011).

d) A análise de cenários consiste em extrair características do ambiente em que estará o objeto a ser localizado como, por exemplo, um corredor. Para localizar o objeto dentro deste ambiente, é usado um algoritmo de comparação/reconhecimento (NUAIMI; KAMEL, 2011). Um exemplo de aplicação deste princípio é o *fingerprinting* de sinais Wi-Fi, em que são realizadas múltiplas leituras de sinais Wi-Fi vindos de diferentes pontos de acesso, com o receptor em várias posições dentro de um cômodo em uma etapa de treinamento, offline. As leituras feitas são salvas em uma base de dados, para que, em uma fase online, quando for feita uma nova leitura, a posição em que esta foi realizada, possa ser estimada (POULOSE; KIM; HAN, 2019).

e) O *dead reckoning* realiza as estimativas de posição de um objeto que se move no espaço a partir do conhecimento de sua posição no instante anterior e dos parâmetros que descrevem o movimento do objeto. Tais parâmetros podem ser: velocidade, aceleração, tempo de realização do movimento, entre outros. Quando o objeto cuja posição a ser estimada é um pedestre este procedimento passa a ser chamado de *PDR* (WU et al., 2019).

Dentre os cinco princípios listados acima, são utilizados, neste projeto, o *dead reckoning* e reconhecimento de cenários nas implementações das técnicas de *PDR* e *fingerprinting* de sinais Wi-Fi respectivamente.

2.5 Pedestrian dead reckoning

Os Algoritmos de *PDR* podem ser implementados utilizando duas abordagens: 1) estimação da ocorrência de passos e direção do movimento e 2) estimação do vetor deslocamento, por meio da integração no tempo das leituras vindas de sensores inerciais, tais como: giroscópio, acelerômetro e magnetômetro, (YAO et al., 2020). Aos sistemas que utilizam a primeira abordagem é dado o nome *Step-and-Heading Systems (SHS)*, enquanto os sistemas que utilizam a segunda abordagem são chamados de *Inertial Navigation Systems (INS)* (WU et al., 2019).

Na abordagem *SHS*, utilizada neste trabalho, a identificação de passos pode ser feita a partir da detecção de valores mínimos e máximos do componente da aceleração, do celular, perpendicular ao movimento do usuário. Além disso, a orientação de cada passo deve ser determinada, para que, com a informação do tamanho de cada passo, o deslocamento possa ser calculado.

Tanto a abordagem *SHS* quanto a *INS* apresentam erros que se acumulam o tempo, ou com cada passo dado pelo usuário. Na abordagem *INS*, é feita uma integral dupla da aceleração em relação ao tempo, o que resulta no acúmulo de erros e ocasiona grandes

discrepâncias entre o deslocamento estimado e o real (WU et al., 2019). Da mesma forma, na abordagem SHS, os passos não detectados e as incertezas na estimação da direção do movimento geram erros que se acumulam durante o movimento (WU et al., 2019; KUNHOTH et al., 2020).

Devido ao acúmulo de erros inerentes à utilização da técnica do PDR, é comum que esta seja utilizada em conjunto com algoritmos que atualizam periodicamente a posição estimada, a fim de mitigar tal acúmulo (KUNHOTH et al., 2020; CHEN et al., 2015; YAO et al., 2020). A posição estimada pode ser atualizada a partir do uso da técnica *fingerprinting* de sinais Wi-Fi, promovendo maior precisão na posição obtida (RAITOHARJU; NURMINEN; PICHÉ, 2015; LI et al., 2016; POULOSE; KIM; HAN, 2019; YAO et al., 2020).

2.6 Wi-Fi Fingerprinting

Em um contexto de localização *indoor*, o *fingerprinting* consiste em fazer a leitura e armazenamento de um conjunto de dados associados a diferentes localizações, fase de coleta. Assim, em um momento futuro, os dados armazenados poderão ser utilizados para determinar as posições em que são feitas novas leituras, fase de uso (NUAIMI; KAMEL, 2011).

No caso do *fingerprinting* de sinais de rádio frequência como *Wi-Fi* ou *bluetooth*, os dados utilizados são as intensidades de múltiplos sinais lidos em diferentes posições (POULOSE; KIM; HAN, 2019), uma vez que são esperadas intensidades de sinal aproximadamente constantes para cada posição em um ambiente que não sofra mudanças (BOORANAWONG et al., 2021).

Cada intensidade do sinal lido é representada por um número denominado *Received Signal Strength Indicator* (RSSI), um número inteiro negativo que quanto mais próximo a zero maior a intensidade indicada. O RSSI é uma aproximação do *Received Signal Strength* (RSS), que por sua vez é um valor em decibéis que representa a potência do sinal recebido (BOORANAWONG et al., 2021).

A partir dos valores de RSSI armazenados, é possível utilizar algoritmos de aprendizado de máquina como o *K Nearest Neighbours* (KNN) e máquinas de vetores de suporte (SVM) para estimar a posição em que novas leituras de sinal são feitas (POULOSE; KIM; HAN, 2019; YAO et al., 2020).

2.7 Algoritmo K nearest neighbours

O algoritmo KNN é comumente utilizado para classificação, isto é, identificação da categoria a qual um determinado dado pertence (KUBAT, 2017). A classificação é realizada em dados, objetos, compostos por um vetor de atributos. Cada atributo é uma variável que

pode ser preenchida por valores categóricos ou numéricos. A classificação consiste em encontrar o valor *a priori* desconhecido de um atributo categórico a partir de outros atributos com valores previamente conhecidos (KUBAT, 2017).

Os atributos utilizados para realizar a classificação são chamados de atributos preditores. A classificação é feita ao avaliar a semelhança entre outros dados já categorizados que são considerados parecidos com o dado de interesse, ou seja, possuem atributos preditores com valores iguais ou próximos (KUBAT, 2017).

O algoritmo *KNN* também pode ser utilizado para regressão, estimação de valores numéricos. No trabalho de Poulouse, Kim e Han (2019), o *KNN* é utilizado em um contexto de localização *indoor* como parte da implantação do *fingerprinting* de sinais Wi-Fi. Neste, os valores *RSSI* de um conjunto de sinais lidos em diferentes posições são registrados durante a fase de coleta do *fingerprinting* e utilizados como atributos preditores para obter as coordenadas x e y das posições em que são realizadas novas leituras.

Na implementação de Poulouse, Kim e Han (2019), ao se realizar uma nova leitura, os valores *RSSI* obtidos nesta são comparados com os valores armazenados, obtidos durante a fase de coleta. A comparação visa encontrar, entre os valores armazenados, os valores *RSSI* mais próximos aos obtidos na nova leitura. Uma vez encontrados os valores mais próximos, a leitura em que esses valores foram encontrados é identificada. A posição em que esta leitura foi feita é então atribuída à posição (x, y) da nova leitura.

Ainda no mesmo trabalho, o ponto de maior proximidade é o que possui menor distância euclidiana mostrada na equação 2.1 (POULOSE; KIM; HAN, 2019).

$$D_{ip} = \sqrt{(RSSI_{i1} - RSSI_{p1})^2 + (RSSI_{i2} - RSSI_{p2})^2 + \dots + (RSSI_{in} - RSSI_{pn})^2} \quad (2.1)$$

Onde, D_{ip} é a distância entre o conjunto de leituras realizadas no ponto i , durante a fase de coleta e a leitura dos sinais realizada no ponto p , na fase de uso, cujas coordenadas devem ser encontradas. O valor $RSSI_{mn}$ corresponde à intensidade do sinal n no ponto m .

2.8 Filtros de Kalman

O filtro de Kalman é um método eficaz de estimação que opera de maneira recursiva, minimizando o erro quadrático médio. Em outras palavras, ele oferece uma solução eficiente para o problema dos mínimos quadrados. (CINTRA; CAMPOS VELHO; TODLING, 2010).

Existem diferentes abordagens e derivações do filtro de Kalman: Filtro de Kalman Linear (LKF), Filtro de Kalman Extendido (EKF), Filtro de Kalman sem cheiro (UFK) e Filtro de Partículas (FP) (SIMON, 2006; POULOSE; KIM; HAN, 2019; RAITOHARJU; NURMINEN; PICHÉ, 2015). Este trabalho utiliza a forma linear do filtro a fim de combinar

as estimativas de posição do PDR e *fingerprinting* de sinais Wi-Fi, como feito por (POULOSE; KIM; HAN, 2019).

Para o uso de filtros de Kalman em um sistema, este deve ser representado no espaço de estados (SIMON, 2006). Esta representação pode ser feita na forma matricial, como nas equações 2.2 e 2.3 (SIMON, 2006).

$$x_k = F_{k-1}x_{k-1} + G_{k-1}u_{k-1} + w_{k-1} \quad (2.2)$$

$$y_k = H_k x_k + v_k \quad (2.3)$$

A equação 2.2 apresenta a mudança de estados do sistema de um instante k-1 para um instante k. Nesta equação, F_{k-1} é a matriz de transição de estados do instante de tempo k-1 para o instante k. G_{k-1} é a matriz de entradas de controle, que descreve como as entradas de controle, representadas pelo vetor u_{k-1} afetam o sistema. Os termos x_k e x_{k-1} são, respectivamente, os vetores que contém as variáveis de estado do sistema nos instantes k e k-1. Por último w_{k-1} é o vetor de ruídos presentes na entrada de controle (SIMON, 2006).

A equação 2.3 apresenta a relação entre as variáveis de estado x_k e o vetor com as variáveis medidas de interesse, y_k . Nesta equação, H_k é a matriz de transformação que descreve a relação entre as variáveis de estado e as variáveis medidas, presentes em y_k . O vetor v_k representa os ruídos de medição (SIMON, 2006).

Para a utilização do filtro, os ruídos w_{k-1} e v_k devem possuir média 0, além disso, os ruídos de diferentes estados não devem estar correlacionados entre si. A aplicação do FK é feita em duas etapas: predição e correção (SIMON, 2006; POULOSE; KIM; HAN, 2019).

2.8.1 Predição

Na etapa de predição, a equação 2.4 é utilizada para estimar o vetor de estados x_k a partir do conhecimento sobre o sistema no instante k-1. A notação $\hat{x}_{k|k-1}$ na equação 2.4 para o vetor de estados x_k se refere ao fato de que está sendo feita uma estimativa de x_k a partir do conhecimento do sistema no instante k-1 (SIMON, 2006).

$$\hat{x}_{k|k-1} = F_{k-1}x_{k-1|k-1} + G_{k-1}u_{k-1} \quad (2.4)$$

Ainda nesta etapa, é feito o cálculo da matriz P de covariância das incertezas na estimativa de estados a partir da equação 2.5.

$$\hat{P}_{k|k-1} = F_{k-1}P_{k-1|k-1}F_{k-1}^T + Q_{k-1} \quad (2.5)$$

A matriz Q_{k-1} representa a matriz de covariâncias do ruído do processo, representado por w na equação 2.2. A matriz Q_k pode ser calculada por: $Q_k = E(w_k \cdot w_k^T)$ (SIMON, 2006).

2.8.2 Correção

Na etapa de correção, são adicionadas informações de medições feitas no instante k à predição de x_k realizada no instante $k-1$ para obter uma nova estimativa para x_k (SIMON, 2006).

O primeiro passo para realizar a correção é obter a diferença entre a medição feita no instante k e a predição deste valor no instante, feita no instante $k-1$. Em seguida, deve ser calculado o ganho K_k . A partir daí, é possível fazer uma nova estimação para \hat{x}_k e P_k .

$$z_k = y_k - \hat{x}_{k|k-1} \quad (2.6)$$

$$K_k = P_{k|k-1} H_k^T R_k^{-1} \quad (2.7)$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k z_k \quad (2.8)$$

$$P_{k|k} = (I - K_k H_k) P_{k|k-1} \quad (2.9)$$

Para utilização do FK no cálculo de posição, as estimativas obtidas neste trabalho pelos métodos de *fingerprinting* e PDR a serem combinadas devem ser representadas a partir do mesmo sistema de coordenadas .

2.9 Sistemas de coordenadas

A posição de um objeto pode ser expressa por um conjunto de números, coordenadas, que podem ser definidas de diferentes maneiras. A forma como as coordenadas são definidas é denominada Sistema de Referência de Coordenadas (SRC) (JANSSEN, 2009).

Um SRC é uma abstração utilizada como modelo para se obter as coordenadas. A implementação de um SRC é chamada de *Datum*. Um *Datum* define, por exemplo, a origem que será usada como referência para as coordenadas, assim como, os valores das coordenadas de um dado ponto na Terra em um período específico de tempo (JANSSEN, 2009).

Um exemplo de *Datum* é o *World Geodetic System 1984 (WGS84)*, que é usado nos sistemas de *Global Positioning System GPS*. O *WGS84* aproxima o formato da Terra como um elipsóide de revolução. Por definição, o planeta Terra tem o formato de um geoide dado pela superfície de potencial gravitacional uniforme que melhor se aproxima do nível médio

do mar. O WGS84 utiliza como referência o elipsoide que melhor se aproxima globalmente deste geóide (JANSSEN, 2009).

2.9.1 Coordenadas geodésicas

As coordenadas geodésicas, utilizadas no WGS84, são expressas como latitude, longitude e altura elipsoidal. A latitude de um ponto na superfície da Terra é dada pelo ângulo, em graus, entre o plano equatorial e a reta que cruza o ponto em questão e é normal à superfície do elipsoide definido.

A longitude de um ponto é o ângulo entre o Meridiano de Referência Internacional (IRM) e o meridiano que passa este. Por último, a altura elipsoidal de um ponto é a distância entre este e a superfície do elipsoide de revolução (SAJEEVAN, 2008). As coordenadas latitude, longitude e altura elipsoidal são ilustradas na Figura 1 por ϕ , λ , e h respectivamente.

O elipsoide de revolução, por sua vez, é definido por dois parâmetros, sendo eles, o semi-eixo maior (a) e o inverso do achatamento (f^{-1}) que é dado pela equação 2.10. Nesta equação, a é o semi-eixo maior, b é o semi-eixo menor e f^{-1} é o inverso do achatamento (JANSSEN, 2009). Os semi-eixos a e b são ilustrados na 1.

$$f^{-1} = \frac{a}{a - b} \quad (2.10)$$

Outra maneira de expressar a posição de um ponto na superfície da Terra, além das coordenadas geodésicas, é a partir da utilização do SRC *Earth-Centered Earth-Fixed* (ECEF), que utiliza coordenadas cartesianas X , Y e Z com origem no centro de massa de gravidade da Terra. O eixo Z atravessa o eixo de rotação da Terra, o eixo X atravessa o IRM no plano equatorial e o eixo Y pode ser obtido a partir da regra da mão direita aplicada do eixo Z ao eixo X (MARKIČ; DONAUBAUER; BORRMANN, 2018). É possível converter coordenadas geodésicas para o sistema ECEF ao utilizar a equação 2.11 (LIGAS; BANASIK, 2011).

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} (N + h) \cdot \cos(\phi) \cdot \cos(\lambda) \\ (N + h) \cdot \cos(\phi) \cdot \sin(\lambda) \\ [(1 - e^2) \cdot N + h] \cdot \sin(\phi) \end{bmatrix} \quad (2.11)$$

Onde ϕ , λ e h correspondem a latitude, longitude e altura geodésicas respectivamente. Os valores de N e e^2 são calculados por:

$$e^2 = 1 - \left(\frac{b}{a}\right)^2 \quad (2.12)$$

e

$$N = \frac{a}{\sqrt{1 - e^2 \cdot \sin^2(\phi)}} \quad (2.13)$$

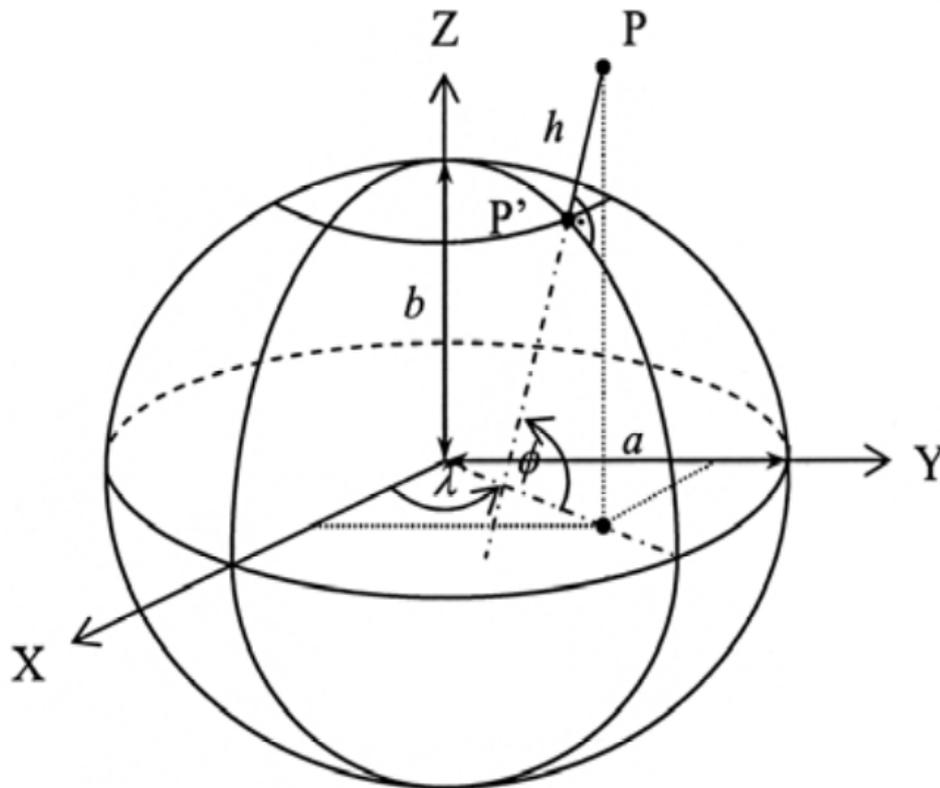


Figura 1 – Elipsoide ilustrando coordenadas cartesianas e geodésicas. (Fonte: Janssen (2009))

O cálculo das coordenadas geodésicas de um ponto P a partir das coordenadas cartesianas X , Y e Z pode ser realizado por meio do método proposto por Lin e Wang (1995). Neste método, são encontrados os valores de X_e , Y_e e Z_e , que são as coordenadas do ponto P' que se encontra sobre a superfície do elipsoide e é atravessado pela reta que também passa pelo ponto P e é normal à superfície do elipsoide. A partir dessas coordenadas são encontrados os valores de latitude e altura elipsoidal a partir das equações 2.14 e 2.15, onde $W_e^2 = X_e^2 + Y_e^2$ e $W^2 = X^2 + Y^2$, enquanto a longitude é calculada a partir da equação 2.16.

$$\phi = \text{Tan}^{-1} \left(\frac{a^2 Z_e}{b^2 W_e} \right) \quad (2.14)$$

$$h = \sqrt{(W - W_e)^2 + (Z - Z_e)^2} \quad (2.15)$$

$$\lambda = \text{Tan}^{-1} \left(\frac{Y}{X} \right) \quad (2.16)$$

Os valores de W_e e Z_e podem ser encontrados ao substituir o valor de m que satisfaz a equação 2.17 nas equações 2.18 e 2.19. O valor de m é encontrado por meio do método de *Newton-Raphson*.

$$\frac{W^2}{\left(a + \frac{2m}{a}\right)^2} + \frac{Z^2}{\left(b + \frac{2m}{b}\right)^2} - 1 = 0 \quad (2.17)$$

$$W_e = \frac{W}{1 + \frac{2m}{a^2}} \quad (2.18)$$

$$Z_e = \frac{Z}{1 + \frac{2m}{b^2}} \quad (2.19)$$

2.9.2 Coordenadas locais

Outro sistema de coordenadas utilizado neste projeto é o sistema de diferenças locais. Este sistema utiliza a origem em um ponto arbitrário e eixos N, E e U que coincidem com as direções norte, leste e cima. Desta forma as coordenadas de um ponto p são descritas pelos valores: Δn , Δe e Δu , Figura 2.

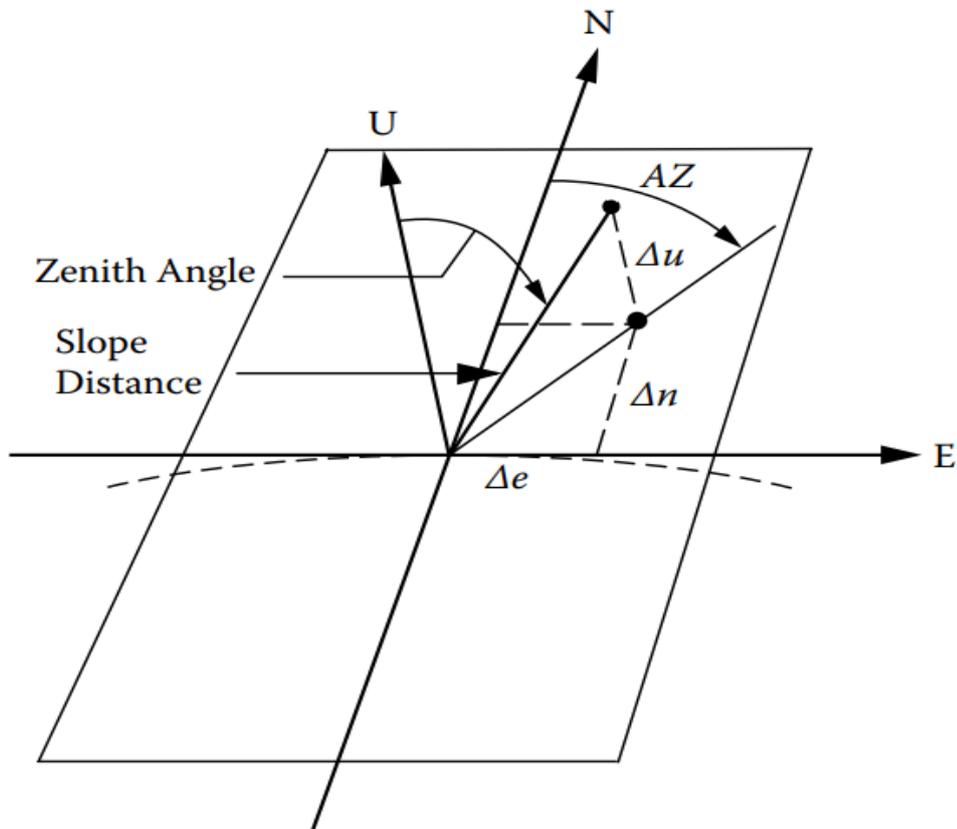


Figura 2 – Coordenadas locais. (Fonte: Burkholder (2008))

A coordenada Δn apresenta módulo igual à distância entre o eixo E e a projeção do ponto p sobre o plano formado pelos eixos E e N e possui valor positivo caso p esteja ao norte da origem. A coordenada Δe corresponde à distância entre o eixo N a projeção de p e tem valor positivo se p está a leste da origem e valor negativo caso contrário. Por último, Δu é

distância entre o plano definido pelos eixos E e N e o ponto p. O sinal de Δu é positivo se p se encontra acima do plano e negativo caso esteja abaixo (BURKHOLDER, 2008).

2.10 Matrizes de rotação e azimutes

Uma matriz de rotação é um conjunto de equações na forma matricial que permite rotacionar um ou mais eixos de coordenadas para atingir uma orientação desejada. Matrizes de rotação podem ser usadas para converter coordenadas cartesianas de um ponto em relação a um sistema de coordenadas para outro (BURKHOLDER, 2008). A equação 2.20 mostra o uso de uma matriz de rotação para converter diferenças em coordenadas representadas no sistema ECEF para coordenadas locais.

$$\begin{bmatrix} \Delta e \\ \Delta n \\ \Delta u \end{bmatrix} = \begin{bmatrix} -\text{sen}(\lambda) & \cos(\lambda) & 0 \\ -\text{sen}(\phi)\cos(\phi) & -\text{sen}(\phi)\text{sen}(\lambda) & \cos(\phi) \\ \cos(\phi)\cos\lambda & \cos(\phi)\text{sen}(\lambda) & \text{sen}(\phi) \end{bmatrix} \cdot \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} \quad (2.20)$$

Na equação 2.20 λ e ϕ são, respectivamente, a latitude e longitude da origem definida no sistema de coordenadas locais. A matriz utilizada como primeiro operando da multiplicação é uma matriz ortogonal, ou seja, sua inversa é igual à sua transposta. Dessa forma, é possível fazer conversão de coordenadas locais para diferenças de coordenadas ECEF ao transpor a matriz de rotação utilizada (BURKHOLDER, 2008).

Outra aplicação de matrizes de rotação para conversão entre sistemas de coordenadas consiste na conversão entre um sistema definido a partir das paredes de um edifício, para o sistema de coordenadas locais descrito na seção 2.9 em conjunto com o ângulo azimute do edifício (RUAN et al., 2018).

Podem ser encontradas diferentes definições de azimute na literatura, sendo elas azimute 3-D, azimute entre linhas geodésicas, azimute magnético, entre outras (BURKHOLDER, 2008; PIMENTEL et al., 2012). Para fins deste trabalho, foi utilizado o azimute 3-D.

O azimute 3-D pode ser definido como um ângulo observado no sentido horário entre um meridiano e uma direção no plano tangente à superfície em que o observador se encontra (BURKHOLDER, 2008; YETKİN; BILGINER, 2021). Ele pode ser calculado a partir das coordenadas locais Δn e Δe descritas na seção 2.9 a partir da equação 2.21 (BURKHOLDER, 2008). Onde α_{3D} corresponde ao ângulo AZ, mostrado na Figura 2 na seção 2.9.1, que ilustra o ângulo azimute de um segmento partindo da origem do sistema de coordenadas locais até um ponto p.

$$\alpha_{3D} = \tan^{-1}\left(\frac{\Delta e}{\Delta n}\right) \quad (2.21)$$

Neste sentido, esta pesquisa visa fomentar a discussão da acessibilidade em ambientes *indoor* para pessoas com deficiência visual.

3 Metodologia e implementação

Neste capítulo, serão tratados os procedimentos utilizados para o desenvolvimento do MovUnB, que se dividem em: 1) criação do caso de uso com base no protótipo de interface, desenvolvido no Figma¹ durante a pesquisa em andamento de Mayra Batista Corrêa; 2) seleção do ambiente testes, corredor da sala dos professores do departamento de engenharia mecânica, localizado no bloco D da FT/UnB; 3) desenvolvimento de algoritmos para cálculo da posição do usuário utilizando as técnicas de *fingerprinting* de sinais Wi-Fi e PDR; 4) obtenção do ângulo azimute do corredor utilizado como ambiente de testes, para que as estimativas de posição feitas na etapa anterior possam ser combinadas; 5) uso do filtro de Kalman para combinação dos resultados obtidos pelos dois algoritmos mencionados na terceira etapa; 6) implantação do algoritmo A* para geração de rotas no ambiente de testes selecionado; 7) desenvolvimento dos elementos visuais e de acessibilidade da interface.

3.1 Criação do caso de uso do aplicativo

Para a criação do caso de uso do MovUnB, figura 4, foram utilizados os requisitos levantados para a construção do protótipo de interface, desenvolvido no Figma, editor de gráficos vetoriais baseado na web, durante a pesquisa em andamento de Mayra Batista Corrêa.

O protótipo da interface permitiu o acesso funcional e estético da aplicação, instruindo o desenvolvimento das ações realizadas pelo aplicativo.

Com base nestas ações, foi criado o caso de uso que apresenta as principais funcionalidades do MovUnB, assim como a relação entre os atores: *Usuário* e *Sistema operacional*. O *Sistema operacional* obtém a posição inicial do *Usuário*, calcula a rota pretendida e fornece instruções para que esse realize a navegação. O *Usuário* se relaciona com o *Sistema operacional* por meio da interface acessível do MovUnB.

3.2 Seleção do ambiente de testes

Após a criação do caso de uso, foi selecionado como ambiente de testes o corredor da sala dos professores do departamento de engenharia mecânica, localizado no bloco D da FT/UnB. A escolha se deu pela natureza do ambiente, que tem estrutura de rede fixa e baixo fluxo de pessoas. Além disso, foi solicitado, que os roteadores que se encontram nas salas dos professores não mudassem de local ou fossem desligados, permitindo replicabilidade

¹ <https://www.figma.com/community/file/1151228795757403534/login-e-cadastro>

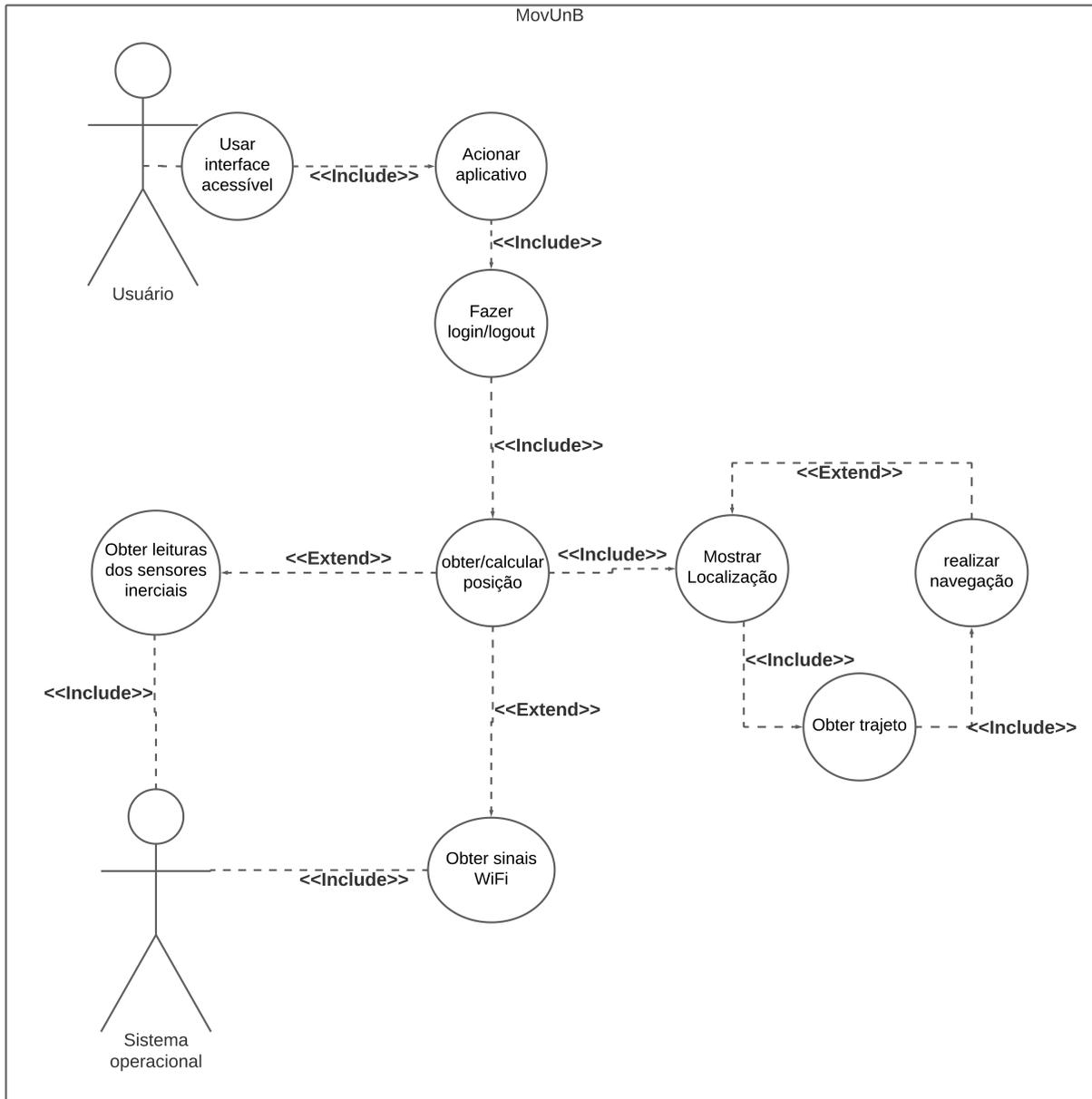


Figura 4 – Diagrama de casos de uso

dos testes em diferentes datas. Visa-se poder replicar tal solução aos demais departamentos e campi.

3.3 Desenvolvimento dos algoritmos para cálculo de posição

Uma vez selecionado o local de testes para validação do aplicativo, foi realizada a implementação da técnica do *fingerprinting* de sinais Wi-Fi para o cálculo da posição do usuário.

3.3.1 Implementação da técnica de *fingerprinting* de sinais Wi-Fi

A técnica de *fingerprinting* foi usada para obter a posição inicial do usuário, assim como, realizar correções periódicas na posição estimada. Isto é necessário devido ao acúmulo de erros característico da técnica de *PDR*, conforme discutido nas seções 2.5 e 2.6.

O *fingerprinting* foi realizado em duas etapas: 1) coleta de dados e 2) desenvolvimento do algoritmo de cálculo de posição. Durante a coleta de dados, foram lidas as intensidades dos sinais *Wi-Fi* emitidos pelos roteadores presentes no ambiente de testes. Foram utilizados sete roteadores, que emitem um total de nove sinais *Wi-Fi*.

Para a realização da coleta, foi desenvolvida uma aplicação dedicada à leitura e registro das intensidades dos sinais disponíveis dentro do ambiente de testes. A aplicação desenvolvida registra a intensidade de cada sinal lido, a posição em que a leitura foi feita, e a direção (norte, sul, leste ou oeste) para a qual estava apontado o celular no momento da leitura.

Foram realizadas 30 leituras para cada direção (norte, sul, leste e oeste) em 112 posições distintas dentro do ambiente de testes, totalizando 120 leituras por posição.

As posições das leituras foram armazenadas em uma planilha *comma separated value* (.csv), enquanto as médias de intensidade dos sinais para cada ponto em cada direção foram armazenadas em arquivo *python*(.py). A tabela 1 mostra os dados armazenados para uma data direção.

Ponto	Rede 1	Rede 2	...	Rede n
1	$RSSI_{1,1}$	$RSSI_{1,2}$...	$RSSI_{1,n}$
2	$RSSI_{2,1}$	$RSSI_{2,2}$...	$RSSI_{2,n}$
3	$RSSI_{3,1}$	$RSSI_{3,2}$...	$RSSI_{3,n}$
...
112	$RSSI_{112,1}$	$RSSI_{112,2}$...	$RSSI_{112,n}$

Tabela 1 – Valores RSSI médios por ponto ponto

Na tabela 1, cada ponto representado pelos números de 1 a 112 está associado a um par de coordenadas (x, y) e os valores $RSSI_{ij}$ representam as médias das leituras feitas no ponto i para a rede j.

Uma vez feita a coleta de dados, foi definida como é realizada a fase de uso. Esta consiste na realização dos seguintes passos: 1) leitura das intensidades de cada um dos sinais *Wi-Fi* utilizados; 2) envio dos valores lidos e da direção do usuário para o servidor por meio de uma requisição *Hyper Text Transfer Protocol* (HTTP). 3) cálculo da posição (x,y) onde as leituras foram feitas, realizado pelo servidor, por meio do algoritmo KNN e, 4) recebimento da posição calculada pelo servidor na resposta da requisição HTTP feita.

Para realizar as leituras dos sinais *Wi-Fi*, é utilizada a *Application Programming*

Interface (API) *WiFiManager*² disponibilizada pelo *Android*. Esta API possui a restrição de que podem ser feitas no máximo 4 leituras a cada 2 minutos. Porém, ao utilizar o modo desenvolvedor do celular, esta frequência é maior, dessa forma, foi necessário habilitar o modo desenvolvedor.

Para fazer a requisição HTTP mencionada no segundo passo, é usada a API *volley*³. Os parâmetros da requisição enviada são: os valores RSSI de cada sinal Wi-Fi utilizado e a direção do usuário. É esperada uma resposta contendo o par de coordenadas (x, y) que representa a posição do usuário no momento em que a leitura foi feita.

Para a realização do cálculo das coordenadas x e y , assim como na abordagem utilizada por [Poulose, Kim e Han \(2019\)](#) e descrita na seção 2.7, é usado o algoritmo KNN para estimar a posição do usuário. A métrica para escolha dos k vizinhos mais próximos é a distância euclidiana, descrita pela equação 2.1 entre valores RSSI da nova leitura e as médias armazenadas na base de dados.

Na implementação do KNN utilizada neste trabalho, é usada a média entre as coordenadas x e y dos 3 pontos considerados como vizinhos mais próximos pelo algoritmo.

3.3.2 Implementação do algoritmo *PDR*

Uma vez concluída a implementação do *fingerprinting*, foi feita a implementação do algoritmo de estimativa da posição utilizando *PDR*, em que é utilizada a abordagem *SHS* com a finalidade de evitar o acúmulo de erros presente na abordagem *INS* vindos da operação de integração dupla da aceleração.

Na abordagem utilizada, assim como feito por [Gowda, Babu e Sekaran \(2017\)](#) e [Cruz \(2021\)](#), são feitas leituras do acelerômetro presente no *smartphone* para monitorar o componente do vetor aceleração perpendicular ao movimento do usuário a fim de realizar a detecção de passos.

A aceleração é obtida inicialmente com relação ao sistema de coordenadas do *smartphone*, mostrado na Figura 5. Em seguida, as coordenadas do vetor de aceleração são convertidas para o sistema de coordenadas mostrado na Figura 6 a partir do uso do método *getRotationMatrix*⁴.

Os eixos x , y e z do sistema de coordenadas ilustrado na Figura 6 possuem, respectivamente, a mesma orientação dos eixos E , N e U do sistema de coordenadas locais descrito em 2.9.2 e mostrado na Figura 2.

² <https://developer.android.com/reference/android/net/wifi/WifiManager>

³ <https://developer.android.com/training/volley?hl=pt-br>

⁴ <https://developer.android.com/reference/android/hardware/SensorManager>

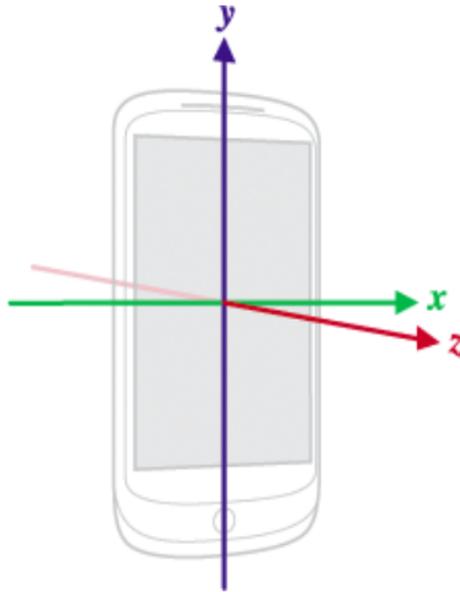


Figura 5 – Sistema de coordenadas do celular. (FONTE: (ANDROID, 2023))

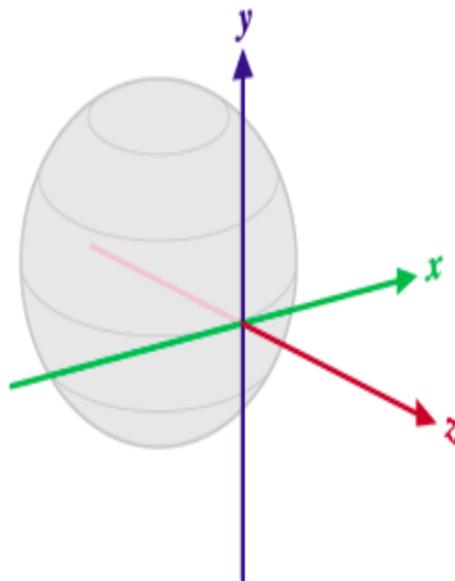


Figura 6 – Sistema de coordenadas locais (FONTE: (ANDROID, 2023))

Ao realizar uma leitura de aceleração e o devido ajuste do sistema de coordenadas, é verificado se foi encontrado um mínimo local, que ocorre caso o componente de aceleração lido seja menor que os valores encontrados tanto na leitura anterior quanto na seguinte. Ao verificar a ocorrência de um mínimo local, é checado se este mínimo foi encontrado após a passagem de um período mínimo de tempo desde o último passo detectado e também se sua amplitude ultrapassa um limiar mínimo.

Para reduzir os efeitos de vibrações indesejadas, assim como feito por [Gowda, Babu e Sekaran \(2017\)](#), é utilizado um filtro passa baixas nas leituras realizadas antes de realizar a detecção de mínimos locais. O filtro utilizado é o filtro *Butterworth* de quinta ordem.

Dessa forma, antes da implementar a função da detecção de passos para a realização do PDR, foi necessário definir os parâmetros do filtro, assim como, o tempo mínimo entre a ocorrência de passos e a amplitude mínima dos mínimos locais. Os parâmetros utilizados no PDR são: 1) a frequência de amostragem de 50 Hz em que são feitas leituras de aceleração pelo acelerômetro; 2) frequência de corte de 3 Hz utilizada no filtro como feito por Cruz (2021); 2) limiar de amplitude, valor 2,8, mesmo valor utilizado por (CRUZ, 2021) e, 3) o limiar de tempo entre a contagem de passos de 0,5 s, descoberto por tentativa e erro.

Além da detecção de passos, é necessário estimar a orientação de cada passo e conhecer a largura de cada passo para poder estimar o deslocamento. Foi utilizado o valor de 0,8m, largura média dos passos do autor deste trabalho. A orientação de cada passo é feita por meio do uso da API `SensorManager`⁵ para obter a direção do movimento, a orientação obtida é o ângulo entre a direção para qual o celular aponta no momento do passo e a direção do eixo N do sistema de coordenadas locais, descrito em 2.9.2 que coincide com o eixo y, Figura 6.

O deslocamento é então calculado a partir da equação 3.1, sendo θ o ângulo entre a direção do deslocamento e a direção do eixo y, das *coordenadas do ambiente*, e L é a largura de passo, que neste trabalho é considerado como constante e vale 0,8 m.

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \text{sen}(\theta) & 0 \\ 0 & \text{cos}(\theta) \end{bmatrix} \cdot \begin{bmatrix} L \\ L \end{bmatrix} + \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} \quad (3.1)$$

A partir do que foi explicado até agora, a estimação do deslocamento por meio da realização do PDR neste trabalho segue os passos listados a seguir:

1. Leitura dos sensores inerciais e magnetômetro.
2. Mudança do sistema de coordenadas que leva o vetor aceleração do sistema de coordenadas do celular, Figura 5, para o sistema de coordenadas locais, Figuras 2 e 6.
3. Filtragem do sinal obtido pela leitura da aceleração, utilizando o filtro *butterworth*.
4. Verificação de valor mínimo local foi encontrado com a leitura.
5. Caso um mínimo local tenha ido encontrado, verificação da amplitude do sinal, se é maior que o limiar de amplitude definido.
6. Caso o limiar a amplitude do mínimo local seja aceita, verificação da diferença entre o instante e que foi detectado o mínimo local o instante em que foi detectado o último passo.

⁵ <https://developer.android.com/reference/android/hardware/SensorManager>

7. caso o passo seja contabilizado, é obtida a direção do deste.
8. Cálculo do deslocamento com base na equação 3.1.

O pseudocódigo 1 descreve a implementação do algoritmo de PDR descrita. O Algoritmo considera os parâmetros *AMPLITUDE_THRESHOLD*, *TIME_THRESHOLD* e *STEP_LENGTH* conhecidos.

Algorithm 1 PDR algorithm

```

1: while (true) do
2:   Accelerations ← readInertialSensors()
3:   Acclerations ← correctReferential(Acclerations)
4:   currentTime ← getCurrentTime()
5:   filteredValue ← Filtter(Acceleration.Z)
6:   if (filttredSginal < AMPLITUDE_THRESHOLD) then
7:     if (currentTime – previousTime) > TIME_THRESHOLD then
8:       theta ← getYaw()
9:       position.Y ← STEP_LENGTH · cos(theta)
10:      position.X ← STEP_LENGTH · sin(theta)
11:      previousTime ← currentTime
12:     end if
13:   end if
14: end while

```

3.4 Determinação do ângulo de azimute

O sistema de coordenadas utilizado no algoritmo de *fingerprinting* é definido pelas paredes do corredor utilizado como ambiente de testes. O sistema de coordenadas utilizado pelo PDR utiliza um conjunto de eixos x, y e z que correspondem, respectivamente, aos eixos E, N e U apresentados no capítulo 2, seção 2.9, conforme discutido em 3.3.2. Por esse motivo, foi necessário encontrar uma maneira de converter as coordenadas de um sistema para o outro.

Os sistemas de coordenadas utilizados no *fingerprinting* e no PDR possuem os respectivos eixos z apontados para a mesma direção. Portanto, é possível converter as coordenadas utilizadas em um sistema para o outro, desde que utilizado o ângulo correto para esta operação. Como o eixo y do sistema de coordenadas utilizado pelo *fingerprinting* é alinhado ao às paredes do corredor. O ângulo a ser utilizado para a conversão corresponde ao azimute do corredor. Para determinar este ângulo, foram seguidos os procedimentos listados abaixo:

1. Coleta das coordenadas geodésicas de pontos dentro da faculdade de tecnologia (FT) do campus Darcy Ribeiro da Universidade de Brasília.

2. Definição de segmentos de reta usando uma combinação diferente de 2 pontos, cujas coordenadas foram coletadas, para cada segmento.
3. Determinar o ângulo azimute de cada um dos segmentos de reta a partir das coordenadas geodésicas dos dois pontos para definição deste.
4. Determinar o ângulo entre o eixo y e cada segmento de reta a partir das coordenadas fornecidas pela planta baixa.
5. Calcular uma estimativa do azimute do eixo y para cada segmento utilizando o valor do azimute deste, assim como seu ângulo em relação ao eixo y.

Em um primeiro momento, foram selecionados 5 pontos para coleta de coordenadas geodésicas (latitude, longitude e altura elipsoidal), os pontos selecionados são mostrados na figura 7.

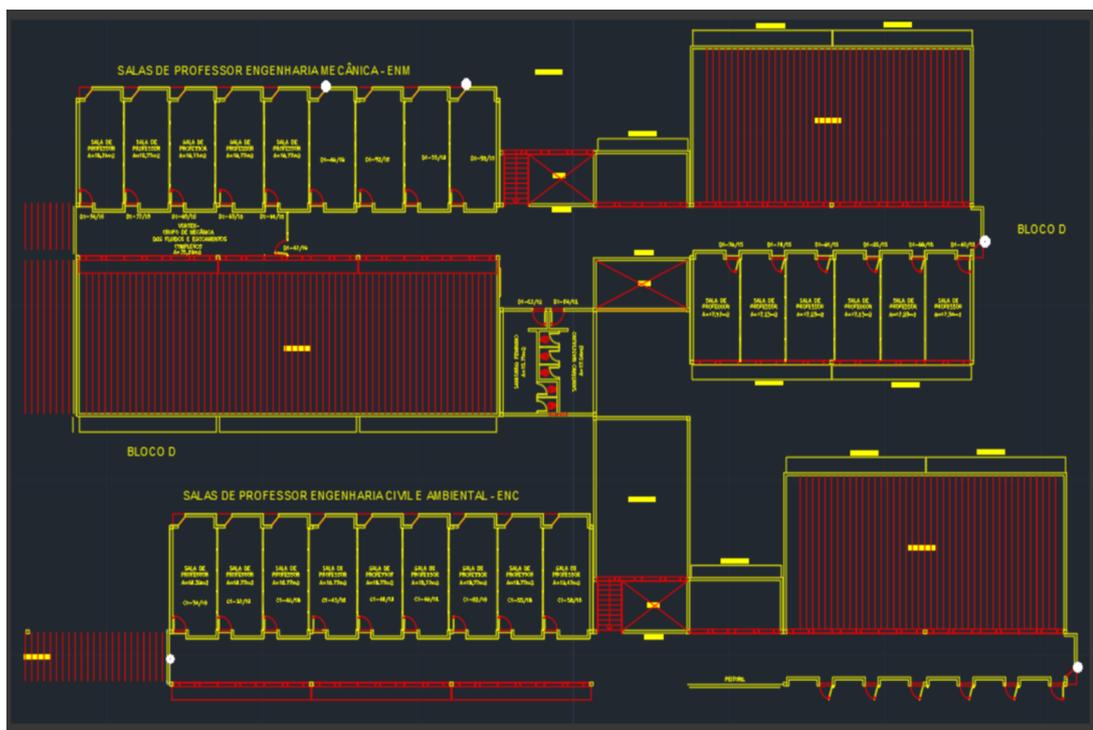


Figura 7 – Pontos selecionados para estimar o azimute

A coleta das coordenadas foi feita com a utilização do *trimble*, um receptor *GNSS* de alta precisão, em conjunto com a rede *Catalyst*, que permite fazer a leitura de sinais obtidos de múltiplas constelações de satélites simultaneamente para uma maior precisão das coordenadas obtidas.

Os valores das leituras de coordenadas são mostradas em um aplicativo próprio para o uso do equipamento ao conectar o celular ao receptor. As figuras 8a e 8b mostram o equipamento utilizado para fazer as leituras das coordenadas, enquanto a tabela 2 mostra as coordenadas obtidas para cada ponto.



(a) Receptor Trimble

(b) Receptor conectado ao celular

Figura 8 – Equipamento trimble para captura de coordenadas

Pontos	Latitude	Longitude	Altura elipsoidal
P1	-15.7635241	-47.872795	1030.3
P2	-15.7631444	-47.872868	1030.5
P3	-15.7634516	-47.872827	1030.2
P4	-15.7635011	-47.872473	1029.4
P5	-15.7630103	-47.872677	1029.7

Tabela 2 – Coordenadas dos pontos coletados.

Com os pontos e suas respectivas coordenadas definidos, foram selecionados 5 pares de pontos, cada um definindo um segmento de reta. Os segmentos definidos são mostrados na figura 9.

A tabela 3 mostra os pontos iniciais e finais de cada segmento de reta. O ângulo em relação ao eixo y em cada segmento e o azimute obtido para cada segmento.

Para obtenção dos azimutes registrados na tabela 3 de cada segmento, foram realizados os seguintes passos:

1. Conversão das coordenadas geodésicas (latitude, longitude e altura elipsoidal) para o sistema ECEF (X, Y, Z) dos pontos P1, P2, P3, P4 e P5 por meio da equação 2.11.

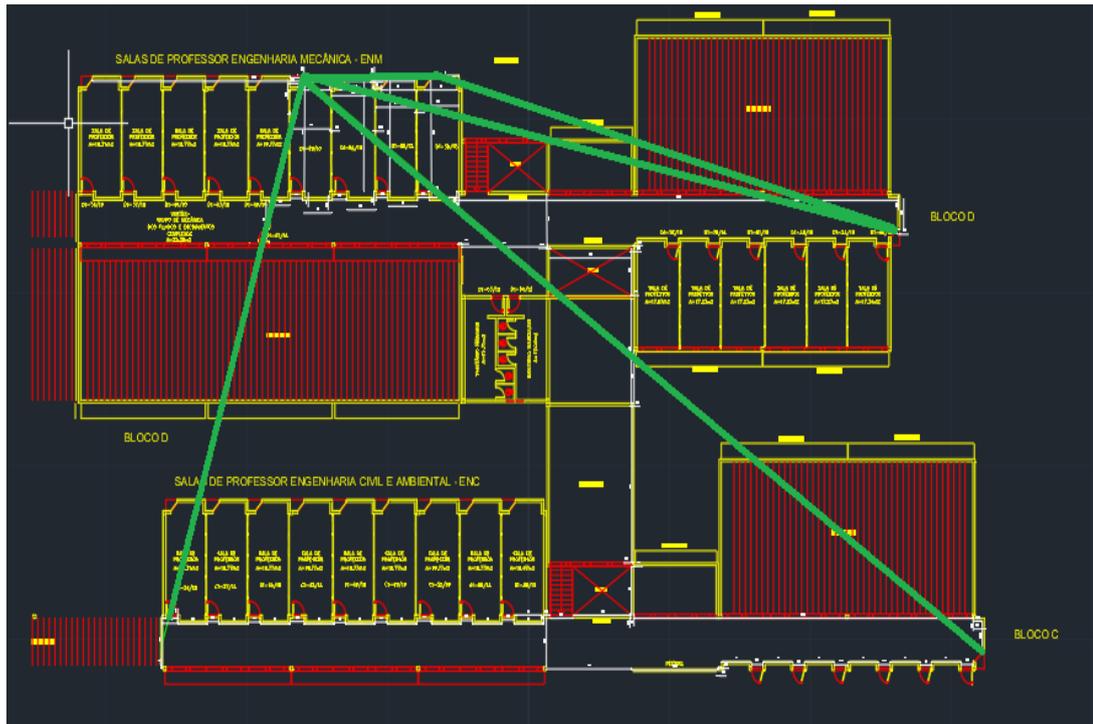


Figura 9 – Pontos selecionados

Pontos	Inclinação em relação a y	Azimute
P2, P3	14,89°	-7.28°
P2, P1	11,81°	-10.43°
P3, P1	0°	-23.15°
P1, P4	106.89°	82.46°
P5, P1	34.82°	12.29°

Tabela 3 – Coordenadas dos pontos coletados.

2. Para cada par de pontos, foram definidas as diferenças ΔX , Δy e Δz realizando a subtração entre coordenadas do primeiro e do segundo ponto de cada par.
3. As diferenças Δ_X, Δ_Y e Δ_Z foram convertidas do sistema ECEF para o sistema de coordenadas locais na forma Δe , Δn , e Δu por meio da equação 2.20.
4. O azimute de cada segmento foi então calculado por pela equação 2.21.

A estimativa do Azimute do eixo y feita a partir de cada segmento é realizada a partir da equação 3.2, onde α_y é a estimativa do azimute do eixo y, α_s é o azimute do segmento de reta e θ_s é a inclinação do segmento em relação ao eixo y.

$$\alpha_y = \alpha_s - \theta_s \quad (3.2)$$

Como foram definidos 5 segmentos de reta, foram obtida 5 estimativas para o azimute do eixo y. A média aritmética dessas estimativas é utilizada como valor do azimute de y. O

valor encontrado foi de 22,9°. Uma vez encontrado o valor do azimute, as estimativas de posição obtidas pelo *fingerprinting* e PDR já podem ser combinadas.

3.5 Implementação do filtro de Kalman

O filtro de Kalman linear foi utilizado para combinar a estimativa de deslocamento obtida pelo PDR e a posição calculada através do *fingerprinting* de sinais Wi-Fi. Para a utilização do filtro, foi considerada a modelagem proposta pela equação 3.3.

$$\begin{bmatrix} x_k \\ y_k \end{bmatrix} = \begin{bmatrix} x_{k-1} \\ y_{k-1} \end{bmatrix} + L \cdot \begin{bmatrix} n_{k-1} \cdot \text{sen}(\theta_{k-1}) \\ n_{k-1} \cdot \text{cos}(\theta_{k-1}) \end{bmatrix} + (L + \sigma_L) \cdot n \cdot \begin{bmatrix} \text{sen}(\theta_{k-1} + \sigma_\theta) - \text{sen}(\theta_{k-1}) \\ \text{cos}(\theta_{k-1} + \sigma_\theta) - \text{cos}(\theta_{k-1}) \end{bmatrix} \quad (3.3)$$

A equação 3.3 é equivalente à equação 2.2, onde as matrizes F_{k-1} , G_{k-1} e w_{k-1} são dadas pelas equações 3.4 e 3.5:

$$F_{k-1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (3.4)$$

$$G_{k-1} = \begin{bmatrix} L & 0 \\ 0 & L \end{bmatrix} \quad (3.5)$$

$$w_{k-1} = (L + \sigma_L) \cdot n \cdot \begin{bmatrix} \text{sen}(\theta_{k-1} + \sigma_\theta) - \text{sen}(\theta_{k-1}) \\ \text{cos}(\theta_{k-1} + \sigma_\theta) - \text{cos}(\theta_{k-1}) \end{bmatrix} \quad (3.6)$$

As variáveis de estado de interesse no sistema são as coordenadas x e y da posição estimada. A matriz F é definida como sendo a identidade, pois as variáveis x e y não mudam de valor com o passar do tempo a menos que o usuário aplique uma entrada, o que equivale a um número de passos dados em uma determinada direção. Da mesma forma, a matriz G é dada pela equação 3.5, pois o valor L é considerado constante (0,8 m) e multiplica a entrada fornecida pelo usuário. A matriz Q pode ser então obtida por:

$$Q_{k-1} = (L + \sigma_L)^2 \cdot n^2 \cdot \begin{bmatrix} (\text{sen}(\theta_{k-1} + \sigma_\theta) - \text{sen}(\theta_{k-1}))^2 & 0 \\ 0 & (\text{cos}(\theta_{k-1} + \sigma_\theta) - \text{cos}(\theta_{k-1}))^2 \end{bmatrix} \quad (3.7)$$

Nas equações de 3.3 a 3.8, L é o comprimento do passo do usuário, n_{k-1} é o número de passos que foi dado entre os instantes $k-1$ e k , θ_{k-1} é o ângulo médio considerando os n passos dados entre os instantes $k-1$ e k , por último, σ_L e σ_θ são as incertezas associadas à largura e direção do passo respectivamente. Foram adotados os valores 0,2 e 5,65 para σ_L e σ_θ obtidos experimentalmente nos testes descritos na seção 4.1.

As equações de 3.3 a 3.8 são suficientes para realização da etapa de predição do filtro. Para a etapa de correção, é utilizada a equação 2.3 substituindo a matriz H pela identidade e o vetor v pelas incertezas (σ_x, σ_y) na estimativa da posição realizadas pelo *fingerprinting*. Os valores utilizados são: $\sigma_x = \sigma_y = 1,41$. Com isso a matriz R é dada por:

$$R_k = \begin{bmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{bmatrix} \quad (3.8)$$

A partir da determinação matriz R e com as matrizes já determinadas na etapa de predição, é realizada a etapa de correção descrita em 2.8.2. A correção é realizada sempre que é obtida uma nova estimativa da posição a partir da técnica de *fingerprinting*. Ao obter a posição do usuário, foi possível criar realizar o desenvolvimento do algoritmo para calcular rotas entre sua posição e um destino desejado.

3.6 Cálculo de rotas

O algoritmo escolhido para calcular as rotas foi o A^* . Este algoritmo tem como objetivo encontrar o menor caminho entre dois pontos em um ambiente e foi escolhido devido a sua simplicidade para implementação.

Uma necessidade para o uso do algoritmo é a discretização do ambiente, já que são usadas células ou nós em um grafo, para representar as possíveis posições do usuário. Dessa forma, o ambiente de testes foi representado como uma matriz, referida como *matriz do ambiente* neste trabalho. Cada posição desta pode assumir os valores 0 ou 1, onde o valor 1 indica uma posição indisponível para o usuário e 0 indica posição disponível.

Cada trajeto dentro do ambiente é representado como um conjunto de posições (i, j) na matriz do ambiente, em que i representa a i -ésima linha e j a j -ésima coluna. Cada posição é também referida como nó.

O primeiro passo para calcular a rota é definir as posições (nós) de início e fim do trajeto, em seguida, o nó inicial é definido como nó atual. O nó atual é então expandido, ou seja, é criada uma lista de nós vizinhos na matriz do ambiente. Cada nó desta lista passa a apontar para o nó atual, sendo este referido como nó pai.

Para cada nó vizinho encontrado, são calculados os valores de custo F , G e H . O custo H de um nó é uma heurística que representa a distância entre este e o nó destino, fim do trajeto. O custo G de um nó é calculado a partir da mesma heurística que H , porém representa a distância entre o referido nó e o nó de início do trajeto. Por último, o custo F é dado pela soma de G e H .

Após o cálculo de F , G e H , todos os nós vizinhos são adicionados a uma lista de nós abertos e o nó atual é retirado dessa mesma lista. Em seguida, é selecionado o nó dentro

da lista de nós abertos com o menor valor de custo F , para que este seja agora considerado como nó atual e seja expandido. Todo o processo é repetido, até que o nó atual seja vizinho do nó destino.

Ao encontrar o nó objetivo, é selecionado o pai do nó atual para que seja adicionado a uma lista que irá conter os nós que fazem parte do trajeto. O nó recém adicionado à lista tem seu nó pai adicionado em seguida, o que se repete até que o nó inicial seja adicionado também e seja gerada uma lista todos os nós que fazem parte do trajeto.

3.7 Desenvolvimento da interface

A interface do aplicativo permite ao usuário acesso às seguintes funcionalidades: 1) login; 2) navegação; 3) configurações; 4) seleção de pontos de partida e destino e, 5) desenho de rotas.

A primeira ação do usuário, ao abrir o aplicativo, é realizar o login. Para isso, é oferecida a opção de utilizar o e-mail fornecido pela conta Google do usuário. Para a implementação da função de *login*, foi usada a API *android.gms.auth.API*, permitindo que seja selecionada uma conta Google previamente salva no dispositivo. Essa ação retira a necessidade de preencher campos de texto contendo usuário e senha, o que reduz o número de toques e passos necessários para acessar as funcionalidades do aplicativo, gerando maior acessibilidade e melhor experiência do usuário. A Figura 10 mostra a tela de login e seleção de conta no aplicativo.

Após a realização do *login* no MovUnB, é exibida a tela de navegação, que apresenta o mapa de onde o usuário está. A posição do usuário é representada por um marcador. As ações que podem ser tomadas a partir da tela de navegação no MovUnB são: 1) definição de partida e destino para cálculo de rota, 2) iniciar trajeto, 3) abrir tela de configurações, 4) abrir o manual do usuário e 5) fechar o aplicativo. Tais ações são ilustradas pelas Figuras 11 e 12.

Para garantir a acessibilidade do MovUnB, são realizadas ações de acessibilidade, a exemplo, o uso da API *android.speech* para permitir a entrada de áudio (voz) durante a seleção do destino do usuário. Desta forma, objetiva-se facilitar o acesso ao aplicativo para pessoas com deficiência visual. Além disso, todos os elementos clicáveis possuem o campo *contentDescription* que possibilita a integração com os leitores de tela. Todos os elementos decorativos, como as *logos* ou imagens serão marcadas como não importantes para acessibilidade; isto é, não precisarão ser lidos pelo leitor, pois não contribuem com a usabilidade do aplicativo.

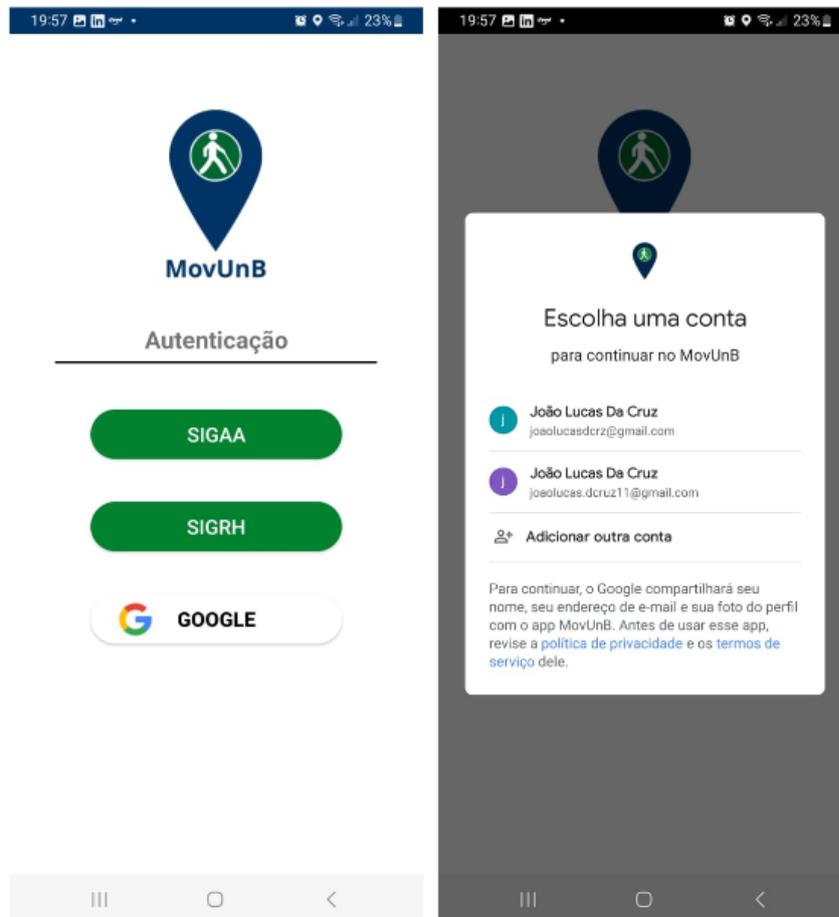


Figura 10 – Tela de login e integração com conta google

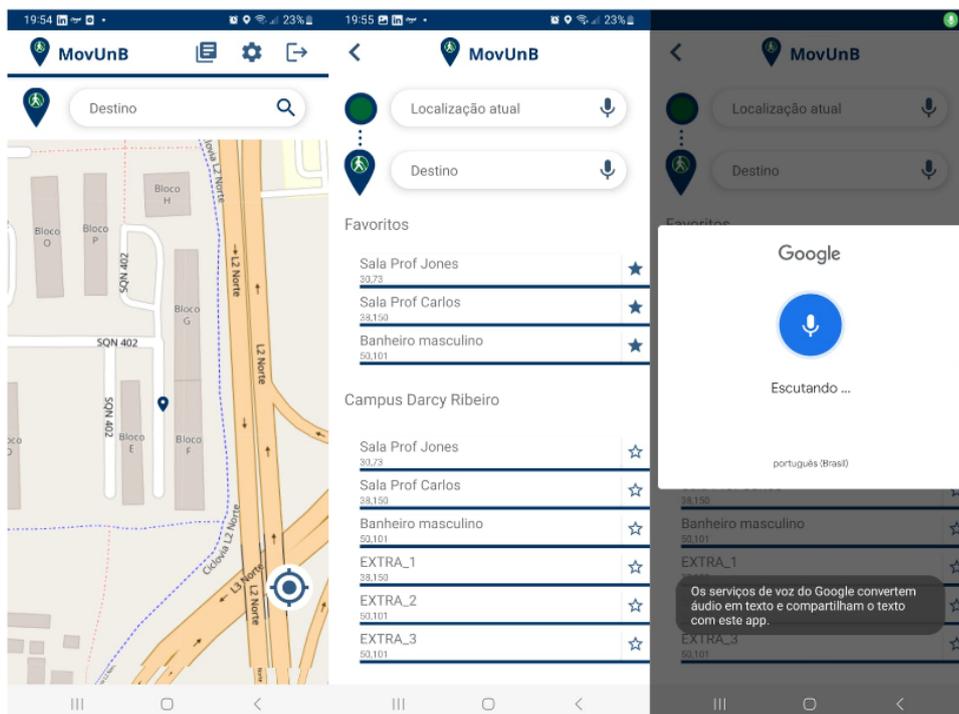


Figura 11 – Telas de navegação e seleção de destino com entrada de áudio

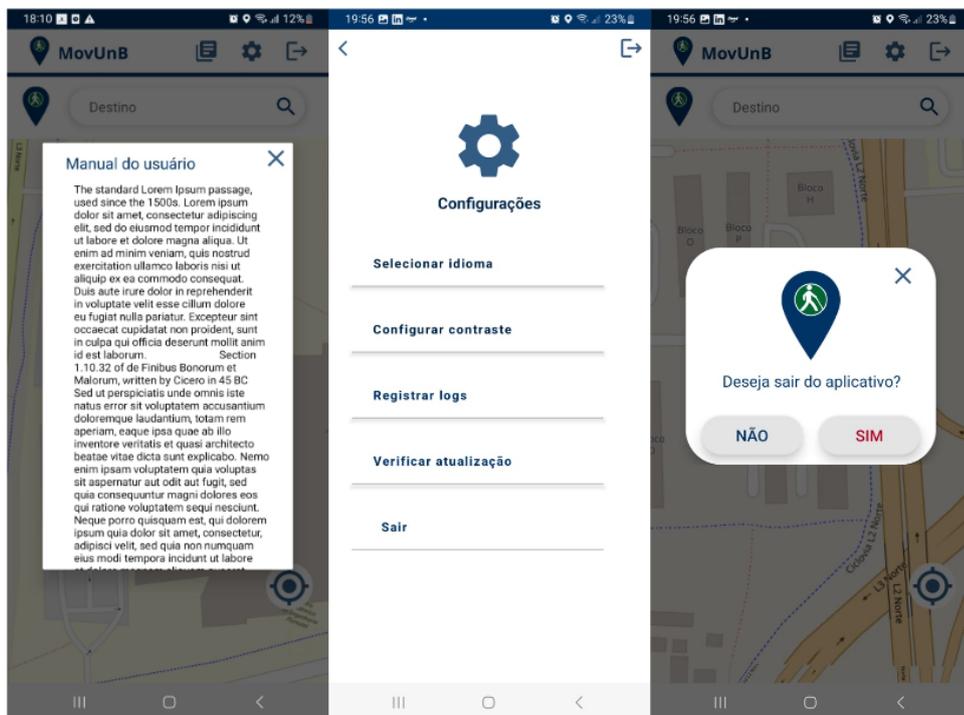


Figura 12 – Manual do usuário, tela de configurações e caixa de saída

4 Testes, resultados e discussão

Nesta seção, são descritos os testes realizados para a estimativa de posição por meio do *fingerprinting*, *PDR* e da combinação das duas técnicas, a partir do uso do filtro de Kalman. Também é feita uma análise dos resultados obtidos em cada teste.

4.1 Testes do PDR

Para o algoritmo de *PDR*, foram realizados dois testes: 1) acertos no número de passos detectados e 2) erro médio na estimação da direção do passo. O teste de detecção no número de acertos que foi realizado em 3 cenários, caminhada com ritmo lento, moderado e acelerado em trajetos aleatórios fora do ambiente de testes. Foram dados 30 passos em cada trajeto, cada um repetido 15 vezes. A tabela 4 mostra o percentual de passos contabilizados para cada forma de caminhada.

Ritmo	Média de passos contados	Percentual médio de acertos
Rápido	27,32	91,06 %
Moderado	28,84	96,13 %
Lento	25,39	84,63 %

Tabela 4 – Frequência de acertos na ocorrência de passos

Na segunda avaliação, foi medida a diferença média entre o ângulo real de passo e o ângulo medido. Ambos os ângulos (real e medido) são tomados em relação ao eixo y, definido no ambiente de testes, e vão de -180° a 180° . O teste foi feito selecionando 15 pares de pontos de posições conhecidas. Cada par de pontos é representado por uma linha na tabela 5, onde as colunas x1 e y1 compõe as coordenadas do primeiro ponto de cada par, enquanto x2 e y2 são as coordenadas do segundo ponto. Em seguida, foi definido um segmento de reta a partir de cada par de pontos, assim como, a direção deste segmento por meio da equação 4.1.

$$\theta_r = \tan^{-1}\left(\frac{y_2 - y_1}{x_2 - x_1}\right) \quad (4.1)$$

A direção obtida é representada por θ_r , na tabela 5, considerada como a direção real do deslocamento entre os pontos. Após isso, para cada par de pontos, foi dado um passo de (x1, y1) para (x2, y2) e a direção do passo foi medida por meio da implementação do *PDR*, a partir dos sensores inerciais do *smartphone*. O procedimento foi repetido 30 vezes para cada segmento de reta e o ângulo médio medido de cada segmento foi registrado na tabela 5 na coluna θ_e . O erro médio no ângulo estimado para cada passo foi registrado na tabela 6.

A presença de erros nos ângulos medidos pode ser explicada por movimentos involuntários da mão que segura o *smartphone* durante a caminhada, o que gera leituras

x1	y1	x2	y2	θ_r	θ_e
1	3	1	3.8	0°	-5.9
1.8	3	1.8	3.8	0°	6.7
1	5	1	5.8	0°	3.4
1	6	1.5	6.5	45°	50
1.8	15	1.3	15.5	-45°	-38
1	50	1.5	49.5	135°	139
1.8	40	1.4	40.6	-33.6°	42,8
1.5	35	1.5	35.8	0°	2,3
1.8	52	1.3	51.5	135°	149,1
1.5	33	1.5	33.7	0°	6,2
3	25	3.8	25	90°	90,4
3.6	24	2.9	24	-90°	-98,3
5	28	4.3	28	-90°	-94,7
5	27	4.5	26.5	-135°	-129,5
5	24	4.5	24.5	-45°	-43

Tabela 5 – Erro angular médio do passo

Erro angular médio	5,65°
Desvio padrão	2,02°

Tabela 6 – Erro angular médio do passo

distorcidas nos sensores inerciais. Além disso, há a presença das incertezas dos próprios sensores utilizados e distorções na declinação magnética dentro do ambiente de testes que são detectadas pelo magnetômetro do celular.

4.2 Testes da implementação do fingerprinting

Para estimar os erros da posição calculada pela implementação do *fingerprinting* de sinais Wi-Fi, foram selecionados 32 pontos dentro do ambiente de testes. Para cada ponto selecionado, foram feitas entre 4 e 5 leituras com o celular apontando para as 4 direções que coincidem com as direções dos eixos +y (norte), +x (leste), -x (oeste) e -y (sul) do sistema de coordenadas utilizado no corredor, ou seja, foram realizadas 16 entre e 20 leituras de sinal Wi-Fi por ponto totalizando 542 leituras.

Para cada leitura realizada, foi utilizado o algoritmo KNN para fazer uma estimativa da posição em que a leitura foi feita. O erro encontrado em cada estimativa foi então calculado por meio da distância euclidiana entre a posição estimada e a posição real, equação 4.2. Na referida equação, e corresponde ao erro estimado, x_e e y_e são as coordenadas da posição estimada, enquanto x_r e y_r são as coordenadas reais do ponto.

$$e = \sqrt{(x_r - x_e)^2 + (y_r - y_e)^2} \quad (4.2)$$

A Figura 13 apresenta a distribuição dos erros encontrados considerando todas as

512 leituras. Cada coluna representa a quantidade de estimações cujos erros estão em um determinado intervalo, mostrado abaixo da respectiva coluna.

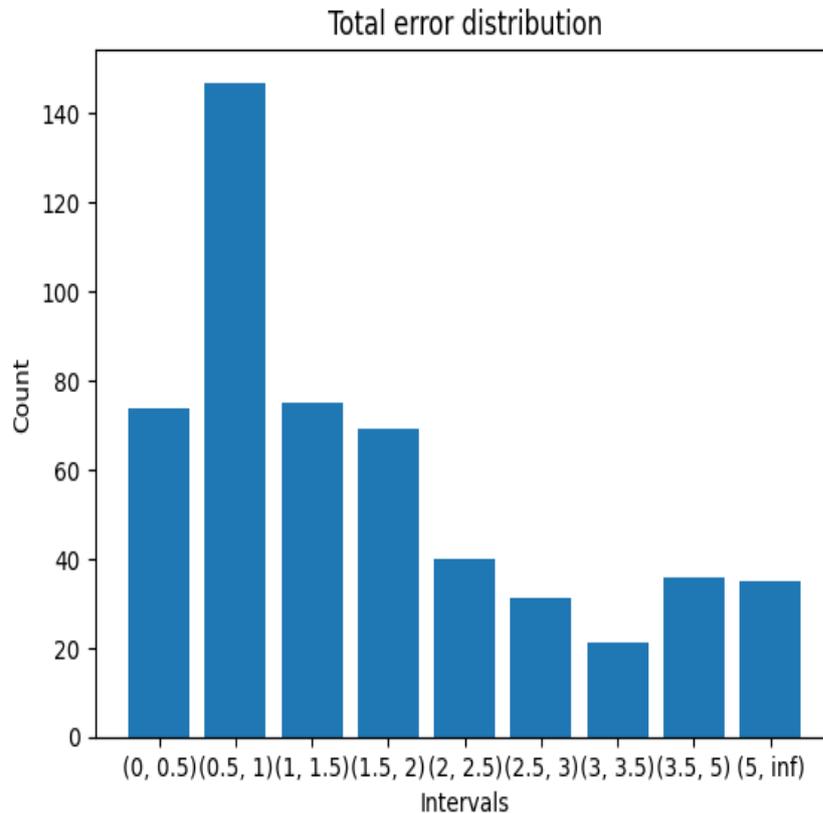


Figura 13 – Distribuição dos erros na estimativa da posição pelo fingerprint

A tabela 7 mostra quantas estimativas foram encontradas em cada intervalo, assim como o erro médio e maior erro encontrado

Intervalo de erro (m)	Número de estimativas
0 a 0,5	76
0,5 a 1	144
1 a 1,5	78
1,5 a 2	70
2 a 2,5	39
2,5 a 3	35
3 a 3,5	23
3,5 a 5	39
> 5	38

Tabela 7 – Distribuição de erros fingerprinting

O erro médio encontrado para as estimativas de posição foi de 1,98 m, já a pior estimativa foi de 11,29 m. O cenário que gerou o pior caso pode ter ocorrido devido a uma oscilação nas redes, uma vez que, ao analisar a distribuição de erros mostrada na Figura 13 e 7 aproximadamente 7% das estimativas apresenta um erro de posicionamento maior que 5 m. Além disso, 67% das leituras apresenta erro menor que 2m.

O *fingerprinting* de sinais Wi-Fi não só é usado para mitigar o acúmulo de erros inerente ao PDR, mas também para obter a posição inicial do usuário. A ocorrência de grandes discrepâncias na primeira estimativa da posição pode comprometer o sistema como um todo. É necessário para um bom funcionamento aplicativo uma forma de contornar a possível ocorrência desses erros na estimativa inicial da posição.

4.3 Teste da trajetória calculada final

Para a realização dos testes de estimativa da posição após a combinação das técnicas de PDR e de *fingerprinting*, a partir da utilização de filtros de Kalman, foi selecionado um trajeto retilíneo e este foi repetido 30 vezes. O trajeto foi iniciado no ponto (1, 2) e concluído no ponto (1, 21). A figura 14 uma foto do ambiente de testes com o percurso realizado que se inicia na trena presente no chão do corredor e vai até a cadeira. As Figuras 15a e 15b mostram o trajeto real em conjunto com as estimativas de posição realizadas pelo *fingerprinting*, PDR e pela combinação dos dois métodos por meio da aplicação do LKF.

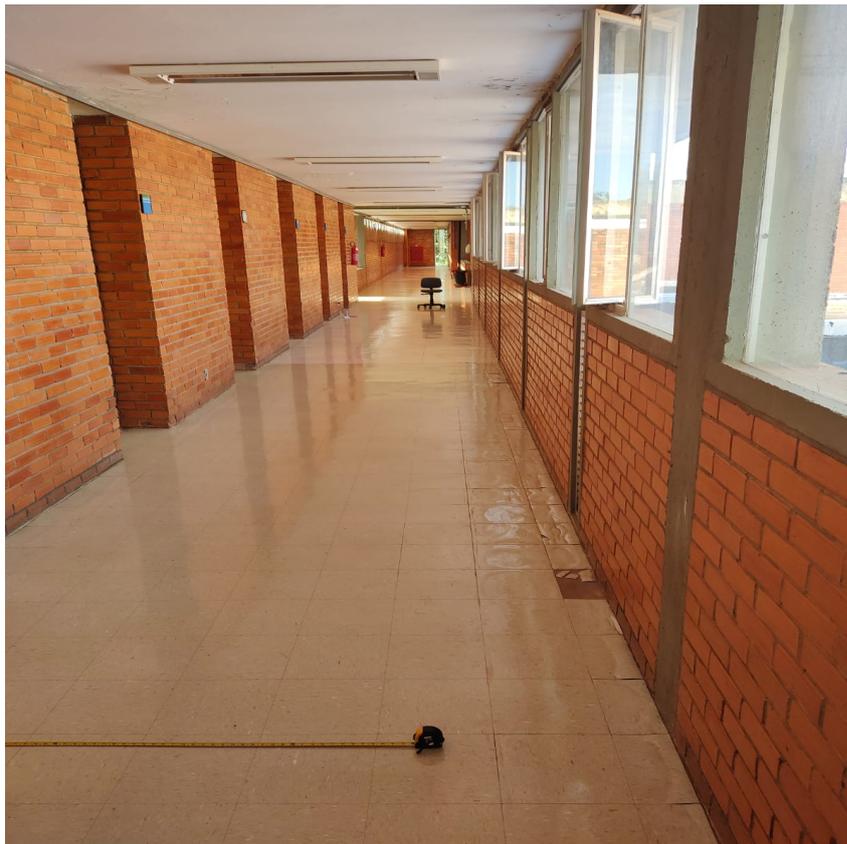


Figura 14 – Ambiente de testes

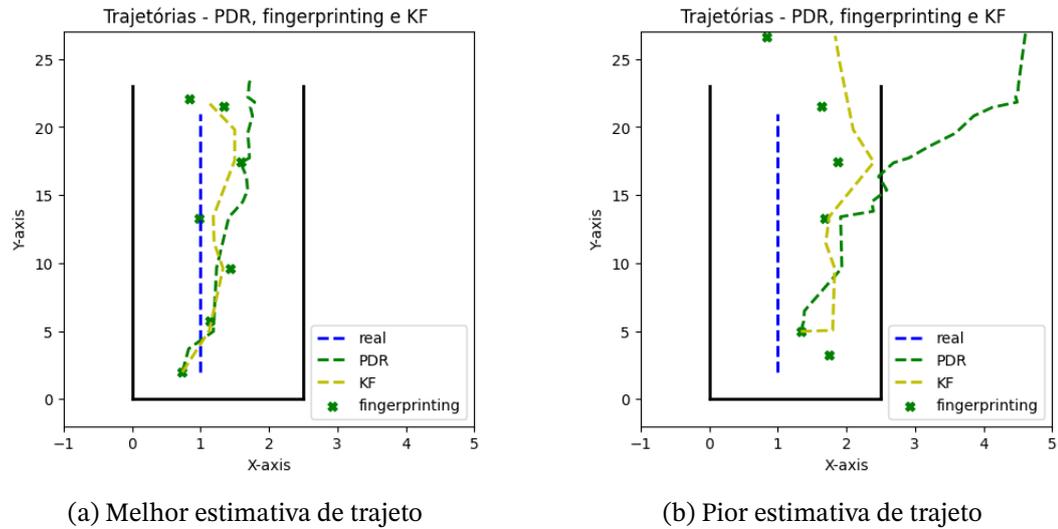


Figura 15 – Trajetos estimados no pior e melhor cenário.

A partir das figuras 15a e 15b, é possível observar que tanto no melhor, quanto no pior cenário, que a estimativa de posição realizada pela combinação dos dois métodos utilizados é melhor que a utilização de ambos separadamente ao final do trajeto.

O maior valor de erro, assim como o valor médio deste são de 5,658 m e 1,73 m respectivamente. Pode-se notar que, a partir da Figura 15b que o erro no pior cenário está relacionado a uma estimativa inicial ruim para a posição, conforme o exposto na seção 4.2.

5 Conclusões

Neste trabalho, foi proposta uma metodologia para desenvolvimento de IPSs a partir do uso das técnicas de PDR, *fingerprinting* e filtro de Kalman linear para obter a posição do usuário. A metodologia proposta possibilitou a construção de um sistema que promove acessibilidade para pessoas com ou sem deficiência visual nas dependências da universidade sem adição de despesas para esta, o que validou as hipóteses um e três descritas na seção 1.3.

Durante os testes com a solução proposta, em um ambiente controlado, foi possível obter erro médio de 1,7m. Ao fornecer o posicionamento dos usuários com esta precisão, o aplicativo desenvolvido tem potencial para auxiliar a comunidade universitária, especialmente aqueles com deficiência visual, no deslocamento em ambientes *indoor*. O que possibilita a validação da hipótese dois, seção 1.3, em um momento futuro, após a realização de testes com usuários.

A promoção da autonomia dos usuários na navegação dentro de edifícios visa facilitar o acesso a locais como salas de aula e banheiros, queixa recorrente entre os alunos. Tais autonomia e facilidade promovidas podem trazer mais acessibilidade para a comunidade universitária, o que é essencial para o sentimento de pertencimento e consequente redução na evasão de alunos.

5.0.1 Trabalhos futuros

Para trabalhos futuros, faz-se necessário:

- Realizar testes de acessibilidade da interface com usuários que possuem deficiência visual.
- Testar a funcionalidade de navegação com usuários em tempo real.
- Adicionar função do cálculo, em tempo real, da largura do passo do usuário no algoritmo que implementa o PDR.
- Possibilitar o uso do aplicativo em outras plataformas.
- Testar o aplicativo em outros ambientes para validar a generalização da solução.

Além disso, a solução pode ser replicada em hardware separado, para contornar as limitações impostas de frequência na leitura de sinais Wi-Fi. Podem também ser testadas diferentes abordagens para o filtro de Kalman como EFK ou UFK, ou mesmo outros algoritmos que implementam o *fingerprinting* de sinais Wi-Fi e o PDR a fim de reduzir o erro na posição estimada.

Referências

- ABREU CYBIS, W. de; BETIOL, A. H.; FAUST, R. **Ergonomia e Usabilidade 3ª edição: Conhecimentos, Métodos e Aplicações**. Novatec Editora, 2015. Citado na p. 17.
- AMUNDSON, I.; KOUTSOUKOS, X. D. A survey on localization for mobile wireless sensor networks. In: SPRINGER. INTERNATIONAL workshop on mobile entity localization and tracking in GPS-less environments. 2009. P. 235–254. Citado na p. 19.
- ANDROID, D. Developers. **developer. android. com**, 2023. Citado na p. 35.
- ANUÁRIO Estatístico da UnB. 2019. <http://www.dpo.unb.br/images/phocadownload/unbemnumeros/anuarioestatistico/>. Citado na p. 14.
- BASIL, C. Os alunos com paralisia cerebral e outras alterações motoras. **Desenvolvimento psicológico e educação**, v. 2, p. 215–233, 2004. Citado na p. 14.
- BOORANAWONG, A.; SENGCHUAI, K.; BURANAPANICHKIT, D.; JINDAPETCH, N.; SAITO, H. RSSI-Based Indoor Localization Using Multi-Lateration With Zone Selection and Virtual Position-Based Compensation Methods. **IEEE Access**, v. 9, p. 46223–46239, 2021. ISSN 2169-3536. DOI: [10.1109/ACCESS.2021.3068295](https://doi.org/10.1109/ACCESS.2021.3068295). Citado nas pp. 16, 19, 21.
- BRASIL. Decreto No 5.296 de 2 de dezembro de 2004, 2004. Citado na p. 17.
- BRASIL. Lei No. 13.146/2015. Lei Brasileira de Inclusão da Pessoa com Deficiência., 2015. Citado na p. 17.
- BRASIL. Lei Nº10.048, de 8 de novembro de 2000. Lei Brasileira de Acessibilidade. **Diário Oficial [da] República Federativa do Brasil**, Brasília, DF, 2000. Disponível em: https://www.planalto.gov.br/ccivil_03/leis/L10098.htm. Citado na p. 14.
- BURKHOLDER, E. F. **The 3-D Global Spatial Data Model: Foundation of the Spatial Data Infrastructure**. CRC Press, 2008. ISBN 978-1-4200-6301-1. Citado nas pp. 27, 28.
- CHEN, G.; MENG, X.; WANG, Y.; ZHANG, Y.; TIAN, P.; YANG, H. Integrated WiFi/PDR/S-martphone using an unscented kalman filter algorithm for 3D indoor localization. **Sensors**, MDPI, v. 15, n. 9, p. 24595–24614, 2015. Citado na p. 21.
- CINTRA, R. S.; CAMPOS VELHO, H. F. de; TODLING, R. Redes neurais artificiais na melhoria de desempenho de métodos de assimilação de dados: Filtro de Kalman. **Trends in Computational and Applied Mathematics**, v. 11, n. 1, p. 29–39, 2010. Citado na p. 22.

- COLOMBO, A.; FONTANELLI, D.; MACII, D.; PALOPOLI, L. Flexible indoor localization and tracking based on a wearable platform and sensor data fusion. **IEEE Transactions on Instrumentation and Measurement**, IEEE, v. 63, n. 4, p. 864–876, 2013. Citado na p. 14.
- CRUZ, E. S. R. da. Desenvolvimento de framework de localização com acessibilidade para usuários com deficiência visual. In: 27º Congresso de Iniciação Científica da UnB e 18º Congresso de Iniciação Científica do DF. Brasília, Brasil, 2021. Citado nas pp. 34, 36.
- ESTUDOS E PESQUISAS EDUCACIONAIS ANÍSIO TEIXEIRA - INEP, I. N. de. **Resumo técnico do Censo da Educação Superior**. 2019. <https://www.gov.br/inep/pt-br/areas-de-atuacao/pesquisas-estatisticas-e-indicadores/censo-da-educacao-superior/>. Citado na p. 14.
- GOMES, J. P.; SOUSA, J. P.; CUNHA, C. R.; MORAIS, E. P. An indoor navigation architecture using variable data sources for blind and visually impaired persons. In: IEEE. 2018 13th Iberian Conference on Information Systems and Technologies (CISTI). 2018. P. 1–5. Citado na p. 13.
- GOWDA, A. K. S. N.; BABU, S. R.; SEKARAN, D. C. UMOISP: Usage Mode and Orientation Invariant Smartphone Pedometer. **IEEE Sensors Journal**, v. 17, p. 869–881, 3 fev. 2017. ISSN 1530-437X. DOI: [10.1109/JSEN.2016.2635691](https://doi.org/10.1109/JSEN.2016.2635691). Citado nas pp. 34, 35.
- IBGE, C. A. **Instituto Brasileiro de Geografia e Estatística (IBGE) Disponível em: <https://www.ibge.gov.br> Acesso: 06. dez. 2023**. 2010. Citado na p. 13.
- ISO, I. 9241-171-Ergonomics of human-system interaction–Guidance on software accessibility. **ISO, Ed**, 2008. Citado na p. 17.
- JANSSEN, V. Understanding coordinate reference systems, datums and transformations. **International Journal of Geoinformatics**, v. 5, n. 4, p. 41–53, 2009. Citado nas pp. 24–26.
- KUBAT, M. **An Introduction to Machine Learning**. Second: Springer International Publishing, 2017. ISBN 978-3-319-63912-3. DOI: [10.1007/978-3-319-63913-0](https://doi.org/10.1007/978-3-319-63913-0). Citado nas pp. 21, 22.
- KUNHOTH, J.; KARKAR, A.; AL-MAADEED, S.; AL-ALI, A. Indoor positioning and wayfinding systems: a survey. **HUMAN-CENTRIC COMPUTING AND INFORMATION SCIENCES**, 10, 1, MAY 2 2020. DOI: [10.1186/s13673-020-00222-0](https://doi.org/10.1186/s13673-020-00222-0). Citado nas pp. 13, 14, 16, 19, 21.
- LI, X.; WANG, J.; LIU, C.; ZHANG, L.; LI, Z. Integrated WiFi/PDR/smartphone using an adaptive system noise extended Kalman filter algorithm for indoor localization. **ISPRS International Journal of Geo-Information**, MDPI, v. 5, n. 2, p. 8, 2016. Citado na p. 21.

- LIGAS, M.; BANASIK, P. Conversion between Cartesian and geodetic coordinates on a rotational ellipsoid by solving a system of nonlinear equations. **Geodesy and Cartography**, v. 60, p. 145–159, 2 jan. 2011. ISSN 2080-6736. DOI: [10.2478/v10277-012-0013-x](https://doi.org/10.2478/v10277-012-0013-x). Citado na p. 25.
- LIN, K.-C.; WANG, J. Transformation from geocentric to geodetic coordinates using Newton's iteration. **Bulletin Géodésique**, v. 69, p. 300–303, 4 dez. 1995. ISSN 0007-4632. DOI: [10.1007/BF00806742](https://doi.org/10.1007/BF00806742). Citado na p. 26.
- MAO, G.; FIDAN, B.; ANDERSON, B. D. Wireless sensor network localization techniques. **Computer networks**, Elsevier, v. 51, n. 10, p. 2529–2553, 2007. Citado na p. 19.
- MARKIČ, Š.; DONAUBAUER, A.; BORRMANN, A. Enabling geodetic coordinate reference systems in building information modeling for infrastructure. In: PROCEEDING of the 17th International Conference on Computing in Civil and Building Engineering, Tampere, Finland. 2018. P. 5–7. Citado na p. 25.
- NASCIMENTO, J. A. M. do; AMARAL, S. A. do. **Avaliação de usabilidade na Internet**. Thesaurus, 2010. Citado na p. 17.
- NIELSEN, J. **Usability engineering**. Morgan Kaufmann, 1994. Citado na p. 17.
- NUAIMI, K. A.; KAMEL, H. A survey of indoor positioning systems and algorithms. In: p. 185–190. ISBN 978-1-4577-0311-9. DOI: [10.1109/INNOVATIONS.2011.5893813](https://doi.org/10.1109/INNOVATIONS.2011.5893813). Citado nas pp. 19–21.
- PAUL, A. K.; SATO, T. Localization in wireless sensor networks: A survey on algorithms, measurement techniques, applications and challenges. **Journal of sensor and actuator networks**, MDPI, v. 6, n. 4, p. 24, 2017. Citado na p. 13.
- PIMENTEL, D. C. C. et al. A modelagem tridimensional no tratamento do problema direto da geodésia. Universidade Federal de Santa Maria, 2012. Citado na p. 28.
- POULOSE, A.; KIM, J.; HAN, D. S. A sensor fusion framework for indoor localization using smartphone sensors and Wi-Fi RSSI measurements. **Applied Sciences**, Multidisciplinary Digital Publishing Institute, v. 9, n. 20, p. 4379, 2019. Citado nas pp. 16, 20–23, 34.
- RAITOHARJU, M.; NURMINEN, H.; PICHÉ, R. Kalman filter with a linear state model for PDR+ WLAN positioning and its application to assisting a particle filter. **EURASIP Journal on Advances in Signal Processing**, SpringerOpen, v. 2015, n. 1, p. 1–13, 2015. Citado nas pp. 21, 22.
- RUAN, L.; ZHANG, L.; LONG, Y.; CHENG, F. Coordinate references for the indoor/outdoor seamless positioning. In: COPERNICUS PUBLICATIONS GÖTTINGEN, GERMANY. PROCEEDINGS of the ICA. 2018. v. 1, p. 97. Citado na p. 28.
- SAJEEVAN, G. Latitude and longitude—A misunderstanding. **Current Science: March**, v. 94, n. 5, p. 568, 2008. Citado na p. 25.

-
- SASSAKI, R. K. Inclusão: acessibilidade no lazer, trabalho e educação. **Revista Nacional de Reabilitação (Reação)**, São Paulo, Ano XII, p. 10–16, 2009. Citado na p. 14.
- SIMON, D. **Optimal state estimation: Kalman, H, and nonlinear approaches**. Hoboken. NJ: Wiley-Interscience, 2006. Citado nas pp. 22–24.
- SRIDHARAN, M.; BIGHAM, J.; CAMPBELL, P. M.; PHILLIPS, C.; BODANESE, E. Inferring micro-activities using wearable sensing for ADL recognition of home-care patients. **IEEE journal of biomedical and health informatics**, IEEE, v. 24, n. 3, p. 747–759, 2019. Citado na p. 13.
- WU, Y.; ZHU, H.-B.; DU, Q.-X.; TANG, S.-M. A survey of the research status of pedestrian dead reckoning systems based on inertial sensors. **International Journal of Automation and Computing**, Springer, v. 16, n. 1, p. 65–83, 2019. Citado nas pp. 20, 21.
- YAO, H.; SHU, H.; SUN, H.; MOUSA, B.; JIAO, Z.; SUO, Y. An integrity monitoring algorithm for WiFi/PDR/smartphone-integrated indoor positioning system based on unscented Kalman filter. **EURASIP Journal on Wireless Communications and Networking**, Springer, v. 2020, p. 1–25, 2020. Citado nas pp. 20, 21.
- YETKİN, M.; BILGINER, Ö. On the Azimuth Determination. **Advanced Engineering Journal**, v. 1, n. 1, p. 17–20, 2021. Citado na p. 28.