

RANCANG BANGUN SISTEM NAVIGASI PADA APLIKASI *ROUTE GUIDANCE* UNTUK TUNANETRA BERBASIS *INDOOR POSITIONING*

PROPOSAL

Diajukan untuk melengkapi tugas-tugas dan
memenuhi syarat-syarat guna pelaksanaan penelitian Tugas Akhir

Oleh:

YUDA ADITYA
1608107010030



**JURUSAN INFORMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS SYIAH KUALA
DARUSSALAM, BANDA ACEH
JULI, 2022**

PENGESAHAN PROPOSAL

RANCANG BANGUN SISTEM NAVIGASI PADA APLIKASI *ROUTE GUIDANCE* UNTUK TUNANETRA BERBASIS *INDOOR POSITIONING*

DESIGN OF NAVIGATION SYSTEM FOR ROUTE GUIDANCE APPLICATION FOR VISUALLY IMPAIRED PERSON BASED ON INDOOR POSITIONING

Oleh:

Nama : Yuda Aditya
NPM : 1608107010030
Jurusan : Informatika

Menyetujui:

Pembimbing I

Pembimbing II

Kurnia Saputra, S.T., M.Sc.
NIP. 198003262014041001

Dalila Husna Yunardi, B.Sc., M.Sc.
NIP. 199006172015042001

Mengetahui:

Ketua Jurusan Informatika FMIPA
Universitas Syiah Kuala,

Dr. Muhammad Subianto, S.Si., M.Si
NIP. 196812111994031005

KATA PENGANTAR

Segala puji dan syukur kepada Allah SWT yang telah melimpahkan rahmat serta hidayah-Nya kepada kita semua dan juga atas izin-Nya penulis dapat menyelesaikan penulisan proposal ini. Tak lupa Shalawat dan Salam penulis sanjung sajikan kepada Nabi Besar Muhammad SAW, karena beliau telah membawa kita semua dari alam jahiliah ke alam yang penuh ilmu pengetahuan.

Proposal yang berjudul **“Rancang Bangun Sistem Navigasi pada Aplikasi *Route Guidance* untuk tunanetra berbasis *Indoor Positioning*”** ini telah dapat diselesaikan atas bantuan banyak pihak. Oleh karena itu, melalui tulisan ini penulis ingin mengucapkan terima kasih dan penghargaan sebesar-besarnya kepada:

1. Orang tua serta keluarga penulis yang senantiasa selalu mendukung aktivitas dan kegiatan penulis lakukan baik secara moral maupun material serta menjadi motivasi terbesar bagi penulis untuk menyelesaikan Proposal ini.
2. Bapak Kurnia Saputra, S.T., M.Sc., selaku Dosen Pembimbing I dan Ibu Dalila Husna Yunardi, B.Sc., M.Sc., selaku Dosen Pembimbing II yang telah banyak memberikan bimbingan dan arahan kepada penulis, sehingga penulis dapat menyelesaikan Proposal ini.
3. Bapak Dr. Muhammad Subianto, S.Si., M.Si., selaku Ketua Jurusan Informatika.
4. Bapak Kurnia Saputra, S.T., M.Sc., selaku Dosen Wali penulis.
5. Seluruh Dosen di Jurusan Informatika Fakultas MIPA atas ilmu dan didikannya selama perkuliahan.
6. Andika Pratama, Budi Gunawan, Fauzy Nisa, dan Muammar Zikri Aksana selaku teman yang telah banyak memberikan dukungan, masukan serta ilmu yang cukup besar dan bermanfaat dalam penulisan Proposal ini.
7. Seluruh teman-teman seperjuangan Jurusan Informatika Unsyiah 2016 lainnya.

Penulis juga menyadari segala ketidaksempurnaan yang terdapat didalamnya baik dari segi materi, cara, ataupun bahasa yang disajikan. Seiring dengan ini penulis mengharapkan kritik dan saran dari pembaca yang sifatnya dapat berguna untuk kesempurnaan Proposal ini. Harapan penulis semoga tulisan ini dapat bermanfaat bagi banyak pihak dan untuk perkembangan ilmu pengetahuan.

Banda Aceh, Juli 2022

Penulis

DAFTAR ISI

PENGESAHAN PROPOSAL	ii
KATA PENGANTAR	iii
DAFTAR ISI	v
DAFTAR TABEL	vi
DAFTAR GAMBAR	vii
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	3
1.3. Tujuan Penelitian	3
1.4. Manfaat Penelitian	3
BAB II TINJAUAN KEPUSTAKAAN	4
2.1. <i>Indoor Positioning System</i>	4
2.2. <i>Route Guidance System / Wayfinding and Navigation System</i> . .	5
2.3. <i>Bluetooth Low Energy</i> (BLE)	5
2.3.1. <i>Beacon</i>	5
2.4. <i>Fingerprinting</i>	6
2.4.1. <i>Received Signal Strength Indicator</i> (RSSI)	7
2.4.2. Kalman Filter	8
2.5. <i>Speech Recognition</i>	10
2.5.1. <i>Text-to-Speech</i> (TTS)	10
2.6. VOSK API	10
2.7. Android	11
2.8. Android Studio	11
2.9. SCRUM	12
2.10. <i>Black Box Testing</i>	14
2.11. <i>Usability Testing</i>	15
2.11.1. <i>System Usability Scale</i> (SUS)	15
BAB III METODOLOGI PENELITIAN	17
3.1. Waktu dan Lokasi Penelitian	17
3.2. Alat dan Bahan	17
3.3. <i>Roadmap</i> Penelitian	18
3.4. Metode Penelitian	21
3.4.1. Identifikasi Masalah	21
3.4.2. Studi Literatur	22
3.4.3. Perancangan Sistem	22
3.4.4. Pengumpulan Data	27
3.4.5. Pembuatan Sistem Aplikasi	28
3.4.6. Pengujian Akurasi	28

3.4.7. Pengujian Fungsionalitas dengan Metode <i>Black Box</i> . .	29
3.4.8. <i>Usability Testing</i>	30

DAFTAR KEPUSTAKAAN	34
---------------------------	-----------

DAFTAR TABEL

Tabel 2.1	Kekuatan sinyal RSSI (Sideeq dan Qasem, 2016)	8
Tabel 3.1	Daftar Pertanyaan Metode SUS (Kaya dkk., 2019)	30
Tabel 3.2	Interpretasi skor sus	31

DAFTAR GAMBAR

Gambar 2.1	Ilustrasi Metode <i>Fingerprinting</i> (Yudha dkk., 2018)	7
Gambar 2.2	Ilustrasi Kalman Filter (Ihsan dkk., 2018)	8
Gambar 2.3	Ilustrasi Metode Pengembangan Menggunakan Scrum (Schwaber dan Sutherland, 2011).	14
Gambar 3.1	<i>Roadmap</i> Penelitian Fase 1	19
Gambar 3.2	<i>Roadmap</i> Penelitian Fase 2	20
Gambar 3.3	Diagram Alir Penelitian	21
Gambar 3.4	<i>Use Case Diagram</i>	23
Gambar 3.5	<i>Deployment Diagram</i>	24
Gambar 3.6	Alur Kerja Sistem	24
Gambar 3.7	<i>Flow Diagram</i> Navigasi <i>Indoor</i>	25
Gambar 3.8	Tampilan Halaman Aplikasi Navigasi Indoor untuk Tunanetra	26

BAB I

PENDAHULUAN

1.1. Latar Belakang

Manusia adalah makhluk sosial yang selalu berinteraksi dan berkomunikasi dengan manusia lainnya, berbagai cara penyampaian informasi dan komunikasi dilakukan untuk saling terhubung dengan lingkungan sekitar. Cara berkomunikasi yang paling sering dilakukan oleh manusia adalah berbicara atau menggunakan media suara, dan beberapa cara lain seperti tulisan, isyarat serta media visual seperti gambar. Dengan adanya suara juga dapat membantu manusia yang memiliki keterbatasan penglihatan, sebagai pengganti indra penglihatan mereka dengan mengandalkan serta mempertajam indra pendengaran mereka. Menurut *World Health Organization* disebutkan dalam laporan *World Report on Vision*, secara global setidaknya sekitar 2,2 Miliar orang di antaranya memiliki gangguan penglihatan atau kebutaan, di mana setidaknya 1 miliar di antaranya memiliki gangguan penglihatan yang dapat dicegah atau belum ditangani (WHO, 2020).

Seiring perkembangan zaman di mana kemudahan berkomunikasi dapat dibantu dengan teknologi yang berkaitan dengan suara telah banyak dikembangkan. Selain itu dengan adanya teknologi ini dapat membantu orang-orang dengan keterbatasan penglihatan dalam berbagai aspek kehidupan seperti, berkomunikasi serta bekerja sebagai pemandu jalan atau arah. Selain berkomunikasi, pemandu arah juga menjadi salah satu kebutuhan penting bagi orang-orang dengan keterbatasan penglihatan bahkan tidak hanya mereka. Untuk mengurangi kesulitan pengguna tunanetra dan untuk memastikan posisi pengguna serta menemukan destinasi tujuannya dibutuhkan teknologi berupa *Global Positioning Systems* (GPS). Namun, saat ini sistem yang menggunakan sensor GPS bekerja dengan baik sebagai alat pencari rute di luar ruangan, sehingga tidak memadai untuk memandu orang di dalam ruangan (Ko dan Kim, 2017). Oleh karena itu tujuan dari penelitian ini adalah untuk membangun sistem pencarian rute yang efektif dengan menggunakan *Indoor Positioning System*.

Indoor Positioning System (IPS) merupakan teknologi untuk melacak suatu objek atau individu di area tertutup atau gedung merupakan bagian dari Location Based Service (LBS) (Brena dkk., 2017). LBS adalah layanan yang menyediakan informasi bagi pengguna berdasarkan lokasi pengguna (Virrantaus dkk., 2001). Teknologi IPS ini nantinya akan di gunakan sebagai dasar dari *Wayfinding System* atau *Route Guidance System* dan akan dipadukan dengan *Speech Command Recognition*.

Speech Recognition merupakan salah satu bentuk dari teknologi yang disebut *Artificial Intelligence (AI)*. *Speech Recognition* sendiri merupakan teknologi pada perangkat komputer agar dapat mengenali dan memahami kata-kata yang diucapkan oleh manusia. Pada sistem *Speech Recognition* memiliki 2 sistem utama yaitu *Speech-to-Text (STT)* dan *Text-to-Speech (TTS)*. *Speech Recognition* juga dikenal sebagai sistem *Speech-to-Text* dimana suara diterima dan diterjemahkan sebagai teks agar dapat dikenali. Sedangkan *Text-to-Speech* merupakan teks yang diterjemahkan oleh perangkat menjadi suara sebagai keluaran dari perangkat sehingga suara dapat didengar oleh pengguna. Seiring berjalannya waktu penerapan *Speech Recognition* yang terkenal adalah Google Assistant oleh Google, Siri oleh Apple dan Alexa dari Amazon. Teknologi ini akan diimplementasikan pada *smartphone* Android yang dimiliki tunanetra untuk memandu rute pada gedung A Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Syiah Kuala (USK).

Android sendiri telah didukung fitur *mobile GPS*, *Geolocation*, *Bluetooth*, *Voice Recorder*, *Speaker*, dsb. Fitur tersebut dapat mendukung teknologi yang akan dibangun dalam penelitian ini. Pada penelitian ini juga menggunakan IPS dengan menggunakan *Bluetooth Low Energy (BLE)* dan *Speech Command Recognition* akan diimplementasikan pada aplikasi android untuk penentu lokasi dan pemandu rute. BLE memiliki kelebihan-kelebihan dibandingkan protokol *Internet of Things (IoT)* lainnya di antaranya konfigurasi yang mudah, metode pengiriman data yang mudah serta jangkauan konektivitas yang luas (Puspitasari, 2020). Pada proses implementasi, Beberapa ruangan pada Gedung A Fakultas Matematika dan Ilmu Pengetahuan Alam USK akan dipasang alat transmisi BLE. BLE bertugas untuk memancarkan gelombang radio untuk mengirimkan informasi secara berkala ke *smartphone* Android yang disebut *Beacon* (Lin dan Lin, 2018). Informasi yang diperoleh berupa kekuatan sinyal atau *signal strength*. Kemudian, *smartphone* pengguna akan menangkap indeks kekuatan sinyal atau *Recieved Signal Strength Indicator (RSSI)* yang dipancarkan dari masing-masing *Beacon* (Li dkk., 2018). Dengan memanfaatkan RSSI dari *Beacon* untuk menentukan lokasi dari pengguna serta sistem akan menggunakan masukan berupa suara untuk menentukan ke mana tujuan pengguna dan keluaran berupa rute menuju tujuan pengguna yang dipandu menggunakan suara dari *smartphone*.

1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan sebelumnya, maka masalah yang akan dikaji pada penelitian ini adalah sebagai berikut:

1. Bagaimana membangun aplikasi *Speech Command Recognition* untuk *Route Guidance* berbasis *Indoor Positioning* untuk tunanetra ?
2. Bagaimana aplikasi dapat membantu serta memandu tunanetra agar sampai ke tujuan berdasarkan lokasi dengan menggunakan teknologi *Bluetooth Low Energy* (BLE) menggunakan metode *Received Signal Strength Indicator* (RSSI) ?
3. Bagaimana proses pengimplementasian Model *Speech Command Recognition* ke dalam aplikasi *Route Guidance* berbasis *Indoor Positioning* ?
4. Bagaimana tingkat akurasi dari lokasi pengguna serta akurasi dari pemandu saat memandu pengguna dengan suara ?

1.3. Tujuan Penelitian

Berdasarkan rumusan masalah yang telah disebutkan sebelumnya, maka dapat dipaparkan tujuan dari penelitian ini adalah sebagai berikut:

1. Membangun aplikasi *Speech Command Recognition* untuk *Route Guidance* berbasis *Indoor Positioning* untuk tunanetra.
2. Membangun aplikasi yang dapat membantu serta memandu tunanetra agar sampai ke tujuan.
3. Mengimplementasi Model *Speech Command Recognition* ke dalam aplikasi *Route Guidance* berbasis *Indoor Positioning*.
4. Menganalisis tingkat akurasi dari lokasi pengguna serta akurasi dari pemandu saat memandu pengguna dengan suara.

1.4. Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah sebagai berikut:

1. Memberikan kemudahan bagi pengguna terkhusus kepada tunanetra untuk menuju ke ruangan dipandu dengan navigasi suara.
2. Memberikan aplikasi *Route Guidance* dengan teknologi *Indoor Positioning* serta *Speech Command Recognition* dengan keakuratan yang baik sehingga pengguna dapat sampai ke tujuan.

BAB II

TINJAUAN KEPUSTAKAAN

Untuk mendukung penelitian ini, maka dalam bab ini akan dikemukakan beberapa rumusan teori pendukung, yang dikutip dari berbagai referensi baik dalam bentuk buku, artikel, maupun tulisan karya ilmiah lainnya termasuk hasil penelitian sebelumnya yang ada kaitannya dengan penelitian yang dilakukan.

2.1. *Indoor Positioning System*

Indoor Positioning System (IPS, atau "*Indoor Location System*") merupakan teknologi yang menyediakan informasi bagi pengguna berdasarkan lokasi pengguna di dalam sebuah gedung (Brena dkk., 2017). IPS merupakan proses mendapatkan perangkat atau lokasi pengguna di dalam ruangan atau lingkungan (Zafari dkk., 2019). Banyak penelitian yang dilakukan dalam mengimplementasi IPS dengan berbagai pemanfaatan seperti WLAN. Namun pemanfaatan WLAN memiliki konsumsi daya yang boros pada smartphone sehingga pada akhirnya implementasi IPS diterapkan pada *Bluetooth low Energy* (BLE) yang memiliki konsumsi daya yang lebih rendah. Metode yang dapat dilakukan dalam pengimplementasian IPS adalah metode *trilateration* (berdasarkan jarak), *triangulation* (berdasarkan sudut) dan *fingerprinting* (Puspitasari, 2020).

Pada luar ruangan untuk mendeteksi suatu lokasi sudah diterapkan dengan adanya teknologi GPS, di mana GPS banyak memberikan dampak yang luar biasa pada kehidupan sehari-hari. Namun kegunaan GPS atau sistem lokasi berbasis satelit hanya terbatas pada lingkungan di luar ruangan. Dengan demikian dibutuhkan suatu metode dan teknologi khusus untuk sistem lokasi pada dalam ruangan yang disebut sebagai *Indoor Positioning System* (IPS) (Brena dkk., 2017). Dengan menggunakan sensor dan teknologi komunikasi sistem penentuan posisi pada IPS dapat menemukan objek di lingkungan dalam ruangan. Peluang pasar pada penggunaan IPS cukup besar sehingga dapat menarik minat ilmiah komunikasi untuk menerapkan teknologi tersebut (Brena dkk., 2017).

Pada teknologi IPS memiliki beberapa pendekatan yaitu, menentukan posisi dengan berbasis WiFi (WPS), menentukan posisi berbasis *Bluetooth Low Energy* (BLE), menentukan posisi berbasis Identifikasi Frekuensi Radio (RFID) dan menentukan posisi dengan menggunakan teknologi *Ultra-Wide Band* (UWB) atau *Visible Light Communication* (VLC) (Cantón Paterna dkk., 2017). IPS juga dapat dipadukan dengan kemampuan *route guidance* atau pemandu rute serta didampingi dengan sistem pemandu dengan suara.

2.2. *Route Guidance System / Wayfinding and Navigation System*

Menurut (Karimi, 2015), *Wayfinding and navigation system* memiliki 2 makna, yaitu merupakan perangkat khusus yang menyediakan solusi pencarian jalan dan navigasi tanpa memerlukan tautan komunikasi atau koneksi ke sistem lain dan tanpa penyedia pihak ketiga. Makna ke dua adalah referensi layanan melalui *smartphone* yang menyediakan solusi pencarian arah dan navigasi hanya melalui tautan komunikasi kabel-nirkabel ke sistem jarak jauh yang didukung oleh penyedia pihak ketiga. Dari perspektif komputasi, *Wayfinding and navigation system* adalah platform terpusat, terdistribusi, dan didistribusikan melalui klien (pada *smartphone*) dan server yang dikelola oleh penyedia layanan.

2.3. *Bluetooth Low Energy* (BLE)

Teknologi *Bluetooth* dikembangkan oleh Ericsson pada tahun 1994 dengan kegunaan sebagai standar komunikasi nirkabel untuk bertukar data dalam jarak dekat (Kaluža dkk., 2017). Teknologi *Bluetooth* memiliki fitur utama yaitu memiliki biaya yang rendah, konsumsi daya rendah, jangkauan kecil, memiliki ketahanan dan penggunaan secara global. Kecepatan transfer data yang diberikan oleh *Bluetooth* sebesar 1 Mbit/s dan menggunakan pita frekuensi tanpa izin 2,4 hingga 2,485 GHz (Kaluža dkk., 2017).

Pertengahan tahun 2010 spesialis otoritas kompeten yang bernama "Bluetooth Special Interest Group" (SIG) mengumumkan spesifikasi *Bluetooth* 4.0, dimana meliputi : *Bluetooth* Klasik, *Bluetooth* dengan kecepatan tinggi, dan protokol *Bluetooth* hemat energi. Karakteristik dari *Bluetooth Low Energy* (BLE) adalah memiliki ukuran yang kecil, menggunakan biaya rendah, konsumsi penggunaan daya yang rendah dengan kemungkinan penggunaan sampai beberapa tahun bekerja dengan menggunakan baterai AAA dan memiliki kompatibilitas dengan perangkat seluler, tablet, dan komputer (Kaluža dkk., 2017).

2.3.1. *Beacon*

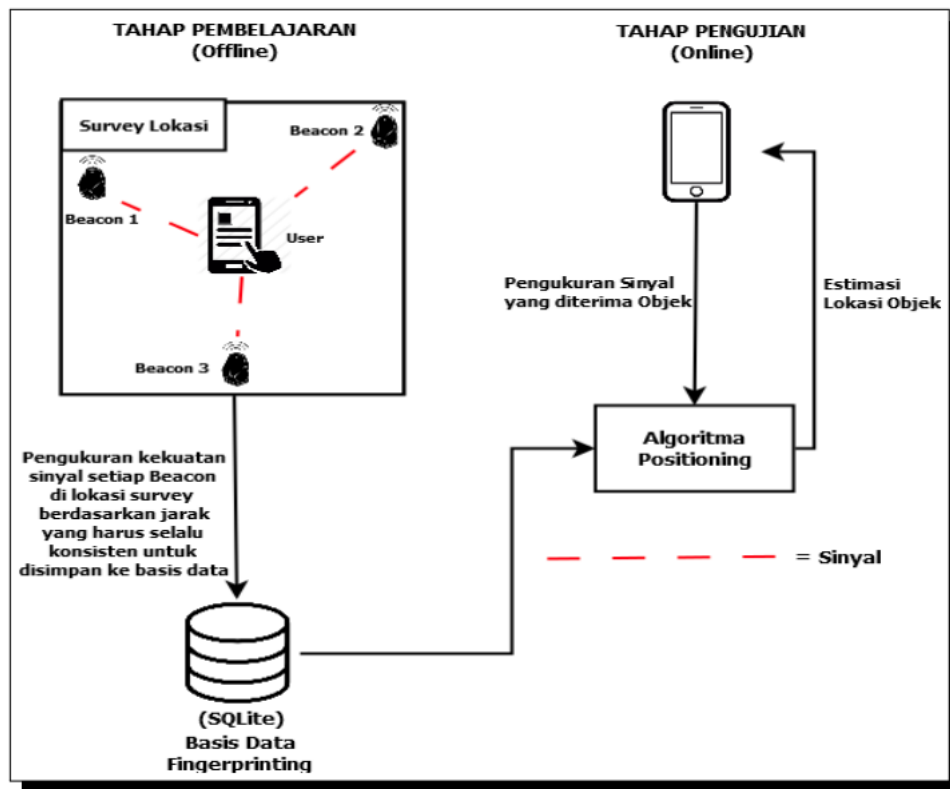
BLE memancarkan sinyal dari alat transmiter yang beroperasi menggunakan baterai. Alat transmiter tersebut disebut dengan *Beacon*. *Beacon* merupakan alat pendeteksi lokasi dengan harga yang terjangkau, ukurannya yang kecil, memiliki daya tahan baterai yang cukup lama, dan tidak membutuhkan energi listrik tambahan (Puspitasari, 2020). Setiap perangkat *smartphone* dan tablet yang mendeteksi sinyal dari *Beacon*, dapat menghitung jarak dan memperkirakan keberadaan lokasi setiap perangkat sekaligus. *Beacon Bluetooth* mengubah pengalaman menggunakan *smartphone* untuk bepergian, berbelanja, bekerja, dan bermain (Kaluža dkk., 2017).

2.4. *Fingerprinting*

Fingerprinting adalah teknik untuk menentukan lokasi dengan pemanfaatan *Radio Signal Strength* (RSS) dari suatu *Access Point* (AP). Menurut (Yudha dkk., 2018), metode ini memperhitungkan atenuasi karena kekuatan sinyal sering berubah-ubah. Setiap titik referensi dikumpulkan yang mengintegrasikan adanya penghalang antara transmitter dan *receiver*.

Transmitter untuk teknologi IPS yang ditunjukkan untuk WLAN disebut dengan AP, sedangkan transmitter untuk teknologi IPS yang ditunjukkan untuk *Bluetooth Low Energy* disebut dengan *Beacon* Puspitasari (2020). Metode fingerprinting berbasis IPS ini terdiri dari 2 tahap. Tahap pertama adalah tahap pembelajaran (*offline*), di mana lokasi fingerprints itu sendiri diperoleh dengan cara mengumpulkan RSSI dalam satuan desibel (dB) yang dipancarkan dari masing-masing AP. Kemudian gelombang radio yang dipancarkan AP yang diletakkan pada posisi yang telah ditentukan sebelumnya ditangkap oleh smartphone yang terintegrasi dengan WLAN ataupun *Bluetooth*.

Selama tahap pembelajaran lokasi yang tidak diketahui data pembelajarannya kemudian dirujuk sebagai titik referensi estimasi lokasi. Tahap kedua adalah tahap pengujian (*online*), di mana keakuratan yang dikumpulkan dalam data pembelajaran. Selama tahap pengujian sistem harus memberikan lokasi setiap objek berdasarkan data RSSI yang diamati. Namun, nilai RSSI bisa dipengaruhi oleh keadaan lingkungan sekitar yang dapat mengganggu keakuratannya (Subhan dkk., 2011). Menurut (Yudha dkk., 2018), metode Fingerprinting berbasis IPS ini melibatkan 2 tahap yang dapat dilihat pada Gambar 2.1.



Gambar 2.1. Ilustrasi Metode *Fingerprinting* (Yudha dkk., 2018)

2.4.1. *Received Signal Strength Indicator (RSSI)*

Received Signal Strength Indicator atau RSSI adalah metode pengukuran jarak yang memperoleh sinyal dari transmitter seperti *Bluetooth* untuk menentukan *distance-loss* model, dan kemudian memperkirakan posisi pengguna melalui beberapa algoritma ((Li dkk., 2018)). Menurut (Puspitasari, 2020) *Positioning System* menghasilkan data yang penting untuk menghitung lokasi pengguna. *Time of Arrival (TOA)*, *Time Difference of Arrival (TDOA)*, *Angle of Arrival (AOA)*. RSSI memperkirakan jarak *node* yang belum diketahui ke referensi *node* dari beberapa kumpulan unit perhitungan dengan menggunakan atenuasi kekuatan sinyal (*signal strength*) yang dipancarkan oleh transmitter (Puspitasari, 2020).

Nilai RSSI didefinisikan dengan bilangan negatif. Semakin tinggi bilangan negatifnya, maka kekuatan sinyal tersebut tergolong lemah. Namun apabila nilainya mendekati 0, maka sinyal tersebut tergolong kuat. RSSI dapat digolongkan menjadi 5 kategori kekuatan sinyal seperti Tabel 2.1

Tabel 2.1. Kekuatan sinyal RSSI (Sideeq dan Qasem, 2016)

No.	<i>Signal Noise Ratio</i> (SNR)	RSSI
1.	Kurang dari -40 dB	Luar Biasa
2.	-40 dB hingga -55 dB	Sangat Baik
3.	-55 dB hingga -70 dB	Baik
4.	-70 dB hingga -80 dB	Cukup Baik
5.	Lebih dari -80 dB	Buruk

2.4.2. Kalman Filter

Kalman Filter adalah seperangkat persamaan matematika yang menyediakan cara komputasi yang efisien untuk memperkirakan keadaan suatu proses, dengan cara meminimalkan rata-rata kesalahan kuadrat. Filter ini sangat berguna dalam beberapa aspek: mendukung estimasi-estimasi keadaan lampau, sekarang dan masa depan, dan dapat melakukannya bahkan ketika sifat pasti dari sistem yang dimodelkan tidak diketahui Welch (2020). Kalman Filter sendiri merupakan algoritma yang menggunakan serangkaian data yang diamati dari waktu ke waktu, yang mengandung *noise* dan ketidakakuratan lainnya, untuk memperkirakan variabel yang tidak diketahui dengan lebih akurat. Ini diusulkan oleh R. E. Kalman pada tahun 1960, dan menjadi pendekatan standar untuk estimasi yang optimal. Karena memiliki keunggulan waktu secara *realtime*, cepat, efisien, dan anti-interferensi yang kuat, Kalman filter telah banyak diterapkan di bidang perhitungan orbit, pelacakan target dan navigasi Li dkk. (2015).

Menurut (Ihsan dkk., 2018) berikut proses Kalman Filter dapat dilihat pada gambar dibawah ini:



Gambar 2.2. Ilustrasi Kalman Filter (Ihsan dkk., 2018)

Pada Kalman Filter dilakukan dengan dua tahapan yaitu tahap prediksi (*time update*) dan tahap pembaruan (*measurement update*) terhadap data-data pengukuran untuk memperbaiki hasil estimasi. Tahap prediksi adalah suatu tahapan untuk

mengubah suatu keadaan menjadi keadaan berikutnya dengan mengabaikan error yang ada (Ihsan dkk., 2018).

Berikut rumus menurut (Ihsan dkk., 2018):

- *Time Update (Predict)*

Predict State

$$\hat{x} = \hat{x} \quad (2.1)$$

Predict error covariance

$$p = p + q \quad (2.2)$$

- *Measurements update (Correct)*

Update the estimate via k

$$\hat{x} = \hat{x} + k * (measurement - \hat{x}) \quad (2.3)$$

Kalman gain

$$k = p / (p + r) \quad (2.4)$$

Update the error covariance

$$p = (1 - k) * p \quad (2.5)$$

Keterangan

x: Nilai yang di filter

p: Error estimasi

q: Noise yang diakibatkan dari proses

k: Kalman Gain

r: Noise dari sensor

Kalman Filter menerima inputan *gain* untuk meminimumkan kovariansi *error*. Kalman gain berupa inputan *high-gain* dan *low-gain*, dimana *high-gain* dapat mendekati posisi sebenarnya karena menghasilkan lintasan perkiraan lebih responsif sementara *low-gain* dapat menghaluskan *noise* namun mengurangi responsifnya. Oleh karena itu, semakin tinggi inputan nilai gain dari Kalman Filter maka semakin mendekati posisi sebenarnya (Ihsan dkk., 2018).

2.5. Speech Recognition

Speech Recognition adalah kemampuan suatu komputer agar dapat mengenali apa yang diucapkan oleh seseorang berdasarkan sinyal suara yang diucapkan oleh seseorang atau bisa disebut sebagai *Automatic Speech Recognition* (ASR)(Jurafsky dan Martin, 2008). *Speech Recognition* merupakan antarmuka alami untuk berkomunikasi dengan peralatan komputer seperti *smart home appliances*, *personal assistants*, atau *smartphone*. ASR Juga berguna untuk *general transcription*, contohnya untuk membuat *captions* secara otomatis untuk audio atau video (mentranskripsikan film atau video atau *live discussions*) (Jurafsky dan Martin, 2008). Menjadi suatu kemudahan bagi manusia jika komputer dapat memahami apa yang diucapkan oleh manusia dan karena hal itu juga manusia dapat dengan mudah mengoperasikan komputer karena adanya teknologi yang *Automatic Speech Recognition*. serta *speech synthesis* atau *text-to-speech* (TTS) (Jurafsky dan Martin, 2008).

2.5.1. Text-to-Speech (TTS)

Speech Syntesis atau text-to-speech merupakan kebalikan dari ASR dalam memetakan teks ke bentuk gelombang akustik. TTS memiliki berbagai macam aplikasi. TTS digunakan dalam agen percakapan yang berdialog dengan orang-orang, berperan dalam perangkat yang membacakan dengan keras untuk tunanetra atau dalam permainan, dan dapat digunakan untuk berbicara bagi penderita gangguan neurologis, seperti mendiang Steven Hawking seorang ahli astrofisika berbicara dengan memanipulasi sistem TTS (Jurafsky dan Martin, 2008).

2.6. VOSK API

Vosk api merupakan toolkit untuk speech recognition, di mana memiliki beberapa kelebihan, yaitu (Cephei, 2019):

1. Memiliki 19 lebih bahasa dan dialek yang didukung vosk.
2. Vosk api merupakan toolkit speech recognition yang bisa digunakan secara offline, yang dapat digunakan pada Raspberry Pi, Android, iOS.
3. Kemudahan untuk menginstalasi vosk dengan menggunakan pip3 install vosk.
4. Model yang portabel pada masing-masing bahasa sebesar 50Mb, namun ada beberapa model server yang lebih besar pula.
5. Menyediakan streaming API untuk pengalaman pengguna terbaik.

6. Memiliki beberapa paket bahasa pemrograman yang berbeda-beda, seperti java, csharp, javascript dll.
7. Untuk akurasi terbaik dapat memungkinkan konfigurasi ulang kosakata dengan cepat.
8. Mendukung identifikasi pembicara selain dengan speech recognition yang sederhana.

2.7. Android

Android merupakan suatu *software* (perangkat lunak) yang digunakan pada *mobile device* (perangkat berjalan) yang meliputi sistem operasi, *middleware*, dan aplikasi inti. Android *Standart Development Kit* (SDK) menyediakan alat dan *Application Programming Interface* (API) yang diperlukan untuk memulai pengembangan aplikasi pada platform Android menggunakan bahasa pemrograman Java, yaitu kode Java yang terkompilasi dengan data dan *file resources* yang dibutuhkan aplikasi dan digabungkan oleh *app tools* menjadi paket Android. *File* tersebut ditandai dengan ekstensi .apk. *File* inilah yang didistribusikan sebagai aplikasi dan dipasang pada perangkat *mobile* (Nasution dkk., 2018).

Menurut (Shaheen dkk., 2017). Ada 4 jenis komponen aplikasi. Setiap jenis memiliki tujuan yang berbeda dan memiliki siklus proses yang berbeda yang menentukan bagaimana komponen di *create* dan di *destroy*. Berikut adalah ke 4 jenis komponen tersebut:

1. *Activities*, merupakan sebuah *activity* merepresentasikan tampilan aplikasi kepada pengguna (*user interface*).
2. *Service*, merupakan komponen yang berjalan pada *background* untuk menjalankan operasi atau proses yang tidak memiliki *user interface*.
3. *Content Providers*, merupakan komponen yang menangani data antar aplikasi.
4. *Broadcast Receivers*, merupakan komponen yang bertanggung jawab atas menerima serta menyampaikan informasi atau notifikasi.

2.8. Android Studio

Android Studio merupakan *Integrated Development Environment* (IDE) untuk pengembangan platform Android. Android Studio diumumkan pada tanggal 16 Mei 2013 di Google I/O *Conference*. Android Studio dapat digunakan secara gratis di bawah pengawasan *Apache License 2.0*. Android Studio merupakan

kolaborasi antara JetBrains dan Google. Android Studio mirip dengan Eclipse yang disertai dengan ADT *Plugin* (Android *Development Tools*)(Craig dan Gerber, 2015).

Fitur-fitur Android Studio menurut (Puspitasari, 2020) adalah sebagian berikut:

- Proyek berbasis pada Gradle *Build*.
- *Refactory* dan perbaikan bug yang cepat.
- *Tools* baru yang bernama "Lint" diklaim dapat memonitor kecepatan, kegunaan, serta kompatibilitas aplikasi dengan cepat.
- Mendukung Proguard and *App-signing* untuk keamanan.
- Memiliki GUI aplikasi Android lebih mudah.
- Didukung oleh Google *Cloud Platform*, sehingga lebih mudah mengintegrasikan Google *Cloud Messaging and Application Engine* untuk setiap aplikasi yang dikembangkan.

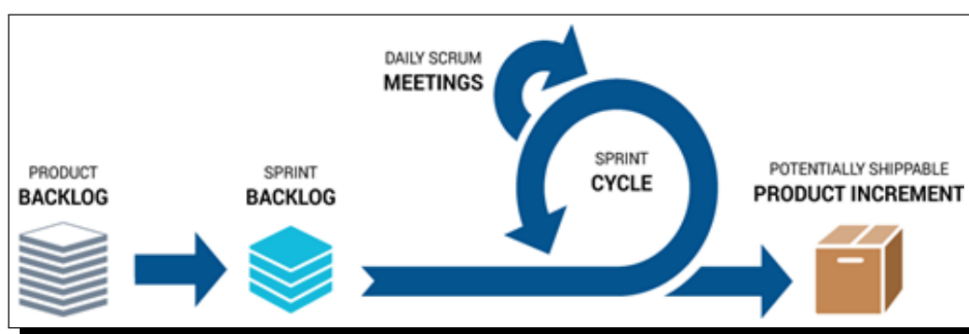
2.9. SCRUM

Scrum adalah *development framework* di mana tim lintas-fungsi mengembangkan produk atau proyek secara berulang dan bertahap. *Scrum* menyusun siklus pengembangan yang disebut *Sprint*. Iterasi ini masing-masing tidak lebih dari empat minggu dan berlangsung satu demi satu tanpa jeda (Deemer dkk., 2012). *Sprint* memiliki durasi tetap atau *Sprint* berakhir pada tanggal tertentu baik selesai atau belum, dan tidak pernah diperpanjang. Oleh karena itu *Sprint* dikatakan *timeboxed* (Schwaber dan Sutherland, 2011). Tahapan-tahapan metode *Scrum* menurut (Schwaber dan Sutherland, 2011) adalah sebagai berikut :

1. Dimulai dengan mengumpulkan *user requirements*, namun tidak harus semua *requirements* diharapkan harus keluar dari pemikiran pengguna di tahap awal proses pengembangan. Pengguna dapat mengubah pikiran mereka di setiap waktu selama proses pengembangan, pengguna dapat menambah fitur-fitur baru, menghapus atau memperbaharui beberapa fitur yang telah ada sebelumnya.
2. Tahapan selanjutnya adalah memprioritaskan *requirements* dan *Product Backlog*. Sebuah perencanaan yang tepat dalam *Sprint* harus dilakukan sesuai dengan jumlah *Sprint* yang dibutuhkan untuk mengembangkan perangkat lunak, yang terdiri dari durasi *Sprint* tersebut dan *requirements* apa saja yang terdapat di *Product Backlog* yang harus diimplementasikan di setiap *Sprint* (dikenal dengan *Sprint Backlog*).

3. *Sprint* diawali dengan *Sprint Planning* di mana *Product Owner*, satu orang yang telah diberikan wewenang dan bertanggung jawab untuk memaksimalkan nilai produk di pasar, bertemu dengan tim *Scrum* (tim dengan jumlah 2-9 orang), kemudian bekerja sama untuk memperkirakan *requirements* dari *Product Backlog* apa saja yang dikerjakan selama satu *Sprint*.
4. *Sprint Planning* difasilitasi dengan *Scrum Master*. *Scrum Master* adalah seorang pemimpin yang melayani (*Servant Leader*). *Sprint Planning* memiliki batasan waktu selama 8 jam di dalam sebuah *Sprint* yang berdurasi selama 30 hari. Keluaran dari *Sprint Planning* adalah daftar pekerjaan dari hasil kesepakatan antara *Product Owner* dan tim *Scrum* di mana pekerjaan itu yang dikerjakan oleh tim *Scrum* nantinya selama satu *Sprint* beserta *Sprint Goal* yang dinamakan dengan *Sprint Backlog*.
5. Setelah *Sprint Planning* berakhir, tim *Scrum* akan mengambil *Sprint Backlog* untuk diri mereka masing-masing dan mengerjakan *Sprint Backlog* setiap hari hingga akhir *Sprint* tanpa campur tangan dari pihak mana pun. *Daily Scrum* akan dikerjakan oleh tim *Scrum* yang tidak lebih dari 15 menit untuk menentukan apa saja yang akan mereka kerjakan selama 24 jam ke depan berdasarkan perkembangan 24 jam terakhir, serta menyampaikan permasalahan yang menghambat mereka untuk bisa mencapai *Sprint Goal*. Tim *Scrum* akan melakukan perbaikan-perbaikan item dari *Product Backlog* pada *Sprint* yang akan datang selama proses pengembangan berlangsung, dengan tujuan membuat *Sprint Planning* menjadi lebih efektif.
6. Di akhir *Sprint* saat acara *Sprint Review*, *Product Owner* akan mempresentasikan hasil pekerjaan tim *Scrum* selama satu *Sprint* dan juga menjelaskan apa saja pencapaian tim *Scrum* menuju *Sprint Goal* di dalam *Sprint* tersebut kepada para pemegang kepentingan (*stakeholder*) agar mendapatkan *feedback*. *Feedback* ini akan dimasukkan ke dalam *Product Backlog* agar meningkatkan nilai dari sebuah produk. *Sprint Review* memiliki batasan waktu tidak lebih dari 4 jam untuk *sprint* yang memiliki durasi selama 30 hari.
7. Setelah *Sprint Review*, *Scrum Master* memfasilitasi acara yang bernama *Sprint Retrospectives* agar tim *Scrum*, *Product Owner* bekerja sama menentukan apa saja peningkatan yang akan mereka implementasikan di *Sprint* berikutnya. *Definition of Done* adalah salah satu hal yang ditekankan oleh tim *Scrum* pada saat *Sprint Retrospectives*.

8. *Sprint Retrospectives* merupakan acara yang paling penting dalam *Scrum* dikarenakan sifatnya yang menekankan *continuous learning* yang dapat meningkatkan tingkat *agility* perusahaan. *Sprint Review* memiliki batasan waktu tidak lebih dari 3 jam untuk *Sprint* yang memiliki durasi selama 30 hari. Setelah *Sprint Retrospectives* berakhir, maka *Sprint* berikutnya akan langsung dilakukan tanpa jeda antar *Sprint*. Pada setiap *Sprint*, *Product Owner* akan memastikan agar produk mencapai nilai setinggi mungkin saat pengembangan produk diakhiri. *Product Owner*, *Scrum Master* dan tim *Scrum* memegang komitmen, keberanian, saling menghargai satu sama lain, keterbukaan dan fokus. Ilustrasi tahapan-tahapan metode *Scrum* dapat dilihat pada 2.3 di bawah ini.



Gambar 2.3. Ilustrasi Metode Pengembangan Menggunakan Scrum (Schwaber dan Sutherland, 2011).

2.10. *Black Box Testing*

Salah satu jenis pengujian fungsional adalah *blackbox testing*. *Blackbox testing* merupakan suatu pengujian yang tidak menggunakan pengetahuan tentang struktur interior aplikasi. Saat melakukan pengujian, penguji akan berinteraksi langsung dengan *user interface*, lalu penguji akan memberikan masukan dan memeriksa hasil dari aplikasi yang digunakan tanpa mengetahui bagaimana proses dari hasil tersebut (Xu dkk., 2016). Pada *blackbox testing* mendapatkan pengujian dari deskripsi eksternal perangkat lunak, termasuk spesifikasi, persyaratan, dan desain (Ammann dan Offutt, 2016).

Blackbox testing memiliki beberapa keuntungan, di antaranya penguji tidak perlu mengetahui suatu bahasa pemrograman, pengujian dilakukan dari sudut pandang pengguna agar dapat mengetahui suatu inkonsisten pada spesifikasi kebutuhan serta yang terakhir programmer dan penguji dapat saling bergantung (Jaya, 2018). *Blackbox testing* juga memiliki beberapa kekurangan yaitu suatu

pengujian sulit didesain tanpa spesifikasi yang jelas, memiliki kemungkinan untuk pengulangan *testing* dan tidak ada pengujian pada bagian *back-end* (Jaya, 2018).

2.11. Usability Testing

Usability Testing adalah kegiatan pengujian untuk mengumpulkan data mengenai sebuah produk dalam tahap pengembangan. Tujuan utama dari *usability testing* adalah untuk mengumpulkan data kuantitatif dengan mengukur waktu untuk mengidentifikasi dan memperbaiki kekurangan yang ada dalam produk serta bahan pendukung yang menyertainya sebelum produk dirilis (Rubin dan Chisnell, 2008). Dengan *usability testing*, didapati apa yang sebenarnya dilakukan pengguna, apa yang berhasil untuk mereka, dan apa yang tidak dipikirkan akan mereka lakukan atau bahkan apa yang mereka pikir akan mereka lakukan jika menggunakan produk (Barnum, 2020).

Pengujian ini diharapkan akan mendapatkan kekuatan dan kelemahan dari setiap aspek yang ada pada aplikasi itu sendiri. Maka dari itu, perlu adanya dokumentasi pengalaman aktual para calon pengguna aplikasi atau produk saat dievaluasi (Wesfix, 2017). Tujuan lain dilakukannya pengujian ini adalah untuk mengumpulkan data kualitatif yang berhubungan dengan produk yang diuji. Data kualitatif tersebut terdiri dari komentar yang dibuat oleh partisipan, jawaban dari kuesioner pertanyaan dan tanggapan dari partisipan saat proses wawancara. *Usability testing* telah terbukti dapat mengurangi waktu pada tahap pengembangan, mengurangi jumlah bugs, dan menghasilkan produk yang lebih berkualitas untuk meningkatkan nilai jual (Wahl, 2000).

2.11.1. System Usability Scale (SUS)

Kuesioner standar yang paling sering digunakan untuk penilaian suatu kegunaan merupakan *System Usability Scale* (SUS). Dari awal kuesioner ini relatif tidak menguntungkan, namun dimasa depan kemungkinan besar SUS akan menjadi suatu ukuran yang populer dari persepsi kegunaan. Mendapatkan nilai pada SUS cukup rumit, karena pola pada *item* yang bergantian dan memiliki keputusan awal untuk memanipulasi skor berada di antara 0 sampai 100 (Lewis, 2018). John Brooke mengembangkan SUS pada tahun 1996 yang berisi 10 pertanyaan dasar dan sederhana tentang kegunaan suatu sistem, berikut adalah pertanyaannya (Kaya dkk., 2019):

1. Saya pikir saya ingin menggunakan sistem ini dengan sering.
2. Saya menemukan sistem yang tidak rumit.
3. Saya pikir sistemnya mudah digunakan.

4. Saya pikir saya akan membutuhkan bantuan dari orang teknis untuk dapat menggunakan sistem ini.
5. Saya menemukan berbagai fungsi dalam sistem ini terintegrasi dengan baik.
6. Saya pikir ada terlalu banyak inkonsistensi dalam sistem ini.
7. Saya membayangkan bahwa kebanyakan orang akan belajar menggunakan sistem ini dengan sangat cepat.
8. Menurut saya sistem ini sangat rumit untuk digunakan.
9. Saya merasa sangat percaya diri menggunakan sistem ini.
10. Saya perlu belajar banyak hal sebelum saya bisa menggunakan sistem ini.

Pertanyaan tersebut akan diberikan kepada peserta uji dan menjawab dengan skala antara 1 (Sangat tidak setuju) dan 5 (Sangat setuju). Menurut Brooke hasil dari jawaban akan dievaluasi dalam kisaran 0 sampai 4. Dapat dilihat pada pertanyaan yang memiliki nomor ganjil maka memiliki makna positif dan pertanyaan bernomor genap memiliki makna negatif. Pada pertanyaan positif dilakukan penilaian yang mana skor pengguna dikurangi 1 poin dan pada pertanyaan negatif dilakukan penilaian yang mana skor pengguna dikurangi 5 poin. Setelah itu jumlah skor yang didapat dikalikan 2,5 untuk membuat kisaran antar 0 sampai 100. Skor SUS rata-rata dan skala peringkat kata sifat dihitung untuk setiap aplikasi seluler di Android dan iOS (Kaya dkk., 2019).

BAB III

METODOLOGI PENELITIAN

3.1. Waktu dan Lokasi Penelitian

Penelitian ini akan bertempat pada Gedung A Fakultas Matematika dan Ilmu Pengetahuan Alam. Waktu yang dibutuhkan agar penelitian ini dapat diimplementasikan adalah 3 bulan terhitung dari Juli 2022 hingga September 2022.

3.2. Alat dan Bahan

Alat dan Bahan yang akan digunakan pada penelitian ini terdiri dari beberapa perangkat keras (*Hardware*) dan perangkat lunak (*Software*) yang dijabarkan sebagai berikut:

1. Perangkat Keras

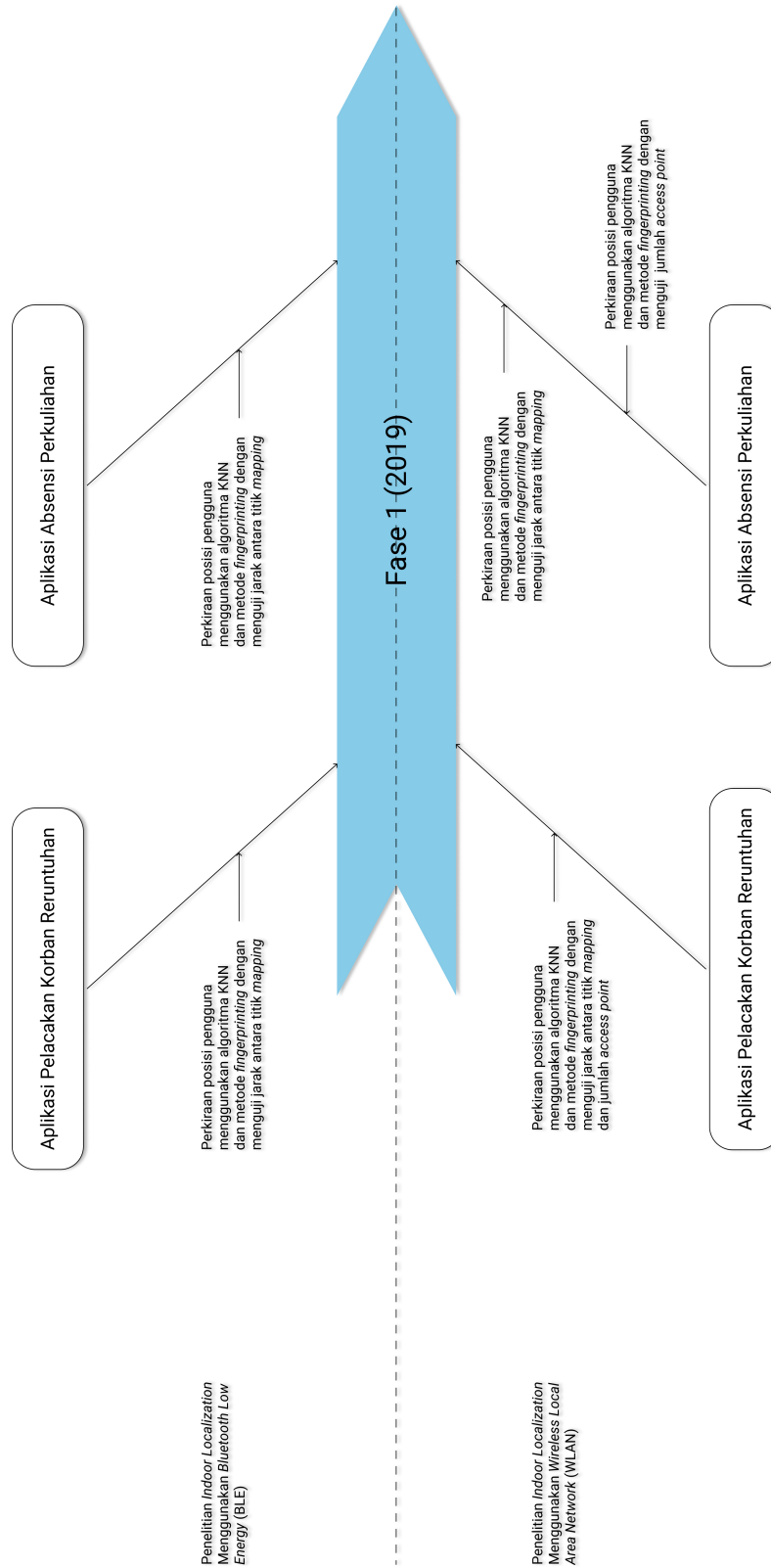
- Laptop Acer Aspire E5-475g dengan RAM 16GB, Intel Core i5-7200U 2.5GHz, Nvidia GeForce 940MX 2GB, *Harddisk* (HDD) 1500Gb, *Solid State Drive* (SSD) 250GB.
- *Smartphone* Xiaomi Poco F3 dengan RAM 6GB, *Internal Storage* 128GB.
- *Personal Computer* dengan AMD Ryzen 7 2700x, RAM 16GB, Nvidia GeForce RTX 2080 8GB, *Solid State Drive* (SSD) 500GB.
- *Beacon Bluetooth*.

2. Perangkat Lunak

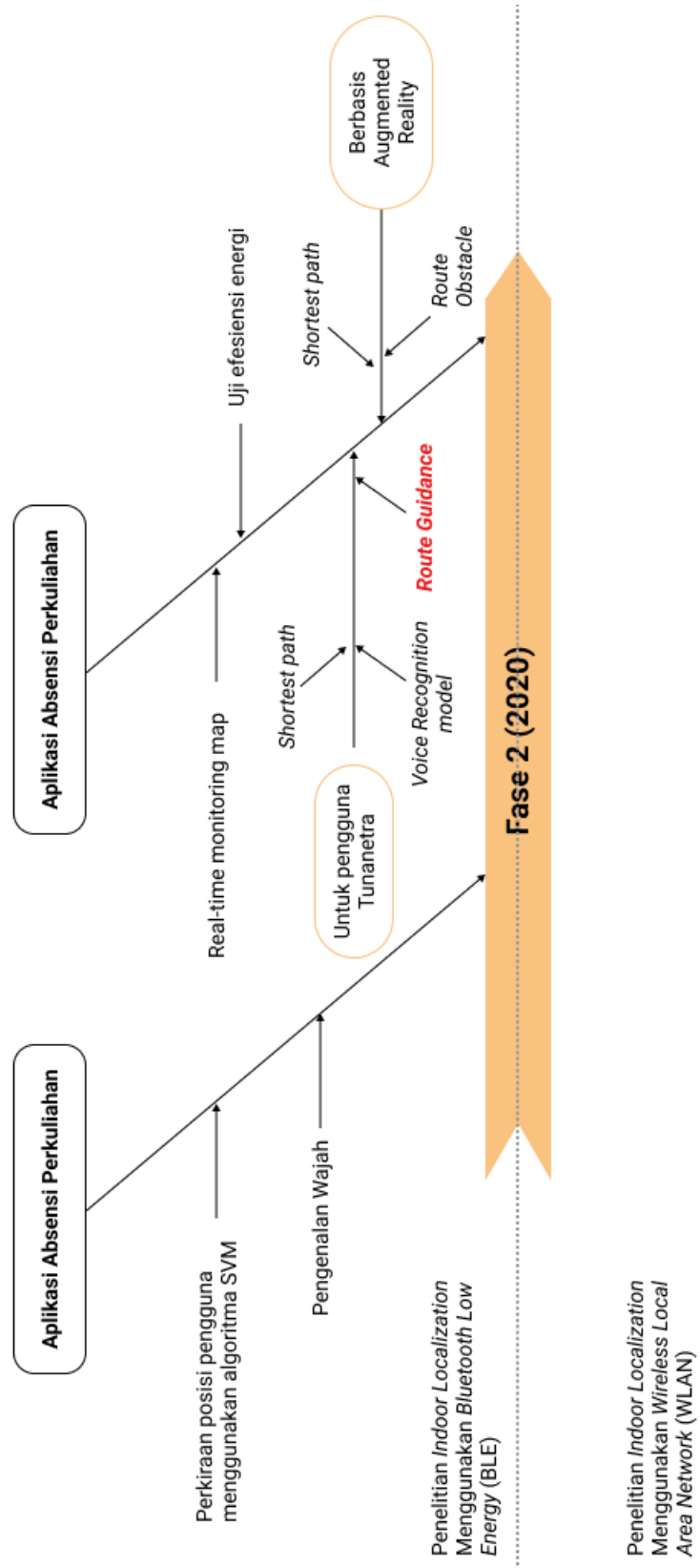
- Windows 11 Pro
- Linux Ubuntu 20.04.1 LTS
- Android Studio 2021.2.1.15
- Visual Studio Code
- Figma
- Notion
- Vosk API

3.3. Roadmap Penelitian

Roadmap penelitian merupakan diagram yang menggambarkan rangkaian beberapa penelitian yang saling berkesinambungan dalam rentang waktu tertentu. *Roadmap* penelitian biasa dibuat untuk memberikan batasan kepada peneliti agar menghindari pengamatan yang tidak perlu dan fokus terhadap bagian penelitiannya saja. Pada penelitian ini dibagi ke dalam 2 fase. Fase pertama pada tahun 2019 memiliki fokus penelitian pada *indoor localization* dengan menggunakan *Bluetooth Low Energy* (BLE) dan menggunakan *Wireless Local Area Network* (WLAN) dapat dilihat gambar 3.1. Pada fase kedua pada tahun 2020 lebih berfokus pada penelitian *Indoor Localization* dengan menggunakan BLE. Penelitian ini terletak pada fase 2 di tahun 2020 dengan topik utama yaitu Aplikasi Navigasi *Indoor* dengan sub topik untuk pengguna Tunanetra. Penelitian ini memiliki batasan berupa pembangunan aplikasi *mobile* untuk *Route Guidance* untuk Tunanetra, seperti yang ditunjukkan pada gambar 3.2 dengan kata yang dicetak tebal berwarna merah sebagai berikut.



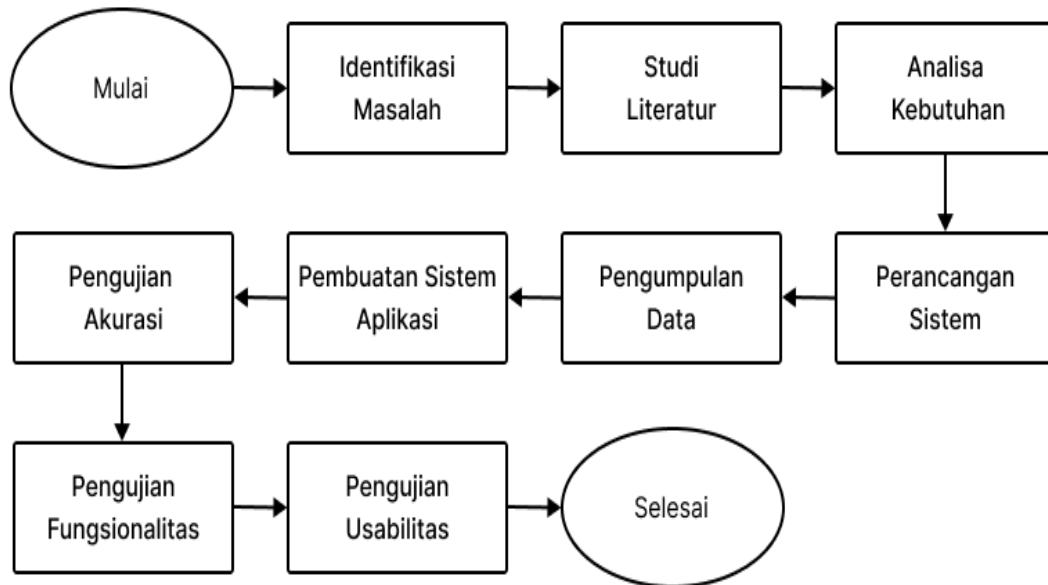
Gambar 3.1. Roadmap Penelitian Fase 1



Gambar 3.2. Roadmap Penelitian Fase 2

3.4. Metode Penelitian

Metode penelitian yang dilakukan dalam penelitian ini ditunjukkan pada Gambar 3.3.



Gambar 3.3. Diagram Alir Penelitian

3.4.1. Identifikasi Masalah

Tahapan ini merupakan tahapan yang dilakukan untuk mengidentifikasi masalah pada lingkungan yang berhubungan dengan aplikasi yang akan dibuat baik secara langsung maupun tidak langsung dan menjadi landasan mengapa aplikasi ini harus dibuat. Masalah-masalah yang berhasil diidentifikasi adalah sebagai berikut:

1. Kebutuhan Fungsional Kebutuhan fungsional mendefinisikan fungsionalitas sistem. Kebutuhan fungsional dari identifikasi masalah yang telah dilakukan adalah sebagai berikut:
 - Melakukan proses pemanduan kepada pengguna secara *background process* dengan bantuan *speech command recognition* dan *route guidance* dengan aplikasi berbasis Android.
 - Menampilkan prediksi lokasi pengguna serta rute terbaik menuju ke lokasi tujuan pengguna.

2. Kebutuhan Non-Fungsional Kebutuhan non-fungsional memastikan batasan eksternal yang harus dipenuhi oleh sistem. Batasan-batasan tersebut antara lain:

- Proses pemanggilan aplikasi dimulai dengan memanggil *hotword*.
- Proses pencatatan dan penentuan lokasi dilakukan secara berkala dalam interval yang sesingkat-singkatnya.
- Hanya dapat melakukan proses pemanduan rute apabila *Bluetooth* pada perangkat hidup dan terhubung dengan *Beacon*.
- Sistem hanya dapat mendeteksi lokasi pengguna di dalam gedung yang telah di petakan terlebih dahulu.

3.4.2. Studi Literatur

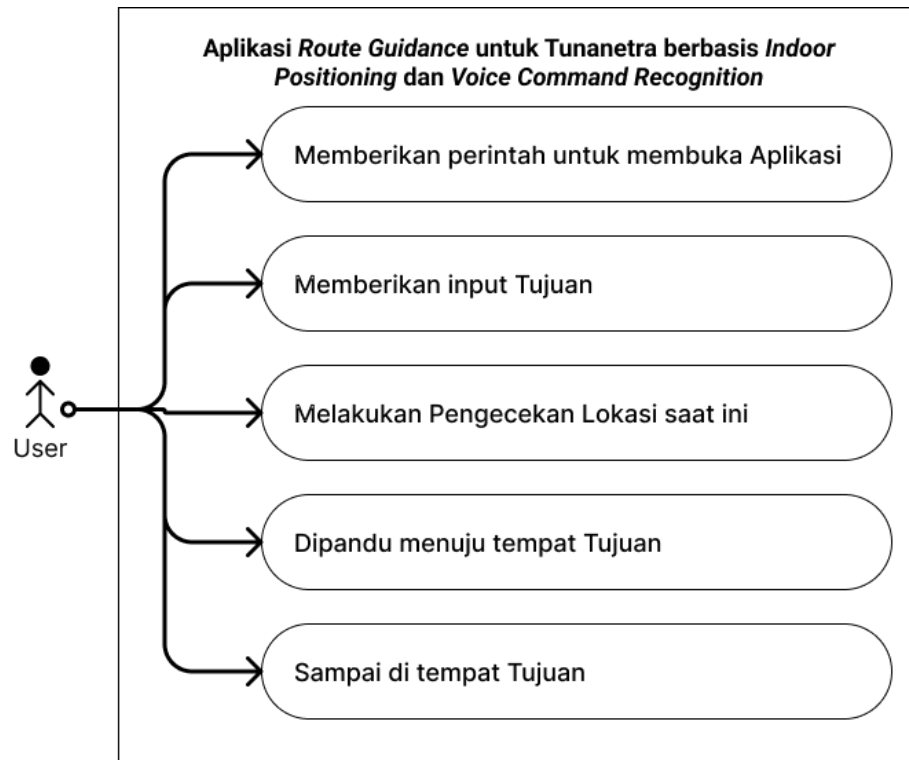
Studi literatur digunakan sebagai bahan referensi selama proses penelitian. Studi literatur dilakukan dengan cara mencari situs *website* dan jurnal-jurnal terkait tentang penelitian, baik jurnal nasional maupun internasional, buku-buku yang telah diterbitkan, serta situs-situs internet yang berkaitan dengan permasalahan yang dikaji dalam penelitian. Studi literatur dapat dikembangkan untuk menyempurnakan kekurangan dari penelitian sebelumnya.

3.4.3. Perancangan Sistem

Tahap perancangan sistem ini meliputi perancangan alur kerja sistem yang berfungsi untuk memastikan sistem yang dibangun dapat digunakan secara baik oleh pengguna. Tahap perancangan sistem ini terbagi menjadi beberapa bagian yaitu diagram Use Case , Diagram Deployment, alur kerja sistem, *Flow Diagram* dan desain prototipe berdasarkan analisa kebutuhan yang telah dijabarkan. Berikut ini merupakan tahapan tersebut:

1. Use Case Diagram

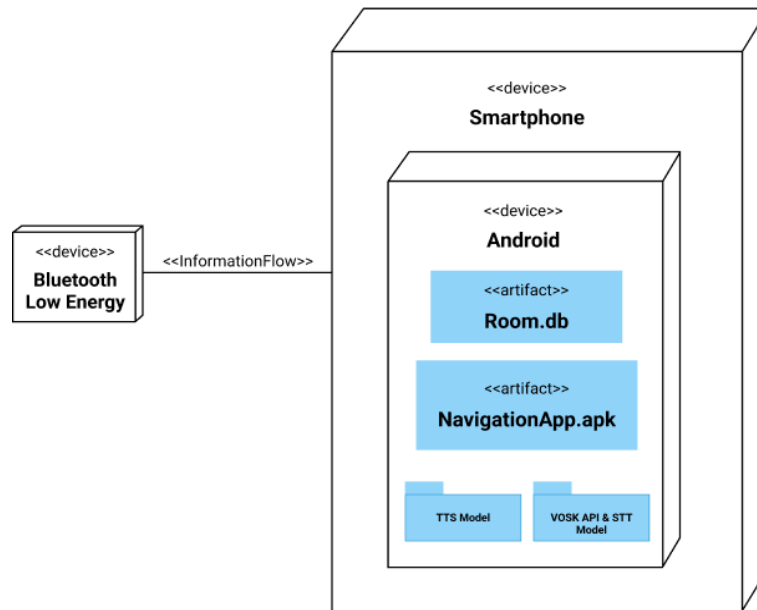
Use Case Diagram merupakan gambaran interaksi antara pengguna dan sistem. Pengguna dari sistem yang akan dibangun ini adalah Tunanetra dan pengunjung gedung FMIPA USK. *Use Case Diagram* dari sistem ini dapat dilihat pada Gambar 3.4



Gambar 3.4. *Use Case Diagram*

2. *Deployment Diagram*

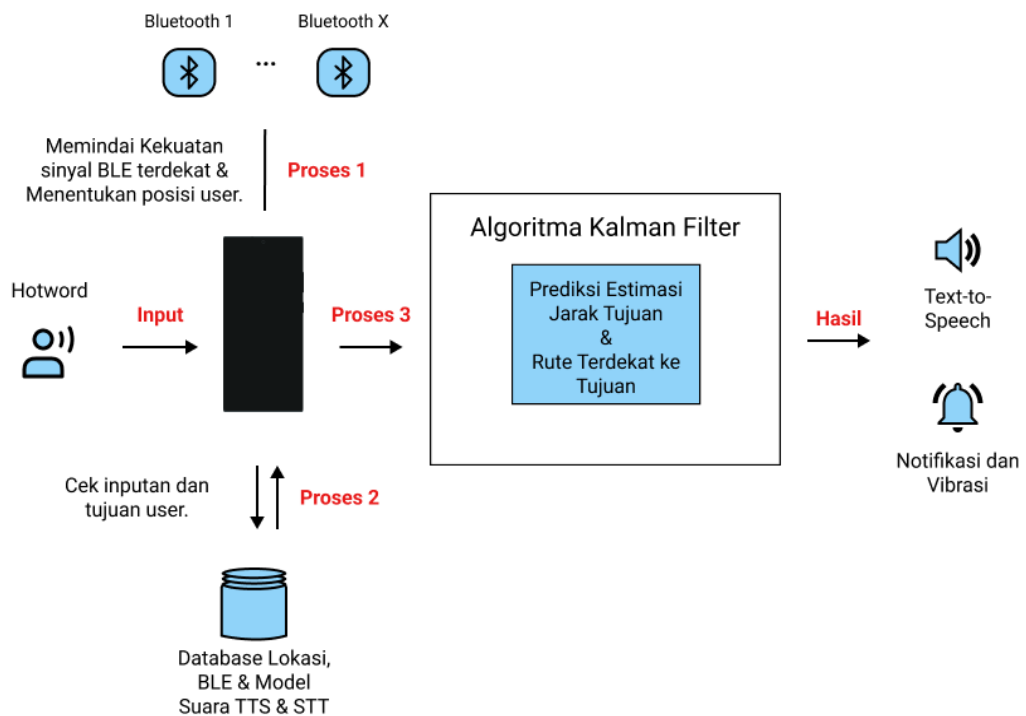
Deployment Diagram merupakan gambaran hubungan antara perangkat lunak dan perangkat keras yang digunakan dalam sebuah sistem. Seperti yang telah disebutkan sebelumnya, penelitian ini merupakan bagian dari penelitian lain yang saling terintegrasi sehingga tidak semua *node* dalam diagram menjadi fokus dalam penelitian ini. Bagian dalam diagram yang menjadi jangkauan dari penelitian ini ditandai dengan warna biru seperti yang ditunjukkan pada Gambar 3.5



Gambar 3.5. *Deployment Diagram*

3. Alur Kerja Sistem

Alur kerja sistem merupakan langkah-langkah yang dilalui sistem hingga fungsionalitas sistem dapat dimanfaatkan oleh pengguna. Alur kerja sistem ini dapat dilihat pada Gambar 3.6

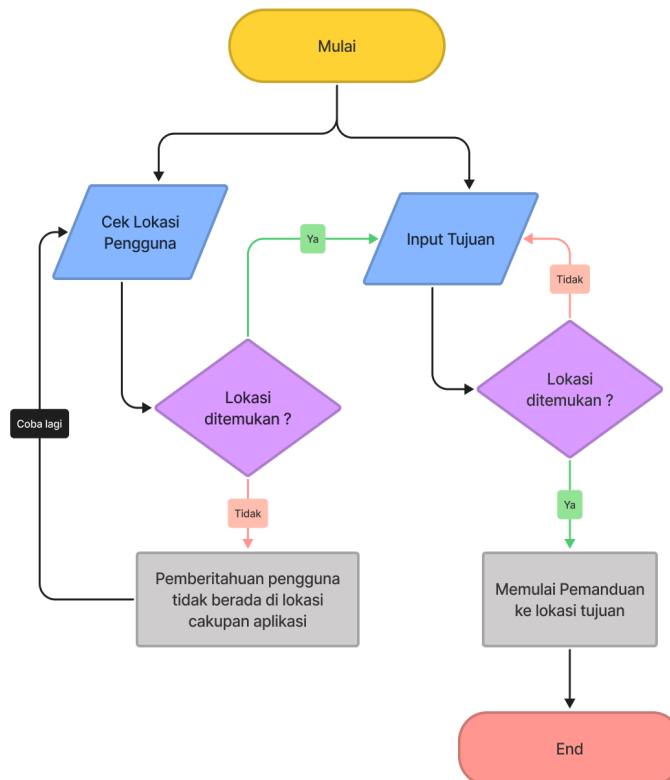


Gambar 3.6. Alur Kerja Sistem

- *Input* diterima dari pengguna setelah mengucapkan *hotword*.
- Pada proses 1 aplikasi dan perangkat akan memindai kekuatan sinyal BLE terdekat serta menentukan posisi pengguna.
- Pada proses 2, setelah posisi pengguna ditentukan, aplikasi akan melakukan pengecekan *input* dari pengguna pada *database* lokasi menggunakan model yang telah tersedia oleh penelitian lainnya.
- Pada proses 3, aplikasi akan memprediksi jarak tujuan serta rute terdekat menuju tujuan pengguna menggunakan algoritma Kalman Filter.
- Hasil yang dihasilkan dari proses-proses sebelumnya berupa notifikasi dan vibrasi serta *text-to-speech* atau berupa ucapan dari rute yang akan di tempuh oleh pengguna, seperti "Belok ke kanan dalam 5 langkah", "Telah sampai di tujuan , Ruangn Jurusan Informatika", dsb.

4. Flow Diagram

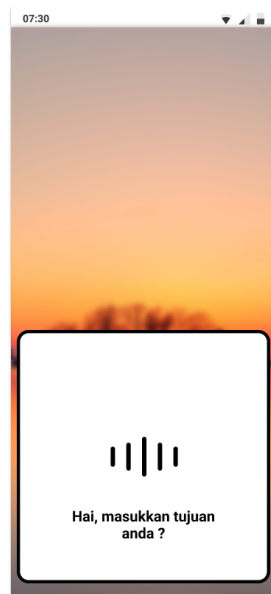
Flow Diagram menunjukkan bagaimana proses tahapan melakukan navigasi pada sistem seperti pada gambar 3.7 berikut ini.



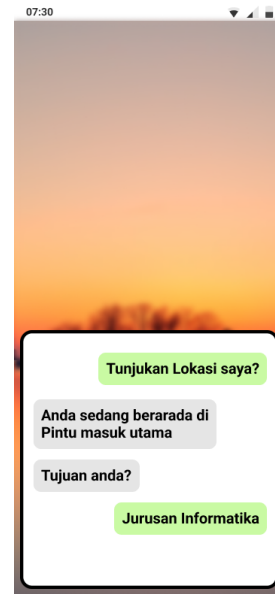
Gambar 3.7. *Flow Diagram Navigasi Indoor*

5. Desain Prototipe

Desain prototipe menunjukkan perkiraan tampilan aplikasi yang akan dibuat. Desain ini dirancang berdasarkan beberapa skenario pada sisi pengguna baik tunanetra dan bukan tunanetra.



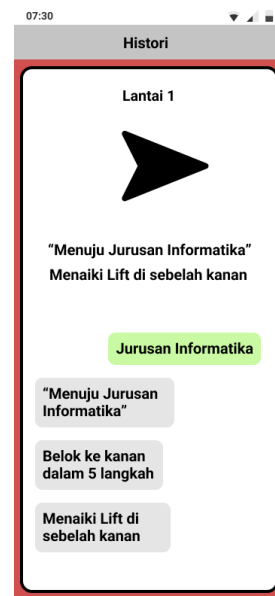
(a) Pop up trigger by hotword



(b) Input tujuan dan lokasi pengguna



(c) Proses Navigasi ke Tujuan



(d) Histori Navigasi

Gambar 3.8. Tampilan Halaman Aplikasi Navigasi Indoor untuk Tunanetra

3.4.4. Pengumpulan Data

1. Data Model *Voice Recognition*

Data yang digunakan dalam penelitian ini terdiri atas data *pre-trained* model yang diperoleh dari penelitian yang berada dalam satu penelitian yang sama pada *roadmap* pada gambar 3.2. *Pre-trained* model dimanfaatkan sebagai titik awal untuk merespons perintah suara dari pengguna, kemudian perintah suara akan di terjemahkan ke dalam bentuk teks yang akan diterima oleh *smartphone* pengguna. Teks yang diterima akan digunakan sebagai input yang akan menjadi lokasi tujuan serta pemilihan rute.

2. Data Rute dan Denah Lokasi Penelitian

Rute yang akan digunakan berlokasi di gedung A FMIPA lantai 1 sampai dengan lantai 3. Data rute yang digunakan akan di petakan dengan menggunakan bantuan BLE. Data / titik-titik BLE disimpan dalam *database* lokal berupa *MAC Address*, nilai *RSSI*, dan nama perangkat BLE. Dengan adanya *RSSI* atau kekuatan sinyal yang disimpan perangkat *smartphone* pengguna akan menangkap sinyal dan menyesuaikan dengan titik-titik yang telah dipetakan sehingga mendapatkan lokasi pengguna berada serta menjadikan titik lokasi pengguna sebagai pemilihan rute terdekat ke lokasi tujuan pengguna.

3. Data *Text-to-Speech*

Data yang akan digunakan menggunakan *text-to-speech* yang disediakan oleh library Google speech. Sehingga proses pemanduan menuju ruangan akan dipandu dengan suara seperti "Menuju Ruangan Jurusan Informatika", "Belok ke kanan dalam 5 langkah", "Telah sampai di tujuan , Ruangan Jurusan Informatika".

4. Data akurasi lokasi pengguna serta akurasi rute

Data yang akan diperoleh dari proses koneksi BLE terdekat dengan *RSSI* sebagai tolak ukur akurasi lokasi pengguna, sedangkan untuk akurasi rute menggunakan data pemetaan rute dan lokasi tujuan yang telah dipetakan pada aplikasi dengan memanfaatkan *RSSI* dari beberapa koneksi BLE terdekat yang saling terhubung menggunakan Kalman Filter.

3.4.5. Pembuatan Sistem Aplikasi

Pada Proses pembuatan sistem aplikasi, metode pengembangan aplikasi yang digunakan adalah metode *Scrum*, dikarenakan sistem ini dikembangkan bersama dengan tim dan membutuhkan fleksibilitas terhadap perubahan yang terjadi dalam proses pengembangannya, sehingga memerlukan iterasi secara berkala agar berjalan dengan baik. Salah satu tahap yang merupakan paling awal dari metode ini adalah *product backlog*. Pada tahap ini, semua kebutuhan akan ditransformasikan menjadi rancangan pekerjaan untuk diselesaikan agar aplikasi dapat bekerja secara fungsional. Adapun *product backlog* dari keseluruhan aplikasi ini adalah sebagai berikut.

- Menampilkan halaman beranda.
- Menampilkan *Pop up* yang muncul dari *input* suara.
- Meluncurkan aplikasi dengan proses pengenalan suara.
- Menampilkan dan mengeluarkan suara daftar ruangan yang bisa di gunakan sebagai tujuan.
- Menampilkan rute terdekat ke ruangan yang di tuju oleh pengguna.
- Mengeluarkan suara navigasi ke ruangan yang di tuju.

3.4.6. Pengujian Akurasi

Pengujian akurasi posisi pengguna dan pemilihan rute akan menggunakan gabungan titik BLE sebagai titik awal dengan mengukur kekuatan RSSI, kemudian menggunakan letak dan posisi BLE yang telah dipetakan sebelumnya untuk melakukan pemilihan rute serta proses navigasi yang akan diprediksi menggunakan algoritma Kalman Filter disertai dengan fase pembaharuan prediksi titik lokasi pengguna ke tujuan.

Berikut perhitungan yang akan digunakan untuk menguji akurasi pengguna serta rute pengguna, rumus menurut (Ihsan dkk., 2018):

- *Time Update (Predict)*

Predict State

$$\hat{x} = \hat{x} \quad (3.1)$$

Predict error covariance

$$p = p + q \quad (3.2)$$

- *Measurements update (Correct)*

Update the estimate via k

$$\hat{x} = \hat{x} + k * (measurement - \hat{x}) \quad (3.3)$$

Kalman gain

$$k = p / (p + r) \quad (3.4)$$

Update the error covariance

$$p = (1 - k) * p \quad (3.5)$$

Keterangan

x: Nilai yang di filter

p: Error estimasi

q: Noise yang diakibatkan dari proses

k: Kalman Gain

r: Noise dari sensor

3.4.7. Pengujian Fungsionalitas dengan Metode *Black Box*

Metode yang digunakan untuk melakukan pengujian fungsionalitas adalah dengan *black box testing*. Pengujian ini dilakukan untuk memeriksa fungsi-fungsi pada sistem yang dibangun pada penelitian ini apakah sudah bekerja sesuai dengan kebutuhan pengguna ataupun tidak. Pengujian ini dilakukan dengan cara menjalankan setiap fungsi yang ada pada sistem dan memastikan agar sistem dapat bekerja dengan semestinya sesuai alur yang telah dirancang. Pengujian ini juga dapat melibatkan target pengguna dan pengembang.

3.4.8. Usability Testing

Usability Testing merupakan pengujian aplikasi untuk melihat apakah pengguna dapat dengan mudah dan nyaman dalam menggunakan aplikasi tersebut serta melihat dan mengevaluasi keberhasilan dari sebuah produk atau jasa. Pengujian ini akan menggunakan metode SUS. SUS terdiri dari 10 pertanyaan dengan menggunakan skala 1-5. Berikut daftar pertanyaan-pertanyaan dengan metode SUS dapat dilihat pada Tabel 3.1.

Tabel 3.1. Daftar Pertanyaan Metode SUS (Kaya dkk., 2019)

R1	Saya pikir saya ingin menggunakan sistem ini dengan sering.
R2	Saya menemukan sistem yang tidak rumit.
R3	Saya pikir sistemnya mudah digunakan.
R4	Saya pikir saya akan membutuhkan bantuan dari orang teknis untuk dapat menggunakan sistem ini.
R5	Saya menemukan berbagai fungsi dalam sistem ini terintegrasi dengan baik.
R6	Saya pikir ada terlalu banyak inkonsistensi dalam sistem ini.
R7	Saya membayangkan bahwa kebanyakan orang akan belajar menggunakan sistem ini dengan sangat cepat.
R8	Menurut saya sistem ini sangat rumit untuk digunakan.
R9	Saya merasa sangat percaya diri menggunakan sistem ini.
R10	Saya perlu belajar banyak hal sebelum saya bisa menggunakan sistem ini.

Rumus untuk menghitung skor akhir metode SUS (Brooke dkk., 1996), dapat dilihat dari persamaan berikut.

$$skorSUS = \left(\sum_9^i (Ri - 1) + \sum_{10}^j (5 - Rj) \right) * 2,5 \quad (3.6)$$

Keterangan:

R = daftar pertanyaan pada metode SUS.

i = angka ganjil 1, 3, 5, 7, dan 9.

j = angka genap 2, 4, 6, 8, dan 10.

Tingkat nilai skala pengujian menentukan apakah sistem tersebut layak digunakan, bermanfaat, diterima oleh pengguna dan bertahan lama penggunaannya. Sebuah sistem dengan nilai pengujian yang tinggi membuat sistem tersebut menjadi populer dalam waktu yang lama dan penggunaannya yang luas, dikarenakan banyak individu akan merasakan manfaat dari kehadiran sistem tersebut. Sedangkan sistem

dengan nilai pengujian yang rendah, sering kali diabaikan walaupun dibuat berdasarkan kebutuhan dan menghasilkan sumber daya yang banyak. Berdasarkan dari skor akhir SUS, dapat diketahui seberapa tinggi tingkat usability rancangan sistem yang dikembangkan. Tabel 3.2 menunjukkan nilai interpretasi yang digunakan.

Tabel 3.2. Interpretasi skor sus

Skor SUS	Interpretasi
<50	Tidak Dapat Diterima
50-70	Marginal
>70	Dapat Diterima

DAFTAR KEPUSTAKAAN

- Ammann, P. dan Offutt, J. (2016). *Introduction to software testing*. Cambridge University Press.
- Barnum, C. M. (2020). *Usability testing essentials: ready, set... test!* Morgan Kaufmann.
- Brena, R. F., García-Vázquez, J. P., Galván-Tejada, C. E., Muñoz-Rodriguez, D., Vargas-Rosales, C., dan Fangmeyer, J. (2017). Evolution of indoor positioning technologies: A survey. *Journal of Sensors*, 2017.
- Brooke, J. dkk. (1996). Sus-a quick and dirty usability scale. *Usability evaluation in industry*, 189(194):4–7.
- Cantón Paterna, V., Calveras Auge, A., Paradells Aspas, J., dan Perez Bullones, M. A. (2017). A bluetooth low energy indoor positioning system with channel diversity, weighted trilateration and kalman filtering. *Sensors*, 17(12):2927.
- Cephei, A. (2019). Introduction: Vosk.
- Craig, C. dan Gerber, A. (2015). *Learn Android Studio: Build Android Apps Quickly and Effectively*. Apress.
- Deemer, P., Benefield, G., Larman, C., dan Vodde, B. (2012). A lightweight guide to the theory and practice of scrum. *Ver*, 2:2012.
- Ihsan, M., Suwastika, N. A., dan Putrada, A. G. (2018). Analisis implementasi kalman filter untuk peningkatan akurasi pendeteksian pada sistem monitoring area terlarang sepeda motor berbasis m2m. *eProceedings of Engineering*, 5(1).
- Jaya, T. S. (2018). Pengujian aplikasi dengan metode blackbox testing boundary value analysis (studi kasus: kantor digital politeknik negeri lampung). *Jurnal Informatika: Jurnal Pengembangan IT*, 3(1):45–48.
- Jurafsky, D. dan Martin, J. (2008). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, volume 2.
- Kaluža, M., Beg, K., dan Vukelić, B. (2017). Analysis of an indoor positioning systems. *Zbornik Veleučilišta u Rijeci*, 5(1):13–32.
- Karimi, H. A. (2015). *Indoor wayfinding and navigation*. CRC Press.
- Kaya, A., Ozturk, R., dan Gumussoy, C. A. (2019). Usability measurement of mobile applications with system usability scale (sus). In *Industrial engineering in the big data era*, pages 389–400. Springer.
- Ko, E. dan Kim, E. Y. (2017). A vision-based wayfinding system for visually impaired people using situation awareness and activity-based instructions. *Sensors*, 17(8):1882.

- Lewis, J. R. (2018). The system usability scale: past, present, and future. *International Journal of Human-Computer Interaction*, 34(7):577–590.
- Li, G., Geng, E., Ye, Z., Xu, Y., Lin, J., dan Pang, Y. (2018). Indoor positioning algorithm based on the improved rssi distance model. *Sensors*, 18(9):2820.
- Li, Q., Li, R., Ji, K., dan Dai, W. (2015). Kalman filter and its application. In *2015 8th International Conference on Intelligent Networks and Intelligent Systems (ICINIS)*, pages 74–77. IEEE.
- Lin, Y.-W. dan Lin, C.-Y. (2018). An interactive real-time locating system based on bluetooth low-energy beacon network. *Sensors*, 18(5):1637.
- Nasution, A. dkk. (2018). Perancangan aplikasi push notification berbasis android. *JURTEKSI (Jurnal Teknologi dan Sistem Informasi)*, 4(2):149–154.
- Puspitasari, R. (2020). Rancang Bangun Aplikasi Kehadiran Perkuliahan Berbasis Teknologi Indoor Positioning System Menggunakan Bluetooth Low Energy dan Metode Klasifikasi K-NN. *ETD Unsyiah*.
- Rubin, J. dan Chisnell, D. (2008). *Handbook of usability testing: how to plan, design and conduct effective tests*. John Wiley & Sons.
- Schwaber, K. dan Sutherland, J. (2011). The scrum guide. *Scrum Alliance*, 21(19):1.
- Shaheen, J. A., Asghar, M. A., dan Hussain, A. (2017). Android os with its architecture and android application with dalvik virtual machine review. *International Journal of Multimedia and Ubiquitous Engineering*, 12(7):19–30.
- Sideeq, M. M. dan Qasem, N. (2016). Smart wall based on active frequency selective wallpaper. *ZANCO Journal of Pure and Applied Sciences*, 28(2).
- Subhan, F., Hasbullah, H., Rozyyev, A., dan Bakhsh, S. T. (2011). Indoor positioning in bluetooth networks using fingerprinting and lateration approach. In *2011 International Conference on Information Science and Applications*, pages 1–9. IEEE.
- Virrantaus, K., Markkula, J., Garmash, A., Terziyan, V., Veijalainen, J., Katanosov, A., dan Tirri, H. (2001). Developing gis-supported location-based services. In *Proceedings of the Second International Conference on Web Information Systems Engineering*, volume 2, pages 66–75. IEEE.
- Wahl, N. J. (2000). Student-run usability testing. In *Thirteenth Conference on Software Engineering Education and Training*, pages 123–131. IEEE.
- Welch, G. F. (2020). Kalman filter. *Computer Vision: A Reference Guide*, pages 1–3.
- Wesfix, T. (2017). *Branding itu Dipraktikin*. Gramedia Widiasarana Indonesia.
- WHO (accessed Oktober 23, 2020). World report on vision. <https://www.who.int/publications-detail-redirect/world-report-on-vision>.

- Xu, S., Chen, L., Wang, C., dan Rud, O. (2016). A comparative study on black-box testing with open source applications. In *2016 17th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*, pages 527–532. IEEE.
- Yudha, D. P., Hasbi, B. I., dan Sukarna, R. H. (2018). Indoor positioning system berdasarkan fingerprinting received signal strength (rss) wifi dengan algoritma k-nearest neighbor (k-nn). *ILKOM Jurnal Ilmiah*, 10(3):274–283.
- Zafari, F., Gkelias, A., dan Leung, K. K. (2019). A survey of indoor localization systems and technologies. *IEEE Communications Surveys & Tutorials*, 21(3):2568–2599.