

## Implementasi Sistem Penentuan Lokasi Dalam Gedung (Indoor Localization) Menggunakan Metode Fingerprinting Berjenis Monitor Based Localization (MBL) Dengan Teknologi Bluetooth Low Energy (BLE)

Muhammad Hasbi Ash Shiddieqy<sup>1</sup>, Adhitya Bhawiyuga<sup>2</sup>, Kasyful Amron<sup>3</sup>

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya  
Email: <sup>1</sup>hasbi12.muhammad@gmail.com, <sup>2</sup>bhawiyuga@ub.ac.id, <sup>3</sup>kasyful@ub.ac.id

### Abstrak

Selain terkenal karena akurasi dalam menentukan lokasi di luar ruangan, teknologi GPS juga memiliki kelemahan dalam menentukan lokasi di dalam ruangan karena adanya pelemahan sinyal. Oleh karena itu, untuk menentukan lokasi di dalam ruangan, salah satu metode yang dapat digunakan adalah metode *fingerprinting*. Metode *fingerprinting* adalah metode untuk menentukan lokasi dengan mengklasifikasikan pola sinyal yang diperoleh menjadi nama lokasi berdasarkan karakteristik sinyal pada suatu lokasi. Salah satu teknologi yang dapat dimanfaatkan untuk menghasilkan sinyal dalam metode *fingerprinting* adalah teknologi *Bluetooth Low Energy* (BLE). Teknologi BLE digunakan karena memiliki beberapa keunggulan seperti durasi *advertisement packet* yang cepat, ukuran perangkat yang kecil, dapat menggunakan baterai yang dapat bertahan lama, dsb. Pada penelitian ini, metode *fingerprinting* dan teknologi BLE menggunakan konsep *Monitor Based Localization* (MBL). Konsep MBL adalah konsep dimana perangkat yang akan ditentukan lokasinya hanya memancarkan sinyal secara pasif, sementara perangkat lain yang akan melakukan penentuan lokasi. Akurasi kesalahan sistem dalam penentuan lokasi ini mencapai 15,24% yang artinya memiliki keakuratan dalam penentuan lokasi sebesar 84,76%. Kesalahan dalam penentuan lokasi dapat disebabkan karena adanya kemiripan pola sinyal pada titik tertentu pada 2 lokasi yang berbeda.

**Kata kunci:** *Indoor Localization, BLE, Fingerprinting, Monitor Based Localization, GPS*

### Abstract

Besides being famous for its accuracy in determining outdoor location, GPS technology also has weaknesses in determining indoor location due to signal attenuation. Therefore, to determine the indoor location, one of the methods that can be used is the fingerprinting method. The fingerprinting method is a method for determining location by classifying the obtained signal patterns into location names based on the signal characteristics at a location. . One technology that can be utilized to produce signals in the fingerprinting method is Bluetooth Low Energy (BLE) technology. BLE technology is used because it has several advantages such as fast advertisement packet duration, small device size, can use long-lasting batteries, etc. In this research, the fingerprinting method and BLE technology use the concept of Monitor Based Localization (MBL). The MBL concept is a concept where the device to be located only emits a passive signal, while other devices will determine the location. The accuracy of the system error in determining location reaches 15.24% which means it has accuracy in determining location of 84.76%. The errors in determining location can be caused by the similarity of signal patterns at certain points at 2 different locations.

**Keywords:** *Indoor Localization, BLE, Fingerprinting, Monitor Based Localization, GPS*

## 1. PENDAHULUAN

Localization adalah suatu mekanisme untuk menemukan dan menentukan lokasi dari suatu objek. Secara umum, penentuan posisi atau lokasi dibagi menjadi dua, yakni outdoor dan indoor localization. Penentuan lokasi di luar ruangan atau outdoor menggunakan teknologi Global Positioning System (GPS). Sedangkan penentuan lokasi di dalam ruangan atau indoor tidak bisa menggunakan GPS karena pelemahan sinyal GPS saat memasuki gedung sehingga proses penentuan lokasi menjadi tidak memungkinkan. (Brena et al., 2017). Oleh karena itu, dalam penentuan lokasi dalam ruangan atau gedung digunakan teknologi nirkabel seperti WiFi dan Bluetooth. (Zafari, Gkelias and Leung, 2019).

Indoor Localization saat ini dapat menggunakan 2 metode, yakni metode propagasi sinyal dan fingerprinting. Metode propagasi sinyal memperoleh koordinat dari objek berdasarkan jarak antara target dan perangkat yang berfungsi sebagai *reference points*. Kelemahan metode ini adalah rentan terhadap lingkungan yang kompleks (tembok, pintu, jendela, orang berjalan), sehingga menyebabkan gangguan sinyal yang mempengaruhi koordinat yang diperoleh. Akibatnya, koordinat yang diperoleh tidak sesuai dengan koordinat sebenarnya. Salah satu metode untuk mengatasi kelemahan metode propagasi sinyal pada struktur lingkungan yang kompleks adalah metode fingerprinting. Metode fingerprinting adalah teknik penentuan lokasi dengan mengklasifikasikan pola sinyal yang diperoleh berdasarkan *data training* yang telah didapat sebelumnya. (Subedi dan Pyun, 2017). Metode fingerprinting dibagi menjadi 2 tahap, yakni tahap *offline* dan tahap *online*. Tahap *offline* adalah tahap pembentukan *data training*, yakni data yang terdiri dari kumpulan pola sinyal yang diberi label sesuai dengan nama lokasi dimana pola sinyal tersebut diperoleh. *Data training* ini nantinya akan dipakai untuk melatih model yang digunakan untuk mengklasifikasikan pola sinyal. Tahap *online* adalah tahap penentuan lokasi dengan cara mengklasifikasikan pola

sinyal yang diperoleh berdasarkan model yang telah dilatih dengan *data training* sebelumnya sehingga memperoleh hasil klasifikasi berupa nama lokasi. (Subedi dan Pyun, 2017). Keuntungan dari metode fingerprinting adalah dapat menghindari efek samping dari gangguan sinyal yang dapat mengakibatkan kesalahan klasifikasi. Keuntungan lainnya adalah tidak perlu mengetahui posisi dari *reference points* sehingga praktis untuk digunakan. (Jiang et al., 2015)

Saat ini, teknologi yang paling banyak digunakan dengan metode fingerprinting adalah WiFi dan Bluetooth Low Energy (BLE). Penggunaan WiFi dengan metode fingerprinting menghasilkan akurasi penentuan lokasi mencapai 85,9%. (Jiang et al., 2015). Sayangnya, durasi advertisement SSID dari Wifi yang lambat, yakni sekitar 100ms tiap access point menyebabkan satu periode scanning membutuhkan waktu beberapa detik. Hal ini tentunya dapat mengakibatkan kesalahan dalam penentuan lokasi saat perangkat bergerak. (Faragher dan Harle, 2015). Teknologi lain yang dapat digunakan untuk Indoor Localization adalah Bluetooth Low Energy (BLE). Menurut Faragher dan Harle (2015), advertisement packet dari BLE memiliki durasi lebih cepat dari WiFi, sehingga BLE dapat digunakan sebagai pengganti alternatif dari WiFi. Keunggulan lain dari BLE adalah ukuran perangkat yang kecil dibandingkan dengan perangkat WiFi, sumber energi menggunakan baterai yang dapat bertahan berbulan-bulan bahkan bertahun-tahun, harganya murah, didesain untuk komunikasi "machine-to-machine" dan mudah untuk digunakan.

Penggunaan teknologi BLE dan metode fingerprinting saat ini menggunakan konsep Device Based Localization (DBL), yakni sebuah konsep dimana perangkat yang dilacak atau dicari lokasinya secara aktif melakukan scanning terhadap *reference points* sekaligus melakukan proses klasifikasi pola sinyal menjadi nama lokasi. (Zafari, Gkelias and Leung, 2019). Konsep ini tidak dapat digunakan pada kondisi dimana objek yang dilacak tidak dibolehkan atau tidak memungkinkan membawa perangkat yang

dapat melakukan pemindaian aktif (contoh perangkat: smartphone) seperti pada rumah sakit jiwa, penjara, dsb. Untuk itulah diperlukan konsep lain yang dapat mengatasi permasalahan tersebut.

Berdasarkan permasalahan di atas, pada penelitian ini penulis menawarkan solusi untuk mengatasi permasalahan pada konsep DBL, yakni dengan mengimplementasikan indoor localization menggunakan metode fingerprinting dengan teknologi Bluetooth Low Energy berbasis Monitor Based Localization (MBL). Konsep MBL adalah konsep dimana perangkat yang dilacak hanya memancarkan sinyal secara simultan, sedangkan komponen lain yang akan melakukan proses penentuan lokasi. Konsep MBL bekerja dengan menggunakan setidaknya 3 komponen. Komponen pertama sebagai perangkat yang dilacak, penulis menggunakan *passive tag* yang akan memancarkan sinyal bluetooth secara terus menerus. Komponen yang kedua adalah perangkat yang berperan sebagai *reference points* yang akan menangkap sinyal bluetooth yang dipancarkan oleh *passive tag* kemudian mengirimkan data RSSI-nya menuju server. Komponen ketiga adalah server yang akan menerima data RSSI dari *reference points* untuk diklasifikasikan menjadi lokasi ruangan berdasarkan *data training* yang telah dikumpulkan sebelumnya.

## 2. LANDASAN KEPUSTAKAAN

### 2.1 Indoor Localization

*Indoor Localization* adalah sebuah sistem atau layanan untuk penentuan lokasi seseorang atau benda menggunakan sebuah koordinat relatif pada sebuah ruangan atau gedung.(Chan dan Sohn, 2012). Secara umum, komponen dari sistem *indoor localization* terdiri dari beberapa *anchor node* yang dapat berfungsi sebagai pemancar sinyal atau dapat juga sebagai penangkap sinyal dari perangkat yang dilacak. Perangkat yang dilacak berfungsi sebagai perangkat yang akan dicari lokasi/posisi-nya sekaligus juga sebagai perangkat yang menginterpretasikan pola sinyal yang didapat menjadi nama lokasi atau koordinat. Pada beberapa kasus, terdapat komponen tambahan sebagai perangkat yang menginterpretasikan pola sinyal menjadi nama lokasi atau posisi.

### 2.2 Fingerprinting

*Fingerprinting* adalah sebuah metode untuk menginterpretasikan sinyal yang didapat dari beberapa *reference points* menjadi sebuah nama lokasi. Metode ini bekerja dengan mencocokkan pengukuran sinyal yang diperoleh dari beberapa *reference points* dengan data sinyal yang terdapat pada database. (Jiang *et al.*, 2015).

Secara umum, *Fingerprinting* bekerja dalam 2 tahap, yakni tahap *offline* dan tahap *online*. Tahap *offline* adalah tahap pengumpulan data pola sinyal. Pada tahap ini karakteristik sinyal pada masing-masing ruangan diukur dan disimpan pada database bersamaan dengan nama ruangan yang diukur. Data pola sinyal dan nama ruangan yang disimpan disebut dengan *reference points* (RPs). Tahap kedua adalah tahap *online*, yakni tahap pelacakan atau tahap klasifikasi. Pada tahap ini, data sinyal yang diperoleh dari *reference points* dicocokkan dengan *Reference points* (RPs) dengan algoritma tertentu. *Reference points* yang memiliki kecocokan dengan pengukuran pada tahap *online* adalah hasil dari pelacakan. (Zafari, Gkelias dan Leung, 2017).

### 2.3 Bluetooth Low Energy (BLE)

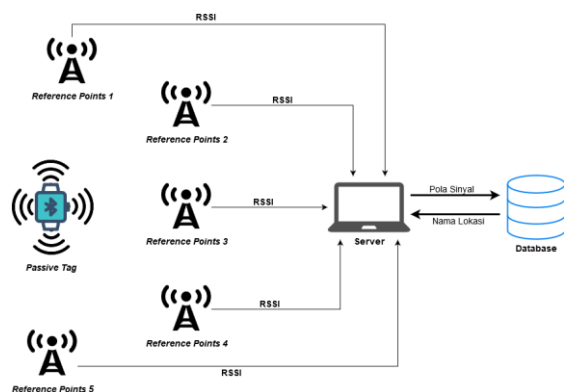
Bluetooth adalah teknologi komunikasi nirkabel yang menggunakan informasi yang tertanam secara digital pada sinyal frekuensi radio. Awalnya dimaksudkan untuk pertukaran data dalam jarak pendek, hal ini didefinisikan oleh standar IEEE 802.15.1. Tujuan utama dari teknologi ini adalah untuk memfasilitasi komunikasi antara perangkat bergerak dan perangkat diam atau dua perangkat seluler, untuk menghilangkan kabel dan konektor antar perangkat (misal, dalam penggunaan headphone nirkabel), dan untuk memfasilitasi sinkronisasi data antara perangkat pribadi. (Chatschik, 2001)

Teknologi Bluetooth telah dipertimbangkan untuk sistem posisi dalam ruangan sebagai pesaing Wi-Fi, khususnya sejak ditemukan Bluetooth Low Energy (BLE), karena availability-nya (didukung oleh sebagian besar smartphone modern), biaya rendah, dan mengkonsumsi daya yang sangat rendah, yang memungkinkan alat tetap bekerja menggunakan baterai selama beberapa bulan atau bahkan bertahun-tahun. (Faragher dan Harle, 2015).

### 3. PERANCANGAN

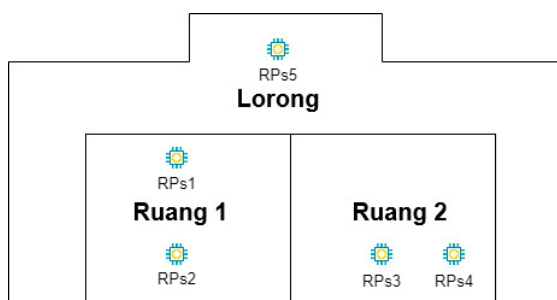
#### 3.1 Perancangan Sistem

Perancangan sistem yang dilakukan meliputi arsitektur sistem dan alur sistem yang akan diimplementasikan. Arsitektur sistem dirancang berdasarkan pada komponen yang akan digunakan dan alur kerja dari sistem yang akan diimplementasikan. Pada Gambar 3.1 di bawah ini menjelaskan mengenai arsitektur sistem yang akan diimplementasikan.



Gambar 3.1 Arsitektur Sistem

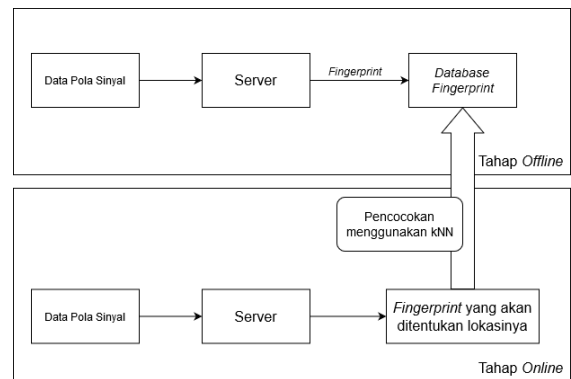
Pada Gambar 3.1 di atas menjelaskan bahwa pada sistem *indoor localization* ini, peneliti menggunakan 3 komponen, yakni *passive tag*, *Reference points*, dan *Server*. *Passive tag* berfungsi untuk memancarkan sinyal bluetooth secara terus menerus, kemudian *Reference points* akan menangkap sinyal bluetooth tersebut dan mengirimkannya menuju server. Kemudian, server mengklasifikasikan pola sinyal bluetooth yang dikirimkan oleh beberapa *Reference points* menjadi nama lokasi.



Gambar 3.2 Denah Implementasi

Lokasi untuk implementasi dibagi menjadi 3. Lokasi pertama dan kedua berupa ruangan, sedangkan lokasi ketiga adalah lorong. Lokasi implementasi berada pada Gedung F lantai 9, Fakultas Ilmu Komputer, Universitas Brawijaya. *Reference points* akan ditempatkan pada ketiga lokasi tersebut. Pada Gambar 3.2 diatas, 2

*reference points* diletakkan pada Ruang 1 dan Ruang 2, sedangkan pada Lorong ditempatkan 1 buah *reference points*.



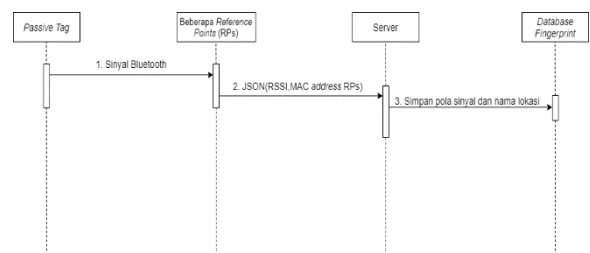
Gambar 3.3 Arsitektur *Fingerprinting*

Untuk alur sistem akan dibagi 2 tahap berdasarkan pada mekanisme kerja metode fingerprinting. Seperti yang disajikan pada Gambar 3.3, tahap pertama disebut sebagai tahap *offline* adalah tahapan untuk membuat *data training*. Tahap ini berfungsi untuk merekam pola sinyal dari sinyal bluetooth yang dipancarkan *passive tag* pada masing-masing ruang. Pola sinyal yang diterima disimpan pada server dan diberi label nama ruangan dimana pola sinyal tersebut diukur.

Tahap kedua adalah tahap *online*, dimana pada tahap ini *passive tag* akan ditentukan lokasinya. Penentuan lokasi dari *passive tag* tersebut menggunakan algoritma KNN berdasarkan pada *data training* yang telah dibuat. Hasil dari penentuan lokasi berupa nama ruangan dimana *passive tag* tersebut berada. Sebagai tambahan, penentuan lokasi dilakukan secara *real-time* dan hasil dari penentuan lokasi akan dicatat pada sebuah file log.

#### 3.2 Perancangan Tahap Offline

##### a. Skema Tahap Offline



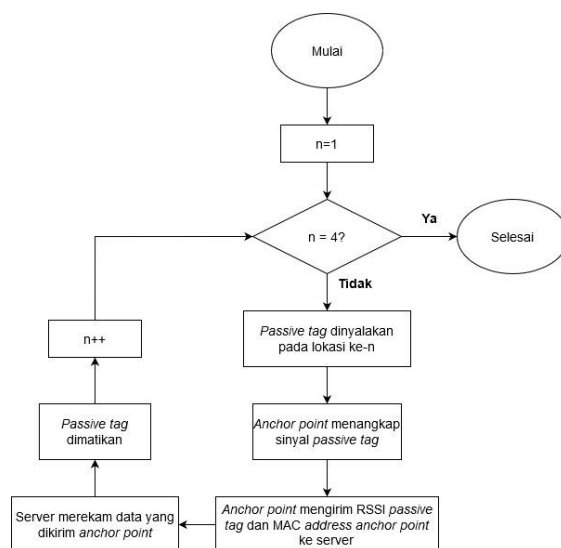
Gambar 3.4 Skema Tahap *Offline*

Pada Gambar 3.4 di atas merupakan skema implementasi tahap *offline*. Tahap *offline* dimulai dengan *Passive tag* yang sudah menyala dan



akan memancarkan sinyal bluetooth. Beberapa *reference points* akan menangkap sinyal bluetooth yang dipancarkan *passive tag*, kemudian akan diukur RSSI dari *passive tag* berdasarkan kekuatan sinyal *passive tag* terhadap *reference points*. Selanjutnya, *reference points* akan mengirimkan data RSSI *passive tag* dan juga MAC address dari *reference points* menuju server. Untuk bisa mengirim 2 tipe data yang berbeda secara bersamaan, digunakan tipe data JSON. Data JSON yang dikirim oleh beberapa *reference points* akan diterima oleh server. Oleh server data JSON dari beberapa *reference points* akan dikumpulkan. Data RSSI pada data JSON tadi akan diurutkan berdasarkan urutan *reference points* kemudian disimpan dengan memberi label nama lokasi dan waktu ke database.

### b. Alur Implementasi Tahap Offline



Gambar 3.5 Alur Tahap Offline

Di bawah ini merupakan penjelasan dari Gambar 3.5 tentang alur tahap *offline* di atas :

1. Pertama, pada lokasi ke-1 *passive tag* dinyalakan. Pada tahap ini *passive tag* akan dinyalakan selama  $\pm 3$  menit
2. Kemudian, *reference points* menangkap sinyal bluetooth yang dipancarkan oleh *passive tag*
3. Selanjutnya, *reference points* akan mengirimkan data RSSI dari *passive tag* dan MAC address *reference points* menuju server dalam format data JSON
4. Server menerima data JSON yang dikirim *reference points* dan menuliskan RSSI *passive tag* pada file csv sesuai urutan dari *reference points* (format urutan pada

Tabel 3.1). Label nama lokasi dan waktu juga dituliskan bersamaan dengan penulisan RSSI.

5. Langkah terakhir, *passive tag* dimatikan sebelum pindah ke lokasi berikutnya. Hal ini untuk menghindari kesalahan penulisan RSSI pada saat berpindah lokasi
6. Langkah 1-5 diulang sampai ketiga ruangan selesai direkam pola sinyal-nya.

### c. Format Data training

Pada bagian ini, akan dijelaskan format dari *data training* yang akan dibuat. *Data training* yang dibuat terdiri dari :

1. Label nama lokasi
2. Label waktu pengambilan pola sinyal dalam format HH:MM:SS
3. RSSI dari *passive tag* terhadap *reference points* (RPs)

Tabel 3.1 dibawah ini merupakan format dari *data training* yang akan dibuat.

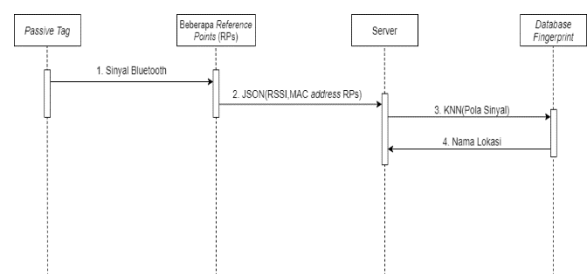
Tabel 3.1 Format Data training

Nama Ruang	Waktu	RSSI RPs 1	RSSI RPs 2	RSSI RPs 3	RSSI RPs 4	RSSI RPs 5
Ruang 1						
.....						
Ruang 2						
.....						
Lorong						

*Data training* ini nantinya akan disimpan dalam sebuah file dengan ekstensi (.csv) pada server. Format csv digunakan untuk memudahkan dalam menggunakan *data training* pada saat proses penentuan lokasi.

## 3.3 Perancangan Tahap Online

### a. Skema Tahap Online

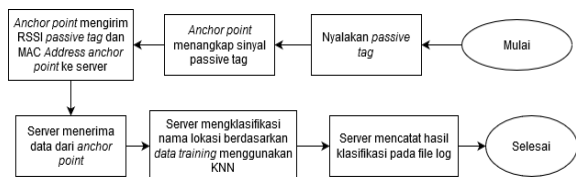


Gambar 3.6 Skema Tahap Online

Pada Gambar 3.6 di atas merupakan skema dari implementasi tahap *online*. Pertama, *passive tag* akan dinyalakan pada salah satu lokasi. Untuk memudahkan, lokasi pertama yang digunakan adalah Ruang 1. Kemudian *reference*

*points* akan menangkap sinyal yang dipancarkan oleh *passive tag* dan mengirimkan data RSSI beserta MAC address dari *reference points* menuju server. Oleh server, data yang diterima dari beberapa *reference points* akan ditentukan lokasi nya berdasarkan *data training* menggunakan algoritma KNN. Nama lokasi yang telah ditentukan akan ditampilkan oleh server.

#### b. Alur Tahap Online



Gambar 3.7 Alur Tahap Online

Pada Gambar 3.7 di atas adalah alur dari implementasi tahap *online*. Berikut adalah penjelasan dari Gambar 3.7 di atas :

1. Pertama, *passive tag* dinyalakan pada salah satu lokasi
2. Saat *passive tag* dinyalakan, *reference points* akan secara otomatis menangkap sinyal bluetooth yang dipancarkan *passive tag*
3. Kemudian *reference points* akan mengirimkan RSSI dari *passive tag* dan MAC address dari *reference points* menuju server
4. Server menerima data yang dikirimkan *reference points* kemudian mengklasifikasikan nama lokasi dari pola sinyal yang dikirimkan oleh beberapa *reference points* dengan menggunakan algoritma KNN.
5. Hasil klasifikasi akan ditampilkan pada layar monitor dan akan dicatat oleh server pada sebuah file log.

#### c. Format File Log Tahap Online

Tabel 3.2 di bawah ini merupakan format dari *file log* tahap *online*. *File* ini nantinya akan disimpan pada server dengan format csv. *File log* yang akan dibuat akan terdiri dari :

1. Nama lokasi hasil klasifikasi
2. Waktu penentuan lokasi (klasifikasi) dengan format HH:MM:SS

Tabel 3.2 Format File Log Tahap Online

Hasil klasifikasi (lokasi)	Waktu
Ruang 1	00:00:00
Ruang 1	00:00:01
Ruang 2	00:03:00

### 3.4 Perancangan Pengujian

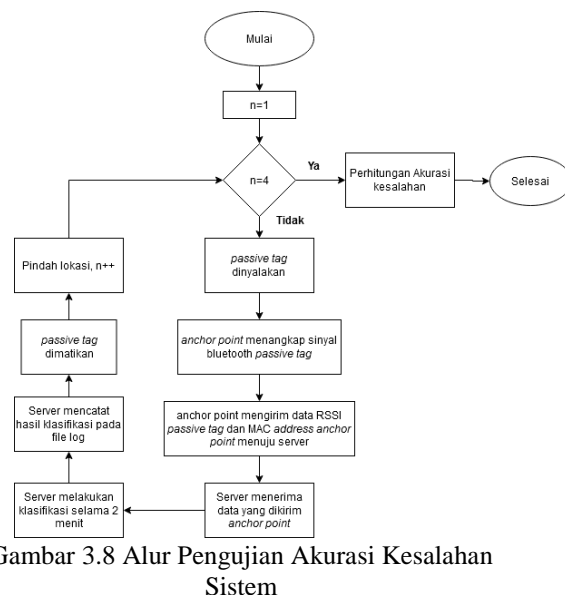
Pengujian dilakukan untuk mengetahui akurasi dari sistem yang telah dibangun. Pada pengujian ini menggunakan persamaan di bawah ini.

$$\text{Akurasi Kesalahan} = \frac{\text{Jumlah Kesalahan Prediksi}}{\text{Total Prediksi}} \times 100 \% \quad (1)$$

Hasil perhitungan diatas digunakan untuk menunjukkan akurasi dari sistem penentuan lokasi yang diimplementasikan. Rumus diatas juga digunakan pada pengujian untuk menghitung akurasi kesalahan tiap sub-lokasi dalam ruang.

#### a. Pengujian Akurasi Kesalahan Sistem

Pada Gambar 3.8 di bawah ini, merupakan alur dari pengujian akurasi kesalahan sistem.



Gambar 3.8 Alur Pengujian Akurasi Kesalahan Sistem

1. Pada ruang ke-n, *passive tag* dinyalakan
  2. *Reference points* akan menangkap sinyal bluetooth dari *passive tag*
  3. *Reference points* akan mengirim RSSI dari *passive tag* dan MAC address *reference points* menuju server menggunakan format data JSON
  4. Server menerima data yang dikirim *reference points*
  5. Server melakukan klasifikasi selama  $\pm 2$  menit dan mencatat hasil klasifikasi pada *file log*
  6. *Passive tag* dimatikan
  7. Proses nomor 1-6 diulang sampai semua lokasi selesai dilakukan pengambilan data.
- Kemudian, untuk format *file log* yang akan

digunakan untuk perhitungan akurasi kesalahan dalam penentuan lokasi terdiri dari :

1. Hasil klasifikasi (nama lokasi)
2. Waktu klasifikasi (HH:MM:SS)
3. Nama lokasi sebenarnya
4. Kecocokan, maksudnya adalah apabila hasil klasifikasi sama dengan lokasi sebenarnya, maka hasil pencocokan = cocok (match). Sebaliknya, jika hasil klasifikasi tidak sama dengan lokasi sebenarnya, maka hasil kecocokan = tidak cocok (mismatch).

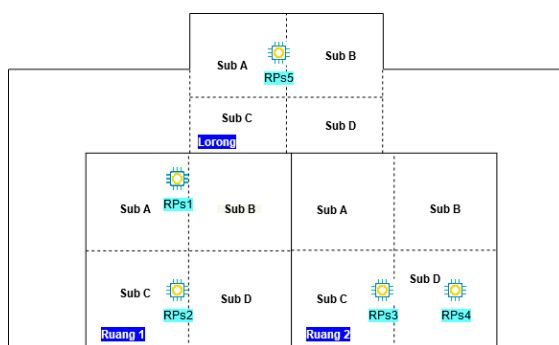
Pada Tabel berikut ini, merupakan format *file log* untuk pengujian ini.

Tabel 3.3 Format *File Log* pengujian

Hasil klasifikasi	Waktu	Lokasi sebenarnya	Kecocokan
Ruang 1	00:00:01	Ruang 1	Match
Lorong	00:00:02	Ruang 1	Mismatch
Ruang 1	00:00:03	Ruang 1	Match

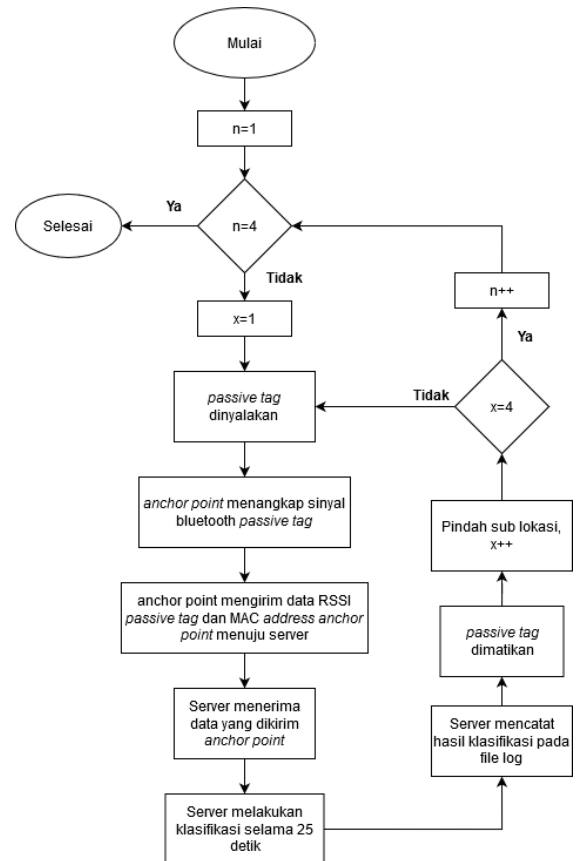
#### b. Pengujian Akurasi Kesalahan Sistem Pada Tiap Sub-Lokasi

Pengujian ini bertujuan untuk mengetahui pada sub lokasi mana pada suatu ruangan yang memiliki tingkat akurasi kesalahan yang tinggi, sehingga nantinya dapat dijadikan objek permasalahan pada penelitian mendatang. Pada tiap lokasi implementasi akan dibagi menjadi 4 sub lokasi seperti Gambar 3.9 di bawah ini



Gambar 3.9 Pembagian Sub Lokasi

Kemudian, pada Gambar 3.10 di bawah ini menggambarkan tentang alur dari pengujian akurasi kesalahan tiap sub lokasi.



Gambar 3.10 Alur Pengujian Akurasi Sistem pada tiap sub lokasi

1. Variabel *n* mewakili ruangan dan variabel *x* mewakili sub lokasi
2. Pada ruang ke-*n* dan sub lokasi ke-*x* *passive tag* dinyalakan
3. *Reference points* akan menangkap sinyal bluetooth dari *passive tag*
4. *Reference points* akan mengirim RSSI dari *passive tag* dan *MAC address reference points* menuju server menggunakan format data JSON
5. Server menerima data yang dikirim *reference points*
6. Server melakukan klasifikasi selama  $\pm 25$  detik pada sub lokasi *x* dan mencatat hasil klasifikasi pada *file log*
7. *Passive tag* dimatikan
8. Apabila *x* bukan sub lokasi terakhir maka pindah pada sub lokasi lain.
9. Apabila *x* adalah sub lokasi terakhir, maka pindah lokasi
10. Proses pada nomor 2-9 diulang sampai semua sub lokasi pada masing-masing ruangan tercatat hasil klasifikasinya pada *file log*

Selanjutnya, untuk format *file log* yang digunakan pada pengujian ini tersusun dari :

1. Hasil klasifikasi

2. Waktu klasifikasi (HH:MM:SS)
3. Nama Ruangan sebenarnya
4. Sub lokasi pada ruangan
5. Kecocokan, maksudnya adalah apabila hasil klasifikasi sama dengan lokasi sebenarnya, maka hasil pencocokan = cocok (match). Sebaliknya, jika hasil klasifikasi tidak sama dengan lokasi sebenarnya, maka hasil kecocokan = tidak cocok (mismatch)

Pada Tabel di bawah ini menunjukkan format dari *file log* yang digunakan.

Tabel 3.4 Format *File Log* pengujian akurasi kesalahan sub lokasi

Hasil klasifikasi	Waktu	Lokasi sebenarnya	Sub lokasi	Kecocokan
Ruang 1	00:00:01	Ruang 1	A	Match
Lorong	00:00:02	Ruang 1	B	Mismatch
Ruang 1	00:00:03	Ruang 1	C	Match
Ruang 1	00:00:03	Ruang 1	D	Match

Setelah pengumpulan data pada *file log* telah selesai, langkah selanjutnya adalah menghitung akurasi kesalahan pada tiap lokasi sehingga diketahui pada sub lokasi mana di suatu ruangan yang memiliki tingkat akurasi kesalahan paling tinggi. Pada Tabel 3.5 di bawah ini merupakan format untuk hasil perhitungan akurasi kesalahan pada tiap sub lokasi.

Tabel 3.5 Tabel Akurasi Kesalahan tiap sub lokasi

No	Ruang	Sub-Lokasi	Akurasi kesalahan
1	Ruang 1	Sub Lokasi A	
		Sub Lokasi B	
		Sub Lokasi C	
		Sub Lokasi D	
2	Ruang 2	Sub Lokasi A	
		Sub Lokasi B	
		Sub Lokasi C	
		Sub Lokasi D	
3	Lorong	Sub Lokasi A	
		Sub Lokasi B	
		Sub Lokasi C	
		Sub Lokasi D	

Pada Tabel 3.5 disajikan data berupa nama ruang, sub lokasi dan akurasi kesalahan. Tiap-tiap ruang yang digunakan untuk implementasi akan dibagi menjadi 4 bagian, kemudian akurasi kesalahannya dihitung menggunakan Persamaan (1).

## 4. IMPLEMENTASI

### 4.1 Tahap Offline

Hasil pada tahap ini berupa *data training* yang berisi data nama lokasi beserta pola sinyal dari masing-masing lokasi tersebut. Pada Tabel 4.1 berikut ini merupakan isi dari *file data training*.

Tabel 4.1 Hasil implementasi Tahap Offline

Ruang 1,13:26:56,-75,-120,-120,-120,-120
Ruang 1,13:26:57,-83,-94,-120,-95,-120
Ruang 1,13:26:58,-90,-89,-120,-120,-120
....
....
Lorong,13:35:50,-120,-120,-120,-120,-89
Lorong,13:35:50,-120,-87,-91,-120,-81
Lorong,13:35:52,-120,-86,-120,-120,-89
....
....
Ruang 2,13:40:03,-120,-120,-120,-120,-92
Ruang 2,13:40:04,-120,-88,-93,-92,-91
Ruang 2,13:40:05,-92,-98,-84,-87,-120

Pada data yang ditampilkan di atas, terlihat bahwa sistem dapat mengumpulkan pola sinyal dari passive tag pada tiap-tiap lokasi. Pola sinyal yang diperoleh dicatat bersama dengan waktu pengambilan pola sinyal, kemudian diberi label sesuai dengan nama lokasi. Kumpulan pola sinyal yang diperoleh tersebut kemudian disimpan dengan format file .csv. Berdasarkan penjelasan di atas, hasil implementasi tahap *offline* sudah sesuai dengan perancangan pada bab 3. Hasil implementasi tahap *offline* ini akan digunakan pada tahap *online* untuk menentukan lokasi.

### 4.2 Tahap Online

Pada tahap ini dilakukan proses penentuan lokasi. Hasil penentuan lokasi dicatat dan disimpan oleh server pada *file log*. Pada Tabel 4.2 di bawah ini merupakan isi dari *file log* tahap *online*.

Tabel 4.2 Hasil Implementasi Tahap Online

Ruang 1,14:58:09
Ruang 1,14:58:09
Ruang 1,14:58:09
....
....
Lorong,14:58:27
Lorong,14:58:27
Lorong,14:58:27

Kemudian pada Gambar 4.1 - Gambar 4.4 di bawah ini merupakan tangkapan layar dari implementasi tahap *online*.



```
PS D:\DATA KULIAH\SEMESTER VIII\skripsi-indoorlocalization\CODE\trial_env_2> py .\real_time_classifier_0.1.py
* Serving Flask app "real_time_classifier_0.1" (lazy loading)
* Environment: production
WARNING: This is a development server. Do not use it in a production deployment.
Use a production WSGI server instead.
* Debug mode: off
Node 3 is Online
Node 1 is Online
Node 4 is Online
Node 5 is Online
Node 2 is Online
Beacon tidak terdeteksi
Beacon tidak terdeteksi
```

Gambar 4.1 Menjalankan Server

```
Beacon tidak terdeteksi
Beacon tidak terdeteksi
Beacon tidak terdeteksi
Pada Jan 14:58:09 beacon berada di Ruang 1
Pada Jan 14:58:09 beacon berada di Ruang 1
Pada Jan 14:58:09 beacon berada di Ruang 1
Pada Jan 14:58:10 beacon berada di Ruang 1
Pada Jan 14:58:10 beacon berada di Ruang 1
Pada Jan 14:58:10 beacon berada di Ruang 1
Pada Jan 14:58:10 beacon berada di Ruang 1
Pada Jan 14:58:10 beacon berada di Ruang 1
Pada Jan 14:58:10 beacon berada di Ruang 1
```

Gambar 4.2 Server menampilkan lokasi *passive tag* pada Ruang 1

```
Pada Jan 14:58:26 beacon berada di Ruang 1
Pada Jan 14:58:26 beacon berada di Ruang 1
Pada Jan 14:58:27 beacon berada di Lorong
Pada Jan 14:58:27 beacon berada di Lorong
Pada Jan 14:58:27 beacon berada di Lorong
Pada Jan 14:58:27 beacon berada di Lorong
Beacon tidak terdeteksi
Beacon tidak terdeteksi
Pada Jan 14:58:28 beacon berada di Lorong
Pada Jan 14:58:28 beacon berada di Lorong
Pada Jan 14:58:28 beacon berada di Lorong
Pada Jan 14:58:28 beacon berada di Lorong
Pada Jan 14:58:28 beacon berada di Lorong
Pada Jan 14:58:28 beacon berada di Lorong
Pada Jan 14:58:28 beacon berada di Lorong
```

Gambar 4.3 Server menampilkan lokasi *passive tag* pada Lorong

```
Pada Jan 14:58:41 beacon berada di Ruang 2
Pada Jan 14:58:41 beacon berada di Ruang 2
Pada Jan 14:58:41 beacon berada di Ruang 2
Pada Jan 14:58:41 beacon berada di Ruang 2
Pada Jan 14:58:41 beacon berada di Ruang 2
Pada Jan 14:58:41 beacon berada di Ruang 2
Pada Jan 14:58:41 beacon berada di Ruang 2
Pada Jan 14:58:41 beacon berada di Ruang 2
Pada Jan 14:58:41 beacon berada di Ruang 2
Pada Jan 14:58:41 beacon berada di Ruang 2
```

Gambar 4.4 Server menampilkan lokasi *passive tag* pada Ruang 2

Pada Gambar 4.1 merupakan tampilan dari server saat pertama menjalankan server. Server menampilkan tulisan “Beacon tidak terdeteksi” karena *passive tag* belum dinyalakan. Kemudian pada Gambar 4.2 menampilkan saat *passive tag* dinyalakan, server menampilkan lokasi dari *passive tag* pada Ruang 1. Pada Gambar 4.3, lokasi dari *passive tag* berpindah menuju Lorong, sehingga server menampilkan lokasi dari *passive tag* pada lorong. Pada Gambar 4.4, *passive tag* berpindah menuju Ruang 2, sehingga server menampilkan lokasi *passive tag* pada Ruang 2.

Pada data di atas, terlihat bahwa sistem dapat melakukan penentuan lokasi dari *passive tag*. Lokasi yang telah didapatkan kemudian ditampilkan pada layar beserta waktu penentuan lokasi. Waktu penentuan lokasi yang ditampilkan memiliki selang waktu tidak sampai 1 detik. Hal ini mengindikasikan bahwa sistem dapat melakukan penentuan lokasi secara *real-time*. Pada data di atas juga ditampilkan apabila *passive tag* tidak terdeteksi, maka sistem akan menampilkan tulisan “Beacon tidak terdeteksi”.

Berdasarkan data di atas, maka dapat disimpulkan bahwa implementasi tahap *online* bekerja sesuai dengan perancangan, sehingga dapat dikatakan bahwa tahap *online* berhasil dilakukan.

## 5. PENGUJIAN

### 5.1 Pengujian Akurasi Kesalahan Sistem

Implementasi pengambilan data hasil klasifikasi pada tahap ini menghasilkan sebuah *file log* yang berisi data hasil klasifikasi beserta nama lokasi sebenarnya. Data ini kemudian akan di hitung menggunakan persamaan (1). Berikut merupakan *file log* dari implementasi pengambilan data hasil klasifikasi yang telah dilakukan.

Tabel 5.1 Hasil pengambilan data pengujian 1

Ruang 1,15:31:46,Ruang 1,Match
Ruang 1,15:31:46,Ruang 1,Match
Ruang 1,15:31:46,Ruang 1,Match
.....
.....
Ruang 2,15:32:00,Ruang 1,Mismatch
Ruang 2,15:32:00,Ruang 1,Mismatch
Ruang 2,15:32:00,Ruang 1,Mismatch

Pada tabel di atas sistem dapat menentukan kecocokan berdasarkan hasil klasifikasi dengan nama lokasi sebenarnya. Nilai kecocokan ditulis dengan ‘Match’ apabila hasil klasifikasi sama dengan nama lokasi sebenarnya. Sebaliknya, apabila hasil klasifikasi tidak sama dengan nama lokasi sebenarnya, maka nilai kecocokan ditulis ‘Mismatch’. Nilai kecocokan ini akan digunakan untuk melakukan perhitungan akurasi kesalahan sistem.

```
C:\Windows\System32\cmd.exe
D:\DATA KULIAH\SEMESTER VIII\skripsi-indoorlocalization\CODE\trial_env_2>py akurasi_pelaca
Akurasi kesalahan sistem = 15.24 %
D:\DATA KULIAH\SEMESTER VIII\skripsi-indoorlocalization\CODE\trial_env_2>
```

Gambar 5.1 Hasil perhitungan akurasi kesalahan sistem

Pada Gambar 5.1 diatas menunjukkan bahwa akurasi kesalahan sistem dalam penentuan lokasi sebesar 15,24%, yang berarti menunjukkan secara tidak langsung bahwa akurasi sistem untuk menentukan lokasi adalah  $100\% - 15,24\% = 84,76\%$ . Pada pengujian ini belum diketahui pada sub lokasi mana pada suatu ruangan yang memiliki tingkat presentase kesalahan akurasi paling tinggi. Untuk mengetahui pada sub lokasi mana yang memiliki

presentase akurasi kesalahan paling tinggi akan dilakukan pada pengujian berikutnya.

## 5.2 Pengujian Akurasi Kesalahan Sistem Pada Tiap Sub Lokasi

Implementasi pengambilan data hasil klasifikasi pada tahap ini menghasilkan sebuah *file log* yang berisi data hasil klasifikasi pada masing-masing sub lokasi. Data ini kemudian akan di hitung menggunakan *script* khusus yang akan menghitung akurasi kesalahan pada tiap sub lokasi. Berikut merupakan *file log* dari implementasi pengambilan data hasil klasifikasi yang telah dilakukan.

Tabel 5.2 Hasil Pengambilan Data Klasifikasi Tiap Sub Lokasi

```
Ruang 2,14:02:21,Ruang 1,A,Mismatch
Ruang 1,14:02:21,Ruang 1,A,Match
Ruang 1,14:02:21,Ruang 1,A,Match
....
....
Ruang 2,14:04:04,Ruang 1,B,Mismatch
Lorong,14:04:04,Ruang 1,B,Mismatch
Ruang 1,14:04:04,Ruang 1,B,Match
```

Pada Tabel 5.2 di atas disajikan hasil dari pengambilan data klasifikasi pada pengujian ini. Pada Tabel 5.2 di atas menunjukkan bahwa sistem dapat melakukan penentuan lokasi dan mencatat hasil penentuan lokasi bersama dengan waktu penentuan lokasi, nama lokasi sebenarnya, sub lokasi dan kecocokan. Nilai kecocokan ditentukan berdasarkan pada kecocokan antara hasil penentuan lokasi dengan nama lokasi sebenarnya. Langkah selanjutnya adalah mengukur akurasi kesalahan sistem pada tiap sub lokasi masing-masing ruangan.

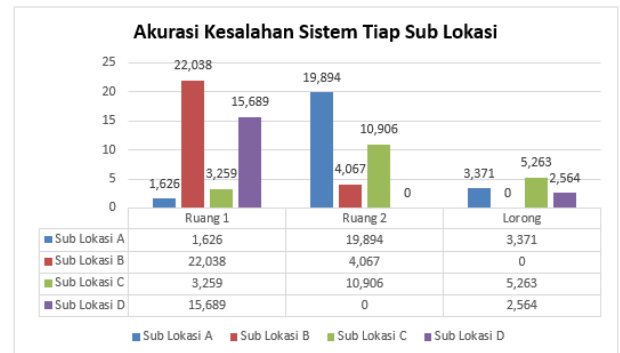
```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.18362.720]
(c) 2019 Microsoft Corporation. All rights reserved.

D:\DATA KULIAH\SEMESTER VIII\skripsi-indoorLocalization\CODE\trial
Kesalahan Prediksi pada Ruang 1 adalah :
Sub-lokasi A = 1.626 %
Sub-lokasi B = 22.038 %
Sub-lokasi C = 3.259 %
Sub-lokasi D = 15.689 %
Kesalahan Prediksi pada Ruang 2 adalah :
Sub-lokasi A = 19.894 %
Sub-lokasi B = 4.067 %
Sub-lokasi C = 10.906 %
Sub-lokasi D = 0.000 %
Kesalahan Prediksi pada Lorong adalah :
Sub-lokasi A = 3.371 %
Sub-lokasi B = 0.000 %
Sub-lokasi C = 5.263 %
Sub-lokasi D = 2.564 %
```

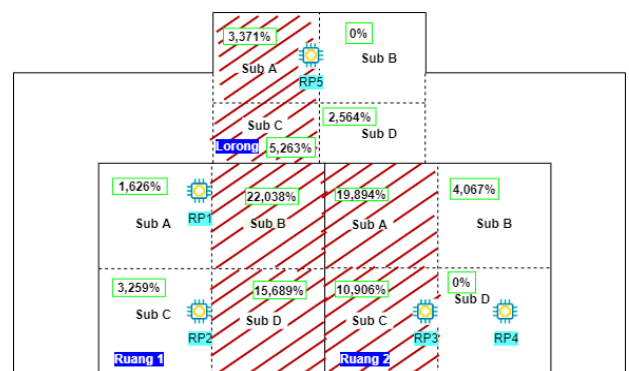
Gambar 5.2 Hasil Perhitungan Akurasi Sistem Pada Tiap Sub Lokasi

Pada Gambar 5.2 di atas dapat terlihat akurasi kesalahan sistem dari masing-masing sub lokasi pada tiap ruangan. Pada grafik di

bawah ini disajikan akurasi kesalahan pada tiap sub lokasi.

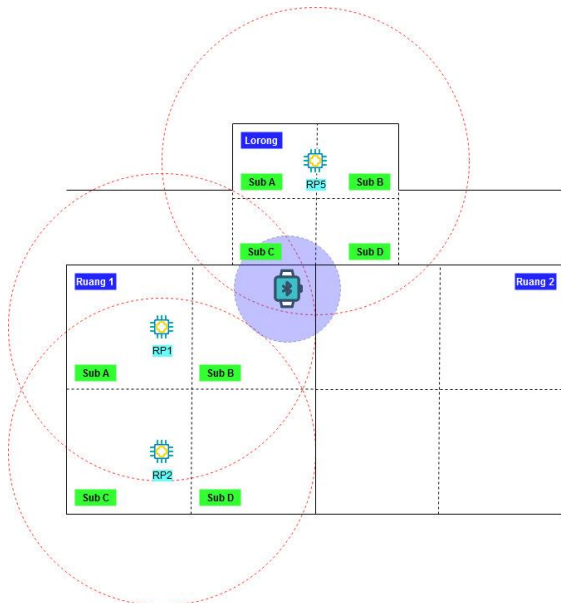


Gambar 5.3 Grafik Akurasi Kesalahan Tiap Sub Lokasi

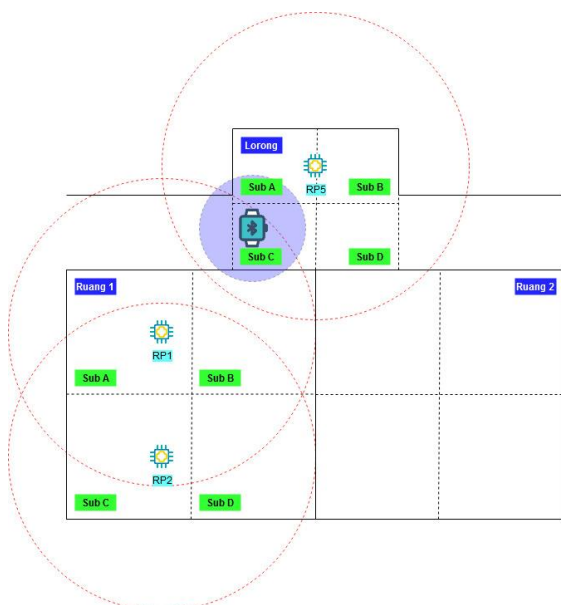


Gambar 5.4 Denah Sub Lokasi Beserta Akurasi Kesalahannya

Pada Gambar 5.3 di atas disajikan grafik dari akurasi kesalahan pada tiap sub lokasi. Sub lokasi yang berdekatan dengan sub lokasi dari lokasi lainnya memiliki tingkat akurasi kesalahan yang cukup tinggi. Pada Gambar 5.4 terlihat bahwa pada sub lokasi B dan D pada Ruang 1 memiliki akurasi kesalahan tertinggi pada ruangan tersebut. Pada Lorong, sub lokasi A dan C memiliki akurasi kesalahan tertinggi pada lokasi tersebut. Pada Ruang 2, sub lokasi A dan C memiliki akurasi kesalahan tertinggi pada lokasi tersebut. Gambar di atas menunjukkan bahwa pada sub-sub lokasi tersebut, terdapat kesalahan klasifikasi yang cukup tinggi. Hal ini dapat terjadi karena sub-sub lokasi tersebut memiliki pola sinyal/*fingerprint* yang cenderung sama dengan sub lokasi yang berdekatan pada ruang yang berbeda. Sebagai contoh, posisi *passive tag* berada pada sub lokasi B pada Ruang 1 namun sistem mendeteksi *passive tag* pada lorong. Hal ini bisa terjadi karena pola sinyal yang terdapat pada sub lokasi B Ruang 1 memiliki kemiripan dengan sub lokasi C Lorong.



Gambar 5.5 Jangkauan Sinyal saat *passive tag* di Ruang 1



Gambar 5.6 Jangkauan Sinyal saat *passive tag* di Lorong

Pada Gambar 5.5 dan Gambar 5.6 menunjukkan posisi dari *passive tag* yang berdekatan. Posisi *passive tag* pada saat berada pada sub lokasi B Ruang 1 dan sub lokasi C pada Lorong berada pada jangkauan terjauh dari *reference points* pada Ruang 1 dan juga berada posisi terjauh dari jangkauan sinyal *reference points* pada Lorong. Hal ini bisa saja menyebabkan pola sinyal yang diterima oleh server memiliki kemiripan karakteristik antara kedua lokasi sehingga dapat terjadi kesalahan saat melakukan klasifikasi. Untuk membuktikannya, penulis mencoba meninjau *file data training fingerprint* yang telah

dikumpulkan pada tahap *offline*. Pada Tabel di bawah ini merupakan contoh dari *fingerprint* yang hampir sama.

Tabel 5.3 *Fingerprint* yang memiliki kemiripan

Nama Lokasi	RP 1	RP 2	RP 3	RP 4	RP 5	Baris Data ke-
Ruang 1	-120	-87	-120	-120	-120	9
	-120	-91	-120	-120	-120	67
	-120	-89	-120	-120	-120	115
Lorong	-120	-90	-120	-120	-120	255
	-120	-92	-120	-120	-95	245
	-120	-94	-120	-120	-120	319

Pada Tabel 5.3 di atas pada kolom pertama dari kiri terdapat nama lokasi, yakni lokasi dimana *fingerprint* tersebut direkam. Kemudian, pada kolom selanjutnya, yakni kolom RP 1 – RP 5 yang merupakan kolom untuk RSSI dari *passive tag* yang diperoleh dari masing-masing *reference points*. Pada kolom paling pojok kanan terdapat kolom yang menjelaskan pada baris data ke berapa pada *data training* data *fingerprint* yang ditampilkan di atas diperoleh.

Pada Tabel 5.3 tersebut dapat dilihat bahwa *fingerprint* pada Ruang 1 dan Lorong memiliki kemiripan. Bahkan RSSI dari *passive tag* yang diperoleh dari RP 1, dan RP 3 – RP 5 memiliki nilai yang sama persis (kecuali pada baris kedua lokasi Lorong). Sedangkan RSSI *passive tag* dari RP 2 hanya memiliki selisih nilai yang sedikit antar lokasi. Maka dapat disimpulkan bahwa kesalahan dalam penentuan lokasi dapat disebabkan karena adanya kemiripan *fingerprint* pada lokasi yang berbeda.

## 6. KESIMPULAN

Berdasarkan hasil dan analisa dari tahap perancangan, implementasi, dan pengujian yang telah dilakukan, maka untuk menjawab pertanyaan pada rumusan masalah, dapat disimpulkan bahwa :

1. Penentuan lokasi dalam gedung menggunakan metode *fingerprinting* dan Bluetooth Low Energy dapat diimplementasikan dengan menerapkan 2 tahap, yakni tahap *offline* dan tahap *online*. Pada tahap *offline* dilakukan pengumpulan data pola sinyal bluetooth pada masing-masing lokasi untuk membentuk sebuah *data training*. Kemudian, pada tahap *online* dilakukan

penentuan lokasi dari perangkat bluetooth sesuai dengan perancangan yang telah dibuat.

2. Implementasi *Monitor Based Localization* pada penentuan lokasi dalam gedung dapat dilakukan dengan menggunakan 3 komponen. Komponen pertama adalah *passive tag* yang berfungsi sebagai perangkat yang akan ditentukan lokasinya. Perangkat ini hanya memancarkan sinyal bluetooth secara terus-menerus. Komponen kedua adalah *reference points* yang berfungsi untuk menangkap sinyal bluetooth yang dipancarkan oleh *passive tag* dan mengirimkannya menuju server. Komponen ketiga adalah server yang berfungsi untuk mengubah kumpulan pola sinyal yang dikirim oleh *reference points* menjadi sebuah nama lokasi. Perangkat ini juga berfungsi sebagai penyimpanan *data training* yang berisi kumpulan pola sinyal dari masing-masing lokasi.
3. Tingkat akurasi dari sistem penentuan lokasi dalam gedung ini sebesar 84,76% dengan akurasi kesalahan sebesar 15,24%. Tingkat akurasi pada tiap sub lokasi dari suatu ruangan memiliki tingkat yang bervariasi. Tingkat akurasi kesalahan penentuan lokasi pada suatu sub lokasi yang berdekatan pada lokasi yang berbeda memiliki tingkat yang cukup tinggi dibandingkan pada sub lokasi yang lain. Hal ini dapat terjadi karena adanya kemiripan karakteristik pola sinyal bluetooth pada sub lokasi - sub lokasi tersebut.

## 7. DAFTAR PUSTAKA

- Brena, R. F. et al. (2017) 'Evolution of Indoor Positioning Technologies: A Survey', *Journal of Sensors*, 2017. doi: 10.1155/2017/2630413.
- Chan, S. and Sohn, G. (2012) 'Indoor Localization Using Wi-Fi Based Fingerprinting and Trilateration Techniques for Lbs Applications', *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XXXVIII-4/, pp. 1–5. doi: 10.5194/isprsarchives-xxxviii-4-c26-1-2012.
- Chatschik, B. (2001) 'An overview of the Bluetooth wireless technology', *IEEE Communications Magazine*, vol.39, no(December), pp. 86–94.
- Faragher, R. and Harle, R. (2015) 'Location Fingerprinting With Bluetooth Low Energy Beacons', *IEEE Journal on Selected Areas in Communications*, 33(11), pp. 2418–2428. doi: 10.1109/JSAC.2015.2430281.
- Heydon, R. and Hunn, N. (2012) *Bluetooth Low Energy: The Developer's Handbook*, CSR Presentation. Available at: <https://www.bluetooth.org/DocMan/handlers/DownloadDoc.ashx>.
- Jiang, P. et al. (2015) 'Indoor mobile localization based on Wi-Fi fingerprint's important access point', *International Journal of Distributed Sensor Networks*, 2015. doi: 10.1155/2015/429104.
- Li, W., Yang, P. and Zhou, B. (2008) 'Internet-Based Spatial Information Retrieval', *Encyclopedia of GIS*, pp. 596–599. doi: 10.1007/978-0-387-35973-1\_650.
- Marques, N., Meneses, F. and Moreira, A. (2012) 'Combining similarity functions and majority rules for multi-building, multi-floor, WiFi positioning', 2012 *International Conference on Indoor Positioning and Indoor Navigation, IPIN 2012 - Conference Proceedings*, (November). doi: 10.1109/IPIN.2012.6418937.
- Subedi, S. and Pyun, J.-Y. (2017) 'Practical Fingerprinting Localization for Indoor Positioning System by Using Beacons', *Journal of Sensors*, 2017, pp. 1–16. doi: 10.1155/2017/9742170.
- Yucel, H., Yazici, A. and Edizkan, R. (2014) 'İç ortam konumlandırma sistemleri üzerine bir derleme', 2014 22nd *Signal Processing and Communications Applications Conference, SIU 2014 - Proceedings*, pp. 1267–1270. doi: 10.1109/SIU.2014.6830467.
- Zafari, F., Gkelias, A. and Leung, K. K. (2019) 'A Survey of Indoor Localization Systems and Technologies', *IEEE Communications Surveys and Tutorials*. IEEE, 21(3), pp. 2568–2599. doi: 10.1109/COMST.2019.2911558.