

LAPORAN AKHIR



JUDUL:

**PERBANDINGAN ALGORITMA FLOYD WARSHALL DAN A
STAR DALAM PENENTUAN RUTE TERPENDEK MENUJU UNIT
GAWAT DARURAT**

TIM PENGUSUL :

**Dr Uung Ungkawa, MT (NIDN: 0411105902)
Dwiki Faizal Mu'minin (15-2015-103)**

Biaya Mandiri

INSTITUT TEKNOLOGI NASIONAL

September 2022

Halaman Pengesahan

Judul Penelitian : PERBANDINGAN ALGORITMA FLOYD WARSHALL DAN A STAR DALAM PENENTUAN RUTE TERPENDEK MENUJU UNIT GAWAT DARURAT

Kode/ Nama Rumpun Ilmu : 538/ Rekayasa Perangkat Lunak

Ketua Peneliti :

- a. Nama Lengkap : Dr Uung Ungkawa, MT.
- b. NIDN : 0411105902
- c. Jabatan Fungsional : Lektor
- d. Program Studi : Informatika
- e. Nomor HP : 08121443095
- f. Alamat surel (e-mail) : uung@itenas.ac.id

Anggota Peneliti (1)

- a. Nama Lengkap : Dwiki Faizal Mu'minin
- b. NRP : 15-2015-103
- c. Instansi : Mahasiswa Institut Teknologi Nasional Bandung

Anggota Peneliti (2)

- a. Nama Lengkap : Dewi Rosmala, MIT
- b. NIDN : 0422106801
- c. Perguruan Tinggi : Dosen Institut Teknologi Nasional Bandung

Lama Penelitian Keseluruhan : 6 bulan

Penelitian Tahun ke : 1 (Satu)

Biaya Penelitian Keseluruhan : Rp. 8.000.000,-

Biaya Tahun Berjalan :

- diusulkan ke DIKTI Rp. 0,-
- dana internal PT Rp.0
- dana institusi lain Rp.0
- inkind sebutkan

Mengetahui
Dekan



itenas
FTI

Jono Suhartono, ST, MT., Ph.D
NIP/NIK

Bandung, 12 – 09 - 2022



Dr Uung Ungkawa, M.T
NIP/NIK. 120071201

Menyetuji
Ketua LPPM Itenas



itenas
LPPM

Iwan Juwana, S.T., MT, Ph.D
NIP/NIK

DAFTAR ISI

DAFTAR ISI.....	i
DAFTAR GAMBAR.....	iv
DAFTAR TABEL	vi
INTISARI	vii
ABSTRACT	viii
BAB I.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	3
1.3 Tujuan.....	4
1.4 Ruang Lingkup.....	4
1.5 Metode Penelitian.....	4
1.6 Tinjauan Pustaka	6
1.7 Kontribusi Penelitian.....	9
1.8 Sistematika Penulisan.....	10
BAB II	11
2.1 Rumah Sakit.....	11
2.2 Pencarian Rute Terpendek	12
2.3 <i>Google Maps API</i>	12
2.4 GPS (Global Positioning System)	13
2.5 Algoritma	14
2.6 Algoritma <i>Floyd Warshall</i>	14
2.7 Algoritma <i>A Star</i>	14
2.8 Graph	15
2.8.1 Macam – macam Graph Menurut Arah dan Bobotnya.....	15
2.9 Lintasan	16
BAB III.....	17
3.1 Alur Peneltiain.....	17
3.1.1 Tahap awal penelitian (analisis)	18
3.1.2 Tahap Implementasi.....	19
3.1.3 Tahap Pengujian.....	24
3.1.4 Kesimpulan dan Saran	24

3.2 Analisis Kebutuhan	24
3.2.1 Requirement Gathering and Refinement	24
3.2.2 Analisis Kebutuhan Pembangunan Sistem.....	25
3.2.3 Analisis Kebutuhan Instalasi Android Studio.....	25
3.3 Perancangan Umum (Quick Design).....	26
3.3.1 Analisis Perancangan Sistem	29
3.4 Membangun Prorotype (Build Prorotype)	29
3.4.1 Blok Diagram	29
3.4.2 Use Case Diagram.....	31
3.4.3 Diagram Alir Sistem	35
3.4.4 Flowchart Sistem.....	41
3.4.5 Flowchart Pengurutan Rumah Sakit	42
3.4.6 Flowchart Subproses Hasil Rute	43
3.4.7 Perancangan Database	44
3.4.8 Squence Diagram	46
3.4.9 State Transisi Diagram.....	49
3.5 Identifikasi Kebutuhan Peneltiain.....	52
3.5.1 Data Layanan Kesehatan Rumah Sakit.....	52
3.5 Analisis Node	53
3.6 Analisis Algoritma A Star	54
3.7 Analisis Algoritma Floyd Warshall.....	60
3.8 Analisis Haversine.....	67
3.9 Evaluasi	69
3.9.1 Perancangan User Interface	69
BAB IV	78
4.1 Lingkungan Pengembangan.....	78
4.1.1 Perangkat Keras	79
4.1.2 Perangkat Lunak	79
4.1.3 Implementasi Perancangan <i>User Interface</i>	84
4.2 Pengujian Fungsionalitas.....	95
4.2.1 Proses Pelaporan Darurat.....	95
4.2.2 Proses Pemilihan Rumah Sakit	96
4.2.3 Proses Rute Algoritma A Star	97
4.2.4 Proses Rute Algoritma FloydWarshall	98
4.3 Pengujian Beta.....	100

4.3.1 Pengujian Titik Awal User Node	100
4.3.2 Pengujian Hasil Algoritma A Star.....	107
4.3.3 Pengujian Hasil Algoritma Floyd Warshall	117
4.3.4 Pengujian Hasil Algoritma Floyd Warshall	126
4.3.5 Pengujian Pengurutan Rumah Sakit	135
4.3.6 Pengujian Hasil Jarak	137
BAB V	142
5.1 Kesimpulan	142
DAFTAR PUSTAKA	143

DAFTAR GAMBAR

Gambar 1 Peta Penelitian.....	9
Gambar 2 Graph dengan 4 verteks 5 edges	15
Gambar 3 Graph berarah dan beraturan	16
Gambar 4 Alur penelitian.....	17
Gambar 5 Pengembangan perangkat lunak prototype	20
Gambar 6 Flowchart algoritma a star.....	21
Gambar 7 Flowchart algoritma floyd warshall	23
Gambar 8 Perancangan umum quick design.....	27
Gambar 9 Blok diagram.....	30
Gambar 10 Use case diagram.....	31
Gambar 11 Diagram alir sistem	36
Gambar 12 Proses login	37
Gambar 13 Proses registrasi.....	38
Gambar 14 Proses laporan darurat	39
Gambar 15 Pemilihan rumah sakit.....	40
Gambar 16 Penentuan rute.....	40
Gambar 17 Flowchart sistem	41
Gambar 18 Pengurutan rumah sakit.....	43
Gambar 19 Subproses hasil rute	44
Gambar 20 Entity relantionship	45
Gambar 21 Table relationship diagram.....	45
Gambar 22 Squence diagram registrasi	47
Gambar 23 Squence diagram login.....	47
Gambar 24 Squence diagram login.....	48
Gambar 25 Squence diagram pemilihan rumah sakit	48
Gambar 26 Squence diagram algoritma.....	49
Gambar 27 Transisi diagram registrasi	49
Gambar 28 Transisi diagram login.....	50
Gambar 29 Transisi diagram laporan daurrat	50
Gambar 30 Transisi pemilihan rumah sait	51
Gambar 31 Transisi run algoritma	51
Gambar 32 Transisi diagram algoritma	52
Gambar 33 Sebaran rumah sakit kota bandung	53
Gambar 34 Daftar rumah sakit.....	53
Gambar 35 Titik node	54
Gambar 36 Analisis node.....	54
Gambar 37 Contoh studi kasus	55
Gambar 38 Model graph antar node	57
Gambar 39 Titik node v0	57
Gambar 40 Titik node v1 bestnode	58
Gambar 41 Titik node v5 sebagai bestnode goal	58
Gambar 42 Open list dan close list	59
Gambar 43 Hasil perhitungan a star.....	59

Gambar 44 Implementasi algoritma a star	59
Gambar 45 Graph floyd warshall.....	61
Gambar 46 Hasil perhitungan floydwrhsall	66
Gambar 47 Hasil implementasi floydwarshall.....	66
Gambar 48 Hasil haversine formula	68
Gambar 49 Perancangan UI registrasi	70
Gambar 50 Perancangan UI login.....	70
Gambar 51 Perancangan UI Screen	71
Gambar 52 Perancangan UI navigasi bar.....	72
Gambar 53 Perancangan menu utama.....	73
Gambar 54 Perancangan UI profil	73
Gambar 55 Perancangan UI paduan.....	74
Gambar 56 Perancangan laporan darurat	75
Gambar 57 Perancangan UI pemilihan rute	75
Gambar 58 Perancangan UI admin RS	76
Gambar 59 Perancangan menu admin	77
Gambar 60 Perancangan menu admin layanan darurat.....	78
Gambar 61 Instalasi android studio	80
Gambar 62 Instalasi android studio	80
Gambar 63 Instalasi android studio	81
Gambar 64 Instalasi android studio	81
Gambar 65 Instalasi android studio	82
Gambar 66 Instalasi xampp	82
Gambar 67 Instalasi xampp	83
Gambar 68 Instalasi xampp	83
Gambar 69 Instalasi xampp	84
Gambar 70 Implementasi menu registrasi	85
Gambar 71 Implementasi menu login.....	85
Gambar 72 Implementasi menu utama	86
Gambar 73 Implementasi navigasi menu.....	87
Gambar 74 Implementasi GPS aktif	87
Gambar 75 Implementasi menu pengguna.....	88
Gambar 76 Implementasi menu laporan darurat.....	89
Gambar 77 Implementasi menu panduan.....	89
Gambar 78 Implementasi menu riwayat	90
Gambar 79 Impelementasi menu informasi.....	91
Gambar 80 Implementasi menu pemilihan rrumah sakit	92
Gambar 81 Implementasi meu a star.....	92
Gambar 82 Implementasi menu floydwarshall	93
Gambar 83 Implementasi menu utama admin	94
Gambar 84 Implementasi menu laporan darurat admin.....	94
Gambar 85 Implementasi menu laporan cetak admin.....	94
Gambar 86 Pengujian alpha	95

DAFTAR TABEL

Table 1 Spesifikasi android.....	25
Table 2 Analisa perancangan sistem.....	29
Table 3 Skenario proses registrasi	32
Table 4 Skenario proses login.....	33
Table 5 Skenario laporan darurat.....	33
Table 6 Skenario pemilihan rumah sakit	34
Table 7 Skenario penentuan rute.....	35
Table 8 Skenario laporan masuk.....	35
Table 9 Latitude dan longitude	55
Table 10 Hasil jarak antar node	56
Table 11 Latitde dan longitude.	60
Table 12 Titik waypoint node	60
Table 13 Perhitungan matrix antar titik	61
Table 14 Perhitungan iterasi pertama.....	62
Table 15 Perhitungan iterasi kedua.....	63
Table 16 Tahapan iterasi ketiga	63
Table 17 Tahapan iterasi keempat	63
Table 18 Tahapan iterasi kelima	64
Table 19 Tahapan iterasi keenam.....	64
Table 20 Hasil perhitungan iterasi	65
Table 21 Pelaporan Darurat	95
Table 22 Pemilihan rumah sakit	97
Table 23 Penentuan rute a star	98
Table 24 Penentuan rute floyd warshall.....	98
Table 25 Hasil pengujian a star.....	107
Table 26 Pengujian hasil floyd warshall	117
Table 27 Hasil perhitungan menampilkan rumah sakit	126
Table 28 Pengujian pengurutan rumah sakit.....	136
Table 29 Hasil jarak a star dan floydwarshall.....	137
Table 30 Hasil jarak a star dan google maps	138
Table 31 Hasil jarak a Floyd warshall.	139

INTISARI

Kecerdasan buatan atau biasa disebut dengan AI (Artificial Intelligence) adalah salah satu cabang ilmu pengetahuan yang bisa diterapkan diberbagai bidang khususnya dalam penentuan rute terpendek, terutama jika mengalami emergency seperti kecelakaan dan sebagainya. Oleh sebab itu pemetaan sebuah lokasi rumah sakit begitu penting karena menjadi tempat yang dituju oleh setiap orang ketika dalam membutuhkan pertolongan medis dengan segera. Untuk membantu masyarakat dalam mengetahui lokasi rumah sakit diperlukan sebuah sistem yang dapat menentukan rute menuju lokasi rumah sakit dengan memanfaatkan smartphone yang terintegasi dengan Google Maps API yang dapat membantu menampilkan rute mana yang dilewati. Pada penelitian ini metode yang digunakan untuk penentuan rute yaitu algoritma A Star dan Floyd-Warshall, algoritma tersebut bekerja dengan mekanisme yang berbeda. Hasil penentuan keduanya dibandingan dan dianalisis berdasarkan parameter jarak baik yang dilakukan sistem ataupun perhitungan Google Maps secara manual. Validasi perbandingan jarak dengan Google Maps menghasilkan akurasi a star 99,56% dan floydwarshall 99,44%.

Kata kunci : *Algoritma A Atar, Floydwarsall , Artificial Inteligence, Rumah Sakit, Penentuan Rute Terpendek*

ABSTRACT

Artificial intelligence or commonly called AI (Artificial Intelligence) is a branch of science that can be applied in various fields, especially in determining the shortest route, especially if you experience an emergency such as an accident and so on, therefore mapping a hospital location is so important because become a place for everyone to go when in need of immediate medical help. So to help the public in knowing the location of the hospital, we need a system that can determine the route to the hospital location by utilizing a smartphone that is integrated with the Google Maps API which can help display which route is passed. In this study, the method used for route determination is the a star and floydwarshall algorithm, the algorithm works with a different mechanism, the difference will be compared and analyzed based on the accuracy of distance and time either by the system or manually, so that the shortest route from starting location to the final destination hospital.

Keywords: *Algorithm A Star, Floydwarsall , Artificial Inteligence, shortest route determination.*

BAB I

PENDAHULUAN

Pada bab ini dijelaskan mengenai latar belakang masalah, rumusan masalah, tujuan, batasan masalah, tinjauan pustaka, metodologi penelitian, dan sistematika penulisan.

1.1 Latar Belakang

Rumah sakit merupakan unit pelayanan kesehatan terkait dengan tanggung jawab memberikan pelayanan gawat darurat sebagaimana disebutkan dalam Undang undang RI no 44 tahun 2009 pasal 1, rumah sakit merupakan institusi pelayanan kesehatan yang menyelenggarakan pelayanan kesehatan perorangan secara menyeluruh, menyiapkan rawat inap, rawat jalan dan gawat darurat. Selanjutnya dalam Permenkes RI No.147/Menkes/ 2010 juga disebutkan bahwa rumah sakit merupakan institusi pelayanan kesehatan yang menyelenggarakan pelayanan kesehatan perorangan secara paripurna yang menyediakan pelayanan rawat inap, rawat jalan, dan gawat darurat. Unit gawat darurat terlaksana dengan adanya unit instalasi gawat darurat. Pelayanan pasien instalasi gawat darurat membutuhkan ketepatan dan kecepatan melakukan diagnosis penyakit sehingga cepat menentukan prioritas pasien yang harus segera mendapatkan tindakan medis dan terapi (R Retnaning, 2018). Keadaan gawat darurat yang sering terjadi di masyarakat adalah keadaan seseorang yang mengalami henti napas dan henti jantung tidak sadarkan diri, kecelakaan, cedera seperti patah tulang, pendarahan, stroke dan korban bencana alam. Kasus gawat darurat karena kecelakaan lalu lintas merupakan penyebab kematian utama didaerah perkotaan (Media Aesculapius, 2007). Pertolongan gawat darurat memiliki sebuah standar waktu pelayanan yang dikenal dengan istilah waktu tanggap (respon time) yaitu maksimal 10 menit. Waktu tanggap gawat darurat merupakan gabungan dari waktu tanggap saat pasien tiba didepan pintu rumah sakit sampai mendapat respon dari petugas instalasi gawat darurat dengan waktu pelayanan yang diperlukan pasien sampai selesai proses penanganan gawat darurat (Haryatun dan Sudaryanto, 2008). Maka dari itu dibutuhkan sebuah perancangan sistem penentuan rute terpendek dari lokasi pasien

ke unit gawat darurat sehingga dapat membantu lebih mudah dalam kebutuhan menentukan rute terpendek sehingga korban mendapatkan tindakan medis dengan segera dan meminimalisir terjadinya kondisi yang serius.

Penentuan rute terpendek telah diterapkan berbagai bidang untuk mengoptimasi kinerja suatu sistem baik untuk meminimalkan biaya ataupun mempercepat jalannya suatu proses. Pencarian rute terpendek termasuk dalam salah satu persoalan dalam teori graf yang berarti meminimalisasi bobot suatu lintasan dalam graf (Y. Rudi Kriswanto, 2014). Dengan banyaknya algoritma pencarian rute terpendek atau *shortest-path* tentu memberikan kita banyak pilihan dalam menentukan algoritma yang kita gunakan. Algoritma tersebut diantaranya adalah, algoritma *A Star* merupakan perbaikan dari metode *best-first search* dengan memodifikasi fungsi *heuristiknya*, yang menjadikan algoritma ini meminimalkan total biaya lintasan dan pada kondisi yang tepat akan memberikan solusi yang terbaik (Imam Ahmad, Wahyu Widodo 2017). Cara kerja dari algoritma *A** yaitu memperhitungkan semua kemungkinan lintasan yang ada, lalu kemudian membandingkan kemungkinan tersebut satu demi satu maka diperoleh lintasan sebagai lintasan terpendek (Johannes Ridho 2015). Sedangkan algoritma *Floydwarsall* merupakan salah satu varian dari pemrograman dinamis yaitu suatu metode yang melakukan pemecahan dengan memandang solusi yang akan diperoleh sebagai keputusan yang saling terkait. Artinya solusi-solusi tersebut terbentuk dari solusi yang berasal dari tahap sebelumnya dan ada kemungkinan solusi lebih dari satu (Friska Widya Ningrum & Tatyantoro Andrasto 2016). Cara kerja dari algoritma *Floyd Warshall* yaitu mencari dan kemudian membandingkan semua lintasan yang mungkin terjadi dalam *graf* pada setiap pasangan simpul dan melalukan pengujian dari kombinasi yang sudah diperoleh (Popa & Popescu, 2016),

Menanggapi dari latar belakang diatas mendorong untuk melakukan penelitian perbandingan dari algoritma penentuan rute terpendek, dalam penelitian ini menggunakan dua algoritma yaitu algoritma *A Star* dan algoritma *Floyd Warshall*. Lalu setelah itu akan dilakukan perbandingan dari kedua algoritma tersebut baik yang dilakukan dengan sistem maupun manual yang dipengaruhi oleh beberapa parameter atau aspek yang dibandingkan yaitu seperti jarak dan waktu tempuh

sehingga dapat mengetahui mana yang lebih efektif dalam penentuan rute terpendek dan *output* yang dihasilkan berupa menampilkan rute terpendek menuju lokasi tujuan.

Pada penelitian ini algoritma tersebut berfungsi untuk menentukan rute terpendek antara posisi *user* sebagai titik awal dan lokasi tujuan sebagai titik akhir, kemudian membandingkan hasil perhitungan dari kedua algoritma tersebut untuk mengetahui tingkat akurasi atau presisi baik yang dilakukan oleh sistem maupun perhitungan secara manual jika digunakan untuk penentuan rute terpendek dalam melakukan pengujinya, definisi akurasi sendiri merupakan pengukuran terhadap nilai yang dihasilkan mendekati nilai nyata atau benar sama dan presisi merupakan pengukuran yang mempunyai nilai hampir sama pada setiap pengukuran yang dilakukan. Selanjutnya untuk menentukan setiap *node* yaitu dengan memanfaatkan Google Maps API. Rumus persamaan *Euclidean* digunakan untuk memberikan bobot *heuristik* atau menghitung jarak antara lokasi user sebagai titik awal ke titik berikutnya dan persamaan *haversine formula* untuk nilai jarak perkiraan garis lurus antara titik awal dan titik akhir.

1.2 Rumusan Masalah

Unit gawat darurat membutuhkan ketepatan dan kecepatan melakukan diagnosis penyakit sehingga dapat menentukan prioritas pasien yang harus segera mendapatkan penanganan tindakan medis dan terapi jika tidak nyawa korban akan terancam atau mengalami kecacatan (R Retnaning, 2018). Hal ini mengakibatkan penentuan rute terpendek merupakan hal yang sangat penting untuk dilakukan ketika mengalami keadaan gawat darurat atau keadaan membutuhkan perjalanan yang sangat cepat permasalahan ini sering dialami oleh setiap orang kesulitan untuk memilih rute dikarenakan banyak faktor seperti jarak dan waktu tempuh.

Berdasarkan identifikasi yang telah diterapkan, maka muncul berbagai masalah yang akan ditemui sebagai berikut :

1. Bagaimana menentukan rute terpendek menggunakan algoritma *Floyd Warshall* dengan parameter seperti jarak dan waktu.
2. Bagaimana menentukan rute terpendek menggunakan algoritma *A Star* dengan parameter seperti jarak dan waktu.

3. Bagaimana tingkat akurasi atau presisi dari hasil algoritma *Floyd Warshall* dan *A Star* dalam menyelesaikan masalah penentuan rute terpendek baik yang dilakukan sistem ataupun secara manual.

1.3 Tujuan

Tujuan dari penelitian ini adalah untuk dapat menerapkan dan membandingkan hasil jarak dari antara dua algoritma yaitu *Floyd Warshall* dan algoritma *A Star* dalam menyelesaikan masalah penentuan rute terpendek (*shortest path*) pada unit gawat darurat baik yang diterapkan pada sistem maupun secara manual, sehingga dapat mengetahui akurasi serta hasil rute dari perbandingan kedua algoritma tersebut berdasarkan beberapa faktor parameter yaitu jarak dan waktu tempuh untuk mencapai lokasi dari titik asal ke titik tujuan.

1.4 Ruang Lingkup

Berdasarkan latar belakang yang telah diuraikan sebelumnya, maka dibuatlah ruang lingkup agar pada saat penelitian tugas akhir ini dapat lebih jelas cakupannya. Adapun ruang lingkup yang dibuat adalah sebagai berikut :

- a. Pemetaan titik sebagai tujuan hanya diambil 27 rumah sakit di wilayah kota Bandung
- b. Penelitian ini dibatasi pada pengembangan aplikasi *smartphone android*.
- c. Sistem yang dibangun adalah aplikasi untuk digunakan oleh sisi pelapor dan admin
- d. Data rumah sakit yang digunakan hanya 27 rumah di wilayah kota Bandung
- e. Penentuan rute terpendek menggunakan *Floydwarsall* dan *A Star* dengan parameter jarak dan waktu tempuh.
- f. Pemetaanrute dan menghubungkan antar titik dibantu dengan menggunakan fasilitas *google maps api*.
- g. Sistem yang dibuat tidak membahas kondisi lalu lintas.
- h. Sistem dibangun dalam bentuk simulasi.
- i. Sistem hanya menggunakan dua pilihan rute yang dapat dilalui.

1.5 Metode Penelitian

Sebagai faktor pendorong untuk keberhasilan di dalam penelitian ini maka digunakanlah pemodelan perangkat lunak berbasis *prototype*. Model penelitian ini

dipilih karena memiliki struktural yang sesuai dengan pembuatan sistem yang akan dibuat. Model ini bersifat berulang yang dimana memungkinkan perbaikan sistem sampai sistem tersebut sesuai dengan rencana diawal pembuatan.

Prototyping perangkat lunak (software prototyping) atau siklus hidup menggunakan prototyping (life cycle using prototyping) adalah salah satu metode siklus hidup sistem yang didasarkan pada konsep model bekerja (working model). Tujuannya adalah mengembangkan model menjadi sistem final. Artinya sistem akan dikembangkan lebih cepat dari pada metode tradisional dan biayanya menjadi lebih rendah (Wanda, Rizal Loa.,2015). Untuk memodelkan sebuah perangkat lunak dibutuhkan beberapa tahapan di dalam proses pengembangannya. Tahap inilah yang akan menentukan keberhasilan dari sebuah sistem yang dibangun.

Pada kasus penelitian ini terdapat objek penelitian yang digunakan yaitu layanan kesehatan seperti rumah sakit yang dikerucutkan hanya 27 saja di wilayah kota Bandung karena rumah sakit di kota bandung sangat banyak sehingga terlalu kompleks dapat memakan lebih banyak waktu. Dengan menggunakan algoritma *Floyd warshall* dan *A Star* ini bertujuan untuk menemukan atau menentukan rute terpendek berdasarkan bobot terkecil dari satu titik ke titik lainnya. Proses pertama membuat graph dan node dengan cara menandai langsung pada maps berdasarkan *latitude* dan *longitude*, menentukan titik layanan rumah sakit yang akan menjadi *node* tujuan.

Tiap *node* tersebut akan terhubung sehingga membentuk lintasan. Penghubung antar node ini memiliki bobot berdasarkan panjang jarak antara node *a* dan node *b*. Tahap selanjutnya mendapatkan nilai koordinat dari lokasi *user* yaitu *longitude* dan *latitude* yang dijadikan titik awal pada proses ini akan menggunakan teknologi *GPS Tracking* pada smarthphone yang bertujuan untuk mendapatkan titik lokasi koordinat *user*. Kemudian digunakan rumus persamaan *Haversine* untuk menghitung jarak *real* atau perkiraan dengan garis lurus antara dua titik dimana titik awal *user* dan titik tujuan layanan rumah sakit yang akan ditampilkan oleh sistem. Tahap berikutnya melakukan perhitungan penentuan rute menggunakan algoritma *Floyd warshall* dan *A Star*, lalu yang dihasilkan ditampilkan oleh sistem berbentuk visualisasi peta.

1.6 Tinjauan Pustaka

Sistem yang akan dibangun mengacu pada penelitian yang dilakukan oleh peneliti terdahulu.

(Anugrah *et al.*, 2017) Penelitian yang dilakukan adalah *Simulasi Algoritma A* dan Djiktra pada WAN*. Penelitian ini menjelaskan tentang. Menerapkan graf pada algoritma A* dan Djiktra yang digunakan pada routing protocol sehingga dapat mengetahui dan membandingkan proses komputasi dalam pencarian rute terpendek pada graf dengan algoritma Dijkstra dan algoritma A* sehingga dapat mengurangi beban kerja pada jaringan. Kontribusi pada penelitian ini adalah untuk memahami proses komputasi pada Algoritma A*.

(Syukriyah *et al.*, 2016) Penelitian yang dilakukan adalah *Penerapan Algoritma A*(star) untuk mencari rute tercepat dengan hambatan*. Penelitian ini menjelaskan tentang mempelajari cara kerja *algoritma A** dengan menggunakan fungsi *heuristic* dalam mencari jarak tercepat, dengan cara disimulasikan seperti kondisi ketika seorang mencari rute dalam keadaan jalanan macet. Kontribusi pada penelitian ini adalah untuk mempelajari cara kerja Algoritma A* dan fungsi *heuristic* dalam pencarian rute terpendek.

(Eka Yulia *et al.*, 2015) Penelitian yang dilakukan adalah *Pencarian SPBU terdekat pada penelitian ini menggunakan algoritma Dijkstra*. Penelitian ini menjelaskan tentang bagaimana cara proses pemetaan lokasi, pencarian lokasi dan penentuan jalur terpendek sehingga didapatkan rekemendasi jalur terpendek dengan jarak tempuh yang berguna memberikan efisien jalur yang terpilih. Kontribusi pada penelitian ini adalah untuk mempelajari cara proses pemetaan lokasi, pencarian lokasi, dan penentuan jalur terpendek.

(Rute *et al.*, 2017) Penelitian yang dilakukan adalah *Pemilihan rute terbaik menggunakan algoritma djikstra untuk mengurangi kemacetan lalu lintas di purwokerto*. Penelitian ini menjelaskan tentang cara menentukan jalur – jalur efektif dan efisien dengan berdasarkan bobot terkecil pada setiap masing-masing ruas jalan yang dapat dilalui oleh pengendara. Kontribusi pada penelitian ini adalah untuk memahami cara menentukan jalur yang efektif dan efisien berdasarkan bobot

terkecil pada setiap masing-masing ruas jalan sehingga dapat mengetahui jalur mana yang dapat dilalui oleh pengendara.

(Anam et al., 2016) Penelitian yang dilakukan adalah *Pencarian rute terbaik menggunakan logika fuzzy dan algoritma semut* Penelitian ini menjelaskan tentang cara menyelesaikan permasalahan pencarian rute terbaik dengan menggunakan logika fuzzy dan algoritma semut dengan membangun sebuah graf berbobot. Kontribusi pada penelitian ini adalah untuk mempelajari sistem pengambilan keputusan dalam menentukan rute.

(Satria Dahni, 2017) Penelitian yang dilakukan adalah *Sistem Informasi Penentuan Jalur Terpendek Bagi Pengantar Surat Menggunakan Algoritma Semut* Penelitian ini menjelaskan tentang proses pencarian rute pengantaran surat bagi kurir dan kurir dapat menemukan lokasi kantor dengan hanya melihat sebuah sistem informasi jalur pengantar surat. Kontribusi pada penelitian ini adalah memahami konsep sistem informasi yang diterapkan pada penelitian tersebut.

(Azan Cahyadi et al., n.d.) Penelitian yang dilakukan adalah *Perbandingan Algortima A*, Djiktra,dan Floyd Warshall untuk menentukan jalur terpendek pada permainan BACTERIA DEFENSE*. Penelitian ini menjelaskan tentang cara menerapkan Algoritma A, Djiktra dan Floyd Warshall pada simulasi sebuah permainan yang kemudian menganalisis dari ketiga algoritma tersebut, sehingga dapat mengetahui algoritma mana yang bekerja lebih efektif diantara ketiga algoritma tersebut dalam menentukan jalur terpendek berdasarkan parameter yang digunakan. Kontribusi pada penelitian ini adalah sebagai referensi dalam proses membandingkan algoritma pada sebuah simulasi permainan.

(Syafiq et al., 2016) Penelitian yang dilakukan adalah *Pemanfaatan Google Maps API untuk pencarian jalur lokasi SPBU terdekat di Kota Jepara dan Kudus dengan Teknologi Node-Js*. Penelitian ini menjelaskan tentang bagaimana penggunaan Google Maps API dalam menampilkan atau memetakan data lokasi SPBU tersebut. Kontribusi pada penelitian ini adalah memahami dan mempelajari dari cara kerja Google Maps API.

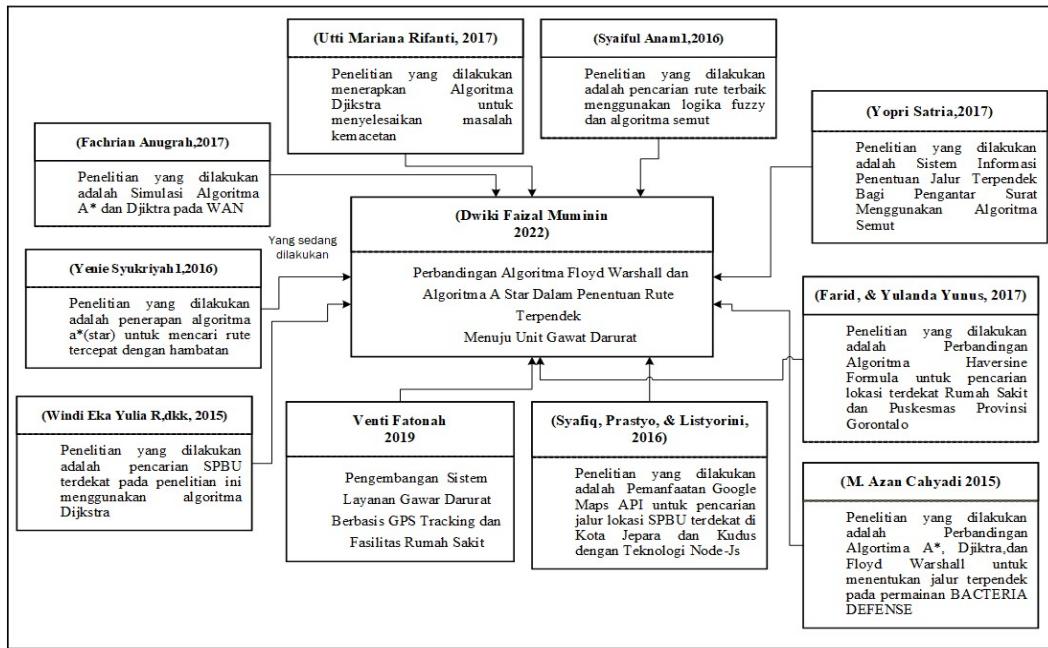
((Farid Dan Yulanda 2017) *Analisa Algoritma Haversine Formula Untuk Pencarian Rute Terdekat Rumah Sakit*, n.d.) Penelitian yang dilakukan adalah

Analisa Haversine Formula untuk pencarian lokasi terdekat Rumah Sakit dan Puskesmas Provinsi Gorontalo. Penelitian ini menjelaskan tentang proses penerapan *Euclidean Distance Formula* dalam menghitung jarak berdasarkan panjang garis lurus antara dua titik kelengkungan dibumi yaitu *Latitude* dan *Longitude*. Kontribusi pada penelitian ini adalah memahami garis lurus antara dua titik yang dapat diimplementasikan.

(Venti Fatonah, 2019) Penelitian yang dilakukan adalah *Pengembangan Sistem Layanan Gawat Darurat Berbasis GPS Tracking dan Fasilitas Rumah Sakit* Penelitian ini menjelaskan tentang sistem pengembalian keputusan dengan mendiagnosa penyakit terlebih dahulu kemudian melakukan proses pencarian pencarian rute terdekat menuju rumah sakit. Kontribusi pada penelitian ini adalah sebagai referensi atau bahan pengembangan penelitian.

Berdasarkan tinjauan pustaka, diambil suatu kesimpulan dari penelitian sebelumnya terdapat banyak algoritma yang dapat digunakan atau diimplementasikan dalam penentuan rute terpendek maka dari itu Dwiki Faizal M, Uung Ungkawa, Dr.,Ir.,M.T., melakukan beberapa pengembangan terhadap penelitian sebelumnya serta melakukan perbandingan antara dua algoritma yang digunakan sehingga penelitian ini yang berjudul *Perbandingan Algoritma Floyd Warshall dan Algoritma A Star Dalam Penentuan Rute Terpendek Menuju Unit Gawat Darurat*.

Adapun ringkasan dari beberapa penelitian yang telah dibahas sebelumnya, dapat dilihat pada Gambar 1.



Gambar 1 Peta Penelitian

1.7 Kontribusi Penelitian

Penelitian ini adalah penelitian yang dapat membantu seseorang dalam menentukan rute terpendek menuju rumah sakit khususnya saat mengalami keadaan darurat sehingga membutuhkan pertolongan secepatnya kepada korban dengan cara melaporkan kejadian kemudian memilih rumah sakit tujuan sehingga dapat mempermudah petugas rumah sakit dalam menangani korban. Selain itu sistem akan membantu saat dalam penentuan rute, dan dapat mengetahui rute mana yang efisien untuk dilalui dari hasil membandingkan dua algoritma yang diimplementasikan sehingga dapat menuju ke titik tujuan dengan cepat. Maka dari itu, penelitian ini berkontribusi pada program *smart city* dibidang *Smart Government*.

1.8 Sistematika Penulisan

Sistematika penulisan yang digunakan untuk memberikan gambaran isi dari laporan ini dijelaskan sebagai berikut :

BAB I PENDAHULUAN

Bab ini menjelaskan mengenai latar belakang masalah, rumusan masalah, batasan masalah, tujuan, batasan masalah, tinjauan pustaka, metodologi penelitian, dan sistematika penulisan.

BAB II LANDASAN TEORI

Bab ini menjelaskan tentang teori-teori yang mendukung dalam melakukan penelitian yang dilakukan serta menjelaskan konsep-konsep yang diperlukan untuk menjelaskan permasalahan yang ada dalam penelitian.

BAB III METODE PENELITIAN

Pada bagian bab ini akan dipaparkan metode yang digunakan dalam penelitian, uraian perancangan dari penelitian yang diusulkan yang dimulai dari alur proses sistem, metode *floyd warshall*, metode *a star*, hingga perancangan antarmuka.

BAB IV IMPLEMENTASI DAN PENGUJIAN

Dalam bab ini akan dipaparkan hasil dari rancangan yang diajukan. Pada bagian ini akan diperlihatkan hasil tampilan dan rincian dari pembangunan sistem serta penggunaan dari sistem yang telah selesai dibuat serta menampilkan hasil evaluasi terhadap pengujian yang telah dilakukan.

BAB V PENUTUP

Pada bab ini berisikan mengenai kesimpulan dan hasil penelitian.

BAB II

LANDASAN TEORI

Pada bab ini dijelaskan tentang teori – teori yang mendukung dalam melakukan penelitian yang dilakukan serta menjelaskan konsep-konsep yang diperlukan untuk menjelaskan permasalahan yang ada dalam penelitian.

2.1 Rumah Sakit

Menurut Undang-undang Indonesia Nomor 44 Tahun 2009. Rumah sakit adalah sebuah institusi perawatan kesehatan yang menyediakan tempat untuk pasien rawat inap dalam jangka waktu tertentu. Rumah sakit biasanya didirikan berdasarkan wilayah oleh suatu organisasi/lembaga kesehatan (baik profit maupun non-profit), badan asuransi maupun badan aman termasuk donator secara langsung bahkan organisasi keagamaan individu atau yayasan (Fajri, Harris, & Widyawati, 2018). Klasifikasi Rumah sakit umum memiliki berbagai macam tipe berdasarkan kemampuannya diantaranya :

1. Rumah sakit Tipe A, merupakan pelayanan kesehatan rujukan tertinggi alias pusat. Rumah sakit yang juga disebut rujukan fasilitas kesehatan (faskes) tingkat tiga ini memberikan pelayanan yang lebih lengkap mulai dari yang umum, subspesialis hingga kedokteran spesialis oleh pihak pemerintah (Aafi Ramadhania Pasha, 2020).
2. Rumah sakit Tipe B, merupakan pelayanan yang diberikan kedokteran medis spesialis luas dan subspesialis terbatas. Rumah sakit ini juga akan dijadikan sebagai pelayanan kesehatan rujukan dari setiap rumah sakit kabupaten (Aafi Ramadhania Pasha, 2020).
3. Rumah sakit Tipe C, merupakan pelayanan faskes tingkat dua ini memberikan pelayanan hanya kedokteran subspesialis, namun sifatnya juga terbatas, misalnya saja pelayanan penyakit dalam, bedah, kesehatan anak, bidan dan kandungan (Aafi Ramadhania Pasha, 2020).
4. Rumah sakit Tipe D, merupakan pelayanan yang hanya sebagai rumah sakit sementara atau transisi. Biasanya, jika pasien yang awalnya melakukan pemeriksaan di puskesmas kemudian dirujuk ke rumah sakit tipe D.

5. Namun, jika dilihat pasien membutuhkan penanganan yang lebih lanjut maka rumah sakit Tipe D ini akan membuat surat rujukan ke faskes yang lebih tinggi (Aafi Ramadhania Pasha, 2020).
6. Rumah sakit Tipe E, merupakan pelayanan yang dikhkususkan hanya memberikan satu pelayanan kesehatan saja, misal khusus jantung,paru, ibu dan anak dan sebagainya. Rumah sakit ini juga tidak tersedia banyak di Indonesia (Aafi Ramadhania Pasha, 2020).

2.2 Pencarian Rute Terpendek

Pencarian rute terpendek merupakan suatu masalah yang paling banyak dibahas dan dipelajari sejak akhir tahun 1950. Pencarian rute terpendek ini telah diterapkan di berbagai bidang untuk mengoptimasi kinerja suatu sistem, baik untuk meminimalkan biaya atau mempercepat jalannya suatu proses. Salah satu aplikasi pencarian rute terpendek yang paling menarik untuk dibahas adalah pada masalah transportasi. Dalam pencarian rute terpendek, penghitungan dapat dilakukan dengan beberapa macam algoritma. Secara garis besar algoritma penghitungan rute terpendek dibagi menjadi dua kelas berdasarkan metode pemberian labelnya, yaitu algoritma label setting dan algoritma label correcting. Metode label setting menentukan label jarak sebagai jarak permanen pada setiap iterasinya, sedangkan metode label correcting menentukan label jarak sebagai temporal pada setiap iterasi sampai langkah akhir ketika semua node telah melewati proses pemeriksaan, maka labelnya akan ditentukan sebagai permanen. (Ramadhan, 2013).

2.3 *Google Maps API*

Google Maps API adalah sebuah layanan (*service*) yang diberikan oleh Google kepada para pengguna untuk memanfaatkan *Google Maps* dalam mengembangkan aplikasi. *Google Maps API* menyediakan beberapa fitur untuk memanipulasi peta dan menambah konten melalui berbagai jenis pelayanan yang dimiliki, serta mengijinkan kepada pengguna untuk membangun aplikasi enterprise di dalam websitenta (Widodo,2018). Sebelum para pengguna menggunakan *API* yang telah disediakan para pengguna harus memiliki *key* yang telah diberikan oleh *google* saat melakukan pendaftaran di laman <https://console.cloud.google.com>.

Selain itu *Google Maps API* menyediakan 15 *library maps* yang dapat digunakan oleh para pengembang diantaranya yaitu :

- *Google Maps Direction API*, digunakan untuk pencarian rute antara 2 titik lokasi.
- *Google Maps Places API*, digunakan untuk pencarian tempat pada suatu wilayah.
- *Distance Matrix API*
- *Geocoding API*
- *Geolocation API*
- *Maps Elevation API*
- *Maps Embed API*
- *Maps JavaScript API*
- *Maps SDK for Android*
- *Maps SDK for iOs*
- *Maps Static API*
- *Roads API*
- *Street View Publish API*
- *Street View Static API*

2.4 GPS (Global Positioning System)

GPS adalah sebuah sistem yang mampu memberikan sebuah lokasi secara tepat di bumi, kapanpun, dalam kondisi apapun dan dimanapun (Rizkidniah, Yamin, & Muchlis, 2016). GPS yang telah dibangun dipandu oleh sekitar 24 satelit yang ditempatkan dalam orbit tertentu.

Satelit tersebut di-monitor secara berkesinambungan melalui ground station yang ada diseluruh bagian bumi. Satelit mentransmisikan sinyal yang dapat ditangkap oleh siapapun yang dilengkapi dengan GPS Receiver. Dengan receiver inilah kita dapat mengetahui secara tepat dimana lokasi kita. Sementara itu dalam proses pengembangan sistem, tidak jarang menghadapi keraguan mengenai efektifitas, efisiensi dan kualitas algoritma yang sedang dikembangkan kemampuan adaptasi sistem terhadap sistem operasinya atau tampilan yang sedang dirancangnya.

2.5 Algoritma

Algoritma merupakan dasar yang harus dikuasai dan dipahami oleh seorang atau pelaku yang akan menyelesaikan permasalahan dengan komputer, dalam hal ini dengan membuat program. Sebuah program hakikatnya dibangun dengan rangkaian instruksi secara terstruktur sehingga bila dijalankan akan menghasilkan output yang diharapkan (Sitorus, 2015). Menurut Donald E. Knuth algoritma harus memenuhi persyaratan :

1. Finiteness : Algoritma harus berakhir (Terminate) setelah melakukan sejumlah langkah proses.
2. Definiteness : Tidak menimbulkan makna ganda (Ambiguous).
3. Input : Setiap algoritma memerlukan data sebagai masukan untuk diolah.
4. Output : Setiap algoritma memberikan satu atau beberapa hasil keluaran.

2.6 Algoritma *Floyd Warshall*

Algoritma *Floyd Warshall* adalah sebuah pemograman dinamis yang diperkenalkan oleh Robert Floyd pada tahun 1962. Tetapi pada dasarnya memiliki kesamaan yang pernah diperkenalkan sebelumnya Bernard Roy pada tahun 1959 dan juga Stephen Warshall pada tahun 1962. Selain itu metode ini melakukan pemecahan masalah dengan solusi yang akan diperoleh sebagai suatu keputusan yang saling terkait, dan solusi-solusi tersebut dibentuk dari solusi yang berasal dari tahap sebelumnya dan kemungkinan solusinya lebih dari satu (Friska Widya Ningrum & Tatyantoro Andrasto 2016).

2.7 Algoritma *A Star*

Algoritma *A Star* merupakan algoritma pencarian rute terpendek yang pertama kali dideskripsikan pada tahun 1968 oleh Peter Hart, Nils Nilsson, dan Bertram Raphael. Algoritma ini dinamakan algoritma A1. Pada tahun 1967 Bertram Raphael mengembangkan lebih jauh algoritma ini dan menyebutnya A2, namun tidak dapat membuktikan keunggulannya dibandingkan algoritma sebelumnya. Kemudian pada tahun 1968 Peter E menunjukkan bukti keoptimalan algoritma A2 dibandingkan A1. Kemudian algoritma A2 dinyatakan sebagai algoritma paling optimal untuk kasus tersebut dan diganti namanya menjadi A* (Naufal

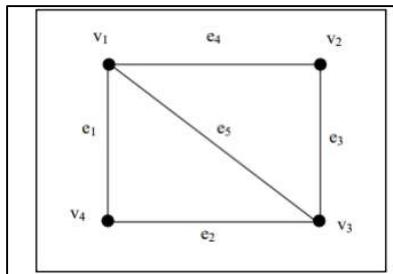
Wafiqurrahman, 2015). Berdasarkan algoritma ini lebih baik daripada algoritma Dijktra dengan pencarian heuristiknya.

Algoritma *A Star* menggunakan dua senarai yaitu *OPEN* dan *CLOSED*. *OPEN* adalah senarai (list) yang digunakan untuk menyimpan simpul-simpul yang pernah dibuat dan nilai heuristiknya telah dihitung tetapi belum terpilih sebagai simpul terbaik (*best node*) dengan kata lain *OPEN* berisi simpul-simpul yang masih memiliki peluang sedangkan *CLOSED* adalah senarai untuk menyimpan simpul-simpul yang sudah pernah dibuat dan sudah pernah terpilih sebagai simpul terbaik (peluang untuk terpilih sudah tertutup) (Ida Bagus Gede Wahyu, 2018).

2.8 Graph

Suatu graph sederhana G adalah suatu pasangan terurut (V, E) , dimana V adalah suatu himpunan berhingga yang tidak kosong yang elemen-elemennya disebut vertex dan E adalah suatu himpunan garis yang menghubungkan dua elemen subset dari E yang disebut edges (Iryanto, 2003). Contoh dapat dilihat pada

Gambar 4 .



Gambar 2 Graph dengan 4 verteks 5 edges

Pada contoh diatas graph $G = (V, E)$ dimana :

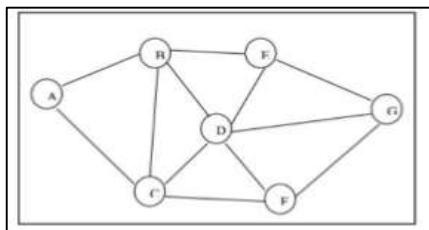
1. V adalah himpunan titik, simpul, vertex atau nodes dari G , yaitu $V = \{v_1, v_2, v_3, v_4\}$.
2. E adalah himpunan rusuk, edges atau sisi dari G , yaitu $E = \{e_1, e_2, e_3, e_4, e_5\}$.

2.8.1 Macam-macam Graph Menurut Arah dan Bobotnya

Menurut (Iryanto. 2003) macam-macam graph menurut arah dan bobotnya, graph dibagi menjadi empat bagian, yaitu:

1. Graph berarah (digraph) dan berbobot : setiap edges mempunyai arah (yang ditunjukkan dengan anak panah) dan bobot. Gambar 5 adalah contoh graph berarah dan berbobot, yang terdiri dari tujuh verteks yaitu

verteks A, B, C, D, E, F, G dan 12 edges. Verteks A mempunyai dua edges yang masing-masing menuju ke verteks B dan verteks C, verteks B mempunyai tiga edges yang masing-masing menuju ke verteks C, verteks D dan verteks E dan seterusnya. Tiap-tiap edges mempunyai arah dan bobot yang telah diketahui.



Gambar 3 Graph berarah dan beraturan

2.9 Lintasan

Lintasan yang panjangnya n dari simpul awal v_0 ke simpul tujuan v_n di dalam graf G ialah barisan berselang-seling simpul-simpul dan sisi-sisi yang berbentuk $v_0, e_1, v_1, e_2, v_2, \dots, v_{n-1}, e_n, v_n$ sedemikian sehingga $e_1 = (v_0, v_1)$, $e_2 = (v_1, v_2)$, \dots , $e_n = (v_{n-1}, v_n)$ adalah sisi – sisi dari graf G (Rinaldi Munir, 2003). Jika graf yang ditinjau merupakan graf sederhana, maka lintasan cukup dituliskan sebagai barisan simpul: $v_0, v_1, v_2, \dots, v_{n-1}, v_n$, karena antara dua buah simpul yang berurutan dalam lintasan tersebut hanya terdapat satu sisi. Jika graf yang ditinjau memiliki sisi ganda, maka, lintasan ditulis sebagai barisan berselang-seling antara simpul dan sisi: $v_0, e_1, v_1, e_2, v_2, e_3, \dots, v_{n-1}, e_n, v_n$. Simpul dan sisi yang dilalui di dalam lintasan boleh berulang. Sebuah lintasan yang semua simpulnya berbeda (setiap sisinya dilalui hanya sekali) dikatakan lintasan sederhana.

Misalkan G adalah graf berarah berlabel dengan titik-titik $V(G) = \{v_1, v_2, \dots, v_n\}$ dan path terpendek yang dicari adalah dari v_1 ke v_n . Algoritma Dijkstra dimulai dari titik v_1 . Dalam iterasinya, algoritma akan mencari satu titik yang jumlah bobotnya dari titik 1 terkecil. Titik-titik yang terpilih dipisahkan, dan titik-titik tersebut tidak diperhatikan lagi dalam iterasi berikutnya (Rendio Halda, 2016).

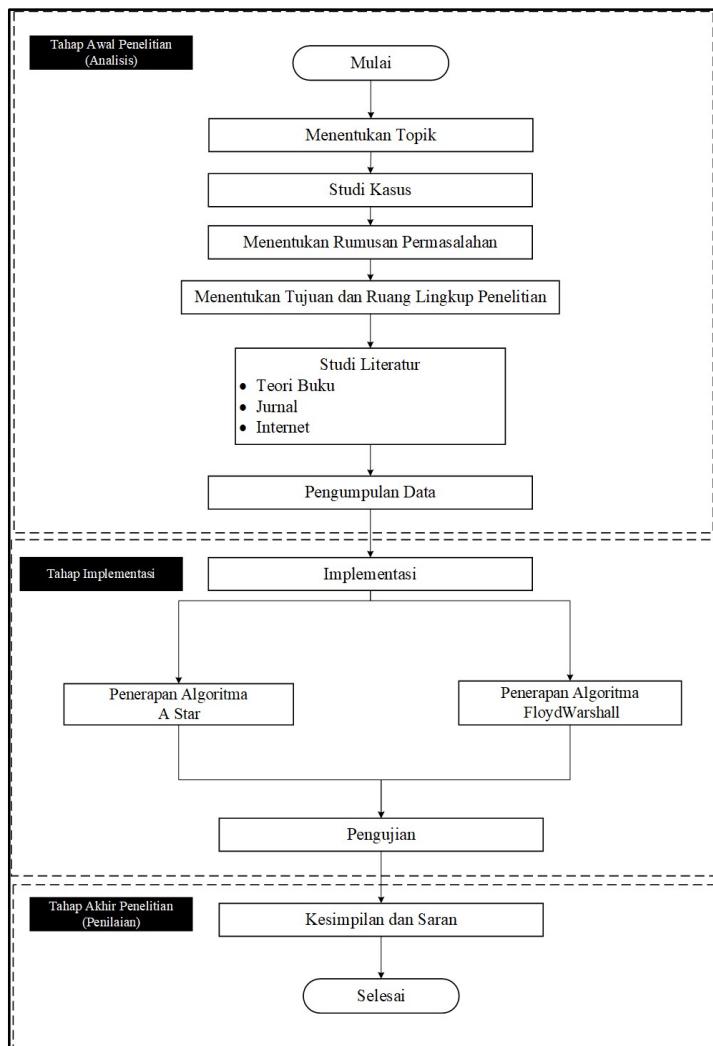
BAB III

METODE PENELITIAN

Pada bab ini menjelaskan tentang bagaimana tahapan – tahapan alur penelitian dari metode penelitian yang dilakukan untuk membangun sistem akan dijelaskan dalam bentuk skema alur penelitian sebagai berikut.

3.1 Alur Peneltiain

Alur penelitian merupakan sebuah langkah pelaksanaan penelitian atau skripsi yang secara sistematis, supaya mudah dipahami alur penelitian akan dijabarkan dengan menggunakan *flow chart diagram*.



Gambar 4 Alur penelitian

3.1.1 Tahap awal penelitian (analisis)

Tahap awal penelitian ini adalah tahapan paling awal dilakukan dalam penelitian yang merupakan identifikasi suatu masalah pada suatu penelitian. Pada tahap ini biasanya dilakukan berdiskusi dengan pembimbing. Berikut merupakan tahap - tahap dalam identifikasi :

a. Menentukan Topik

Merupakan tahapan awal yang biasanya dilakukan pada saat akan melaksanakan penelitian, terlebih dahulu menentukan topik apa yang akan diambil atau digunakan sesuai dengan bidang kemudian mencari referensi sebanyak mungkin dari penelitian sebelumnya atau yang sudah dilakukan. Topik yang diambil dalam penelitian ini adalah perbandingan algoritma Floyd warshall dan A star pada penentuan rute terpendek pada layanan gawat darurat.

b. Studi Kasus

Merupakan proses tahapan pengamatan secara langsung yang dilakukan terhadap fenomena atau isu permasalahan yang terjadi disekitar sehingga dapat mendapatkan solusi dari permasalahan tersebut berhubungan atau sesuai dengan bidang dan topik penelitian ini yang sudah kita ambil atau tentukan akan diberikan pada Bab III.

c. Menentukan Perumusan Permasalahan

Merupakan proses tahapan mengidentifikasi permasalahan masalah – masalah apa saja yang terdapat pada sebuah peristiwa,fenomena atau kejadian yang sesuai dengan bidang dan topik penelitian ini yang telah diambil atau ditentukan sudah diberikan pada Bab I.

d. Menentukan Tujuan dan Ruang Lingkup Penelitian

Setelah melakukan perumusan masalah langkah selanjutnya adalah penentuan tujuan penelitian untuk menentukan langkah langkah yang akan dilaksanakan selanjutnya guna mencapai tujuan penelitian pada bidang dan topik yang telah diambil atau ditentukan. Sedangkan ruang lingkup penelitian untuk, memberikan batasan dan asumsi-asumsi yang jelas dalam pelaksanaan dan penyusunan penelitian sehingga penelitian tersebut tidak melebar keluar dari

ruang lingkup penelitian ini yang telah diambil atau ditentukan sudah dilakukan pada Bab I.

e. Studi Literatur

Merupakan tahapan proses cara penelusuran literatur untuk menghimpun sumber-sumber yang berhubungan dengan topik penelitian yang bersumber dari sumber yang menunjang seperti buku, media, jurnal, internet ataupun referensi yang relevan dari penelitian orang lain terdahulu yang berhubungan dengan topik penelitian. Studi literatur bertujuan untuk menyusun dasar teori atau sebagai landasan teori yang digunakan dalam penelitian ini dan menyelesaikan permasalahan secara ilmiah sudah dilakukan pada Bab I.

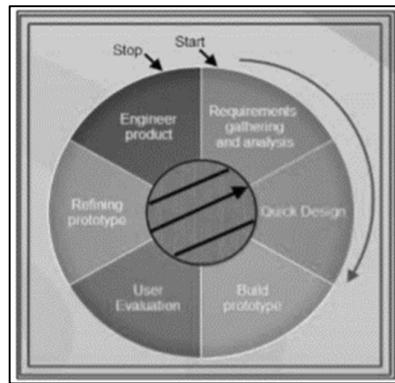
f. Pengumpulan Data

Merupakan tahapan yang dilakukan dalam pengumpulan data yang berhubungan dengan penelitian. Pengumpulan data terdiri dari data primer dan skunder. Data primer yaitu melakukan dengan cara pengamatan dan pencatatan secara langsung pada hal yang berhubungan langsung dengan penelitian ini bisa seperti orang yang dijadikan sebagai sumber mendapatkan informasi ataupun data contohnya wawancara dan sebagainya. Sedangkan data skunder yaitu melakukan pengumpulan data dengan secara tidak langsung namun mengumpulkan sumber data yang tertulis seperti buku-buku, artikel, halaman web, jurnal, makalah, dan laporan penelitian terdahulu serta sumber lain yang berkaitan dengan masalah penelitian atau bisa dalam bentuk apapun dilakukan pada Bab III.

3.1.2 Tahap Implementasi

Merupakan tahapan proses pelaksanaan pengembangan atau penerapan terhadap rancangan sistem yang akan dibuat pada penelitian, serta yang telah disusun dan akan dilakukan secara cermat dan rinci, tahap implementasi dilakukan dengan tujuan untuk mengetahui fungsi dari metode atau algoritma yang akan digunakan terhadap penelitian. Tahapan ini digunakan pada pengembangan sistem ini mengimplementasikan pembangunan sistem dengan metode prototype yakni metode untuk pengembangan software. Proses pemodelan ini bersifat berulang dengan perencanaan dimana terdapat umpan balik yang memungkinkan terjadinya perbaikan sistem sampai sistem tersebut sesuai dengan rencana awal. Untuk

memodelkan sebuah perangkat lunak dibutuhkannya beberapa tahapan proses pengembangannya. Dapat dilihat pada *Gambar* merupakan tahapan dari metode *prototype*.



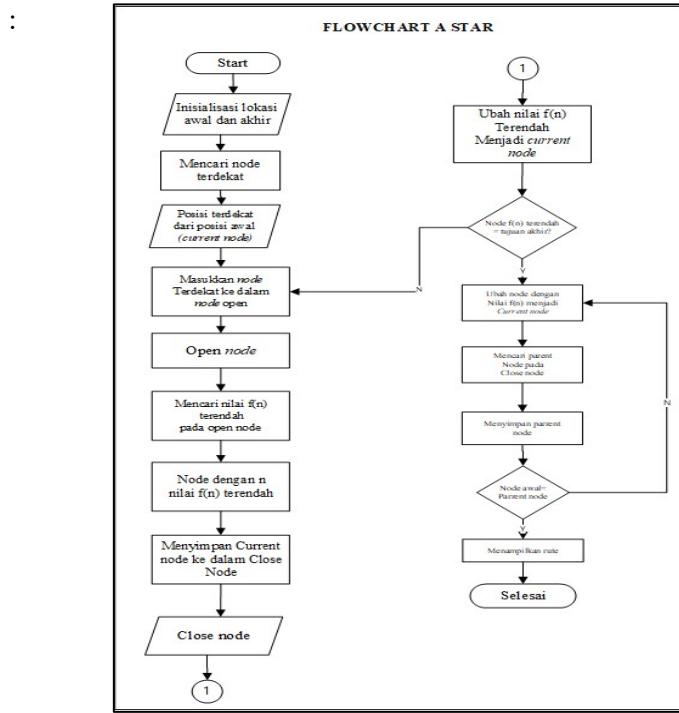
Gambar 5 Pengembangan perangkat lunak prototype

Tahap ini menjelaskan tahapan perancangan suatu sistem penentuan rute terpendek menuju layanan gawat darurat yang dipilih. Pada sistem ini *user* melakukan penginputan koordinat posisi *user* dengan mengaktifkan GPS pada *smartphone* yang digunakan, kemudian dikirimkan ke rumah sakit yang dipilih oleh *user*. Sistem dapat menampilkan berupa daftar list jarak *real* atau perkiraan dengan garis lurus antara dua titik rumah sakit terdekat yang akan dipilih, apabila rumah sakit sudah dipilih sistem akan menentukan dan menampilkan rute yang akan dilalui oleh *user* beserta dengan informasi jarak dan waktu tempuh.

Dalam proses menentukan rute terpendek sistem ini akan menggunakan atau menerapkan sebuah dua algoritma dalam waktu dan sistem yang sama pada penentuan rute terpendek, tentunya banyak berbagai macam algoritma dalam memecahkan masalah menentukan rute akan tetapi pada penelitian ini menggunakan dua algoritma *A Star* dan *Floyd Warshall*.

a. Penerapan Algoritma *A Star*

Merupakan tahapan melakukan proses penerapan algoritma bagaimana langkah berjalannya algoritma yang digunakan pada penelitian. Tahapan tersebut dapat dilihat pada *Gambar* sebagai berikut



Gambar 6 Flowchart algoritma a star

Langkah pertama yang dilakukan yaitu pencarian jalur menentukan posisi awal dan akhir, posisi awal berupa koordinat *longitude* dan *latitude* posisi terkini sebagai node *user* dengan menggunakan *GPS (Global Positions System)*, posisi akhir sebagai node *goal* berupa koordinat *longitude* dan *latitude* lokasi tujuan sebagai titik akhir.

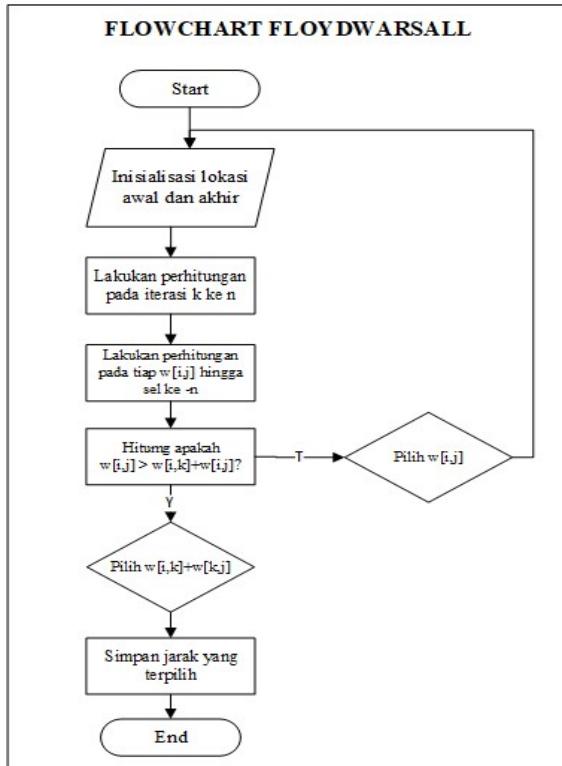
1. Tahap pertama melakukan pencarian *node* terdekat dengan posisi lokasi awal *node user* yaitu dengan menghitung jarak dari posisi titik awal ke setiap *node* persimpangan, *goal* dan jarak yang terkecil dipilih sebagai antrian *current node*. Perhitungan jarak untuk nilai heuristic menggunakan fungsi *euclidiean distance*.
2. *Node* yang terdekat dengan posisi awal dengan jarak terkecil, masuk pada antrian *current node*.
3. Neighboar *node* merupakan *node* yang saling terhubung.
4. *Node* yang saling terhubung tersebut masuk ke pada antrian *Open node* atau *Open list*, *open list* adalah list yang menyimpan kemungkinan path yang akan diperiksa. *open list* dibuat terurut berdasarkan nilai f. *open list* digunakan untuk menentukan secara selektif (berdasarkan nilai f) jalan yang dikira lebih dekat

menuju path tujuan. *open* berisi simpul-simpul yang masih memiliki peluang untuk terpilih sebagai simpul terbaik.

5. *Current Node* sama dengan = posisi akhir, jika tidak maka akan melakukan kembali proses pengulangan ke nomer 5, jika ya lanjut ke tahap proses selanjutnya dan akan dimasukkan ke dalam *closed list* bahwa *node* tersebut merupakan sebagai simpul atau node terbaik.
6. Menghitung nilai $f(n)$ dengan rumus $f(n) = g(n) + h(n)$. $g(n)$ yaitu bobot atau jarak antar *node*. Sedangkan $h(n)$ yaitu nilai *heuristic* atau nilai pertimbangan antar *node* yang saling berhubungan dan diberikan dari *euclidian distance*.
7. Pencarian *parent node* pada *closed node*. *Parent node* adalah node yang memiliki hubungan dengan node lain. *Closed* adalah senarai (list) untuk menyimpan simpul-simpul yang sudah pernah dilewati dan sudah pernah terpilih sebagai simpul terbaik (*best node*) atau meyimpan senarai yang menyimpan jalan yang sudah diperiksa dari *open list*. Artinya, *closed* berisi simpul-simpul yang tidak mungkin terpilih sebagai simpul terbaik (peluang untuk terpilih sudah tertutup). Kedua list (*open list* dan *closed list*) ini bertujuan juga untuk menghindar penelusuran berkali-kali jalan (rute) yang memang sudah diidentifikasi agar tidak masuk lagi ke dalam *open list*.
8. Pemeriksaan akan dihentikan jika telah mencapai tujuan.

b. Penerapan Algoritma *A Star*

Merupakan tahapan melakukan proses penerapan algoritma bagaimana langkah berjalananya algoritma yang digunakan pada penelitian. Tahapan tersebut dapat dilihat pada *Gambar* sebagai berikut.



Gambar 7 Flowchart algoritma floyd warshall

Dalam proses pencarian lintasan terpendek, algoritma *floydwarshall* memulai iterasi dari titik awalnya kemudian memperpanjang lintasan dengan mengevaluasi semua titik demi titik hingga mencapai titik tujuan dan itu menyebabkan proses perhitungan lama. Berikut merupakan penjabaran

1. Pertama, melakukan proses mengidentifikasi titik *node* yang akan dijadikan titik awal dan titik akhir sebagai tujuan yang telah dipilih oleh *client*, lalu dipresentasikan ke dalam sebuah tabel matrix. Terdapat fungsi yang digunakan pada algoritma tersebut yaitu, w = berfungsi untuk menentukan berapa banyak matriks yang dibuat, k = jumlah integrasi yang dibuat pada tabel matrix, i = jumlah simpul yang terbuat pada graf sebagai (titik awal), j = jumlah simpul yang terbuat pada graf sebagai (titik akhir).
2. Lalu, melakukan perhitungan pada iterasi K ke n sampai dengan iterasi terakhir.
3. Selanjutnya melakukan perhitungan pada tiap matriks dan jumlah simpul W $[i,j]$ hingga pada iterasi terakhir.

4. Pada iterasi ke-1 setiap sel matriks dilakukan pengecekan apakah jarak antara dua titik mula-mula lebih besar dari penjumlahan antar jarak titik asal ke titik tujuan (titik tujuan = iterasi ke -1) dengan jarak titik asal (titik asal = iterasi ke-1) ke titik tujuan. Dengan kata lain apakah $W[i,j] > W[i,k] + W[k,j]$.
5. Jika iya maka jarak antar dua titik mula-mula diganti dengan penjumlahan antar jarak titik asal ke titik tujuan (titik tujuan = iterasi ke -1) dengan jarak titik asal (titik asal = iterasi ke -1) ke titik tujuan ($W[i,k] + W[k,j]$).
6. Jika tidak, maka jarak yang digunakan yaitu jarak antar dua titik mula-mula ($W[i,j]$) dan akan terus mengulang sampai dengan iterasi terakhir..
7. Proses iterasi akan terus dilakukan hingga pada iterasi terakhir dan ditemukan bobot terkecil yang diambil pada setiap matixnya (jumlah iterasi = jumlah total titik).

3.1.3 Tahap Pengujian

Merupakan tahapan proses yang harus dilakukan bertujuan untuk memastikan apakah pekerjaan yang dilakukan apakah sistem sesuai dengan rencana awal sehingga dapat dievaluasi dari hasil pengujian tersebut

3.1.4 Kesimpulan dan Saran

Setelah melakukan semua tahapan diatas maka dapat ditarik kesimpulan langkah akhir yang dilakukan adalah penarikan kesimpulan yang berisikan hal hal penting yang sesuai dengan tujuan yang ingin dicapai penelitian tersebut dan sedangkan saran bertujuan untuk pemberian saran atau masukkan untuk penelitian yang sudah dilakukan.

3.2 Analisis Kebutuhan

Setelah melakukan semua tahapan diatas maka dapat ditarik kesimpulan langkah akhir yang dilakukan adalah penarikan kesimpulan yang berisikan hal hal penting yang sesuai dengan tujuan yang ingin dicapai penelitian tersebut dan sedangkan saran bertujuan untuk pemberian saran atau masukkan untuk penelitian yang sudah dilakukan

3.2.1 Requirement Gathering and Refinement

Kebutuhan yang digunakan pada proses penelitian ini yaitu analisis perencanaan suatu sistem, pengumpulan kebutuhan seperti kebutuhan *hardware*,

software dan sistem yang digunakan dalam membangun aplikasi menentukan rute terpendek. Tahap ini mengidentifikasi beberapa literature yang diperlukan untuk menggali teori yang telah berkembang dalam bidang ilmu yang berkepentingan, mencari metode metode, teknik penelitian, baik pengumpulan data atau menganalisa data untuk mengetahui sampai kemana ilmu yang berhubungan dengan penelitian telah berkembang dan sebagai dasar untuk mengaplikasikan nya kedalam penelitian ini.

Peneliti menelusuri literature dari sumber referensi jurnal nasional ataupun, buku dan sumber internet. Bagian ini terus dilakukan beriringan dengan bagian lain sampai bagian akhir penelitian. Hal ini dilakukan supaya jika dalam tahap selanjutnya peneliti membaca atau mempelajari sumber referensi lain yang mendukung penelitian ini. Hal lain yang dilakukan adalah mengumpulkan data-data yang diperlukan dalam penelitian.

Dalam hal ini peneliti mengumpulkan beberapa data-data tentang fasilitas layanan kesehatan seperti rumah sakit sebagai objek penelitian yang berada di wilayah kota Bandung diantaranya nama layanan , titik koordinat (*latitude* dan *longitude*).

3.2.2 Analisis Kebutuhan Pembangunan Sistem

Dalam pembangunan sistem ini, menggunakan sistem operasi Windows 10 Pro 64-bit dan menggunakan beberapa aplikasi pendukung seperti sebagai berikut:

- Android Studio 3.5.2
- Sublime Text 3
- XAMPP
- MySQL Database

3.2.3 Analisis Kebutuhan Instalasi Android Studio

Berikut merupakan Spesifikasi minimum komputer yang dianjurkan oleh developer.android.com untuk installasi android studio yaitu dapat dilihat pada.

Table 1 Spesifikasi android

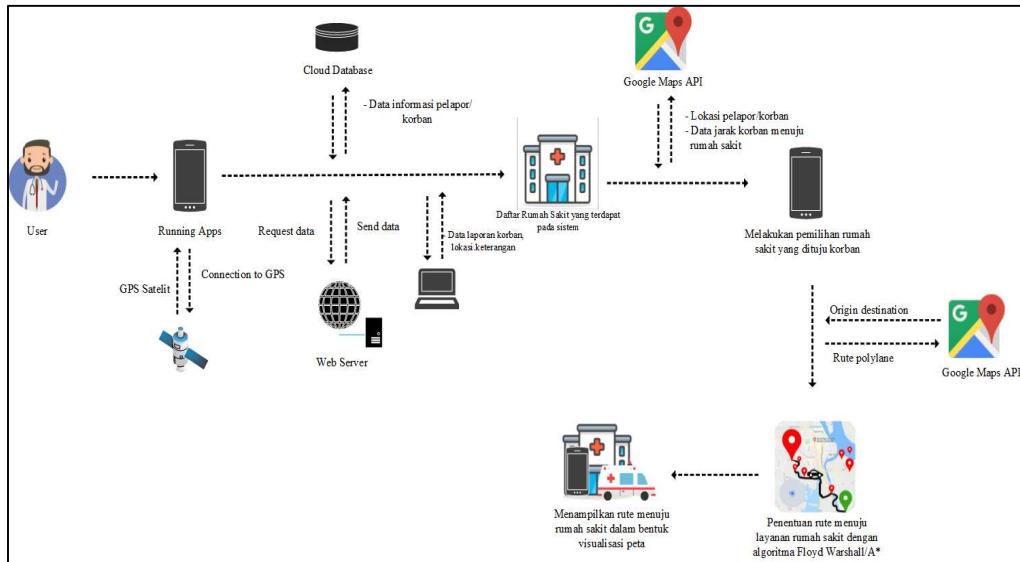
Windows	Mac	Linux
- Microsoft® Windows® 7/8/10 (32- or 64-bit)	- Mac® OS X® 10.10 (Yosemite) or higher, up to 10.13 (macOS High Sierra)	- GNOME or KDE desktop <i>Tested on Ubuntu® 14.04 LTS, Trusty Tahr (64-bit distribution capable of running 32-bit applications)</i>
- 3 GB RAM minimum, 8 GB RAM recommended; plus 1 GB for the Android Emulator	- 3 GB RAM minimum, 8 GB RAM recommended; plus 1 GB for the Android Emulator	- 64-bit distribution capable of running 32-bit applications
- 2 GB of available disk space minimum, 4 GB Recommended (500 MB for IDE + 1.5 GB for Android SDK and emulator system image)	- 2 GB of available disk space minimum, 4 GB Recommended (500 MB for IDE + 1.5 GB for Android SDK and emulator system image)	- GNU C Library (glibc) 2.19 or later
- 1280 x 800 minimum screen resolution	- 1280 x 800 minimum screen resolution	- 3 GB RAM minimum, 8 GB RAM recommended; plus 1 GB for the Android Emulator - 2 GB of available disk space minimum, 4 GB Recommended (500 MB for IDE + 1.5 GB for Android SDK and emulator system image) - 1280 x 800 minimum screen resolution

3.3 Perancangan Umum (Quick Design)

Pada sub bab ini akan menjelaskan prinsip kerja sistem atau cara kerja sistem layananan gawat darurat secara keseluruhan dan untuk menggambarkan langkah – langkah dan urutan prosedur dari implementasi algoritma *FloydWarshall*, *A Star* pada aplikasi.

Penelitian dilakukan disekitar wilayah kota bandung tentu membutuhkan data setiap jalanan dan persimpangan yang ada serta info jalur arah dikota tersebut untuk menuju layanan rumah sakit, membutuhkan juga data koordinat setiap rumah sakit yang akan dijadikan sebagai titik tujuan. Koordinat tersebut berisikan data *latitude* dan *longitude* didapatkan dari *Google Maps* sangat dibutuhkan yang nanti data tersebut akan diproses, dan dipetakan didalam, terdapat data *node-node* yang saling terhubung *node* yang dibuat di setiap persimpangan jalan raya dan ruas jalan sebagai rute dari setiap *node* nya memiliki nilai bobot ruas jalan yang digunakan

untuk melakukan proses menghitung jarak tempuh dari lokasi awal menuju akhir. Seperti berikut ini



Gambar 8 Perancangan umum quick design

Menjelaskan bahwa sistem yang dibuat diimplementasikan pada *mobile device* atau *smartphone* sebagai *client*. Pertama *user* menjalankan aplikasi , lalu dilakukan login serta terhubung dengan koneksi internet dan mengaktifkan koneksi *GPS (Global Posisiton System)* pada *smartphone*, sistem akan membaca dan menerima transmisi sinyal satelit untuk mendapatkan posisi koordinat *user* sebagai variabel masukan input berupa *latitude* dan *longitude* yang di jadikan sebagai posisi awal . *User* sebagai pelapor melakukan proses pelaporan dengan menginputkan laporan darurat berupa informasi kejadian yang dialami oleh pelapor yaitu kecelakaan, tempat kejadian, posisi kemudian data tersebut akan disimpan pada *cloud database* dan dikirimkan dari *smartphone* ke *web server* lalu akan diterima oleh rumah sakit yang dipilih oleh *user*. Setelah itu akan ditampilkan pilihan rumah sakit yang sudah terdapat pada *database* dengan titik *latitude* dan *longitude* yang, sebanyak 27 rumah sakit yang tersedia beserta data informasi setiap rumah sakit, kemudian dengan menggunakan fasilitas persamaan *haversine formula* akan dihitung perkiraan jarak

real terdekat dengan garis lurus antara dua titik permukaan bumi dari lokasi terkini *user* menuju rumah tujuan memanfaatkan koordinat *user* yang sudah didapatkan sebelumnya melalui *GPS* pada *smartphone* yang terintegrasikan dengan *Google Maps API*. Sistem akan menampilkan hasil semua data informasi rumah sakit dan data jarak *real* terdekat dari lokasi terkini *user* menuju rumah sakit yang sudah diproses dalam bentuk *card list*, lalu sistem dapat mengurutkan rumah sakit terdekat secara *ascending* dari mulai yang terkecil sampai terjauh jaraknya dari lokasi *user* terkini guna untuk memudahkan dalam pemilihan rumah sakit yang akan dituju oleh *user*. Kemudian *user* dapat memilih rumah sakit mana saja yang akan dituju sesuai dengan yang dibutuhkan *user*. Setelah salah satu rumah sakit dipilih sistem akan mengarahkan untuk melakukan pencarian rute terpendek, pada proses tersebut dilakukan perhitungan dengan menggunakan dua algoritma yaitu *Floydwarhsall* dan *A Star* sistem akan mengirimkan request pada *Google Maps Direction API* yang bertujuan untuk mencari rute mana yang dilalui beserta jarak dan waktu tempuh dari lokasi *user* ke rumah sakit yang dipilih. *API* akan mengembalikan data berupa *JSON* yang didalamnya terdapat data *waypoint* (*point node* setiap persimpangan), lalu antar *point node* tersebut akan digambar sebuah garis yang akan menjadi rute menuju rumah sakit tujuan, rute tersebut akan ditampilkan dalam bentuk visualisasi peta. Untuk Algoritma *A Star* merupakan perbaikan dari metode *best-first search* dengan memodifikasi fungsi *heuristiknya*, yang menjadikan algoritma ini meminimalkan total biaya lintasan dan pada kondisi yang tepat akan memberikan solusi yang terbaik (**Imam Ahmad, Wahyu Widodo 2017**). Cara kerja dari algoritma *A** yaitu memperhitungkan beberapa kemungkinan lintasan yang ada, lalu kemudian membandingkan kemungkinan tersebut satu demi satu maka diperoleh lintasan sebagai lintasan terpendek (**Johannes Ridho 2015**). Sedangkan algoritma *Floydwarsall* merupakan salah satu varian dari pemrograman dinamis yaitu suatu metode yang melakukan pemecahan dengan memandang solusi yang akan diperoleh sebagai keputusan yang saling terkait. Artinya solusi-solusi tersebut terbentuk dari solusi yang berasal dari tahap sebelumnya dan ada kemungkinan solusi lebih dari satu (**Friska Widya Ningrum & Tatyantoro Andrasto 2016**). Cara kerja dari algoritma *Floyd Warshall* yaitu mencari dan

kemudian membandingkan semua lintasan yang mungkin terjadi dalam *graf* pada setiap pasangan simpul dan melakukan pengujian dari kombinasi yang sudah diperoleh (**Popa & Popescu, 2016**).

3.3.1 Analisis Perancangan Sistem

Tahap tahap perencangan pengembangan sistem yang diperlihatkan pada tabel berikut ini.

Table 2 Analisa perancangan sistem

No	Tahap	Keterangan
1.	Pengumpulan kebutuhan	Melakukan pengumpulan hal hal yang dibutuhkan dalam sistem.
2.	Perancangan cepat	Membuat perancangan pada sistem aplikasi seperti flowchart ,workflow, use case dan kebutuhan dalam perancangan.
3.	Membangun <i>prototyping</i>	Membuat <i>prototype</i> sistem aplikasi penentuan rute terpendek.
4.	Evaluasi <i>prototyping</i>	Melakukan evaluasi pada <i>prototype</i> yang sudah dibuat.
5.	Kode sistem	<i>Prototype</i> yang sudah dibuat diterjemahkan kedalam bahasa pemrograman.

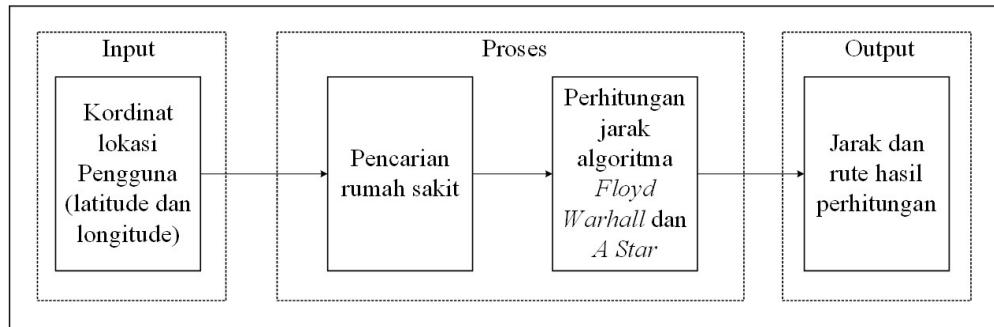
3.4 Membangun Prorotype (Build Prorotype)

Pada tahap ini dibuat sebuah perencangan sementara yang merupakan implementasi dari tahap *design* cepat , dimana akan dijelaskan tentang perancangan dan cara kerja dari sistem yang akan dibangun. Dalam penelitian ini , perancangan sistem yang digunakan ialah *Unified Modeling Language* (UML).

UML merupakan sebuah bahasa menggambarkan, memspesifiksikan, membangun dan mendokumentasikan sebuah sistem pengembangan *software* seperti blok diagram , flowchart, diagaram alir dan lain sebagainya.

3.4.1 Blok Diagram

Pada Gambar 9 berikut ini merupakan blok diagram dari alur cara kerja sistem yang akan dibangun dengan mewakili semua aspek perangkat keras dan lunak yang dibutuhkan dan menjadi dasar pembuatan sistem hingga tahap implementasi dan pengujian pada sistem.



Gambar 9 Blok diagram

Sebagaimana ditunjukkan pada Error! Reference source not found. Proses-proses yang berlangsung pada sebuah pembangunan sistem, dapat dijelaskan dibawah ini sebagai berikut:

- Input

Proses input adalah sebuah proses untuk memasukan variabel atau parameter dari penelitian berikut penjelasan mengenai tahapan input.

1. Input atau masukan yang digunakan pada sistem adalah nilai koordinat *latitude* dan *longitude* posisi *user* yang didapatkan dari smartphone dengan menggunakan *GPS*.

- Proses

Pada tahap ini sistem melakukan proses dari data yang sudah didapatkan dari tahapan input. Terdapat beberapa proses yang diolah sebelum melakukan kesimpulan.

2. Inputan atau masukkan didapatkan kemudian dilakukan perhitungan perkiraan jarak *real* menggunakan *Haversine Formula* dengan menghitung garis lurus antara dua titik permukaan bumi antara lokasi *user* dengan lokasi tujuan, bertujuan untuk mengetahui rumah sakit terdekat dari lokasi terkini *user*, lalu

setelah proses selesai hasil data jarak tersebut akan ditampilkan berbentuk *list*.

Dengan begitu *user* dapat memilih rumah sakit yang akan dituju.

3. Setelah pilihan rumah sakit terpilih maka masuk ke dalam proses perhitungan algoritma baik algoritma *floydwarshall* dan *a star* sistem akan melakukan request data pada *Maps Direction API* untuk mengetahui rute mana yang dilalui dari kedua algoritma tersebut berdasarkan jarak dan waktu tempuh.

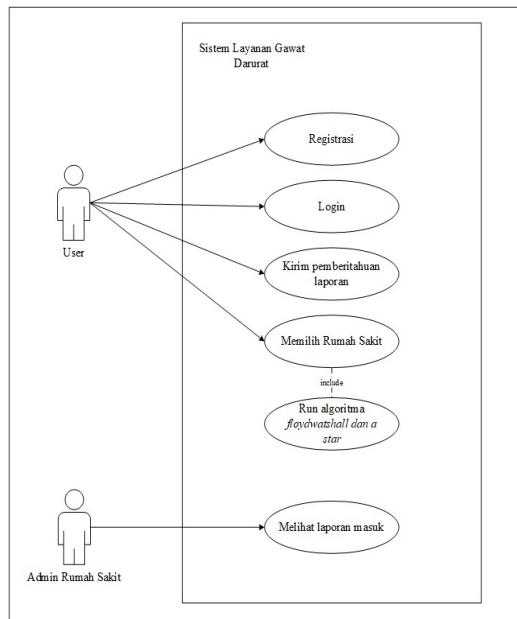
- Output

Pada tahap ini merupakan hasil dari sistem yang dibuat.

4. Hasil keluaran dari proses yang dilakukan sistem akan ditampilkan rute beserta jarak dan waktu tempuh dalam bentuk visualisasi peta pada *smartphone user*.

3.4.2 Use Case Diagram

Use Case Diagram digunakan untuk mengetahui interaksi antara aktor dengan sistem. Aktor yang dapat berinteraksi dengan sistem ini adalah *user*.



Gambar 10 Use case diagram

Dapat dilihat bahwa *user* dapat melakukan proses registrasi, login, pengiriman pemberitahuan laporan, pemilihan rumah sakit dan pemilihan algoritma pencarian rute. Berikut scenario interaksi antara aktor dengan sistem.

1. Skenario Melakukan Proses Registrasi

Registrasi ini dilakukan *user* yang bertujuan untuk memiliki akun sehingga dapat melakukan proses login dan masuk ke dalam aplikasi.

Table 3 Skenario proses registrasi

Use Case	Registrasi
Aktor	<i>User</i>
Kondisi Awal	Aktor dapat melakukan membuka halaman proses registrasi
Kondisi Akhir	Registrasi berhasil melakukan proses registrasi dan aktor memiliki akun
Deskripsi	Aktor melakukan registrasi dengan mengisi informasi data diri yang terdapat pada halaman yang bertujuan untuk mendapatkan akun dapat menggunakan aplikasi
Skenario	
Aktor	Sistem/Aplikasi
1. Memasukkan data diri	2. Validasi data masukkan dari aktor Memyimpan data <i>user</i>
2. Skenario Melakukan Proses Login	

Proses login dilakukan *use* ini bertujuan untuk masuk ke dalam aplikasi jika sudah melakukan proses registrasi sebelumnya dengan memasukan informasi data diri.

Use Case	Login
Aktor	<i>User</i> .
Kondisi Awal	Aktor dapat melakukan membuka halaman proses login.
Kondisi Akhir	Login berhasil, aktor masuk kedalam aplikasi.
Deskripsi	Aktor melakukan login bertujuan untuk bisa menggunakan aplikasi dengan akun yang sudah terdaftar dengan melakukan proses registrasi sebelumnya.
Skenario	

Aktor	Sistem/Aplikasi
1. Memasukkan username dan password	2. Validasi data masukkan dari aktor 3. Masuk ke dalam menu utama aplikasi

Table 4 Skenario proses login

3. Skenario Melakukan Pengiriman Pemberitahuan Laporan Darurat

Proses ini dilakukan saat *user* membutuhkan layanan gawat darurat ke pihak rumah sakit yang dipilih, dan dapat ditangani dengan segera mungkin maka dibutuhkan data pemberitahuan laporan kepada pihak layanan rumah sakit

Table 5 Skenario laporan darurat

Use Case	Laporan
Aktor	<i>User</i>
Kondisi Awal	<ul style="list-style-type: none"> - Berhasil login - Membuka layanan gawat darurat - Mengaktifkan <i>GPS</i> pada <i>smartphone</i>
Kondisi Akhir	Login berhasil, aktor masuk kedalam aplikasi dan masuk kedalam fitur input laporan darurat
Deskripsi	Data laporan <i>user</i> dibutuhkan untuk memberikan informasi kepada pihak layanan rumah sakit yang dipilih dengan melakukan proses penginputan data laporan darurat pada aplikasi, kemudian dikirimkan kepada pihak layanan rumah sakit.
Skenario	
Aktor	Sistem/Aplikasi
1. Membuka fitur input laporan darurat	2. Mengirim data laporan ke server dan ke pihak rumah sakit.

4. Skenario Pemilihan Rumah Sakit

Sistem menampilkan seluruh informasi data rumah sakit sesuai dengan kebutuhan *user* yang dalam bentuk daftar *list* pada *smartphone*. *User* dapat memilih

seluruh rumah sakit mana yang dibutuhkan sesuai dengan kondisi *user* dan dilengkapi dengan informasi jarak, waktu tempuh, kategori dan tipe rumah sakit sehingga dapat memudahkan *user*.

Table 6 Skenario pemilihan rumah sakit

Use Case	Pemilihan Rumah Sakit
Aktor	<i>User</i>
Kondisi Awal	Aktor selesai melakukan proses input laporan darurat dan mengaktifkan <i>GPS</i> pada <i>smartphone</i>
Kondisi Akhir	Sistem menampilkan dalam bentuk <i>list</i> rumah sakit beserta jarak,kategori,nama rumah sakit pada <i>smartphone</i> yang kemudian nantinya akan didipilih oleh <i>user</i> sesuai kebutuhan.
Deskripsi	<i>User</i> dapat dengan bebas memilih rumah sakit mana yang dinginkan atau dibutuhkan sesuai dengan yang direkomendasikan sistem berdasarkan lokasi jarak dari <i>user</i> .
Skenario	
Aktor	Sistem/Aplikasi
1. Membuka fitur layanan darurat	2. Menampilkan <i>list</i> data rumah sakit pada <i>smartphone</i>

5. Skenario Proses Penentuan/Menampilkan Rute

Sistem akan menampilkan rute dan melakukan perhitungan menggunakan algoritma yang terdapat didalam sistem dan sistem akan menampilkan hasil dari perhitungan rute terpendek berdasarkan algoritma yang dipilih oleh *user*

Use Case	Menampilkan rute
Aktor	<i>User</i>
Kondisi Awal	Aktor akan memilih algoritma dari pilihan algoritma yang ada, yaitu <i>A Star</i> dan <i>Floyd Warshall</i>
Kondisi Akhir	Sistem akan menampilkan rute hasil dari algoritma <i>Floyd Warshall</i> dan <i>A Star</i>

Deskripsi	<i>User</i> memilih algoritma setelah itu sistem akan menampilkan rute menuju lokasi tujuan dari posisi <i>user</i> dengan mengaktifkan <i>GPS</i> .
Skenario	
Aktor	Sistem/Aplikasi
1. Memilih rumah sakit yang diinginkan atau tujuan	2. Menampilkan rute dan informasi menuju rumah sakit tujuan.

Table 7 Skenario penentuan rute

6. Skenario Laporan Masuk

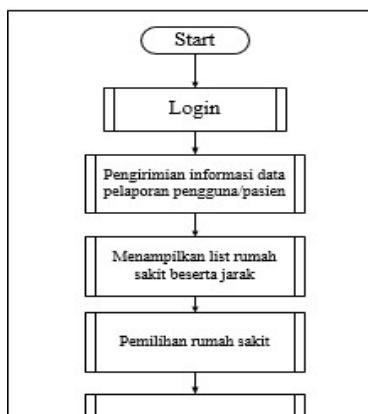
Proses ini berfungsi untuk menampilkan seluruh informasi data laporan *user* sebagai pelapor kepada pihak rumah sakit yang dipilih oleh *user*.

Table 8 Skenario laporan masuk

Use Case	Riwayat Rumah Sakit
Aktor	Admin
Kondisi Awal	Aktor masuk ke dalam server
Kondisi Akhir	Sistem akan mampulkan riwayat informasi seluruh laporan pemberitahuan yang masuk pada setiap masing rumah sakit.
Deskripsi	Manampulkan seluruh informasi laporan masuk pada setiap rumah sakit berbeda yang berisikan lokasi kejadian,tanggal, jenis, dan keterangan lainnya.
Skenario	
Aktor	Sistem/Aplikasi
1. Melakukan proses login kedalam server menu unit layanan darurat	2. Menampilkan seluruh informasi laporan pada setiap masing masing rumah sakit.

3.4.3 Diagram Alir Sistem

Diagram alir sistem akan menjelaskan alur kerja sistem secara rinci pada setiap proses yang ada pada sistem layanan gawat darurat

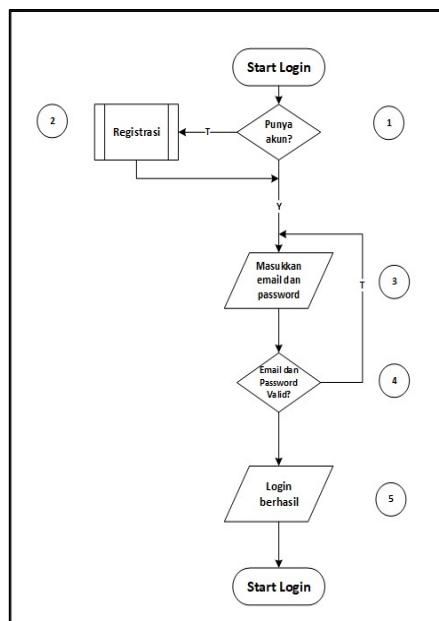


Gambar 11 Diagram alir sistem

Diagram alir sistem program terdiri dari beberapa sub- proses yang akan dijelaskan secara rinci sebagai berikut.

a. Diagram Alir Proses *Login*

Untuk melakukan proses *login*, terlebih dahulu dibutuhkannya akun yang didapatkan dengan melalui proses registrasi. Sistem akan mengecek apakah *email* dan *password user* yang dimasukkan valid, jika *user* dapat masuk kedalam sistem. Setelah melakukan *login* sistem akan terus menyimpannya sehingga *user* yang akan menggunakan aplikasi tidak perlu melakukan proses *login* kembali kecuali *user* melakukan *logout* pada sistem sebelumnya.



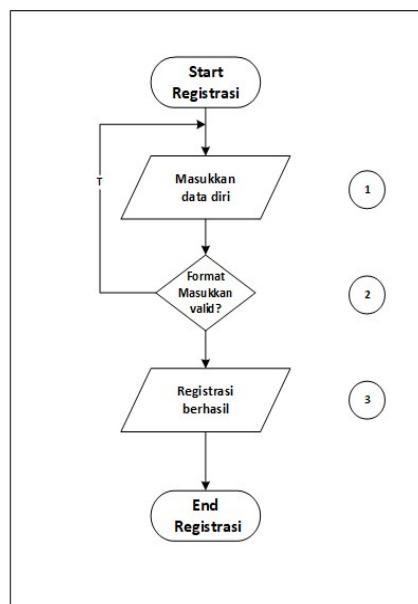
Gambar 12 Proses login

Proses *login* akan dijelaskan sebagai berikut :

1. Supaya dapat melakukan proses login, terlebih dahulu dibutuhkan akun yang didapat dari proses registrasi. Jika *user* belum mempunyai akun, maka *user* melakukan ke proses registrasi yang akan dijelaskan pada proses registrasi bagian b. Jika *user* sudah melakukan registrasi sebelumnya, maka *user* akan langsung masuk ke proses login.
2. Proses registrasi dijelaskan pada bagian b diagram proses registrasi.
3. Pada proses login ini, *user* diminta untuk memasukkan *email/username* dan *password* yang sudah didaftarkan sebelumnya.
4. Sistem akan mengecek apakah *email/username* dan *password* yang dimasukkan valid ataupun kesesuaian dengan *email/username* dan *password* tidak valid, disarankan untuk memasukkan kembali *email/username* dan *password*.
5. Jika *email/username* dan *password* yang dimasukkan *user* telah valid, maka proses *login* berhasil.

b. Diagram Alir Proses *Register*

Pada proses registrasi ini dibutuhkannya data diri dari setiap *user* seperti nama, alamat, nomer telepon, email, *username* dan *password*. Sistem terlebih dahulu akan mengecek apakah format data yang telah diinputkan *user* sesuai valid ataupun tidak. Apabila semua data yang telah dimasukkan valid maka proses registrasi berhasil dan dapat melakukan proses login yang sudah dijelaskan pada bagian a.

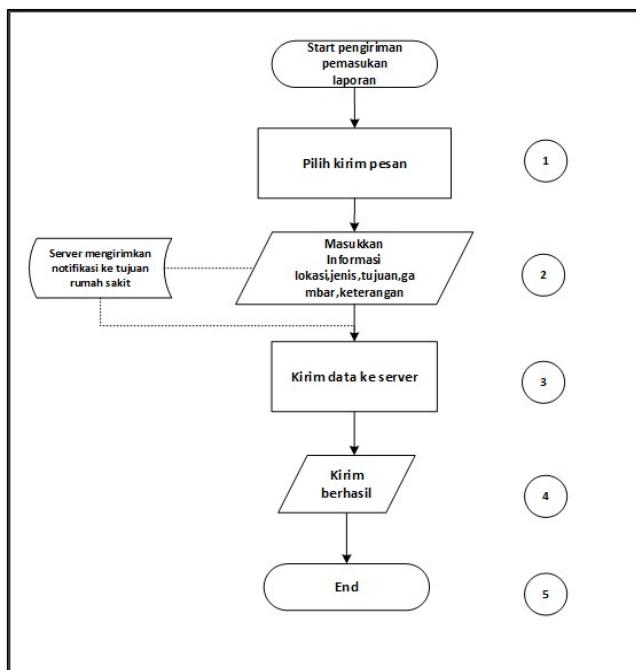


Gambar 13 Proses registrasi

Proses registrasi akan dijelaskan secara detail sebagai berikut :

1. Dalam melakukan proses pendaftaran/registrasi pada sistem dibutuhkannya data diri dari setiap *user*, termasuk *email/username* dan *password* yang akan digunakan untuk proses *login*.
 2. Dan sistem akan memeriksa apakah data yang sudah diinputkan oleh *user* sudah valid atau tidak. Jika masih belum sesuai data yang diinputkan, maka sistem akan meminta untuk mengisi kembali data diri *user*.
 3. Setelah sistem mendeteksi data diri yang dimasukkan *user* telah valid maka proses registrasi berhasil dan *user* dapat melakukan proses *login*.
- c. Diagram Alir Proses Laporan Darurat

Merupakan data laporan yang diterima oleh sistem yang dididalamnya berisikan masukkan lokasi, jenis laporan, foto, kordinat, tanggal dan keterangan lain tentang kejadian tersebut. Kemudian sistem akan mengirimkan ke server, admin rumah sakit yang dipilih setelah proses pengiriman selesai sistem akan memberikan pemberitahuan darurat kepada pihak rumah sakit yang dipilih.



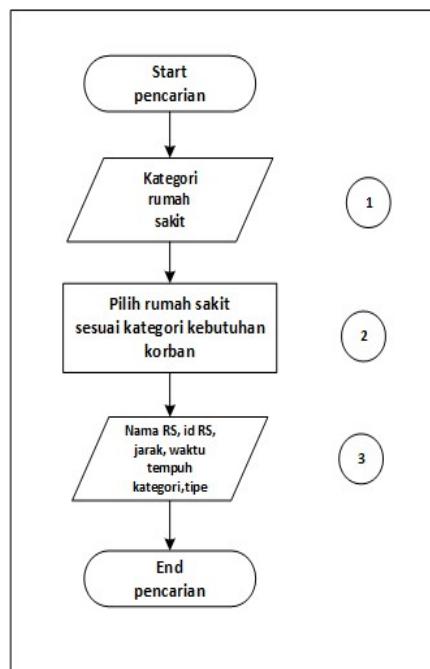
Gambar 14 Proses laporan darurat

Proses pengiriman pesan pemberitahuan data laporan pada *user* akan dijelaskan secara detail sebagai berikut :

1. Untuk melakukan proses pengiriman lalu menginputkan pesan pemberitahuan data laporan darurat pada korban dibutuhkan informasi.
2. Memasukkan informasi yang terdiri dari lokasi tempat ,jenis laporan,tujuan alamat rumah sakit, foto, kordinat,tanggal dan keterangan lain.
3. Setelah itu melakukan proses pengiriman data laporan berupa koordinat user dan informasinya ke tujuan rumah sakit yang dipilih oleh user, setelah melakukan pengiriman sistem akan memberikan pemberitahuan darurat kepada rumah sakit tujuan yang dipilih.
4. Pengiriman berhasil
5. Selesai.

d. Diagram Alir Proses Pemilihan Rumah Sakit

pemilihan rumah sakit sesuai dengan kebutuhan *user* sehingga sistem akan menampilkan informasi berupa nama rumah sakit, id rumah sakit, jarak, ,alamat dan kategori yg sesuai dengan kebutuhan *user* yang kemudian akan ditampilkan di *smartphone user* dalam bentuk list

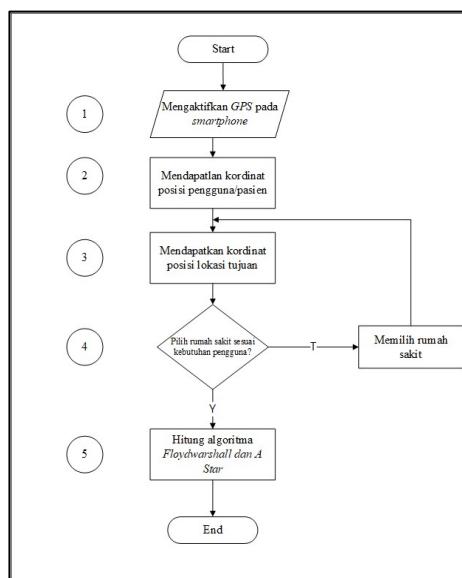


Gambar 15 Pemilihan rumah sakit

Pemilihan rumah sakit akan dijelaskan detail sebagai berikut :

1. Sistem akan menampilkan daftar rumah sakit beserta kategori yang dapat dipilih oleh *user*.
 2. Lalu *user* melakukan proses pemilihan rumah sakit sesuai dengan kebutuhan.
 3. Data yang didapatkan berupa nama rumah sakit , id rumah sakit, jarak, kategori, alamat dan tipe rumah sakit.
- e. Diagram Alir Penentuan Rute

Data rumah sakit yang terdapat pada sistem akan ditampilkan pada *smartphone* dalam berbentuk list dilengkapi informasi data jarak, kategori, tipe, dan informasi lainnya yang nantinya akan di pilih *user*



Gambar 16 Penentuan rute

Proses diagram alir penentuan rute terpendek akan dijelaskan secara detail sebagai berikut :

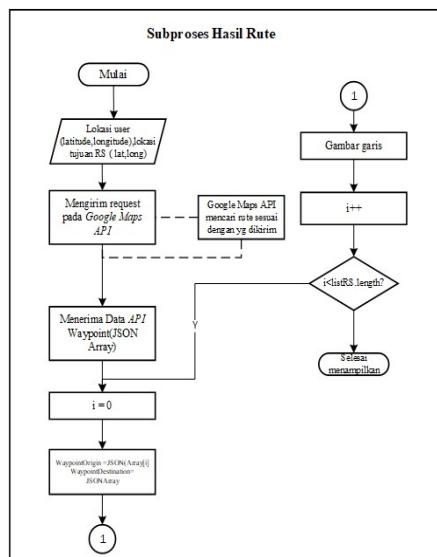
1. Proses pertama,*user* mengaktifkan *GPS* terlebih dahulu pada *smartphone*.
2. Sistem menerima kordinat lokasi setelah mengaktifkan *GPS* pada *smartphone* yang bertujuan sebagai titik awal.
3. Mendapatkan kordinat posisi lokasi tujuan rumah sakit, disini sistem melakukan persamaan *Haversine* untuk mendapatkan data jarak dari lokasi

user dengan lokasi rumah sakit tujuan yang kemudian diinputkan ke dalam list daftar rumah sakit.

4. Dan *user* dapat memilih rumah sakit yang sesuai dengan kebutuhan yang terdapat pada sistem, jika sudah dipilih masuk ke proses penentuan atau pencarian rute jika tidak kembali lagi ke proses nomer3.
5. Setelah mendapatkan *node* awal dan tujuan, selanjutnya melakukan perhitungan dengan algoritma *Floyd Warshall* dan *A Star*.
6. Lalu menampilkan hasil

3.4.4 Flowchart Sistem

Hasil pemilihan rumah sakit ditetapkan oleh *user* sistem akan kedalam proses perhitungan pencarian rute menggunakan algoritma *floydwarshall* dan *star*. lalu akan ditampilkan nya rute dari lokasi terkini *user* ke lokasi rumah sakit tujuan sesuai dengan data nilai *latitude* dan *longitude*. Setelah perhitungan selesai sistem melakukan *request* pada *google maps API* untuk penentuan rute. API akan mengembalikan data bentuk *JSON* berupa persimpangan (*waypoint*) dan sistem menggambarkan *polyline* (garis) antar titik *waypoint node* sehingga menjadi rute menuju tujuan rumah sakit pada *smartphone user*.



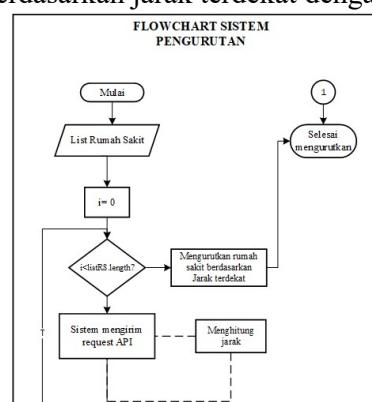
Gambar 17 Flowchart sistem

Berikut penjelasan dari *flowchart* sistem pada *Gambar 4* sebagai berikut :

1. Proses konfigurasi pada *Google Maps* digunakan *API key* yang tersedia pada *Google Maps*, bertujuan untuk membantu dalam pemetaan rute pada aplikasi.
2. *User* menjalankan aplikasi
3. Supaya aplikasi dapat berjalan terlebih dahulu melakukan konfigurasi *GPS* agar selalu aktif pada *smartphone* dan terkoneksi dengan internet untuk menginisialisasi atau pembacaan guna mendapatkan nilai titik koordinat posisi terkini *user*.
4. Sistem akan memberikan notifikasi setelah koordinat *user* terdeksi.
5. Dari koordinat lokasi awal dan lokasi tujuan yang didapatkan, sistem memproses dengan melakukan perhitungan jarak *real* atau perkiraan jarak garis lurus antara dua titik permukaan bumi dihitung dari lokasi terkini *user* dengan rumah sakit dengan menggunakan persamaan *haversine formula* dan fasilitas *google maps API Distance matrix*.
6. Selesai melakukan proses tersebut sistem akan menampilkan pilihan rumah sakit dari *database* dalam bentuk *list* dilengkapi data jarak yang dapat dipilih oleh *user* sesuai dengan kebutuhan. Selain itu sistem dapat mengurutkan *list* rumah sakit berdasarkan jarak terdekat dari lokasi terkini *user*.
7. Tujuan rumah sakit terpilih sistem akan masuk ke proses perhitungan algoritma pencarian rute terpendek dengan menggunakan dua algoritma yaitu *floydwarsall* dan *a star* dan sistem melakukan *request* pada *Google Maps Direction API* untuk mengetahui rute yang dapat dilalui menuju rumah sakit terpilih.
8. Setelah dilakukan proses perhitungan dan analisis dari masing-masing algoritma pencarian rute terpendek, rute akan ditampilkan pada visualisasi peta disertakan dengan jarak dan waktu tempuh menuju lokasi tujuan.

3.4.5 Flowchart Pengurutan Rumah Sakit

Data rumah sakit yang diterima, ditampilkan dalam bentuk *list* pada *smartphone* dilengkapi dengan dengan data jarak. Rumah sakit yang tersedia ditampilkan diurutkan berdasarkan jarak terdekat dengan lokasi *user*.



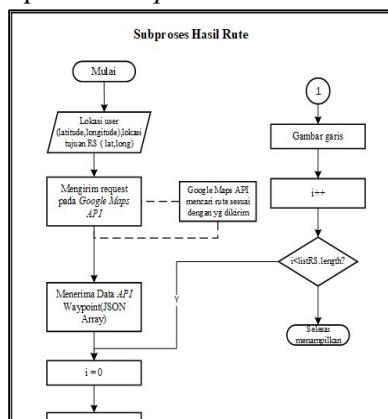
Gambar 18 Pengurutan rumah sakit

Proses pengurutan rumah sakit dijelaskan sebagai berikut :

1. Sistem menerima data list rumah sakit yang didapat dari proses perhitungan jarak terdekat menggunakan *haversine formula* dan melakukan *request* pada *google maps distance matrix API* guna mendapatkan hasil data jarak dari lokasi terkini *user* menuju rumah sakit terpilih sebagai lokasi tujuan.
2. Sistem akan menerima data jarak kemudian dimasukkan kedalam *list* rumah sakit.
3. Sistem melakukan pengurutan data jarak secara *ascending*. Dan akan mengulang proses kembali ke no 1 sebanyak data *list* rumah sakit.
4. Selesai pengurutan selesai dapat *user* dapat mengetahui rumah sakit mana yang terdekat dari lokasi terkini *user*.

3.4.6 Flowchart Subproses Hasil Rute

Dari hasil pemilihan rumah sakit ditetapkan oleh *user* sistem akan kedalam proses perhitungan pencarian rute menggunakan algoritma *floydwarshall* dan *star*. lalu akan ditampilkan nya rute dari lokasi terkini *user* ke lokasi rumah sakit tujuan sesuai dengan data nilai *latitude* dan *longitude*. Setelah perhitungan selesai sistem melakukan *request* pada *google maps API* untuk penentuan rute. API akan mengembalikan data bentuk JSON berupa persimpangan (*waypoint*) dan sistem menggambarkan *polyline* (garis) antar titik *waypoint node* sehingga menjadi rute menuju tujuan rumah sakit pada *smartphone user*.



Gambar 19 Subproses hasil rute

Subproses menampilkan rute dijelaskan sebagai berikut :

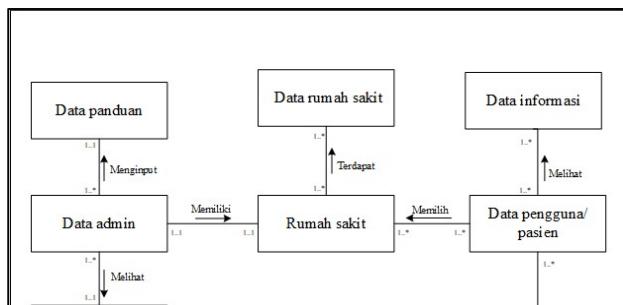
1. Sistem akan menerima data lokasi *user* dan rumah sakit terpilih.
2. Data tersebut didapatkan dari setelah proses perhitungan algoritma selesai dilakukan
3. Sistem akan mengirim *request* pada *google maps API direction* untuk mendapatkan rute.
4. Lalu *google maps API* akan mengembalikan data bentuk *JSON* berupa *polyline*.
Dan menerima data *polyline* yang kemudian data tersebut disimpan dalam *array JSONArray*. *Polyline* adalah data titik waypoint (node) dari lokasi *user* sebagai titik awal dan menuju rumah sakit sebagai lokasi tujuan.
5. Rute yang terbentuk yaitu dengan menghubungkan garis antar titik *waypoint*.
Data waypoint yang dijadikan titik awal(*origin*) merupakan data array ke-*i* lalu data waypoint yang dijadikan titik tujuan (*destination*) merupakan data array ke *i*+1.
6. Setelah data *waypoint* didapat , sistem akan menggambar garis berwarna dan membentuk rute. Tahap proses 5 dan 6 akan diulang sebanyak data *JSONArray*.

3.4.7 Perancangan Database

Di dalam sistem yang sedang dibangun, dibutuhkan suatu database yang berfungsi untuk menampung seluruh data sebagai berikut:

- *Entity Relationship Diagram (ERD)*

Digunakan untuk mengetahui hubungan antar entitas dalam database. Pada sistem ini terdapat beberapa entitas dan dapat dilihat secara detail sebagai berikut.

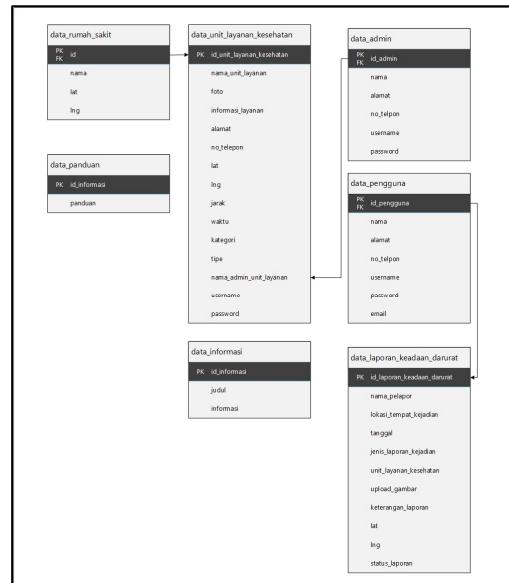


Gambar 20 Entity relationship

Relasi antara setiap entitas yaitu pada *user* bisa memilih lebih dari 1 rumah sakit sesuai kebutuhan, setiap rumah sakit bisa dirujuk atau dipilih lebih dari 1 *user*, dan *user* dapat mengirimkan lebih dari 1 laporan kepada rumah sakit yang dipilih, dan laporan dapat dikirim lebih dari 1 *user*, dan setiap rumah sakit hanya memiliki 1 admin dan 1 admin hanya dimiliki 1 rumah sakit, sedangkan admin dapat melihat lebih dari 1 laporan yang masuk pada rumah sakit yang dipilih oleh *user*, pada masing masing rumah sakit terdapat lebih dari 1 informasi data rumah sakit.

- *Table Relanrstionship Diagram (TRD)*

Digunakan untuk mengetahui hubungan antar table dalam database. Pada sistem ini terdapat beberapa entitas dan dapat dilihat secara detail lebih jelasnya sebagai berikut.



Gambar 21 Table relationship diagram

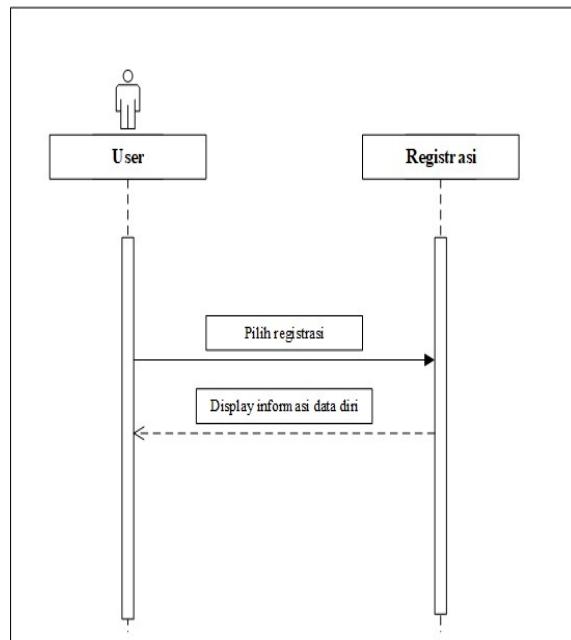
Dapat dilihat pada gambar entitas rumah sakit yang dipresentasikan pada tabel data_rumah_sakit memiliki atribut id,nama,lat dan lng. Atribut id menjadi *foreign key* yang terhubung pada atribut id_unit_layanan_kesehatan pada tabel data_unit_layanan_kesehatan yang presentasikan pada entitas data_rumah sakit. Dan tabel data_admin yang memiliki atribut id_admin menjadi *foreign key* yang terhubung dengan tabel data_layanan_kesehatan yang bertujuan untuk digunakan mendapatkan data admin. Sedangkan tabel data_user memiliki atribut id_user yang terhubung langsung dengan tabel data_laporan_keadaan_darurat yang digunakan untuk mendapatkan data user.

3.4.8 Sequence Diagram

Kegiatan *sequence diagram* dari sistem yang dibangun menggambarkan langkah-langkah yang dilakukan pada sistem langsung dengan *user*. Yang terdiri dari beberapa pengelolaan yaitu pengelolaan registrasi,login,laporan darurat, pemilihan rumah sakit, run algoritma *FloydWarsall* dan *A Star*, laporan masuk, lokasi *user*, lokasi tujuan. Berikut penjelasan dari masing-masing bagian.

1. Sequence Diagram Registrasi

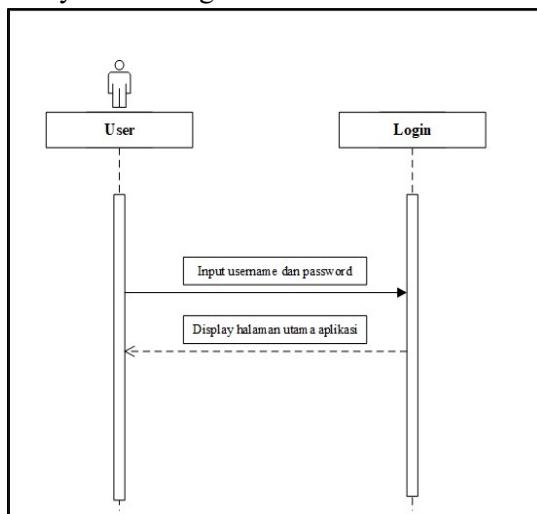
Kegiatan pada pengelolaan registrasi ini diantaranya *user* membuka halaman registrasi pada aplikasi dan *user* kemudian mengisi informasi data diri yang terdapat pada halaman tersebut



Gambar 22 Squence diagram registrasi

2. Squence Diagram Login

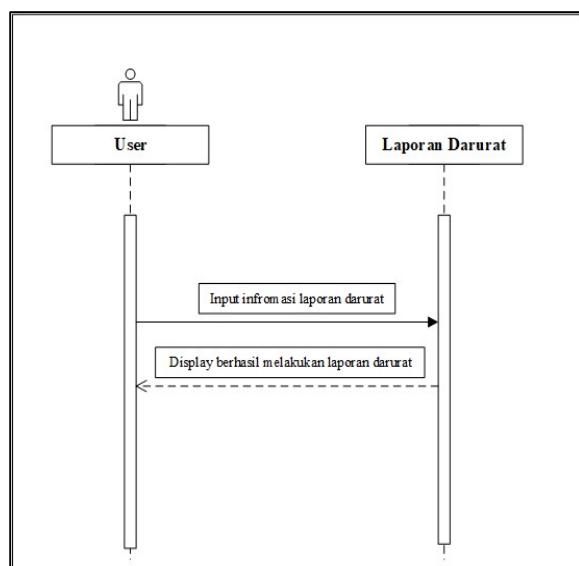
Kegiatan pada pengelolaan login ini diantaranya *user* membuka halaman login pada aplikasi dan *user* lalu dengan cara memasukkan *username* dan *password* yang sudah dibuat sebelumnya dalam registrasi



Gambar 23 Squence diagram login

3. Squence Diagram Login

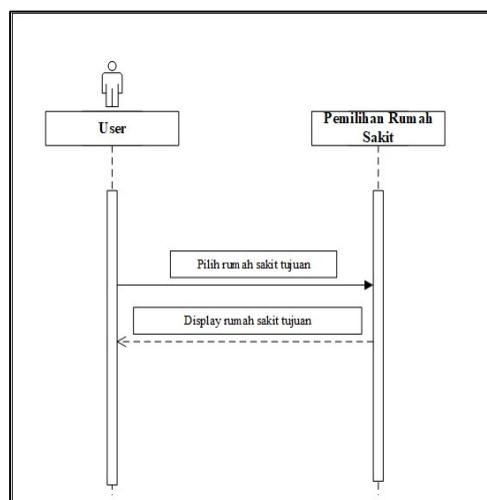
Kegiatan pada pengelolaan laporan darurat ini diantaranya *user* melakukan penginputan laporan darurat.



Gambar 24 Squence diagram login

4. Squence Diagram Pemilihan Rumah Sakit

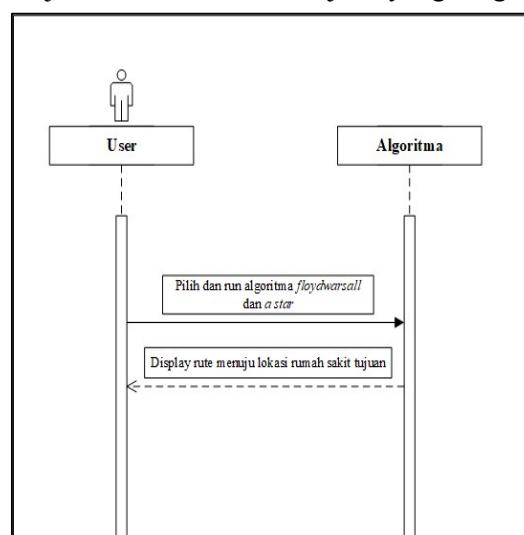
Kegiatan pada penglolaan pemilihan rumah sakit ini diantaranya *user* melakukan pemilihan rumah sakit sesuai dengan kebutuhan *user*, kemudian sistem akan menampilkan berupa list rumah sakit pada *smartphone*



Gambar 25 Squence diagram pemilihan rumah sakit

5. Squence Diagram Algoritma Floydwarshall dan A Star

Kegiatan pada penglolaan pemilihan rumah sakit ini dilanjutkan *user* melakukan pemilihan algoritma *FloydWarsall* dan *A Star* pada aplikasi, kemudian sistem akan menampilkan rute menuju lokasi rumah sakit tujuan yang dinginkan oleh *user*



Gambar 26 Sequence digram algoritma

3.4.9 State Transisi Diagram

Adalah *state* diagram dari aplikasi yang menggambarkan langkah-langkah yang dilakukan pada sistem. *State* transisi diagram terdiri dari 6 pengolahan diantaranya yaitu pengelolaan registrasi,login,laporan darurat,pemilihan rumah sakit,run algoritma *FloydWarsall* dan *A Star*, laporan masuk. Berikut penjelasan masing- masing dari setiap bagian.

1. *State* Transisi Diagaram Registrasi

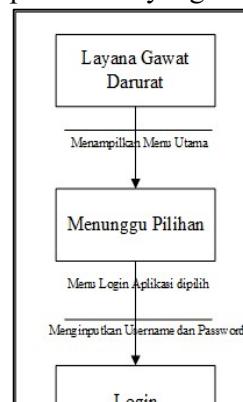
Berikut ini adalah *state* transisi diagram pada Pengelolaan Registrasi diantaranya proses *user* membuka halaman awal yang terdapat pilihan tombol registrasi



Gambar 27 Transisi diagram registrasi

2. *State* Transisi Diagaram Login

Berikut ini adalah *state* transisi diagram pada Pengelolaan Login diantaranya proses *user* membuka halaman awal yang terdapat pilihan tombol login. Pada halaman login terdapat username dan password yang diisi oleh *user*



Gambar 28 Transisi diagram login

3. State Transisi Diagram Laporan Darurat

Berikut ini adalah *state* transisi diagram pada Pengelolaan Laporan Darurat diantaranya proses *user* membuka halaman laporan darurat. Pada halaman laporan darurat *user* melakukan penginputan informasi laporan darurat



Gambar 29 Transisi diagram laporan daurrat

4. State Transisi Diagram Pemilihan Rumah Sakit

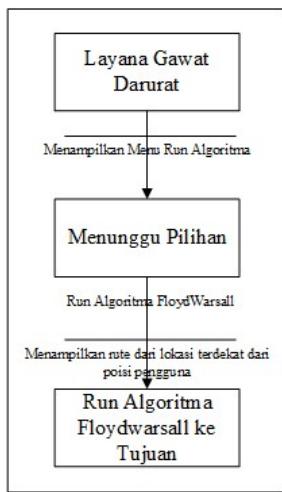
Berikut ini adalah *state* transisi diagram pada Pemilihan Rumah Sakit diantaranya proses *user* membuka halaman pemilihan rumah sakit. Pada halaman pemilihan rumah sakit *user* memilih rumah sakit yang akan dituju



Gambar 30 Transisi pemilihan rumah sait

5. State Transisi Diagram Run Algoritma *FloydWarsall*

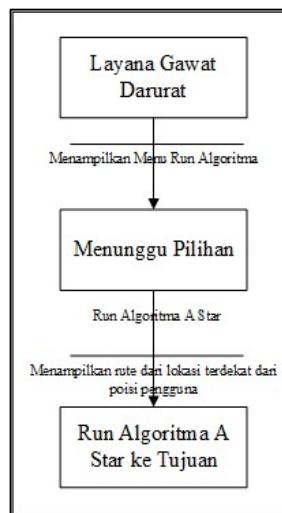
Berikut ini adalah *state* transisi diagram pada Pengelolaan Run Algoritma *FloydWarsall* yang terdapat *user* melakukan pencarian rute menuju rumah sakit tujuan dituju.



Gambar 31 Transisi run algoritma

6. State Transisi Diagram Run Algoritma *A Star*

Berikut ini adalah *state* transisi diagram pada Pengelolaan Run Algoritma *A Star* yang terdapat *user* melakukan pencarian rute menuju rumah sakit tujuan dituju



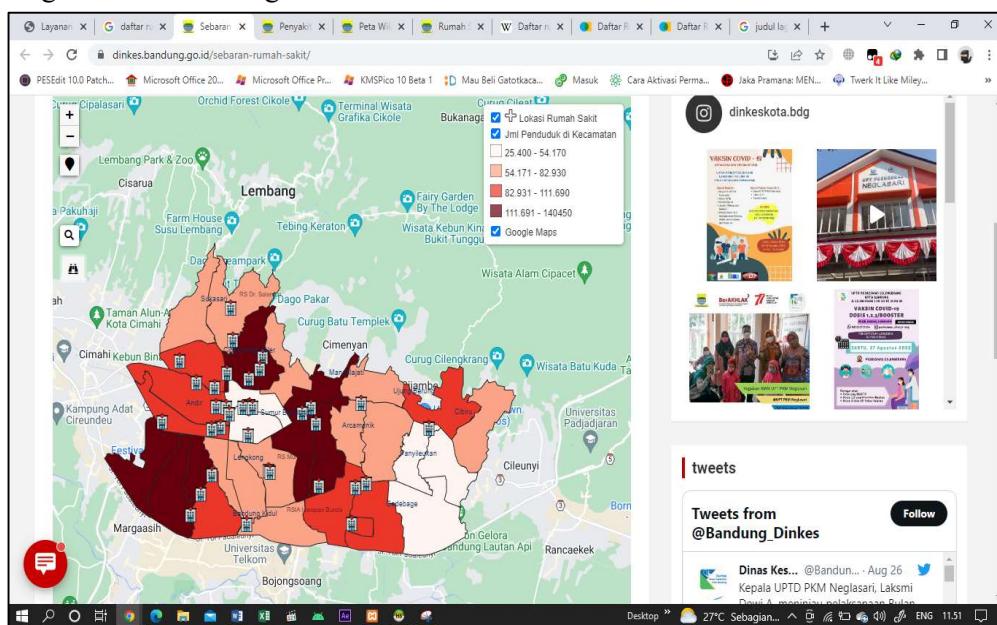
Gambar 32 Transisi diagram algoritma

3.5 Identifikasi Kebutuhan Peneltiain

Tahap ini mengidentifikasi beberapa literatur yang diperlukan untuk menggali teori yang telah berkembang dalam bidang ilmu yang berkepentingan, mencari metode metode, teknik penelitian, baik dalam mengumpulkan data atau menganalisa data untuk mengetahui sampai kemana ilmu yang berhubungan dengan penelitian telah berkembang dan sebagai dasar untuk mengaplikasikannya kedalaam penelitian ini. Dalam hal ini peneliti menelusuri literature dan sumber referensi dari jurnal jurnal sebagai tinjauan pustaka. Dalam hal ini juga peneliti mengumpulkan data data yang diperlukan seperti nama rumah sakit, alamat rumah sakit, titik koordinat yang digunakan seperti (*latitude* dan *longitude*).

3.5.1 Data Layanan Kesehatan Rumah Sakit

Data disini berupa nama rumah sakit, alamat rumah sakit, dan pengambilan titik kordinat rumah sakit seperti (*latitude* dan *longitude*). Menurut dinas wilayah Kota Bandung hanya terdapat 27 rumah sakit yang sudah terdaftar di kementerian Kesehatan Republik Indonesia yang terakhir di akses pada tanggal 18-07-2021. Berikut merupakan sebaran 27 rumah sakit dengan tingkat jumlah penduduk dengan berwarna sebagai berikut.



Gambar 33 Sebaran rumah sakit kota bandung

Dari gambar sebaran rumah sakit di wilayah bandung yang terdapat pada website resmi dinas kesehatan kota bandung berikut merupakan daftar rumah sakit sebagai berikut :

1	Nama Rumah Sakit	Alamat Rumah Sakit	Kategori	Tipe	Lat	Long
2	R.S.Mata Cicendo	Jl. Cicendo No.4 Kota Bandung	RSU	A	-6909973	107,60477
3	RS Paru Dr. H. A. Rotinsulu	Jl. Bukit Jarian No. 40 Kota Bandung 140141	RSU	A	-6878753	107,60005
4	RS Santosa Hospital Bandung Central	Jl. Kebon Jati No.38 Kota Bandung 40181	RSU	A	-6915057	107,60021
5	RSK Gigi dan Mulut FKG UNPAD	Jl. Sekeloa Selatan I Kota Bandung 40132	RSK	A	-6890299	107,61926
6	RSUP Dr. Hasan Sadikin	Jl. Pasteur No. 38 Kota Bandung 40161	RSUP	A	-6896436	107,59089
7	RS Advent Bandung	Jl. Cithampelas 161 Kota Bandung 40131	RSU	B	-6891984	107,60306
8	RS Al-Islam Bandung	Jl. Soekarno Hatta 644 Kota Bandung 40286	RSU	B	-6938975	107,66918
9	RS Immanuel Bandung	Jl. Kopo 161 Kota Bandung 40234	RSU	B	-6934907	107,59763
10	RS Khusus Ibu & Anak Kota Bandung	Jl. Astanasyar 224 Kota Bandung 40241	RSU	B	-6943301	107,59143
11	RS Lamud Dr. M. Salamun	Jl. Ciumbuleuit 203 Kota Bandung 40142	RSU	B	-6864124	107,60484
12	RS Santos Hospital Bandung Kopo	Jl. Kh. Wahid Hasyim (Kopo) NO. 461-463 Kota Bandung 40227	RSU	B	-6952495	107,58831
13	RS St. Borromeus	Jl. Ir. H.Juanda No.100 Kota Bandung 40135	RSU	B	-6893804	107,61388
14	RSK Gigi dan Mulut Maranatha	Jl. Prof Drg. Suria Sumamtri, MPH No.65 Bandung 40164	RSK	B	-6886844	107,58092
15	RSUD Al Ihsan Provinsi Jawa Barat	Jl. Kiatramanggala Bale Endah Bandung 40375	RSUD	B	-7007278	107,62388
16	RS Hermina Arcamanik	Jl. AH Nasution KM.7 No. 30 , Antapani Kota Bandung 40291	RSU	C	-6904909	107,60676
17	RS Hermina Pasteur	Jl. DR. Djundjungan No.107, Kota Bandung 40173	RSU	C	-6895926	107,58887
18	RS Kebonjati	Jl. Kebonjati No 152 Kota Bandung 40181	RSU	C	-6916018	107,5983
19	RS Mata Bandung Eye Center	Jl. Buah Batu No.147 RT 007 RW 006 Kel. Turwanga Kec. Lengkong Bandung 40264	RSU	C	-6937675	107,62384
20	RS Melinda 2	Jl. Dr. Cipto No.1 Kota Bandung 40171	RSU	C	-6906223	107,59827
21	RS Muhammadiyah Bandung	Jl. K.H. Ahmad Dahlan 93 Kota Bandung 40262	RSU	C	-6933486	107,62328
22	RS Rajawali	Jl. Rajawali No.38 Kota Bandung	RSU	C	-6912684	107,57331
23	RS St Yusup Bandung	Jl. Cikutra No. 7 Kota Bandung 40124	RSU	C	-6906696	107,64314
24	RSIA Grha Bunda	Jl. Terusan Jakarta No. 15 – 17 Kota Bandung 40281	RSIA	C	-6913256	107,64548
25	RSIA Humana Prima	Jl. Rancabolang No. 21 Kota Bandung 40286	RSIA	C	-6940779	107,66392
26	RSIA Limijati	Jl. RE. Marunduatu No.39, Kota Bandung	RSIA	C	-6906342	107,61356
27	RSK Bedah Halmahera Siaga	Jl. RE Martadinata 28 Kota Bandung 40115	RSK	C	-6906743	107,6157

Gambar 34 Daftar rumah sakit

3.5 Analisis Node

Dalam proses pencarian rute terpendek menggunakan algoritma *FloydWarsall* dan *A Star* bekerja dengan cara melakukan proses mencari bobot yang paling kecil. Pada proses pembuatannya node berbobot ini akan menggunakan titik-titik kordinat dan membuat dengan cara menarik garis atau path.



Gambar 35 Titik node

Merupakan *node* yang dibuat pada setiap persimpangan jalan raya yang kemudian nantinya sebagai rute. Pada gambar tersebut bobot yang diambil adalah panjangnya rute pada setiap *waypoint/node* yang saling terhubung dan kemudian akan digambarkan sebuah garis dari *node* ke *node* lainnya akan membentuk rute dan menghasilkan bobot pada setiap ruas jalan raya.



Gambar 36 Analisis node

Dan kordinat jalur tersebut merupakan titik - titik penghubung antara *node* satu ke *node* lainnya agar mendapatkan rute tersebut. Dari graph yang sudah digambarkan dalam visualisasi peta akan disimpan pada setiap titiknya. Berisikan titik-titik kordinat pada setiap *node* yang saling terhubung antara *node* lainnya.

3.6 Analisis Algoritma A Star

Penerapan proses algoritma *A Star* ini berupa penentuan rute terpendek menuju lokasi rumah sakit yang telah dipilih. Setelah itu algoritma *A Star* akan melakukan proses pencarian rute terpendek setelah mengetahui kordinat lokasi *user* dan lokasi tujuan.

Berikut ini adalah studi kasus pada implementasi dari algoritma *A Star* dan percobaan yang dilakukan yaitu menggambarkan dengan dari titik *user* yang di presentasikan sebagai titik asal/*origin* menuju titik akhir yang di presentasikan sebagai tujuan/*destination* dengan jalur yang dapat dilalui oleh kendaraan, cakupan daerah kota Bandung seperti pada dibawah ini:



Gambar 37 Contoh studi kasus

Dalam proses pencarian rute terpendek menggunakan algoritma *FloydWarsall* dan *A Star* bekerja dengan cara melakukan proses mencari bobot yang paling kecil ,Percobaan yang dilakukan dari titik *origin* atau awal menuju titik destination atau tujuan dengan 2 jalur yang dapat dilalui. Pada proses pembuatannya graph berbobot ini akan menggunakan titik-titik kordinat dan membuat dengan cara menarik garis atau path yang terdapat pada *Google Maps API* dan persimpangan jalan raya yang terdapat pada peta *Google Maps* akan sebagai *node*. Dan melakukan pendataan setiap titik waypoint atau persimpangan. Dari gambar di atas didapatkan data *latitude* dan *longitude* sebagai berikut :

Table 9 Latitude dan longitude

Rumah Sakit Mata Cicendo	Node	Latitude	Longitude
	v0	-6,91261	107,60314
	v1	-6,91236	107,59797
	v2	-6,90647	107,59755
	v3	-6,90717	107,60443
	v4	-6,91048	107,60411
	v5	-6,90956	107,604773

Didapatkan data *latitude* dan *longitude* pada tabel diatas selanjutnya melakukan pencarian rute. Terdapat persimpangan pada setiap *node* yang dapat dilalui dan melakukan perhitungan jarak antara titik untuk mendapatkan nilai *heuristik* dengan menggunakan rumus persamaan *Euclidean Distance* berikut.

$$d = \sqrt{(lat_1 - lat_2)^2 + (long_1 - long_2)^2}$$

Pada persamaan diatas diketahui :

- d adalah Jarak *Euclidean*
- lat_1 adalah Nilai *latitude* titik asal/origin
- lat_2 adalah Nilai *latitude* titik tujuan/destination
- $long_1$ adalah Nilai *longitude* titik asal/origin
- $long_2$ adalah Nilai *longitude* titik tujuan/destination

Dari perhitungan jarak antara dua titik dengan diketahui satuan jarak 1 derajat *latitude longitude* sama dengan 111.32 km yang dikoversikan dalam satuan km, kemudian dikalikan dengan jarak *Euclidean* yang sudah didapatkan jadi seperti : jarak (km) = $d \times 111.32$. Hasil tersebut akan dilakukannya pembulatan yang bertujuan untuk menyesuaikan dengan hasil dari *Google Maps*. Data jarak yang diperoleh pada Tabel selanjutnya akan melakukan proses pencarian rute pada Algoritma *A Star* dengan rumus persamaan sebagai berikut:

$$f(n) = g(n) + h(n)$$

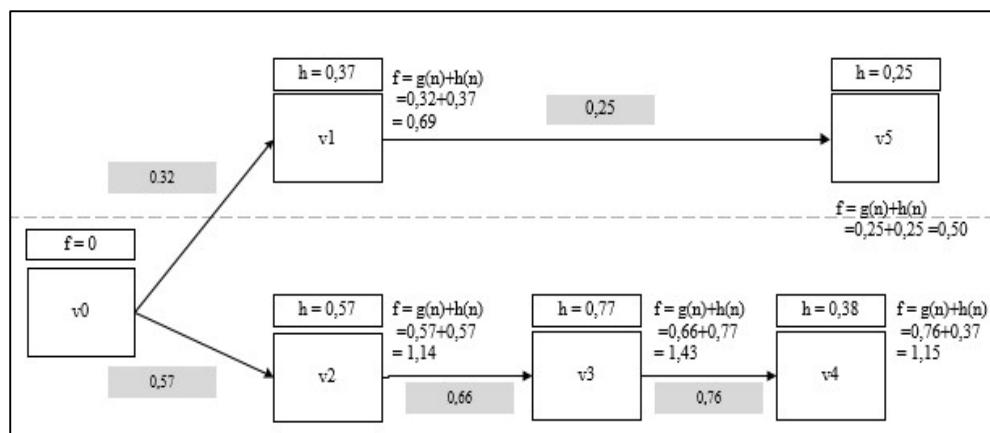
Pada persamaan diatas diketahui :

- $f(n)$ adalah Total biaya
- $g(n)$ adalah Biaya yang sudah dihitung dari sebelumnya hingga keadaan n
- $h(n)$ adalah Estimasi biaya untuk sampai menuju tujuan dari keadaan n

Table 10 Hasil jarak antar node

Rumah Sakit Mata Cicendo	Node	Jarak Euclidean Distance(d)	$(d) \times 111,32$	Jarak (km)/h(n)	jarak (m)	Peta (m)	Peta (km)
	v0-v1	0,0051761	0,32619232	0,32	320	335	0,35
	v2-v3	0,0069152	0,7697778	0,77	770	660	0,66
	v3-v4	0,0033263	0,38025032	0,38	380	766	0,76
	v4-v5	0,0009991	0,11120868	0,11	110	260	0,26
	v1-v5	0,0005669	0,25503308	0,25	250	201	0,2
	v0-v2	0,00517604	0,57619677	0,57	570	579	0,57

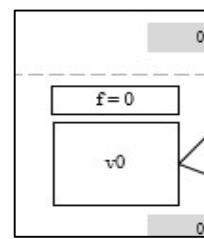
Supaya mempermudah dalam melakukan proses pencarian rute maka dipresentasikan ke dalam model *graph* sebagai berikut :



Gambar 38 Model *graph* antar node

Algoritma *A Star* ini akan melakukan beberapa tahapan yaitu mengecek pada setiap titik dengan fungsi *open list* dan *close list* yang mencari biaya minimum atau terkecil.

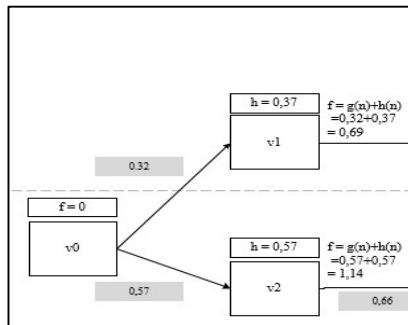
Langkah pertama, karena di *open list* hanya terdapat 1 simpul (yaitu v_0) v_0 sebagai titik awal titik *client* atau *user*.



Gambar 39 Titik node v_0

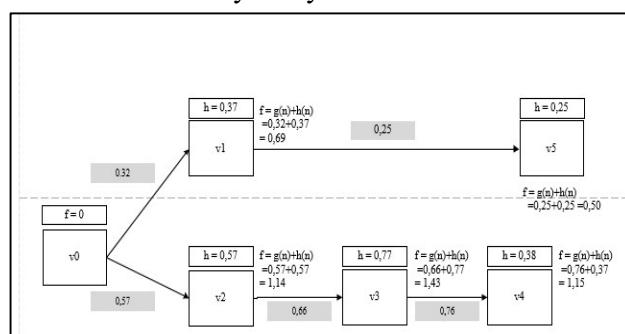
Langkah kedua, pada graf diatas terdapat dua *node* yaitu *node* v_1 dan v_2 yang kemudian dimasukkan kedalam *open list*, pada *node* v_2 dengan biaya minimum/terkecil (yaitu 1,14) sedangkan *node* v_1 dengan biaya minimum/terkecil (yaitu 0,69) terpilih sebagai *BestNode* karena memiliki biaya minimum/terkecil dari

node v2 dan kemudian dipindahkan kedalam *close list* karena *node* v1 tersebut selesai diproses kemudian suksesor/cabang pada v1 dibuka yaitu v5 dan dimasukkan ke *open list*.



Gambar 40 Titik node v1 bestnode

Langkah ketiga, pada *node* v1 dengan biaya minimum/terkecil (yaitu 0.69) kemudian suksesor/cabang v1 dibuka yaitu v5 dan sebagai *BestNode Goal*. Karena *BestNode* sama dengan *Goal* atau tujuan, artinya solusi untuk mencapai tujuan sudah ditemukan. Rute dan total biaya bisa ditelusuri kembali dari v5 ke v0 karena pada setiap simpul hanya mempunyai satu parent dan pada setiap linstasannya itu mempunyai informasi besar kecilnya biaya.



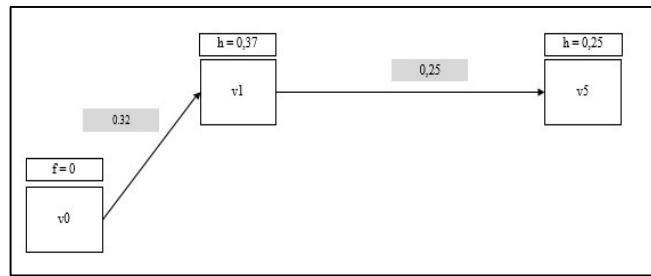
Gambar 41 Titik node v5 sebagai bestnode goal

Rute dan total biaya bisa ditelusuri kembali dari v5 ke v0 karena pada setiap simpul hanya mempunyai satu parent dan pada setiap linstasannya itu mempunyai informasi besar kecilnya biaya.

[1]			[2]		
Open List			Close List		
Tujuan	Harga	Via	Tujuan	Harga	Via
v1	0,69	v0	v0	0	v0
v2	1,14				
Open List			Close List		
Tujuan	Harga	Via	Tujuan	Harga	Via
v2	1,14	v0	v0	0	v0
v3	1,43	v2	v1	0,69	v1
v4	1,15	v3	v5	0,5	v1

Gambar 42 Open list dan close list

Dari hasil proses yang telah dilakukan diatas algoritma *a star* menggunakan fungsi nilai *heuristic* maka dapat disimpulkan rute mana yang mempunyai biaya minimum menuju lokasi tujuan dengan biaya minimum v0-v1-v5 serta total biaya (f) yaitu $32+0,25$ didapatkan hasil (f) = 0,57.



Gambar 43 Hasil perhitungan *a star*

Hasil implementasi dari *smartphone* dari algoritma *a star* yang sudah diintregasikan dengan *Google Maps* di definisikan A sebagai titik awal dan titik B sebagai titik akhir sistem akan memberikan dengan garis warna biru.



Gambar 44 Implementasi algoritma *a star*

Pada *Gambar* diatas menunjukkan rute yang ditentukan oleh sistem yang dibangun dengan menggunakan algoritma a star dalam proses menentukan rute didapatkan rute utama dari titik A yang didefinisikan sebagai titik awal posisi *user* dengan lokasi Jln.Kebon Kawung No.14 d.pasir Keliki Kec.Cicendo Kota Bandung Jawa Barat 40171, Indonesia, menuju titik B yang didefinisikan sebagai titik akhir tujuan dengan lokasi Jln.Cicendo No.4b babakan Ciamis Kec.Sumur Bandung Kota Bandung Jawa Barat 40117, Indonesia, dengan jarak yang di tempuh 0,50 KM dan estimasi waktu tempuh durasi selama 6 menit.

3.7 Analisis Algoritma Floyd Warshall

Penerapan proses algoritma *FloydWarsall* ini yaitu pencarian rute terpendek menuju lokasi rumah sakit yang dipilih oleh *user*. Algoritma *Floydwarsall* menghitung dan membandingkan semua lintasan pada *graph* setiap dari semua simpul, setelah ditemukannya lintasan lalu informasi akan dikembalikan oleh sistem ke *user*.

Selain itu algoritma ini menggunakan matriks $n \times n$. Masih dengan studi kasus graph yang sama seperti sebelumnya dalam penentuan rute terpendek atau pencarian rute terpendek didapatkan setiap titik *waypoint* persimpangan.

Table 11 Latitde dan longitude.

Node	Latitude	Longitude
v0	-6,91261	107,60314
v1	-6,91236	107,59797
v2	-6,90647	107,59755
v3	-6,90717	107,60443
v4	-6,91048	107,60411
v5	-6,909973	107,604771

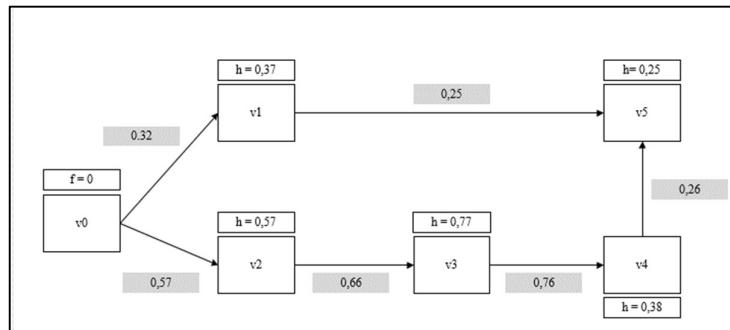
Setiap node titik waypoint tersebut saling terhubung dan memiliki nilai bobot yang didapatkan dari *eucludien distance*.

Table 12 Titik waypoint node

No	Node	Bobot
1	v0-v1	0,32

2	v0-v2	0,57
3	v1-v5	0,25
4	v2-v3	0,66
5	v3-v4	0,76
6	v4-v5	0,26

Dari hasil tabel yang berisikan hubungan setiap *node* dan bobotnya yang telah ditentukan, agar mempermudah dipresentasikan dalam bentuk graph



Gambar 45 Graph floyd warshall

Selanjutnya melakukan proses perhitungan matriks antar titik seperti berikut dibawah ini.

Table 13 Perhitungan matrix antar titik

		Node Tujuan					
		v0[1]	v1[2]	v2[3]	v3[4]	v4[5]	v5[6]
Node	v0[1]	0	0,32	0,57	—	—	—
	v1[2]	0,32	0	—	—	—	0,25
yang	v2[3]	0,32	—	0	0,66	—	—
Dipilih	v3[4]	—	—	0,66	0	0,76	—
W=0	v4[5]	—	—	—	0,76	0	0,26
	v5[6]	—	0,25	—	—	0,26	0

Terdapat rute berbobot dan persimpangan, pada penyelesaian studi kasus ini akan menggunakan tabel, kolom pada tabel akan menjelaskan *node* yang terdapat pada *graph* sedangkan baris menjelaskan *node* yang dipilih.

Pada tabel diatas setelah itu akan dilakukan tahapan iterasi mengunjungi seluruh rute yang akan dilalui sampai dengan iterasi terakhir atau jumlah total titik yang terbuat pada sebuah *graph* yang sudah terbentuk dipresentasikan ke dalam matriks bobot berukuran $n \times n$ dengan menggunakan persamaan $w[i,j] > w[i,k] + w[k,j]$. Dimulai dengan iterasi pertama.

Diketahui $w=0$ selanjutnya melakukan tahapan iterasi hingga pada iterasi terakhir jumlah iterasi yaitu jumlah total titik yang akan dicari dari titik asal iterasi ke n dengan menggunakan persamaan sebagai berikut.

Persamaan						
$w[i,j] > w[i,k] + w[k,j]$						
Berdasarkan Gambar Graph						
$k = 0$	1	2	3	4	5	6
$i = 1$	2	3	4	5	6	
$j = 1$	2	3	4	5	6	
Matriks Hubungan Graph						
$K=0$						

Diketahui :

- w adalah untuk menentukan berapa banyak matrix yang dibuat
- i adalah jumlah simpul yang terbuat pada graph/ titik awal
- j adalah jumlah simpul yang terbuat pada graph / titik akhir
- k adalah jumlah integrasi pada simpul yang dibuat

Table 14 Perhitungan iterasi pertama

		Node Tujuan					
		$v0(1)$	$v1(2)$	$v2(3)$	$v3(4)$	$v4(5)$	$v5(6)$
k1	$v0(1)$	0	0,32	0,57	—	—	—
	$v1(2)$	0,32	0	—	—	—	0,25
	$v2(3)$	0,32	—	0	0,66	—	—
	$v3(4)$	—	—	0,66	0	0,76	—
	$v4(5)$	—	—	—	0,76	0	0,26
	$v5(6)$	—	0,25	—	—	0,26	0

Pada tabel diatas melakukan iterasi pertama dan integrasi pertama disini node $v0 = 0$ adalah sebagai lokasi awal *user* algoritma ini akan berhenti setelah

melakukan semua iterasi untuk mendapatkan rute mana yang akan dilalui menuju lokasi tujuan dan akan terus melakukan proses yang sama.

Table 15 Perhitungan iterasi kedua

		Node	Node Tujuan					
			v0[1]	v1[2]	v2[3]	v3[4]	v4[5]	v5[6]
k2	Node	v0[1]	0	0,32	0,57	—	—	—
		v1[2]	0,32	0	0,89	—	—	0,25
	yang	v2[3]	0,32	0,64	0	0,66	—	—
		v3[4]	—	—	0,66	0	0,76	—
	Dipilih W=1	v4[5]	—	—	—	0,76	0	0,26
		v5[6]	—	0,25	—	—	0,26	0

Pada Tabel iterasi kedua melakukan iterasi pada *node v1-v1* adalah = 0, *node* yang terhubung langsung dengan *node v1* tidak akan dihitung dan akan bernilai bobot tetap, pada tabel diatas mengalami perubahan nilai bobot *node* dari infinity menjadi 0,04 dan 0,89 pada tahapan iterasi diatas maka nilai bobot yang berubah akan dimasukkan kedalam tabel dan akan di proses pada tahapan iterasi berikutnya..

Table 16 Tahapan iterasi ketiga

		Node	Node Tujuan					
			v0[1]	v1[2]	v2[3]	v3[4]	v4[5]	v5[6]
k3	Node	v0[1]	0	0,32	0,57	—	—	0,57
		v1[2]	0,32	0	0,89	—	—	0,25
	yang	v2[3]	0,32	0,64	0	0,66	—	0,89
		v3[4]	—	—	0,66	0	0,76	—
	Dipilih W=2	v4[5]	—	—	—	0,76	0	0,26
		v5[6]	0,57	0,25	1,14	—	0,26	0

Pada Tabel iterasi ketiga melakukan iterasi pada *node v2-v2* adalah = 0, *node* yang terhubung langsung dengan *node v2* tidak akan dihitung pada matrix dan akan bernilai tetap, pada tabel diatas mengalami perubahan bobot *node* dari infinity menjadi 1,14 pada node tersebut berubah dan akan dimasukkan ke dalam tabel matrix berikutnya dan diproses pada tahap berikutnya dan seterusnya hingga iterasi terakhir.

Table 17 Tahapan iterasi keempat

		Node	Node Tujuan					
			v0[1]	v1[2]	v2[3]	v3[4]	v4[5]	v5[6]
k4	Node	v0[1]	0	0,32	0,57	1,23	—	0,57
		v1[2]	0,32	0	0,89	1,55	—	0,25
	yang	v2[3]	0,32	0,64	0	0,66	—	0,89
		v3[4]	0,98	1,3	0,66	0	0,76	1,55
	Dipilih W=3	v4[5]	—	—	—	0,76	0	0,26
		v5[6]	0,57	0,25	1,14	1,8	0,26	0

Pada Tabel iterasi keempat melakukan iterasi pada node v3-v3 = 0 nol dan node yang terhubung dengan node v3 akan bernilai tetap, pada tabel tersebut terdapat perubahan nilai bobot node infinity = 1,23 , 1,55, dan 1,8, dari hasil perubahan tersebut akan dimasukkan kedalam tabel matix berikutnya dan diproses pada tahap iterasi berikutnya.

Table 18 Tahapan iterasi kelima

		Node Tujuan						
		Node	v0(1)	v1(2)	v2(3)	v3(4)	v4(5)	v5(6)
k5	Node	v0(1)	0	0,62	0,57	1,23	1,99	0,57
	v1(2)	0,32	0	0,89	1,55	2,31	0,25	
	yang	v2(3)	0,32	0,64	0	0,66	1,42	0,89
	Dipilih	v3(4)	0,98	1,3	0,66	0	0,76	1,55
	W=4	v4(5)	1,74	2,06	1,42	0,76	0	0,26
		v5(6)	0,57	0,25	1,14	1,8	0,26	0

Pada Tabel iterasi kelima melakukan iterasi pada node v4-v4=0 nol, node yang terhubung dengan v4 akan bernilai tetap pada tabel diatas mengalami beberapa perubahan nilai bobot seperti nilai infinity = 1,99,2,31, dan 1,42 pada perubahan tersebut akan dimasukkan kedalam matrix iterasi berikutnya dan diproses kembali.

Table 19 Tahapan iterasi keenam

		Node Tujuan						
		Node	v0(1)	v1(2)	v2(3)	v3(4)	v4(5)	v5(6)
k6	Node	v0(1)	0	0,62	0,57	1,23	1,99	0,57
	v1(2)	0,32	0	0,89	1,55	2,31	0,25	
	yang	v2(3)	0,32	0,64	0	0,66	1,42	0,89
	Dipilih	v3(4)	0,98	1,3	0,66	0	0,76	1,02
	W=5	v4(5)	1,74	2,06	1,42	0,76	0	0,26
		v5(6)	0,57	0,25	1,14	1,02	0,26	0

Pada Tabel iterasi keenam merupakan tahapan iterasi terakhir yang dilakukan pada iterasi iterasi sebelumnya dan node $v_5 - v_5 = 0$ artinya node yang terhubung dengan langsung v_5 akan bernilai tetap dan tidak akan berubah, pada tabel diatas terdapat perubahan nilai node $v_3 - v_5$, dan $v_5 - v_3$ seperti infinity menjadi 1,02 dan 1,02 pada perubahan tersebut kedalam matrix dan akan dilihat bobot terkecil yang diperoleh.

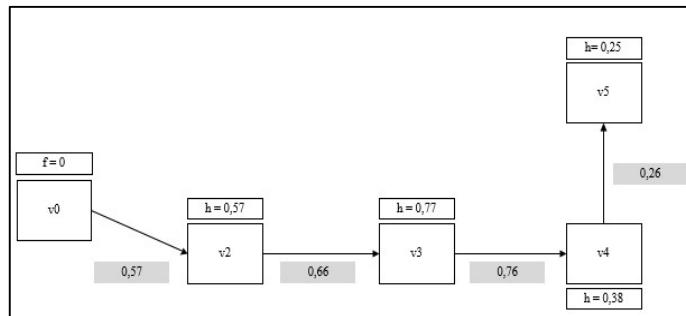
1		$w[i,j] > w[i,k] + w[k,j]$	$w[i,k] + w[k,j]$
$w_0[i,j]$			
$w_0[2,3]$			$w[2,1] + w[1,3]$
—	>		$0,32 + 0,57 = 0,89$
$w_0[2,4]$			$w[2,1] + w[1,4]$
—	=		$0,32 + - = -$
$w_0[2,5]$			$w[2,1] + w[1,5]$
—	=		$0,32 + - = -$
$w_0[2,6]$			$w[2,1] + w[1,6]$
0,25	<		$0,32 + - = -$
$w_0[3,2]$			$w[3,1] + w[1,2]$
—	>		$0,32 + 0,32 = 0,64$

Nilai pada tabel akan berubah setelah melakukan pengcekan pada persamaan seperti pada Tabel persamaan karena hasil nilai bobot tak hingga lebih besar dari hasil jumlah maka nilai bobot pada tabel akan berubah, tetapi jika hasilnya lebih kecil atau sama maka nilai bobot nya akan tetap atau tidak akan berubah begitupun pada tahapan seterusnya.

Table 20 Hasil perhitungan iterasi

		Node Tujuan							
	Node	$v_0(1)$	$v_1(2)$	$v_2(3)$	$v_3(4)$	$v_4(5)$	$v_5(6)$		
	$v_0(1)$	0	0,62	0,57	1,23	0,83	0,57		
	$v_1(2)$	0,32	0	0,69	1,27	2,31	0,25		
yang	$v_2(3)$	0,32	0,64	0	0,66	1,42	0,89		
Dipilih	$v_3(4)$	0,98	1,27	0,66	0	0,76	1,02		
	$v_4(5)$	0,83	0,51	1,4	0,76	0	0,26		
	$v_5(6)$	0,57	0,25	1,14	1,02	0,26	0		

Jadi setelah melakukan semua tahapan iterasi sampai dengan iterasi terakhir dari node v_0 sebagai titik asal menuju ke node v_5 sebagai titik akhir tujuan didapatkan hasil pada tabel diatas atau jadi rute yang didapatkan dan harus dilalui adalah $v_0 - v_2 - v_3 - v_4 - v_5$ dengan total biaya sebesar 2,25 KM. Jika digambarkan maka sebagai berikut :



Gambar 46 Hasil perhitungan floydwrhsall

Hasil implementasi *smartphone* pada *algoritma floydwarshall* yang sudah diintregasikan dengan *Google Maps API* di definisikan A sebagai titik awal dan titik B sebagai titik akhir atau tujuan sistem akan memberikan dengan garis warna hijau dapat dilihat pada *Gambar* sebagai berikut.



Gambar 47 Hasil implementasi floydwarshall

Pada *Gambar* diatas menunjukkan rute yang ditentukan oleh sistem yang dibangun dengan menggunakan algoritma Floyd Warshall dalam proses menentukan rute didapatkan rute utama dari titik A yang didefinisikan sebagai titik awal posisi *user* dengan lokasi Jln.Kebon Kawung No.14 d.pasir Keliki Kec.Cicendo Kota Bandung Jawa Barat 40171, Indonesia, menuju titik B yang didefinisikan sebagai titik akhir tujuan dengan lokasi Jln.Cicendo No.4b babakan Ciamis Kec.Sumur Bandung Kota Bandung Jawa Barat 40117, Indonesia, dengan jarak yang di tempuh 2,49 KM dan estimasi waktu tempuh durasi selama 7 menit.

3.8 Analisis Haversine

Pada proses persamaan *haversine* ini yang bertujuan digunakan untuk menghitung antara dua titik dipermukaan bumi dengan menggunakan garis lintang (*longitude*) dan garis bujur (*latitude*) sebagai inputan, lalu *haversine* ini akan menghasilkan jarak *real* perkiraan terdekat antara dua titik dari lokasi terkini *user* menuju lokasi tujuan rumah sakit. Di asumsikan bahwa bumi itu berbentuk bulat dengan jari – jari yang R 6371 km.

Keterangan :

long = Longitude (lintang)

lat = Latitude (Bujur)

d = Jarak

R = Radius Bumi = 6371 km

1 derajat = 0,0174532925 radian

Dari penjabaran diatas maka didapatkan rumus dalam *query* SQL dibawah ini sebagai berikut.

$$(6371 * ACOS (SIN (RADIANS (lats)) * SIN (RADIANS ($lat)) + COS(RADIANS(lngs-$lng)) *COS*RADIANS(lats)) * COS (RADIANS($lat))))$$

Nilai π adalah nilai bilangan real yang digunakan untuk menghitung lingkaran bisa dengan $22/7$ atau dengan 3,14 maka menjadi :

$$\Delta lat : \frac{3,14}{180} \cdot (-0,120594 - (-0,120647)) = 0,00000092456$$

$$\Delta long : \frac{\pi}{180} \cdot (longitude tujuan - longitude awal)$$

$$\Delta long : \frac{3,14}{180} \cdot (1,878057 - 1,878029) = 0,00000048844$$

Selanjutnya setelah didapatkan nilai delta lat dan long kemudian mencari nilai a:

$$a ; \sin\left(\frac{\Delta lat}{2}\right)^2 + \cos(lat1) \cdot \cos(lat2) \cdot \sin\left(\frac{\Delta long}{2}\right)^2$$

$$a : \sin\left(\frac{\Delta lat}{2}\right)^2 = \left(\frac{0,00000092456}{2}\right) = 0,00000046228$$

$$c : \cos(lat1) \cdot \cos(lat2) \cdot \sin\left(\frac{\Delta long}{2}\right)$$

$$c : \cos(-0,120647) \cdot \cos(-0,120594) \cdot \sin\left(\frac{0,00000048844}{2}\right) = 0,00000024068$$

Setelah diketahui Maka :

$$d : R \cdot 2 \cdot \sin(\sqrt{a + c})$$

$$d : 6371 \cdot 2 \cdot \sin(\sqrt{0,00000046228 + 0,00000024068}) = 0,38$$

Berdasarkan dari hasil perhitungan diatas maka jarak real garis lurus yang dihasilkan dari titik a menuju titik adalah 0,38 km.

Hasil implementasi *smartphone* pada dalam proses perhitungan perkiraan jarak real antara dua titik dari lokasi terkini *user* menuju lokasi tujuan menggunakan *haversine formula*.



Gambar 48 Hasil haversine formula

Dari hasil gambar diatas sistem akan menampilkan informasi perkiraan jarak *real* dari lokasi terkini *user* menuju lokasi tujuan dengan total jarak 0,34km. Selain itu sistem dapat mencari rumah sakit terdekat dari lokasi terkini *user* menuju tujuan berdasarkan jarak terdekat.

3.9 Evaluasi

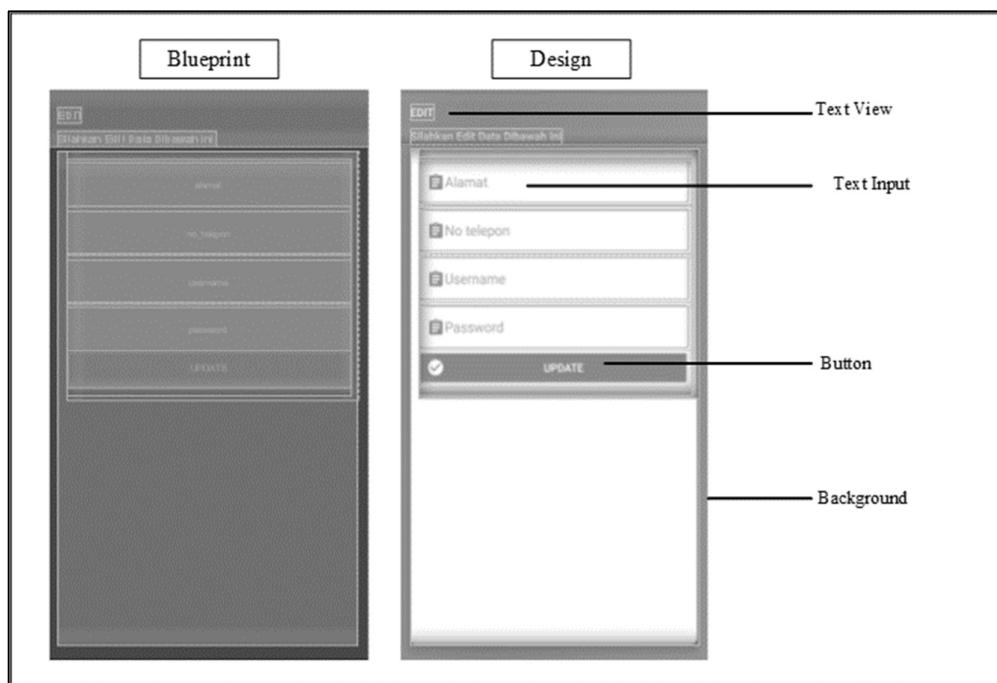
Pada tahap ini yaitu mengevaluasi hasil dari prototype dan bertujuan untuk mengetahui berbagai kekurangan serta kelemahan selama proses pembuatan atau pembangunan sebuah sistem/aplikasi tersebut. Dengan melakukan tahapan evaluasi maka dapat diketahui kekurangan dan kelemahan dari sistem atau aplikasi yang dibangun apakah sesuai dengan perancangan awal, yang pada akhirnya dapat dipergunakan sebagai tolak ukur untuk melakukan perbaikan dan batasan dalam proses pembangunan sistem atau aplikasi

3.9.1 Perancangan User Interface

Tahap ini dibuatnya perancangan desain *user interface* pada aplikasi/sistem yang dibuat, sehingga ditentukan menu dan komponen apa saja yang dibutuhkan pada masing- masing menu, hal tersebut bertujuan untuk mempermudah interaksi *user* dengan aplikasi/sistem pada saat digunakan. Menu tersebut akan dijelaskan secara masing-masing sebagai berikut.

- Perancangan *user interface* menu registrasi

Pada perancangan desain menu registrasi ini terdapat beberapa komponen didalamnya antara lain 2 buah *text view* 4 buah *,text input field*, 1 buah *button* dan *background*.

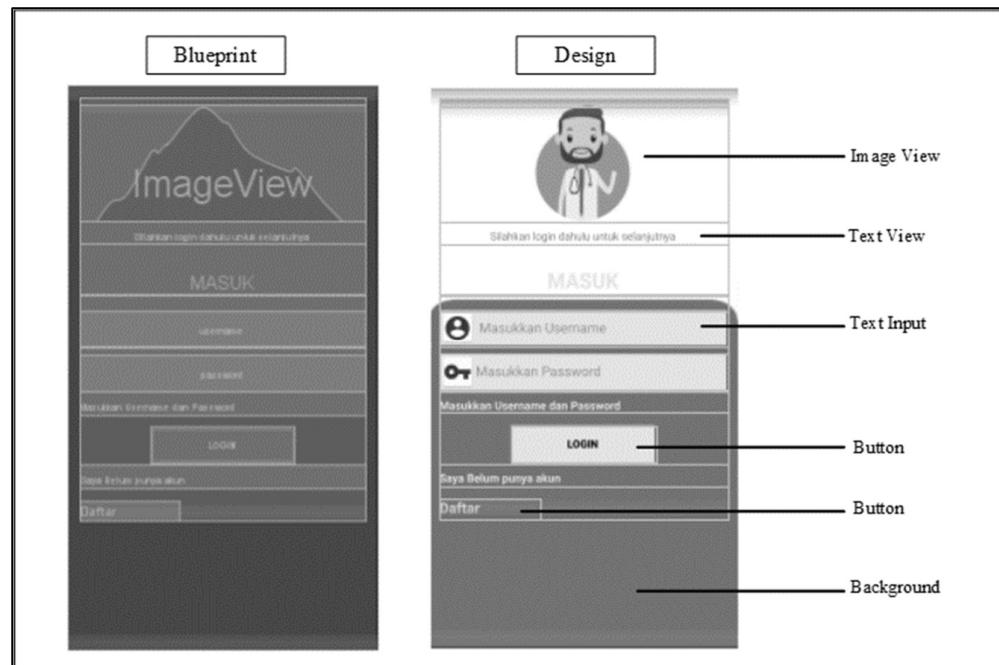


Gambar 49 Perancangan UI registrasi

Dijelaskan pada Gambar 46 bahwa bagian *text* yang digunakan didalam menu ini adalah jenis *font sans-serif*, dibagian *button* menggunakan warna #8A2BE2 sedangkan pada *background* menggunakan warna #5F9EA0. Terdapat *card rectangle* yang bertujuan untuk menampung semua komponen pada menu dengan tinggi 100dp dan radius 8dp, pada setiap fungsi komponennya diberikan *icon* sebagai tambahan.

- Perancangan *user interface* menu login

Pada perancangan desain menu login ini terdapat beberapa komponen didalamnya antara lain 1 buah *image view*, 4 buah *text view*, 2 buah *button*, 2 buah *text input field* dan *background*.

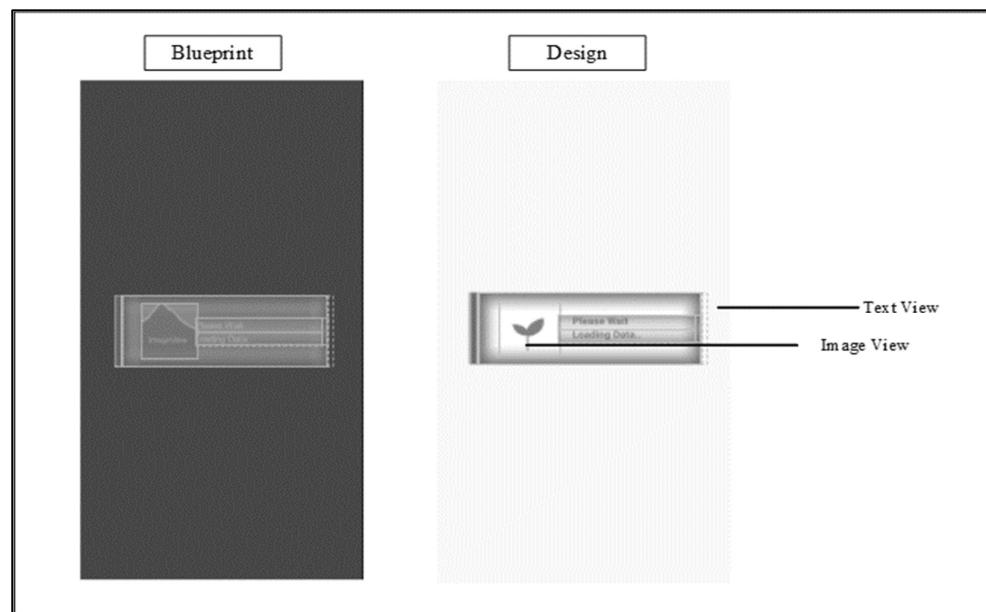


Gambar 50 Perancangan UI login

Dijelaskan pada Gambar 47 text yang digunakan adalah jenis *font sains-serif* serta memiliki 2 background, background utama menggunakan warna #FFFFFF dan background *card* warna #8A2BE2. Memiliki 2 fungsi *button*, *button login* dengan warna #FFFFFF dan *button* kedua warna #FFFFFF, serta terdapat *icon* dengan ukuran 150dp yang digunakan sebagai tambahan dan memperjelas serta tampilan menarik pada menu tersebut.

- Perancangan *user interface splash screen*

Pada peracangan ini terdapat splash screen yang dibuat sebagai transisi antara *activity* pada setiap menu, terdapat *image view* dan *text view*.

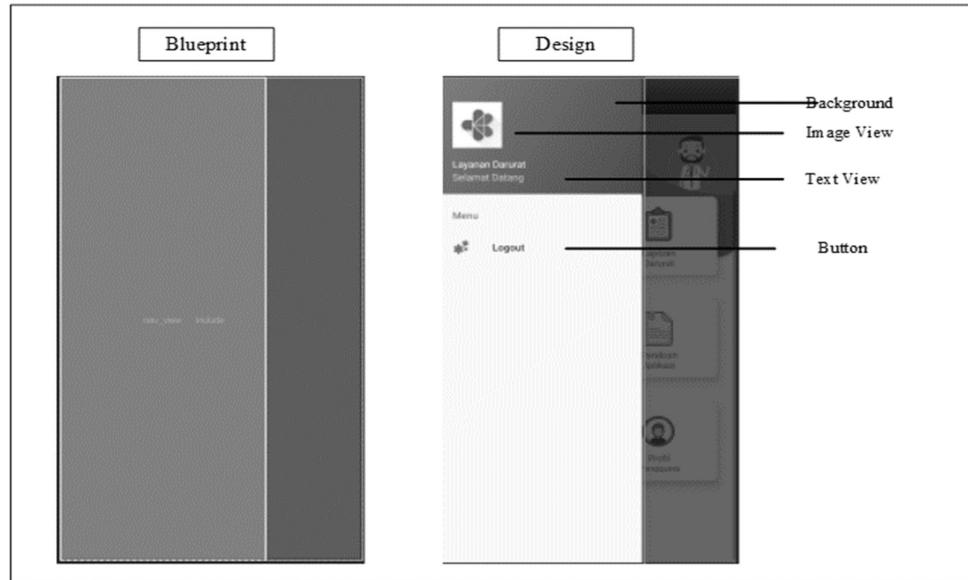


Gambar 51 Perancangan UI Screen

Dijelaskan pada Gambar 48 terdapat *icon* yang digunakan dengan ukuran 30dp pada *splash screen* ini sebagai transisi penghubung antara *activity*.

- Perancangan *user interface* navigasi bar

Pada perancangan desain navigasi bar ini terdapat komponen diantaranya *background, image view, text view* dan *button*.

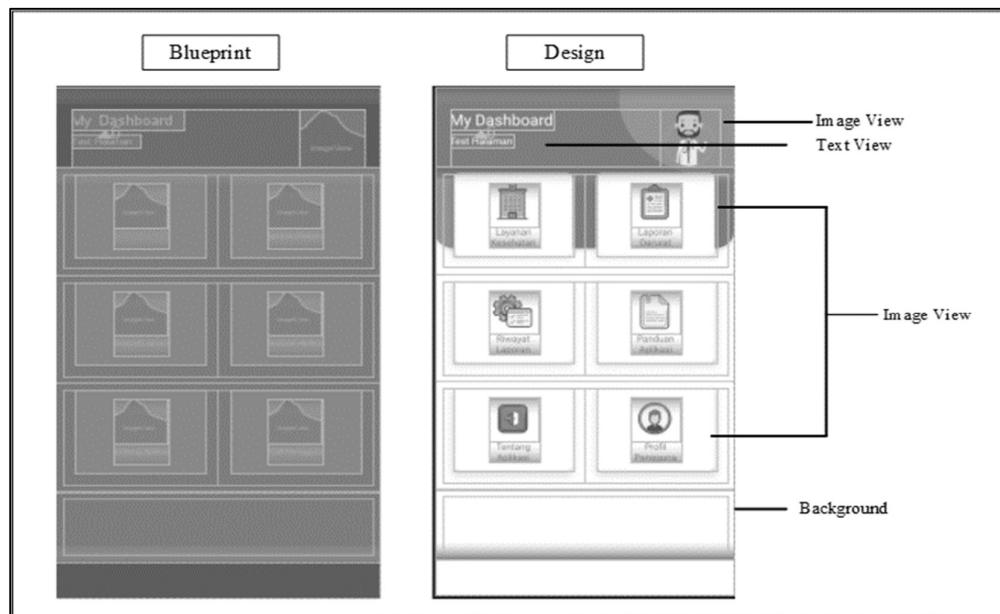


Gambar 52 Perancangan UI navigasi bar

Dijelaskan pada Gambar 49 text yang digunakan adalah jenis *font sains-serif*, pada *background* dengan warna #8A2BE2 dan *icon* yang digunakan berukuran 30x30dp seta *button* dengan lebar 80dp.

- Perancangan *user interface* menu utama

Pada perancangan desain menu utama ini terdapat beberapa komponen didalam diantaranya 7 buah *image view*, 2 buah *text view* dan *background*.

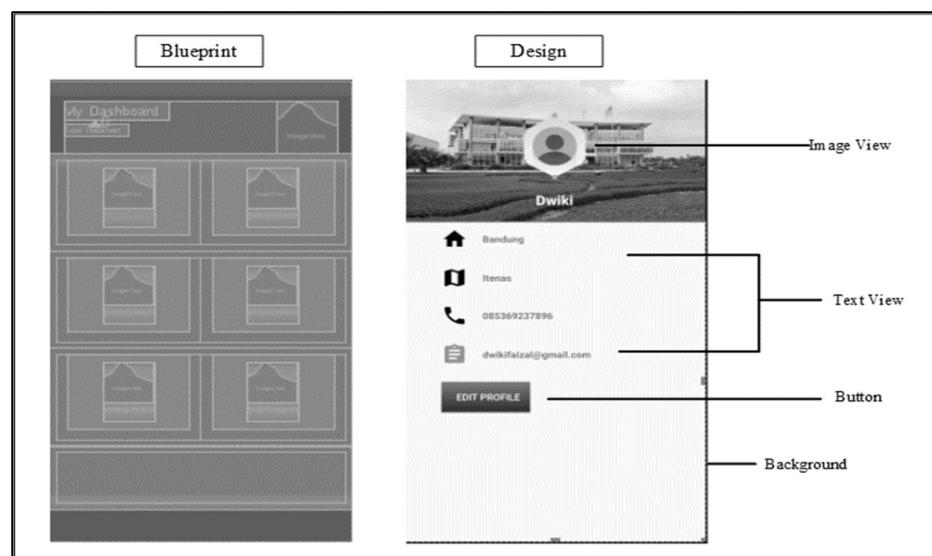


Gambar 53 Perancangan menu utama

Dijelaskan pada Gambar 53 diatas text yang digunakan adalah jenis *font sains-serif*, pada menu utama ini terdapat background terdiri dari 2 warna yaitu #5F9EA0 dan #FFFFFF selain itu pada menu ini terdapat banyak *icon* yang digunakan dengan ukuran 30x30 dan serta dalam *button* semuanya memakai *image button* sehingga jadi lebih menarik.

- Perancangan *user interface* menu profil

Pada perancangan desain menu profil ini memiliki beberapa komponen didalam yaitu *image view*, 4 buah *text view*, *button*, dan *background*.



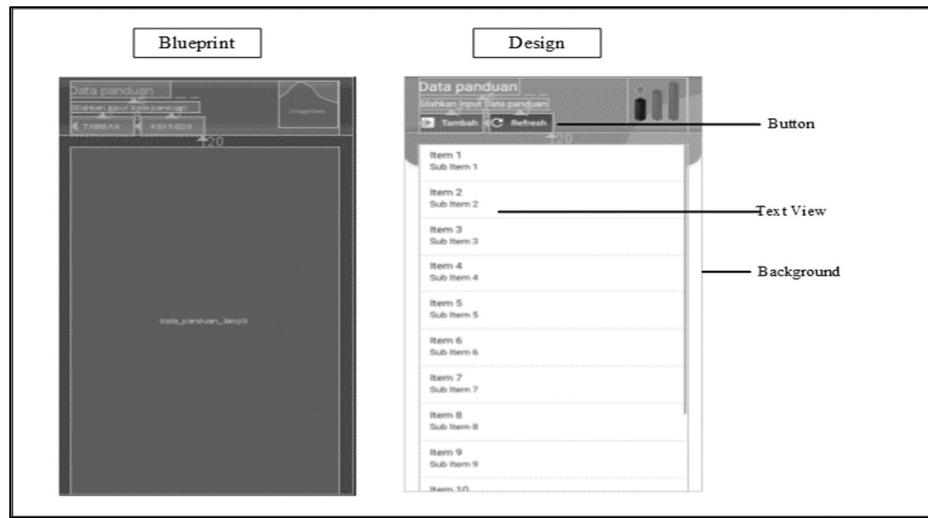
Gambar 54 Perancangan UI profil

Dijelaskan pada Gambar 54 diatas text yang digunakan adalah jenis *font sains-serif*, dimenu profil ini memiliki *background* warna #FFFFFF dan *image view*

menampilkan foto, dan terdapat *button* berwarna #5F9EA0 dengan panjang *rectangle* 30dp.

- Perancangan *user interface* menu panduan

Pada perancangan desain menu panduan ini memiliki beberapa komponen didalam yaitu 2 buah *button*, *text view*, dan *background*.

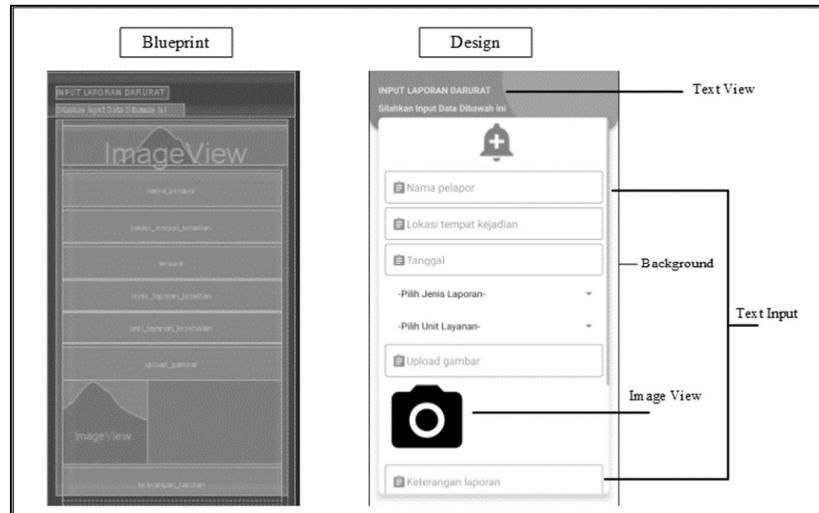


Gambar 55 Perancangan UI paduan

Dijelaskan pada Gambar 55 diatas text yang digunakan adalah jenis *font sans-serif*, dimenu panduan ini memiliki *background* dengan degradasi 2 warna #FFFFFF dan #5F9EA0, terdapat *button* berwarna #5F9EA0 dengan panjang *rectangle* 25dp.

- Perancangan *user interface* menu input laporan darurat

Pada perancangan desain menu input laporan darurat ini memiliki beberapa komponen didalamnya yaitu 1 buah *button*, *text view*, 7 buah *text input filed*, 2 buah *combobox* dan *background*.

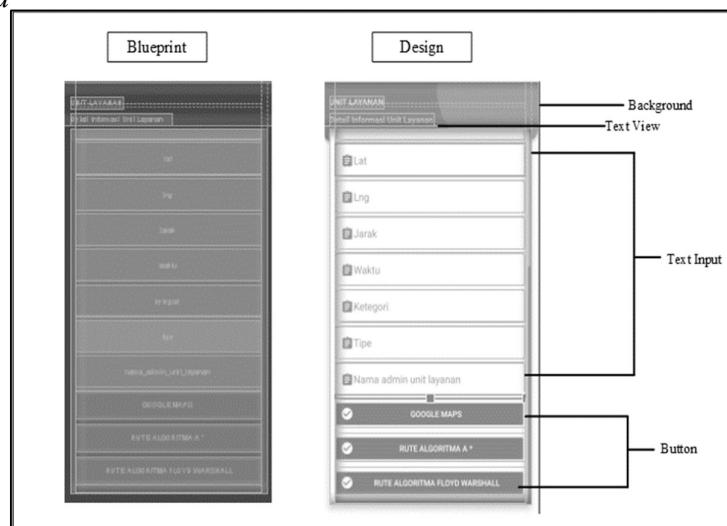


Gambar 56 Perancangan laporan darurat

Dijelaskan pada Gambar 56 diatas text yang digunakan adalah jenis *font sains-serif*, pada menu ini dilakukannya penacarian/pemilihan rumah sakit dalam berbentuk *list* dan dapat mempermudah dalam memilih, *list* rumah sakit tersebut akan ditampung dalam sebuah *card* sesuai jumlah data rumah sakit yang tersedia. Terdapat *background* pada menu ini dengan degradasi 2 warna #FFFFFF dan #5F9EA0, lalu terdapat *button* berwarna #5F9EA0 dengan panjang *rectangle* 5-10dp, lalu *image view* akan muncul jka tidak tersambung dengan internet.

- Perancangan *user interface* menu pemilihan rute

Pada perancangan desain menu detail unit layanan kesehatan ini memiliki beberapa komponen didalamnya yaitu 3 buah *button*, 10 buah *text input*, *text view* dan *background*

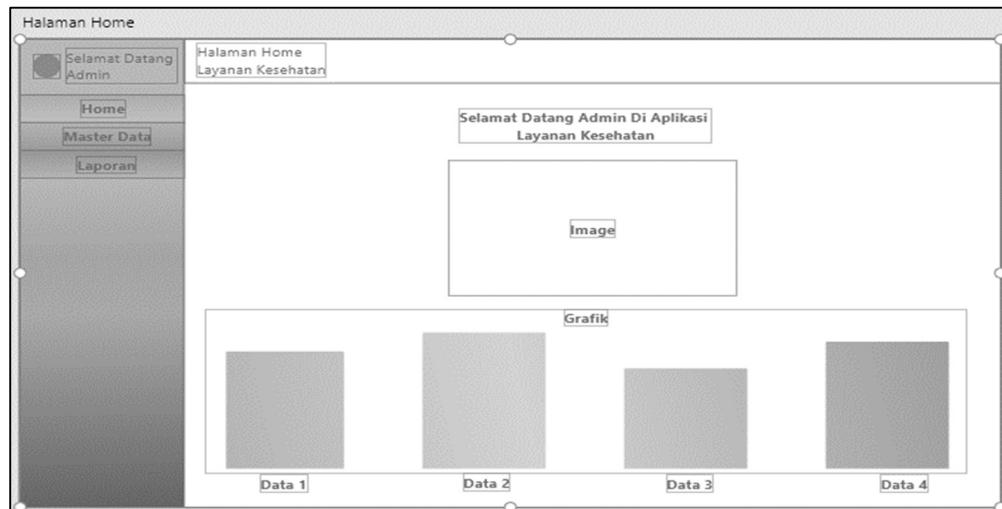


Gambar 57 Perancangan UI pemilihan rute

Dijelaskan pada Gambar 57 diatas bahwa text yang digunakan adalah jenis *font sains-serif*, pada menu ini dilakukannya run pada masing masing algoritma dengan memilih tombol *button* satu untuk algoritma *floydwarsall* dan *button* dua untuk algoritma *a star*. Ditampilkan informasi yang dipresentasikan dalam bentuk *text view* yang berisikan informasi tentang unit rumah sakit yang dipilih oleh *user*, terdapat *background* pada menu ini dengan degradasi 2 warna #FFFFFF dan #5F9EA0 dengan transparan.

- Perancangan *user interface* menu admin layanan gawat darurat

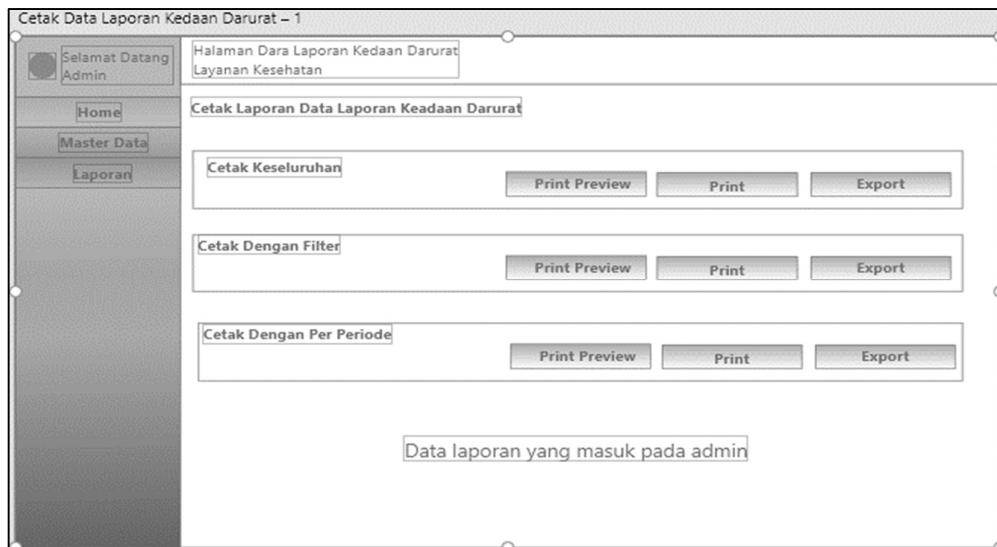
Berikut perancangan *user interface* untuk halaman utama admin layanan gawat darurat yang memiliki beberapa komponen diantaranya *image*, *text*, *button* dan grafik untuk menampilkan jumlah data.



Gambar 58 Perancangan UI admin RS

- Perancangan *user interface* menu admin cetak data laporan kedaan darurat

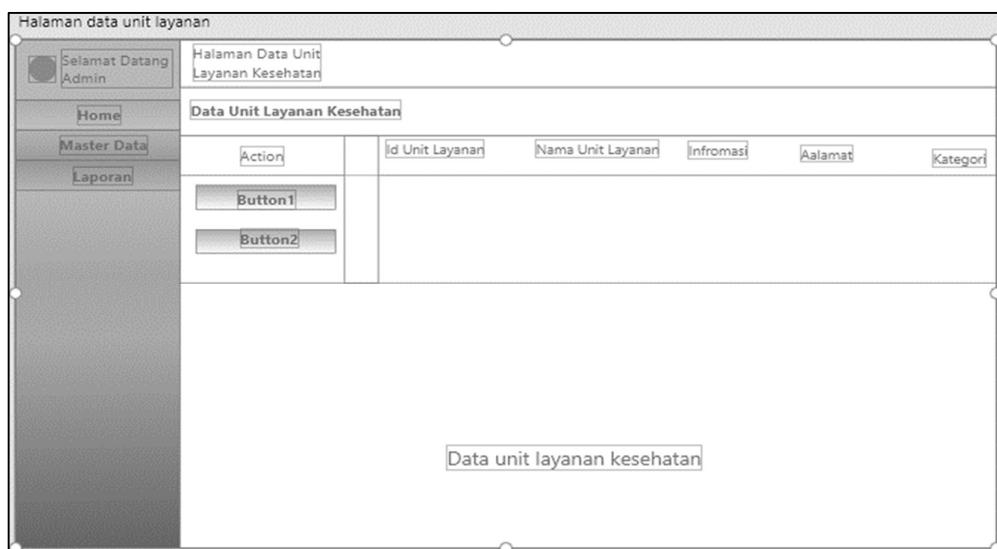
Berikut perancangan *user interface* untuk halaman utama admin cetak data laporan keadaan darurat yang memiliki beberapa komponen diantaranya 3 buah *button* yang fungsinya sama yaitu sebagai *print preview*, *print* dan *export*.



Gambar 59 Perancangan menu admin

- Perancangan *user interface* menu admin data unit layanan kesehatan

Berikut perancangan *user interface* untuk halaman ini yang memiliki beberapa komponen yaitu *button action*, dan tabel.



Gambar 60 Perancangan menu admin layanan darurat

BAB IV

IMPLEMENTASI DAN PENGUJIAN

Pada sub bab ini berisikan tentang mengenai implementasi hasil penelitian dari sistem yang telah dibangun dari pengujian terhadap fungsionalitas sistem dan perangkat yang digunakan didalam pembangunan sistem. Setelah melakukan implementasi maka di tersusun pada tahapan pengujian terhadap sistem dengan pengoperasian, dan kecocokan data.

4.1 Lingkungan Pengembangan

Pada lingkungan pengembangan, dibutuhkannya perangkat – perangkat yang digunakan dalam mendukung pembangunan sistem seperti perangkat keras (*Hardware*) dan perangkat lunak (*Software*). Berikut merupakan spesifikasi perangkat yang digunakan.

4.1.1 Perangkat Keras

Dalam pembangunan sistem ini menggunakan *tools* *Android Studio* serta menggunakan bahasa pemrograman JAVA dan PHP. Berikut ini merupakan perangkat keras (*Hardware*) yang digunakan dalam pembangunan sistem berupa laptop sebagai berikut.

- Laptop Lenovo G360 dengan spesifikasi laptop :
 - Processor : AMD A8-6410 APU with AMD R5 Graphics
 - Display : 14-inch
 - Graphics : AMD Radeon R5 Graphics
 - RAM : 8 GB

Berikut merupakan perangkat keras (*Hardware*) yang digunakan dalam implementasi sistem berupa *smartphone* sebagai berikut.

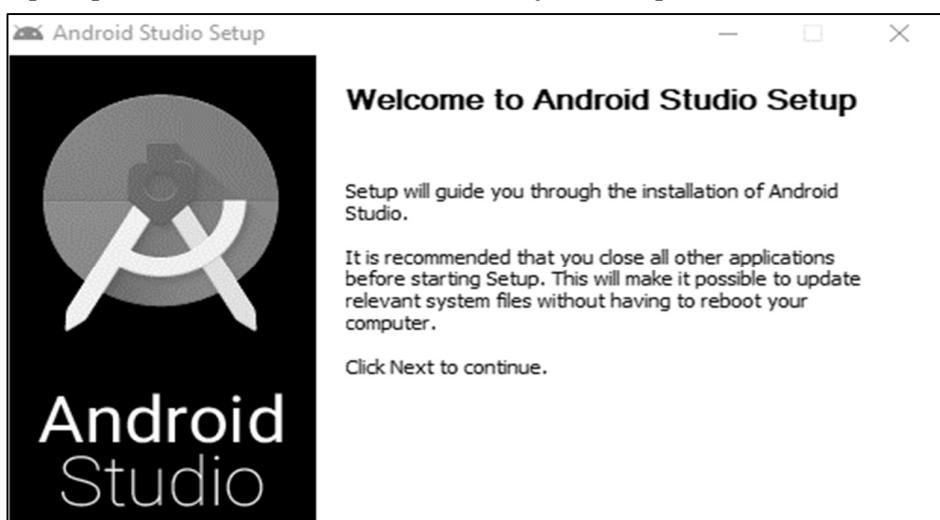
- Smartphone Oppo F7 dengan spesifikasi sebagai berikut :
 - CPU : Eight Core
 - RAM : 4 GB
 - Versi : Android 8

4.1.2 Perangkat Lunak

Perangkat lunak (*software*) atau *tools* yang digunakan didalam pembangunan sistem ini yaitu *Android Studio* dan *Xampp*. Berikut merupakan dijelaskannya proses instalasi perangkat lunak yang akan digunakan.

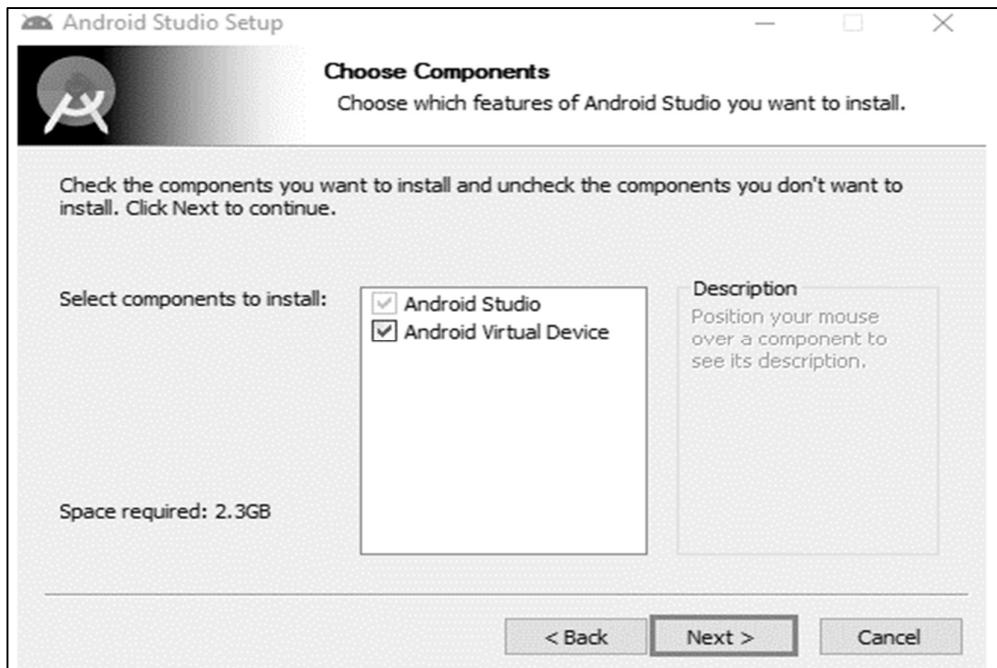
- Proses instalasi android studio

Langkah pertama yaitu melakukan proses instalasi, akan muncul halaman seperti pada Gambar 61 klik next untuk melanjutkan ke proses instalasi.



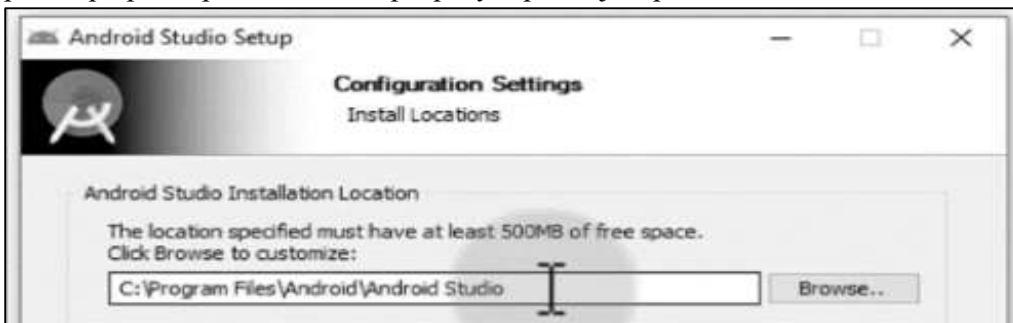
Gambar 61 Instalasi android studio

Langkah kedua memilih komponen tambahan untuk install android studio, yaitu AVD (Android Virtual Device). Sesuaikan komponen yang ditambahkan seperti pada Gambar 62 jika sudah klik next untuk melanjutkan proses instalasi android studio.



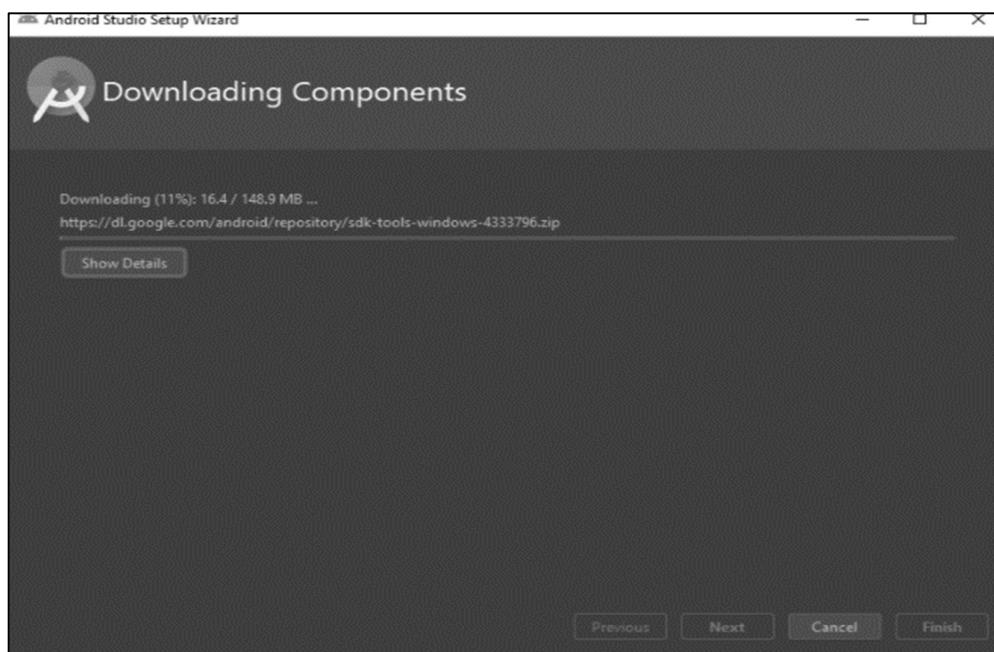
Gambar 62 Instalasi android studio

Kemudian menentukan lokasi instalasi, yaitu dengan pilih lokasi android studio pada laptop/computer untuk tempat penyimpanan jika proses instalasi sudah selesai

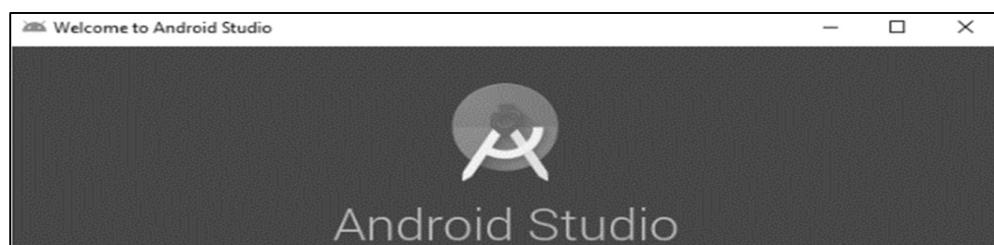


Gambar 63 Instalasi android studio

Selanjutnya tunggu proses instalasi pada laptop/computer selesai dan telah mendownload *component* yang diperlukan sebelumnya pastikan koneksi internet stabil agar tidak terjadi error dalam proses instalasi jika sudah selesai maka akan tampil halaman utama pada android studio.



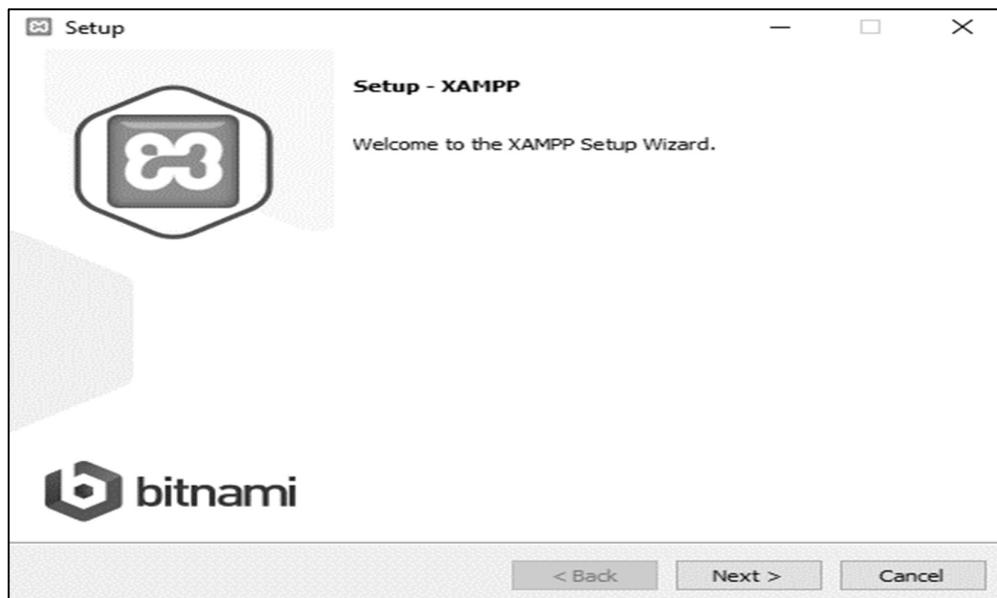
Gambar 64 Instalasi android studio



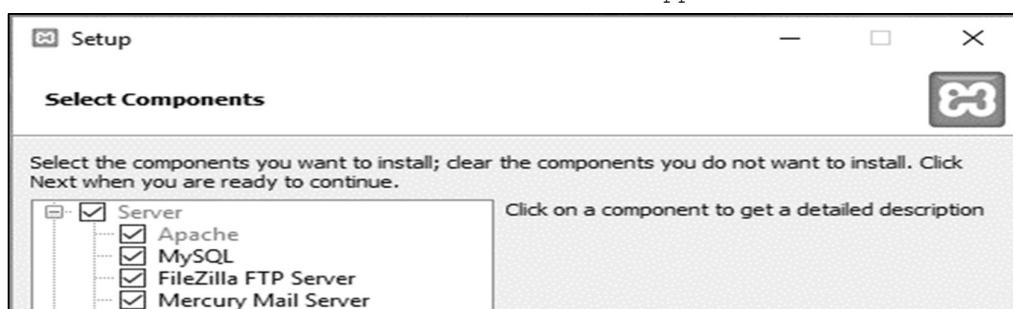
Gambar 65 Instalasi android studio

- Proses instalasi *XAMPP*

XAMPP adalah sebuah prangkat lunak yang berfungsi untuk menjalankan sebuah website di server dengan cara menyimpan file website kedalam hosting local (localhost). Untuk proses instalasinya dapat dilihat sebagai berikut.

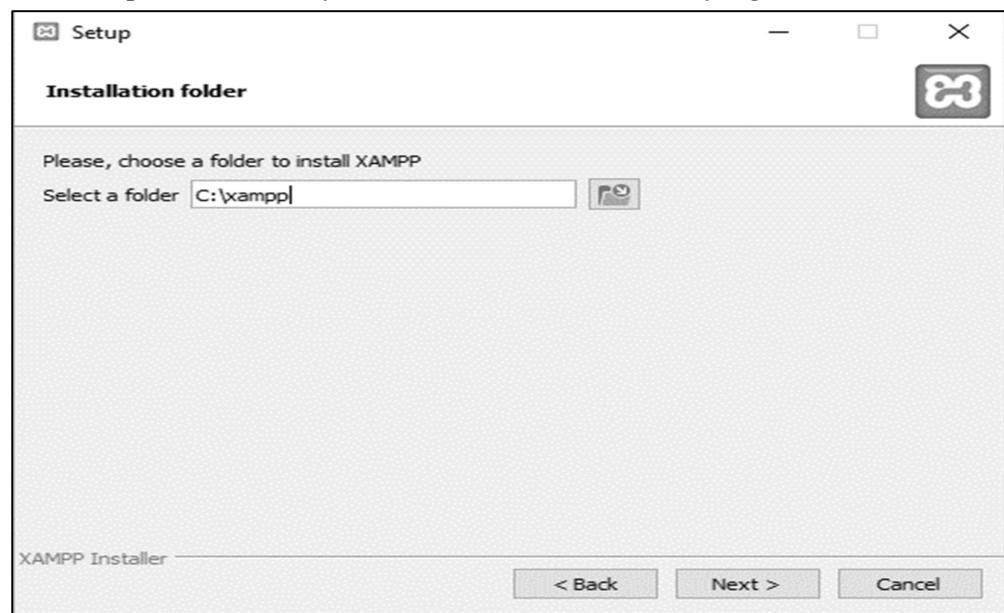


Gambar 66 Instalasi xampp



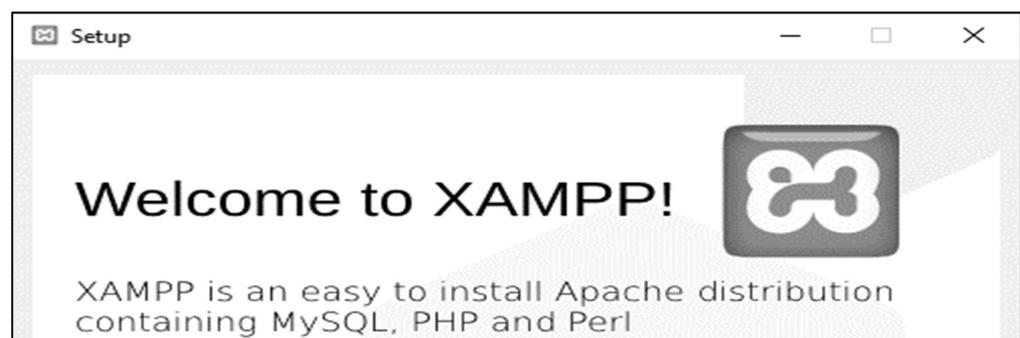
Gambar 67 Instalasi xampp

Pada tahap ini dilakukannya memilih lokasi file XAMPP yang akan diinstall



Gambar 68 Instalasi xampp

. Terakhir tunggu beberapa menit sampai proses instalasi selesai dan dapat digunakan dapat dilihat



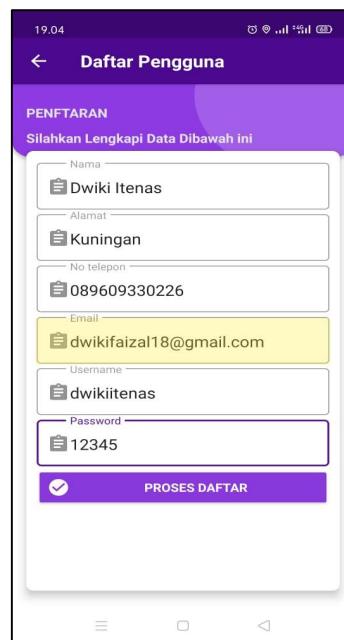
Gambar 69 Instalasi xampp

4.1.3 Implementasi Perancangan *User Interface*

Pada tahap ini dilakukannya implementasi *user interface* berdasarkan hasil dari perancangan sebelumnya yang sudah dijelaskan pada BAB 3.

- Implementasi menu registrasi

Di menu registrasi ini memiliki beberapa komponen diantaranya *text input field* sejumlah 6 buah yang berfungsi untuk mengisikan data diri dari setiap *user* dan 1 buah tombol *button*.



Gambar 70 Implementasi menu registrasi

Pada tombol *button* proses daftar berfungsi untuk mengirimkan data diri yang dimasukkan oleh setiap *user* ke server yang nantinya akan disimpan ke *database*. Setelah selesai melakukan pengisian informasi data diri akan kembali menuju ke menu login untuk melakukan proses login.

- Implementasi menu login

Di menu login ini memiliki komponen diantaranya *text input field* sejumlah 2 buah yang berfungsi untuk input *username* dan *password*.

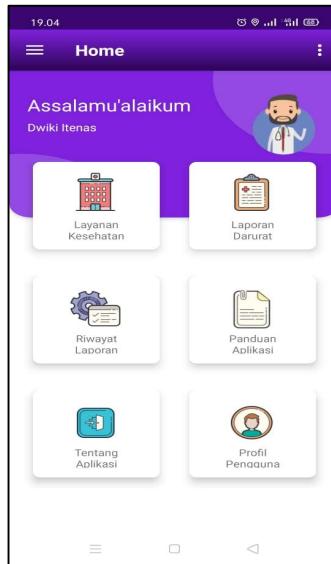


Gambar 71 Implementasi menu login

Pada tombol *button* *login* ini berfungsi untuk berpindah menu dari menu login ke menu utama jika *username* dan *password* yang diinputkan user sesuai.

- Implementasi menu utama

Pada menu ini memiliki komponen sejumlah 6 buah *button* yang bisa digunakan sesuai kebutuhan. *Button* tersebut diantaranya *button* layanan kesehatan, laporan darurat, riwayat, panduan, profil user, dan tentang aplikasi.



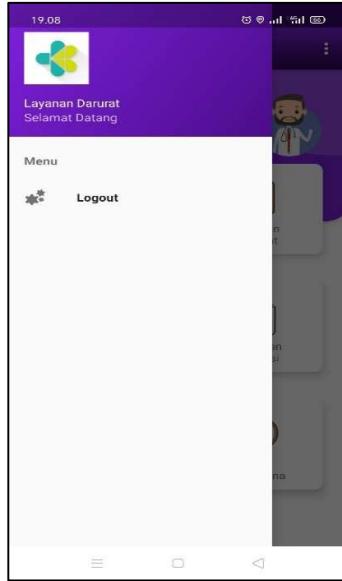
Gambar 72 Implementasi menu utama

Gambar 71 Implementasi menu utama

Pada tombol *button* layanan kesehatan berfungsi untuk berpindah dari menu utama menuju menu layanan gawat darurat yang kemudian dapat memilih rumah sakit. *Button* laporan darurat untuk berpindah dari menu utama menuju menu laporan darurat yang berfungsi untuk melaporkan kejadian kepada pihak rumah sakit. *Button* riwayat laporan untuk berpindah dari menu utama menuju menu riwayat laporan yang berisikan tentang seluruh laporan yang pernah dilakukan oleh *user*. *Button* panduan aplikasi untuk berpindah dari menu utama menuju menu panduan aplikasi yang didalamnya bagaimana cara menggunakan aplikasi. *Button* profil pengguna untuk berpindah dari menu utama menuju menu profil pengguna berfungsi untuk merubah informasi data diri. Dan *button* tentang aplikasi untuk berpindah dari menu utama menuju menu tentang aplikasi, berisikan informasi tentang aplikasi/sistem yang dibuat beserta informasi berita.

- Implementasi navigasi menu

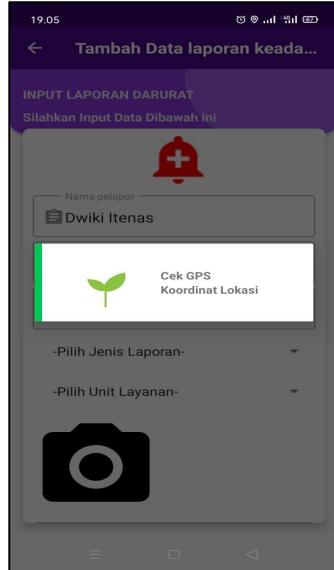
Pada navigasi menu ini yang berfungsi sebagai keluar dari aplikasi dan kembali ke menu login.



Gambar 73 Implementasi navigasi menu

- Implementasi GPS aktif

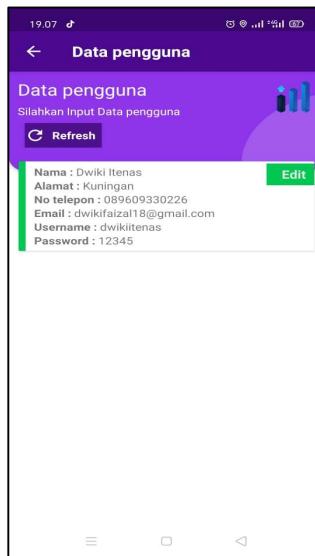
Apabila *user* telah mengaktifkan GPS pada *smartphone* nya sistem akan mengecek GPS koordinat lokasi.



Gambar 74 Implementasi GPS aktif

- Implementasi profil pengguna

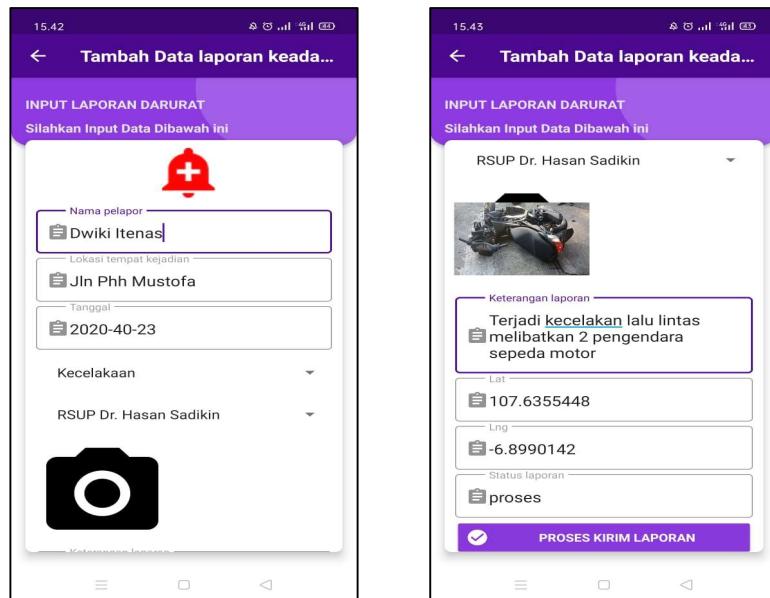
Pada menu profil pengguna ini memiliki komponen diantaranya tombol *button edit* yang berfungsi sebagai merubah informasi data diri setiap user.



Gambar 75 Implementasi menu pengguna

- Implementasi menu laporan darurat

Pada menu laporan darurat ini terdapat beberapa komponen diantaranya *text input*, *image input* dan *combox* yang berfungsi untuk mengisikan laporan darurat

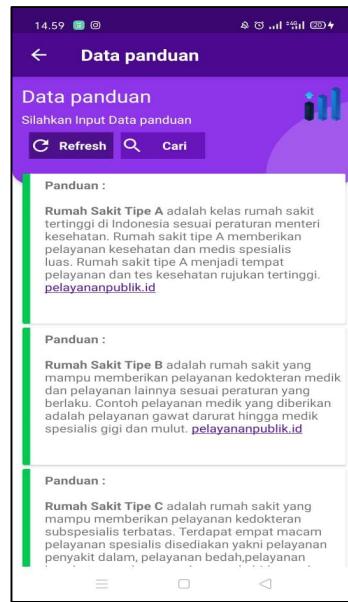


Gambar 76 Implementasi menu laporan darurat

Pada *button* tersebut digunakan untuk mengirimkan data laporan darurat yang telah diinputkan oleh user ke server untuk disimpan pada *database*. Untuk inputan *longitude* dan *latitude* akan terinput jika GPS pada *smartphone user* diaktifkan.

- Implementasi menu panduan

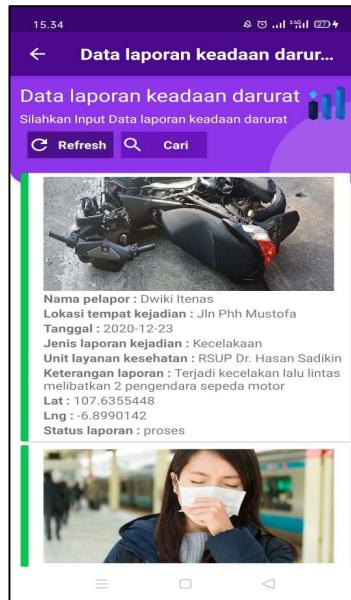
Pada menu panduan ini memiliki beberapa komponen diantaranya seperti tombol *button* dan *card view* berfungsi untuk menampung *text* sedangkan tombol *button* untuk mencari berdasarkan filter yang tersedia pada menu



Gambar 77 Implementasi menu panduan

- Implementasi menu riwayat obrolan atau data laporan darurat

Pada menu riwayat obrolan ini terdapat komponen diantaranya *card view*, menu ini berfungsi untuk menampilkan seluruh informasi laporan darurat yang telah dilakukan oleh *user* kepada pihak rumah sakit yang dipilih.

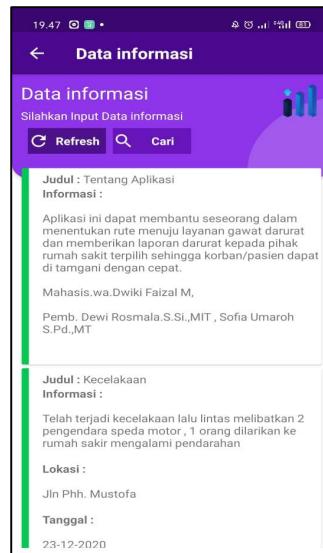


Gambar 78 Implementasi menu riwayat

Informasi laporan darurat ditampilkan dalam bentuk *list* dan menggunakan *card view* yang dimana setiap *card view* berisikan informasi data laporan seperti nama pelapor, lokasi, jenis laporan, keterangan, *latitude,longitude* dan status laporan.

- Implementasi menu data informasi atau tentang aplikasi

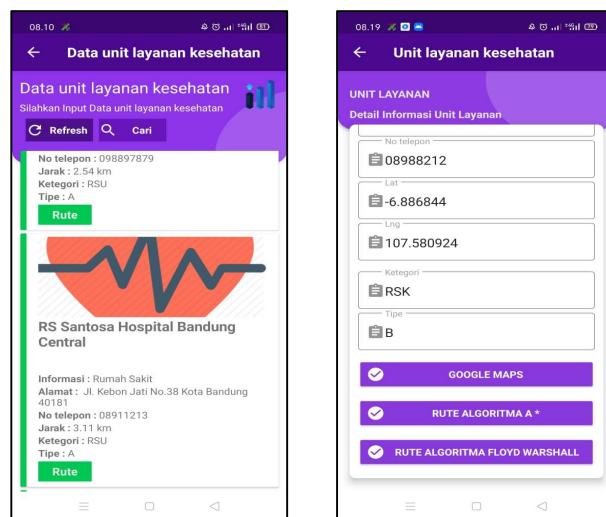
Pada menu ini terdapat komponen diantaranya *card view*, dan menu ini berfungsi sebagai media pemberitaan informasi yang telah dilaporakn oleh *user* atau admin rumah sakit tertentu dalam bentuk *list*.



Gambar 79 Implementasi menu informasi

List ditampilkan dengan menggunakan *card view* dimana setiap *card view* berisikan berita informasi diantaranya seperti judul, informasi, lokasi, tanggal dan unit layanan dan lain lain, sehingga dapat mempermudah dalam melihat berita. Terdapat *button* pencarian yang berfungsi untuk mempermudah dalam pencarian berita berdasarkan filter yang tersedia.

- Implementasi menu pencarian/pemilihan rumah sakit atau layanan kesehatan
- Pada menu ini terdapat komponen diantaranya *button*, *card view* dan pada menu ini berfungsi untuk menampilkan *list* rumah sakit yang tersedia sehingga *user* dapat memilih sesuai kebutuhan.



Gambar 80 Implementasi menu pemilihan rumah sakit

List yang ditampilkan ditampung dalam sebuah *card view* sehingga dapat mempermudah dalam memilih dan dimana pada setiap *card view* berisikan informasi tentang data rumah sakit diantaranya seperti nama rumah sakit, alamat rumah sakit, no telepon, kategori rumah sakit, tipe rumah sakit dan data jarak antara lokasi posisi *user* dengan rumah sakit tersebut. Data jarak readl dengan metode haversine akan tampil jika *user* mengaktifkan GPS pada *smartphone* dan akan mendeteksi, terdapat *button* rute berfungsi untuk memilih rute mana yang akan digunakan.

- Implementasi menu penentuan rute algoritma a star

Pada menu berfungsi untuk menampilkan kemungkinan rute yang bisa dilalui oleh *user* menuju rumah sakti yang dipilih dengan menggunakan algoritma *a star*.



Gambar 81 Implementasi meu a star

Di dalam menu ini terdapat beberapa komponen diantaranya *fragment* yang berfungsi untuk sebagai tempat menampilkan peta. Selain itu terdapat *rectangle*

yang berfungsi sebagai menampung informasi data jarak, perkiraan waktu tempuh dan keterangan jarak asal menuju lokasi tujuan.

- Implementasi menu penentuan rute algoritma floydwarsall

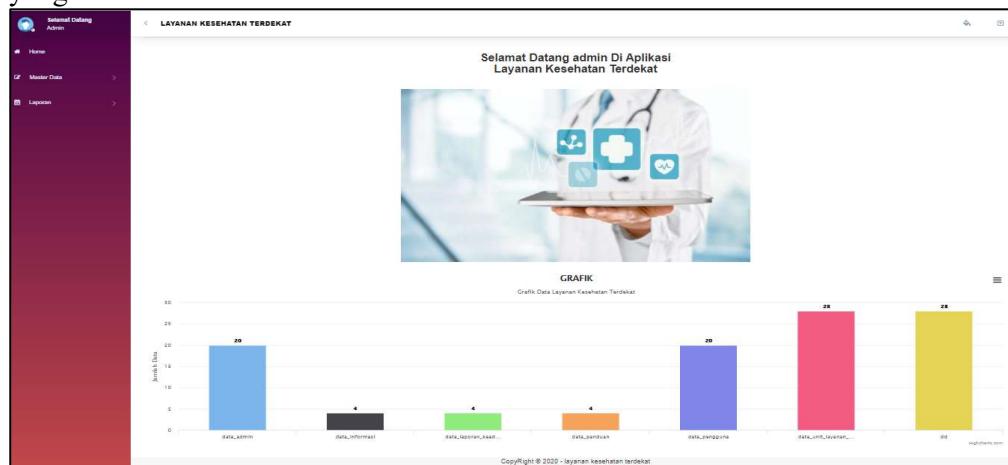
Pada menu berfungsi untuk menampilkan kemungkinan rute yang bisa dilalui oleh *user* menuju rumah sakti yang dipilih dengan menggunakan algoritma *floydwarsall*.



Gambar 82 Implementasi menu floydwarshall

- Implementasi menu utama admin

Pada menu ini adalah tampilan utama admin dari aplikasi yang dibangun, dimana pada menu terdapat komponen diantaranya *button* master data, *button* laporan dan *grafik* berfungsi untuk menampilkan jumlah informasi data layanan yang masuk



Gambar 83 Implementasi menu utama admin

- Implementasi menu data laporan keadaan darurat admin

Menu ini terdapat komponen *button action* dan tabel. Menu ini berfungsi menampilkan seluruh laporan masuk yang berisikan informasi data laporan yang masuk, sehingga admin dapat mengkonfirmasi status laporan dengan memilih laporan pilih *button action* edit/proses.



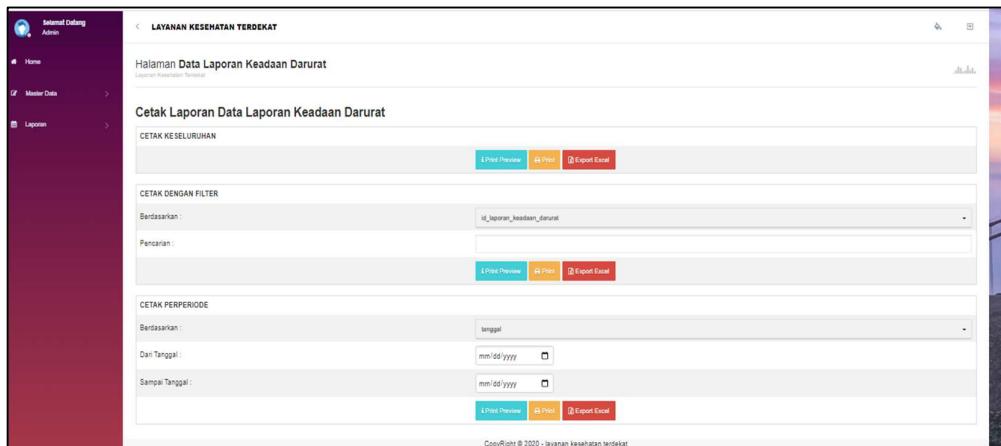
Table showing Laporan Data Unit Layanan Kesehatan:

ACTION	NO	NAMA LAYANAN PESANAN	NAMA UNIT LAYANAN	ALAMAT	KOORDINAT	LAT	LNG	JAMAK	PERAK	PERANGKAT	USIA	PERSAMA
	1	RS. Jawa Province Jawa Barat	RS. Jawa Province Jawa Barat	Jl. Kremi Mulyo KM 1 Cikarang - Bandung	06,-6.93695	-107.56425	-	-	RSU	A	-	rs7
	2	RS Mata Cipta	RS Mata Cipta	Jl. Ciwidey No.4 Kota Bandung	08881979	-107.604771	-	-	RSU	A	-	rs8
	3	RS Samrat Hospital Bandung Central	Rumah Sakit	Jl. Kebon Jati No.38 Kota Bandung 40181	08911219	-107.61007	-107.60025	-	RSU	A	-	rs1
	4	RS Pdt. Dr. H. A. Romualdo	Rumah Sakit	Jl. Budi Jaya No.4 Kota Bandung 40142	08912121	-107.61702	-107.60050	-	RSU	A	-	rs2
	5	RSU Gigris Muad PKU UINQ	Rumah Sakit	Jl. Semina Sosial I Kota Bandung 40132	08977521	-107.61207	-	-	RSU	A	-	rs3
	6	RSUP Dr. H. Hasan Sadikan	Rumah Sakit	Jl. Petaio No. 38 Kota Bandung 40161	08123323	-107.60464	-107.59981	-	RSUP	A	-	rs4
	7	RS Anugerah Bandung	Rumah Sakit	Jl. Champsuis 161 Kota Bandung 40131	08995223	-107.61084	-107.60205	-	RSU	B	-	rs5
	8	RS Almaran Bandung	Rumah Sakit	Jl. Siliwangi No.444 Kota Bandung 40138	08988132	-107.60879	-107.609176	-	RSU	B	-	rs6
	9	RS Anugrah Bandung	Rumah Sakit	Jl. Kopo 161 Kota Bandung 40234	08977877	-107.61407	-107.60727	-	RSU	B	-	rs7
	10	RS Khatib Isu & Andi Kita Bandung	Rumah Sakit	Jl. Amanzay 124 Kota Bandung 40241	08761767	-107.611425	-	-	RSU	B	-	rs10

Gambar 84 Implementasi menu laporan darurat admin

- Implementasi menu cetak data laporan kedaan darurat

Menu ini berfungsi untuk mencetak seluruh laporan keadaan darurat yang masuk dan yang telah dilaporkan oleh *user*.



Cetak Laporan Data Laporan Keadaan Darurat

CETAK KESELURUHAN

CETAK DENGAN FILTER

Berdasarkan:

Pencarian:

CETAK PERIODE

Berdasarkan:

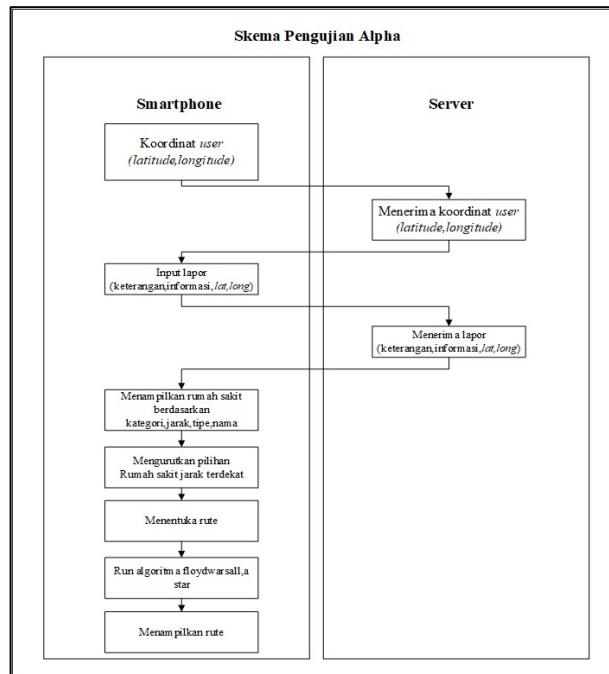
Dari Tanggal:

Sampai Tanggal:

Gambar 85 Implementasi menu laporan cetak admin

4.2 Pengujian Fungsionalitas

Pada sub bab ini, pengujian alpha dilakukan terhadap fungsionalitas yang terdapat pada sistem yang sedang dibangun. Yang diuji yaitu terdapat 2 sistem yaitu pada *smartphone* dan *server*.



Gambar 86 Pengujian alpha

4.2.1 Proses Pelaporan Darurat

Data laporan darurat yang telah diinputkan oleh *user* ke server yang kemudian akan di simpan ke *database* sehingga nantinya dapat di proses oleh pihak admin tertentu

Table 21 Pelaporan Darurat

<p><i>Flowchart</i></p>	<pre> graph TD Start([Start pengiriman pemasukan laporan]) --> Pilih[Pilih kirim pesan] Pilih --> Masukkan[/Masukkan informasi lokasi/jenis tujuan, gambar, keterangan/] Masukkan --> Kirim[Kirim data ke server] Kirim --> Berhasil{Kirim berhasil} Berhasil --> End([End]) Note[Server mengirimkan notifikasi ke tujuan rumah sakit] -.-> Masukkan </pre>																																																																																																																																		
<p><i>Source Code</i></p>	<p>Terlampir pada lampiran L.2</p>																																																																																																																																		
<p>Hasil Pengujian</p>	<table border="1"> <thead> <tr> <th colspan="9">LAYANAN KESAMPAHAN TERDOKAT</th> </tr> <tr> <th colspan="9">Halaman Data Laporan Keadaan Darurat</th> </tr> <tr> <th colspan="9">Cari di kolom Nama</th> </tr> </thead> <tbody> <tr> <td colspan="9"> <table border="1"> <thead> <tr> <th colspan="2">Pencarian</th> <th colspan="7">LAPORAN KEADAAN DARURAT</th> </tr> <tr> <th colspan="2">Berdasarkan</th> <th colspan="7">Berdasarkan</th> </tr> <tr> <th colspan="2">Pencarian</th> <th colspan="7">Pencarian</th> </tr> </thead> <tbody> <tr> <td colspan="2">ACTION</td> <td>ID</td> <td>NO</td> <td>NAMA LELAKU</td> <td>LOKASI LELAKU</td> <td>TANGGAL</td> <td>JENIS LAPORAN KEADAAN</td> <td>UNIT LAYANAN KEDOKTERAN</td> <td>UPLOAD GAMBAR</td> <td>KETERANGAN LAPORAN</td> <td>LAT</td> <td>LNG</td> </tr> <tr> <td colspan="2"> Detail Edit Delete </td> <td>1</td> <td>LAP0412202115190097</td> <td>KH</td> <td>Jl. Pahlawan</td> <td>31 Desember 2020</td> <td>Kebonharjo</td> <td>RS Si Yusuf Bandung</td> <td></td> <td>Tidak tau lokasi yg lokasi ini lebih dekat ke pengendara sepeda motor</td> <td>107.8355145</td> <td>-6.93000</td> </tr> <tr> <td colspan="2"> Detail Edit Delete </td> <td>2</td> <td>LAP0212202154462393</td> <td>Dewi tetep</td> <td>Jl. P. H. Mulyo</td> <td>23 Desember 2020</td> <td>Kebonharjo</td> <td>RSUP Dr. Hasan Sadikin</td> <td></td> <td>Mengalami pingsan saat berjalan kaki sejauh 100m dari rumahnya</td> <td>107.8355446</td> <td>-6.93000</td> </tr> <tr> <td colspan="2"> Detail Edit Delete </td> <td>3</td> <td>LAP0412202132715144</td> <td>Dewi tetep</td> <td>Jl. Pahlawan Kota Bandung</td> <td>24 Desember 2020</td> <td>Lamya</td> <td>RSUP Dr. Hasan Sadikin</td> <td></td> <td>Mengalami pingsan saat berjalan kaki sejauh 100m dari rumahnya</td> <td>107.8355491</td> <td>-6.93000</td> </tr> <tr> <td colspan="2"> Detail Edit Delete </td> <td>4</td> <td>LAP0412202134467639</td> <td>Dewi tetep</td> <td>Jl. Cikutra Baru Kota Bandung</td> <td>24 Desember 2020</td> <td>Ibu Hamil</td> <td>RS Si Yusuf Bandung</td> <td></td> <td>Gejala sakit melahirkan disertai pendarahan</td> <td>107.8355491</td> <td>-6.93000</td> </tr> </tbody> </table> </td> </tr> <tr> <td data-bbox="355 1262 551 1577"> <p>Keterangan</p> </td><td data-bbox="551 1262 1341 1577"> <p>Data laporan darurat yang telah di inputkan oleh <i>user</i> pada <i>smartphone</i> kemudian sistem akan mengirimkan ke server. Lalu akan di simpan ke <i>database</i> sehingga pihak admin rumah sakit yang dipilih dapat melakukan proses konfirmasi status terhadap laporan yang masuk ke rumah sakit.</p> </td></tr> </tbody></table>	LAYANAN KESAMPAHAN TERDOKAT									Halaman Data Laporan Keadaan Darurat									Cari di kolom Nama									<table border="1"> <thead> <tr> <th colspan="2">Pencarian</th> <th colspan="7">LAPORAN KEADAAN DARURAT</th> </tr> <tr> <th colspan="2">Berdasarkan</th> <th colspan="7">Berdasarkan</th> </tr> <tr> <th colspan="2">Pencarian</th> <th colspan="7">Pencarian</th> </tr> </thead> <tbody> <tr> <td colspan="2">ACTION</td> <td>ID</td> <td>NO</td> <td>NAMA LELAKU</td> <td>LOKASI LELAKU</td> <td>TANGGAL</td> <td>JENIS LAPORAN KEADAAN</td> <td>UNIT LAYANAN KEDOKTERAN</td> <td>UPLOAD GAMBAR</td> <td>KETERANGAN LAPORAN</td> <td>LAT</td> <td>LNG</td> </tr> <tr> <td colspan="2"> Detail Edit Delete </td> <td>1</td> <td>LAP0412202115190097</td> <td>KH</td> <td>Jl. Pahlawan</td> <td>31 Desember 2020</td> <td>Kebonharjo</td> <td>RS Si Yusuf Bandung</td> <td></td> <td>Tidak tau lokasi yg lokasi ini lebih dekat ke pengendara sepeda motor</td> <td>107.8355145</td> <td>-6.93000</td> </tr> <tr> <td colspan="2"> Detail Edit Delete </td> <td>2</td> <td>LAP0212202154462393</td> <td>Dewi tetep</td> <td>Jl. P. H. Mulyo</td> <td>23 Desember 2020</td> <td>Kebonharjo</td> <td>RSUP Dr. Hasan Sadikin</td> <td></td> <td>Mengalami pingsan saat berjalan kaki sejauh 100m dari rumahnya</td> <td>107.8355446</td> <td>-6.93000</td> </tr> <tr> <td colspan="2"> Detail Edit Delete </td> <td>3</td> <td>LAP0412202132715144</td> <td>Dewi tetep</td> <td>Jl. Pahlawan Kota Bandung</td> <td>24 Desember 2020</td> <td>Lamya</td> <td>RSUP Dr. Hasan Sadikin</td> <td></td> <td>Mengalami pingsan saat berjalan kaki sejauh 100m dari rumahnya</td> <td>107.8355491</td> <td>-6.93000</td> </tr> <tr> <td colspan="2"> Detail Edit Delete </td> <td>4</td> <td>LAP0412202134467639</td> <td>Dewi tetep</td> <td>Jl. Cikutra Baru Kota Bandung</td> <td>24 Desember 2020</td> <td>Ibu Hamil</td> <td>RS Si Yusuf Bandung</td> <td></td> <td>Gejala sakit melahirkan disertai pendarahan</td> <td>107.8355491</td> <td>-6.93000</td> </tr> </tbody> </table>									Pencarian		LAPORAN KEADAAN DARURAT							Berdasarkan		Berdasarkan							Pencarian		Pencarian							ACTION		ID	NO	NAMA LELAKU	LOKASI LELAKU	TANGGAL	JENIS LAPORAN KEADAAN	UNIT LAYANAN KEDOKTERAN	UPLOAD GAMBAR	KETERANGAN LAPORAN	LAT	LNG	Detail Edit Delete		1	LAP0412202115190097	KH	Jl. Pahlawan	31 Desember 2020	Kebonharjo	RS Si Yusuf Bandung		Tidak tau lokasi yg lokasi ini lebih dekat ke pengendara sepeda motor	107.8355145	-6.93000	Detail Edit Delete		2	LAP0212202154462393	Dewi tetep	Jl. P. H. Mulyo	23 Desember 2020	Kebonharjo	RSUP Dr. Hasan Sadikin		Mengalami pingsan saat berjalan kaki sejauh 100m dari rumahnya	107.8355446	-6.93000	Detail Edit Delete		3	LAP0412202132715144	Dewi tetep	Jl. Pahlawan Kota Bandung	24 Desember 2020	Lamya	RSUP Dr. Hasan Sadikin		Mengalami pingsan saat berjalan kaki sejauh 100m dari rumahnya	107.8355491	-6.93000	Detail Edit Delete		4	LAP0412202134467639	Dewi tetep	Jl. Cikutra Baru Kota Bandung	24 Desember 2020	Ibu Hamil	RS Si Yusuf Bandung		Gejala sakit melahirkan disertai pendarahan	107.8355491	-6.93000	<p>Keterangan</p>	<p>Data laporan darurat yang telah di inputkan oleh <i>user</i> pada <i>smartphone</i> kemudian sistem akan mengirimkan ke server. Lalu akan di simpan ke <i>database</i> sehingga pihak admin rumah sakit yang dipilih dapat melakukan proses konfirmasi status terhadap laporan yang masuk ke rumah sakit.</p>
LAYANAN KESAMPAHAN TERDOKAT																																																																																																																																			
Halaman Data Laporan Keadaan Darurat																																																																																																																																			
Cari di kolom Nama																																																																																																																																			
<table border="1"> <thead> <tr> <th colspan="2">Pencarian</th> <th colspan="7">LAPORAN KEADAAN DARURAT</th> </tr> <tr> <th colspan="2">Berdasarkan</th> <th colspan="7">Berdasarkan</th> </tr> <tr> <th colspan="2">Pencarian</th> <th colspan="7">Pencarian</th> </tr> </thead> <tbody> <tr> <td colspan="2">ACTION</td> <td>ID</td> <td>NO</td> <td>NAMA LELAKU</td> <td>LOKASI LELAKU</td> <td>TANGGAL</td> <td>JENIS LAPORAN KEADAAN</td> <td>UNIT LAYANAN KEDOKTERAN</td> <td>UPLOAD GAMBAR</td> <td>KETERANGAN LAPORAN</td> <td>LAT</td> <td>LNG</td> </tr> <tr> <td colspan="2"> Detail Edit Delete </td> <td>1</td> <td>LAP0412202115190097</td> <td>KH</td> <td>Jl. Pahlawan</td> <td>31 Desember 2020</td> <td>Kebonharjo</td> <td>RS Si Yusuf Bandung</td> <td></td> <td>Tidak tau lokasi yg lokasi ini lebih dekat ke pengendara sepeda motor</td> <td>107.8355145</td> <td>-6.93000</td> </tr> <tr> <td colspan="2"> Detail Edit Delete </td> <td>2</td> <td>LAP0212202154462393</td> <td>Dewi tetep</td> <td>Jl. P. H. Mulyo</td> <td>23 Desember 2020</td> <td>Kebonharjo</td> <td>RSUP Dr. Hasan Sadikin</td> <td></td> <td>Mengalami pingsan saat berjalan kaki sejauh 100m dari rumahnya</td> <td>107.8355446</td> <td>-6.93000</td> </tr> <tr> <td colspan="2"> Detail Edit Delete </td> <td>3</td> <td>LAP0412202132715144</td> <td>Dewi tetep</td> <td>Jl. Pahlawan Kota Bandung</td> <td>24 Desember 2020</td> <td>Lamya</td> <td>RSUP Dr. Hasan Sadikin</td> <td></td> <td>Mengalami pingsan saat berjalan kaki sejauh 100m dari rumahnya</td> <td>107.8355491</td> <td>-6.93000</td> </tr> <tr> <td colspan="2"> Detail Edit Delete </td> <td>4</td> <td>LAP0412202134467639</td> <td>Dewi tetep</td> <td>Jl. Cikutra Baru Kota Bandung</td> <td>24 Desember 2020</td> <td>Ibu Hamil</td> <td>RS Si Yusuf Bandung</td> <td></td> <td>Gejala sakit melahirkan disertai pendarahan</td> <td>107.8355491</td> <td>-6.93000</td> </tr> </tbody> </table>									Pencarian		LAPORAN KEADAAN DARURAT							Berdasarkan		Berdasarkan							Pencarian		Pencarian							ACTION		ID	NO	NAMA LELAKU	LOKASI LELAKU	TANGGAL	JENIS LAPORAN KEADAAN	UNIT LAYANAN KEDOKTERAN	UPLOAD GAMBAR	KETERANGAN LAPORAN	LAT	LNG	Detail Edit Delete		1	LAP0412202115190097	KH	Jl. Pahlawan	31 Desember 2020	Kebonharjo	RS Si Yusuf Bandung		Tidak tau lokasi yg lokasi ini lebih dekat ke pengendara sepeda motor	107.8355145	-6.93000	Detail Edit Delete		2	LAP0212202154462393	Dewi tetep	Jl. P. H. Mulyo	23 Desember 2020	Kebonharjo	RSUP Dr. Hasan Sadikin		Mengalami pingsan saat berjalan kaki sejauh 100m dari rumahnya	107.8355446	-6.93000	Detail Edit Delete		3	LAP0412202132715144	Dewi tetep	Jl. Pahlawan Kota Bandung	24 Desember 2020	Lamya	RSUP Dr. Hasan Sadikin		Mengalami pingsan saat berjalan kaki sejauh 100m dari rumahnya	107.8355491	-6.93000	Detail Edit Delete		4	LAP0412202134467639	Dewi tetep	Jl. Cikutra Baru Kota Bandung	24 Desember 2020	Ibu Hamil	RS Si Yusuf Bandung		Gejala sakit melahirkan disertai pendarahan	107.8355491	-6.93000																															
Pencarian		LAPORAN KEADAAN DARURAT																																																																																																																																	
Berdasarkan		Berdasarkan																																																																																																																																	
Pencarian		Pencarian																																																																																																																																	
ACTION		ID	NO	NAMA LELAKU	LOKASI LELAKU	TANGGAL	JENIS LAPORAN KEADAAN	UNIT LAYANAN KEDOKTERAN	UPLOAD GAMBAR	KETERANGAN LAPORAN	LAT	LNG																																																																																																																							
Detail Edit Delete		1	LAP0412202115190097	KH	Jl. Pahlawan	31 Desember 2020	Kebonharjo	RS Si Yusuf Bandung		Tidak tau lokasi yg lokasi ini lebih dekat ke pengendara sepeda motor	107.8355145	-6.93000																																																																																																																							
Detail Edit Delete		2	LAP0212202154462393	Dewi tetep	Jl. P. H. Mulyo	23 Desember 2020	Kebonharjo	RSUP Dr. Hasan Sadikin		Mengalami pingsan saat berjalan kaki sejauh 100m dari rumahnya	107.8355446	-6.93000																																																																																																																							
Detail Edit Delete		3	LAP0412202132715144	Dewi tetep	Jl. Pahlawan Kota Bandung	24 Desember 2020	Lamya	RSUP Dr. Hasan Sadikin		Mengalami pingsan saat berjalan kaki sejauh 100m dari rumahnya	107.8355491	-6.93000																																																																																																																							
Detail Edit Delete		4	LAP0412202134467639	Dewi tetep	Jl. Cikutra Baru Kota Bandung	24 Desember 2020	Ibu Hamil	RS Si Yusuf Bandung		Gejala sakit melahirkan disertai pendarahan	107.8355491	-6.93000																																																																																																																							
<p>Keterangan</p>	<p>Data laporan darurat yang telah di inputkan oleh <i>user</i> pada <i>smartphone</i> kemudian sistem akan mengirimkan ke server. Lalu akan di simpan ke <i>database</i> sehingga pihak admin rumah sakit yang dipilih dapat melakukan proses konfirmasi status terhadap laporan yang masuk ke rumah sakit.</p>																																																																																																																																		

4.2.2 Proses Pemilihan Rumah Sakit

Pada server akan melakukan pencarian *database* rumah sakit sesuai berdasarkan kategori lalu akan diambil berupa data setiap rumah sakit dan lokasi rumah sakit berdasarkan latitude dan longtiutude.

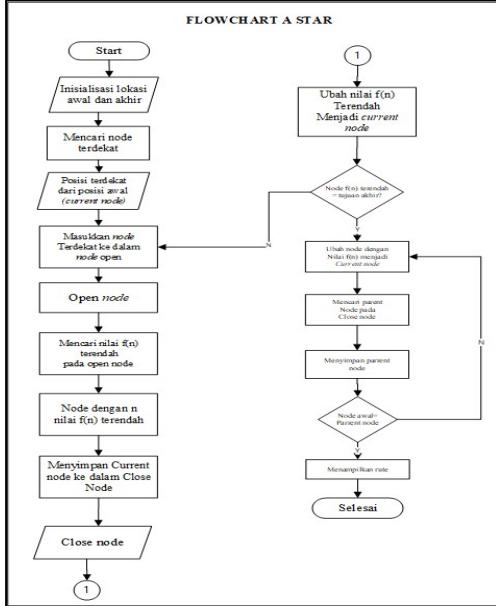
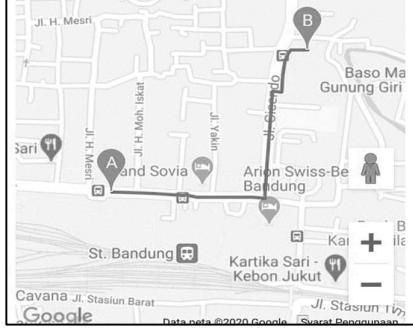
Table 22 Pemilihan rumah sakit

<p><i>Flowchart</i></p>	<pre> graph TD Start([Start pencarian]) --> Kategori[/Kategori rumah sakit/] Kategori --> PilihRumahSakit[Pilih rumah sakit sesuai kategori kebutuhan korban] PilihRumahSakit --> DataRS[/Nama RS, id RS, jarak, waktu tempuh kategori,tipe/] DataRS --> End([End pencarian]) </pre>
<p><i>Source Code</i></p>	<p>Dan terlampir pada lampiran L.2</p> <pre> <?php \$no = 0; \$startRow = (\$page-1)*\$dataPerPage; \$startRow; if (isset(\$_GET['Berdasarkan']) && !empty(\$_GET['Berdasarkan']) && isset(\$_GET['isi']) && !empty(\$_GET['isi'])) { \$berdasarkan = mysql_real_escape_string(\$_GET['Berdasarkan']); \$isi = mysql_real_escape_string(\$_GET['isi']); \$querytable="SELECT * FROM data_unit_layanan_kesehatan where \$berdasarkan like '%\$isi%' LIMIT \$startRow ,\$dataPerPage"; \$querypagination="SELECT COUNT(*) AS total FROM data_unit_layanan_kesehatan where \$berdasarkan like '%\$isi%'"; } else { \$querytable="SELECT * FROM data_unit_layanan_kesehatan LIMIT \$startRow,\$dataPerPage"; \$querypagination="SELECT COUNT(*) AS total FROM data_unit_layanan_kesehatan"; } \$proses = mysql_query(\$querytable); while (\$data = mysql_fetch_array(\$proses)) { <tr class="event2"> <td class="th_border cell" align="center" width="200"> <table border="0"><tr> <td> <a href=<?php index(); ?>?input=detail&proses=<?= encrypt(\$data['id_unit_layanan_kesehatan']); ?>"> <?php btn_detail('Detail'); ?> </td> <td> <a href=<?php index(); ?>?input=edit&proses=<?= encrypt(\$data['id_unit_layanan_kesehatan']); ?>"> <?php btn_edit('Edit'); ?> </td> <td> <a href=<?php index(); ?>?input=hapus&proses=<?= encrypt(\$data['id_unit_layanan_kesehatan']); ?>"> <?php btn_hapus('Hapus'); ?> </td> </tr> </td> </tr> } </pre>
<p><i>Hasil Pengujian</i></p>	<p></p> <p>Keterangan</p> <p>Dari pengujian terdapat listRS yang merupakan array list yang berisikan id, nama rumah sakit, alamat, notelp, kategori,tipe jarak. Sistem melakukan request pada <i>google maps API distance matrix</i> sehingga mendapatkan informasi durasi dan jarak yang diberikan oleh <i>haversine</i> dari lokasi <i>user</i> dengan mendeteksi <i>GPS smartphone</i> menuju rumah sakit. Setelah itu sistem akan menyimpan data jarak dalam bentuk listRS dan menampilkannya pada <i>cardview</i> berdasarkan kategori tertentu.</p>

4.2.3 Proses Rute Algoritma A Star

Proses ini dilakukan setelah *user* menentukan rumah sakit yang akan dituju, lalu sistem akan menampilkan hasil dari rute dari algoritma *a star* disertasi dengan informasi jarak,waktu tempuh dan keterangan.

Table 23 Penentuan rute a star

Flowchart	Source Code
 <pre> graph TD Start([Start]) --> Inisialisasi[Inisialisasi lokasi awal dan akhir] Inisialisasi --> MencariNode[Mencari node terdekat] MencariNode --> Posisi[Posisi terdekat dari posisi awal (current node)] Posisi --> Masukan[Masukan node Terdekat ke dalam node open] Masukan --> OpenNode[Open node] OpenNode --> MencariF[Mencari nilai f(n) terendah pada open node] MencariF --> NodeF[Node dengan nilai f(n) terendah] NodeF --> Menyimpan[Menyimpan Current node ke dalam Close Node] Menyimpan --> CloseNode[Close node] CloseNode --> 1((1)) 1 --> 1((1)) 1((1)) --> Ubah1[Ubah nilai f(n) Terendah Menjadi current node] Ubah1 --> NodeF1{Node f(n) terendah = tujuan akhir?} NodeF1 --> Selesai([Selesai]) NodeF1 --> Ubah2[Ubah node dengan Nilai f(n) menjadi Current node] Ubah2 --> MenyanParent[Menyan parent Node pada Close node] MenyanParent --> MenyimpanParent[Menyimpan parent node] MenyimpanParent --> NodeAwal{Node awal = Parent node?} NodeAwal --> MenyimpanRute[Menyimpan rute] MenyimpanRute --> Selesai NodeAwal --> 1((1)) </pre>	Terlampir pada lampiran L.2
	<p>Dari hasil pengujian sistem akan mengirim request kepada google maps API direction, pertama melakukan pembuatan matriks untuk lintasan jalur/node yang akan dilalui, berikutnya menginisialisasi titik asal dan titik tujuan berfungsi untuk mengetahui rute awal dan akhir, jika sudah lanjut inisialisasi persimpangan setiap simpangan dijadikan node $h(n)$ dan simpul yg berhubungan antar node sebagai bobot node $g(n)$, lalu setelah mendapatkan perkiraan jarak antar node $f(n)$, setiap node yang akan dilewati menuju tujuan akan dimasukan openlist yang mempunyai peluang untuk terpilih sebagai bestnode, selanjutnya sesudah memasukkan node ke openlist yaitu dengan mempertimbangkan bobot antar node yg terhubung bobot tersebut akan di tumpang dalam array, lalu node mana yg bisa dilalui dan memiliki nilai bobot yg kecil akan dilalui, berikutnya setelah mempertimbangkan bobot antar node, node yg memiliki nilai terkecil dan dapat dilalui maka akan dimasukkan ke closelist dan tidak bisa dilewati lagi sehingga peluang untuk terpilih tertutup begitupun seterusnya. Dalam tahap menentukan node yang memungkinkan ke arah tujuan, jika node yang dipilih tidak dapat dilalui atau memiliki nilai bobot besar maka sistem akan melakukan kembali menghitung ulang dengan mempertimbangkan antar node yang terhubung. Lalu menampilkan hasil</p>

4.2.4 Proses Rute Algoritma FloydWarshall

Table 24 Penentuan rute floyd warshall

Flowchart	Source Code
-----------	-------------

<pre> graph TD Start([Start]) --> Inisialisasi[Inisialisasi lokasi awal dan akhir] Inisialisasi --> IterasiLama[Lakukan perhitungan pada iterasi k ke n] IterasiLama --> IterasiKecil[Lakukan perhitungan pada step w[i,j] hingga sel k < n] IterasiKecil --> HitungApakah{Hitung apakah w[i,j] > w[i,k] + w[k,j] ?} HitungApakah --> PilihWij{Pilih w[i,j] ?} PilihWij --> PilihWik{Pilih w[i,k] + w[k,j] ?} PilihWik --> SimpanJarak[Simpan jarak yang terpilih] SimpanJarak --> End([End]) </pre>	Terlampir pada lampiran L.2
<p>Hasil Pengujian</p>	<p>Keterangan</p> <p>Dari hasil pengujian, sistem akan mengirim request pada google maps API direction, pertama dalam mepresentasikan sebuah graph yaitu dengan menggunakan matriks $n \times n$, lalu menginisialisasi titik node awal dan titik node tujuan yang bertujuan untuk mengetahui titik awal dan akhir, kemudian sistem atau algoritma akan menghitung bobot dari iterasi $k-n$ atau sampai dengan iterasi terakhir sesuai dengan jumlah total simpul, kemudian setiap sel matriks akan dilakukannya pengecekan pada persamaan dan membandingkan semua kemungkinan lintasan dari semua simpul sampai ke titik tujuan, setelah ditemukannya lintasan yang dapat dilalui lalu akan disimpan dan dikembalikan oleh sistem ke <i>user</i> menampilkan hasil.</p>

Dari hasil pengujian alpha yang sudah dilakukan. Dapat ditarik kesimpulan bahwa sistem :

1. Fungsi pencarian rumah sakit berdasarkan filter kategori,tipe,nama,lokasi jarak sudah berhasil diimplementasikan.
2. Fungsi penentuan rute menuju rumah sakit berhasil dilakukan
3. Fungsi pelaporan darurat terhadap rumah sakit berhasil dilakukan
4. Fungsi menampilkan rumah sakit beserta data informasi berhasil dilakukan

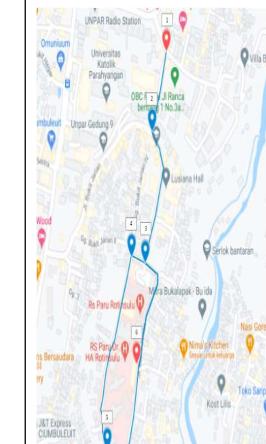
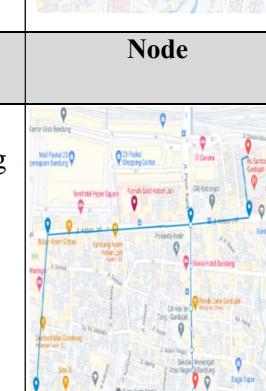
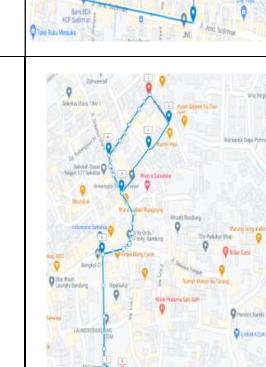
4.3 Pengujian Beta

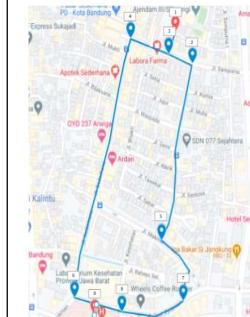
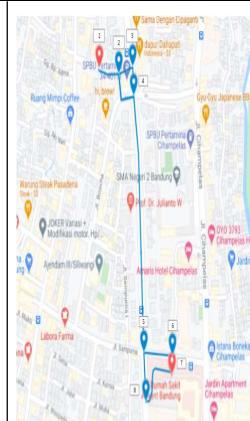
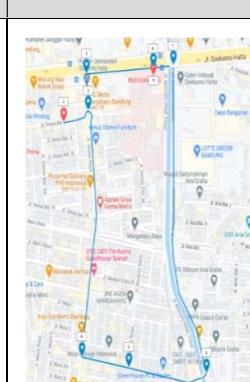
Pada implementasi pengujian algoritma a star dan floydwarsall ini maka dilakukan pencarian rute terpendek menuju layanan darurat seperti rumah sakit di kota bandung. Di implementasi yang dilakukan yaitu menerapkan dari kedua algoritma tersebut dengan memanfaatkan titik koordinat yang telah dibuat terhadap node-node yang akan dilewati pada aplikasi, dan perhitungan jarak real dihitung dengan menggunakan *haversine*. Sedangkan perbandingan dilakukan dengan membandingkan hasil dari aplikasi dengan perhitungan yang dilakukan secara manual sehingga mengetahui hasil rute baik dari tingkat akurasi, hasil perhitungan jarak dan waktu tempuh.

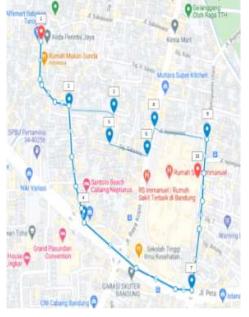
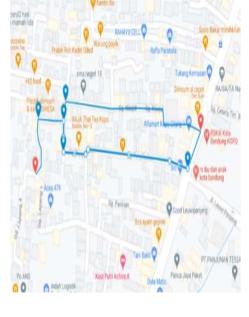
4.3.1 Pengujian Titik Awal User Node

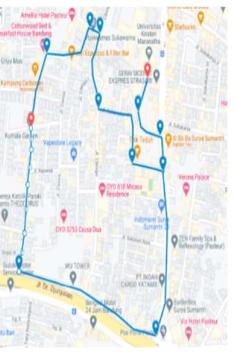
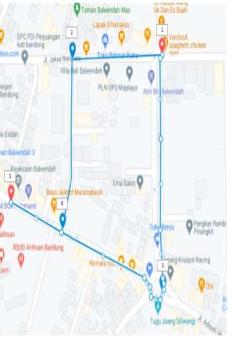
Pengujian dilakukan pada 27 titik kordinat awal user dan 27 tujuan layanan gawat darurat yang berbeda pada setiap pengujinya, dalam proses pengambilan titik awal.

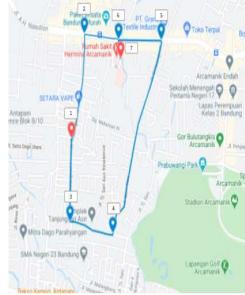
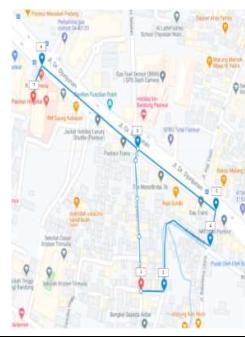
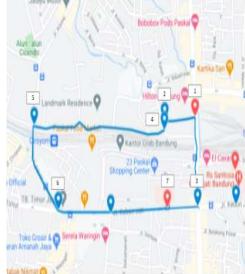
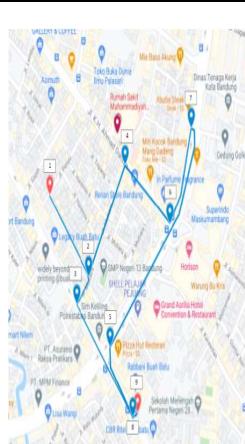
Uji		Titik Kordinat Awal	Titik Kordinat Tujuan	Node
1		Jalan Kebon Kawung	RS Mata Cicendo Kota Bandung	

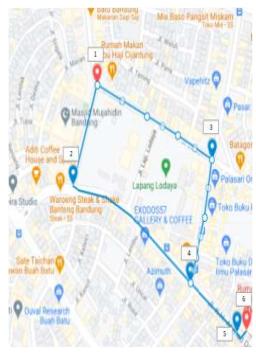
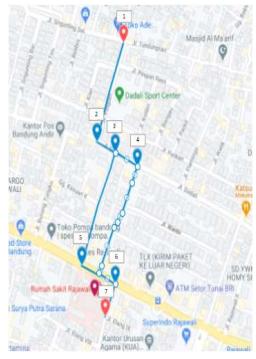
2		Jalan Raya Bentang 1	RS Dr H Rotinsulu	
Uji		Titik Kordinat Awal	Titik Kordinat Tujuan	Node
3		SMAN 4 Bandung	RS Santosa Hospital Bandung	
4		Jalan Cihulang	RSK Gigi dan Mulut FKG Unpad	

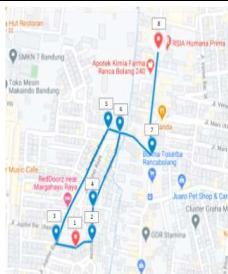
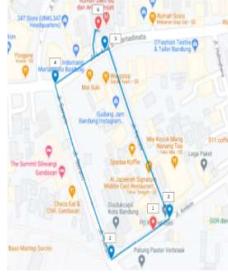
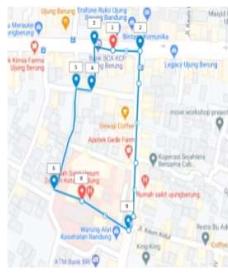
5	Jalan Boscha	RSUP Hasan Sadikin	
6	Jalan Lamping	RS Advent Bandung	
Uji	Titik Kordinat Awal	Titik Kordinat Tujuan	Node
7	Jalan Venus Bar	RS Al Islam Bandung	

8	Alfamart Babakan Tarogong	RS Immanuel Bandung	
9	Gg. Lapang 4	RS Khusus Ibu dan Anak Kota Bandung	
10	Jalan Ab Tuguh	RS.Lanud Dr M Salaman	
Uji	Titik Kordinat Awal	Titik Kordinat Tujuan	Node
11	Jalan Caringin	RS Santosa Hospital Bandung Raya	

12		Unpad Bandung	RS Boromeus	
13		Jalan Mustang	RSK Gigi dan Mulut Maranatha	
14		Jalan Jaksa Naranata	RSUD Al Ihsan Prov Jawa Barat	
Uji		Titik Kordinat Awal	Titik Kordinat Tujuan	Node

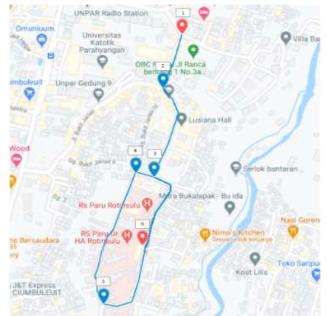
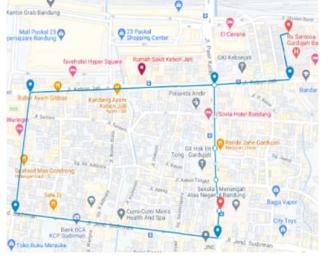
15		Jalan Sindang Sari 3	RS Hermina Arcamanik	
16		Jalan Cipedes Hilir	RS Hermina Pasteur	
17		Jalan Pasir Kaliki	RS Kebonjati	
18		Jalan Penyu	RS Mata Bandung eyes Center	
Uji		Titik Kordinat Awal	Titik Kordinat Tujuan	Node

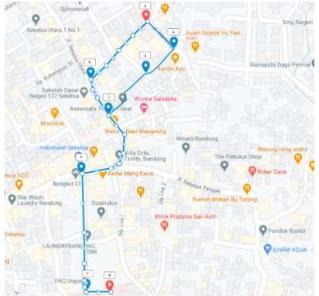
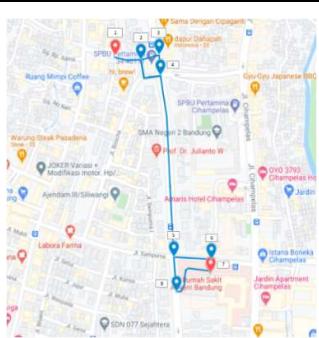
Uji	Titik Kordinat Awal	Titik Kordinat Tujuan	Node
19	Jalan Dr Rum	RS Melinda 2	
20	Jalan Lodaya	RS Muhammadiyah Bandung	
21	Jalan Tandungsari	RS Rajawali	
22	Jalan Lemah Neundeut	RS Santo Yusup	

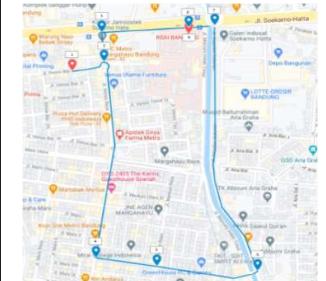
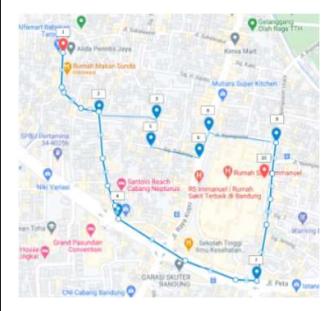
23		Jalan Kiara Condong	RSIA Graha Bunda	
24		Jalan Jupiter Utama	RSIA Humana Prima	
25		Jalan Ambon	RSIA Limijati	
26		Jalan Cimanuk	RSK Bedah Halmahera Niaga	
27		Jalan AH Nasution	RSUD Kota Bandung	

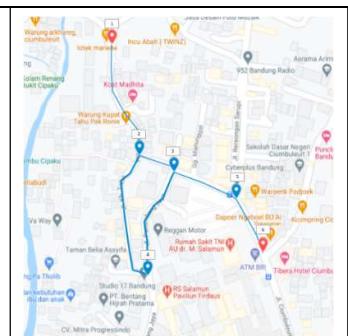
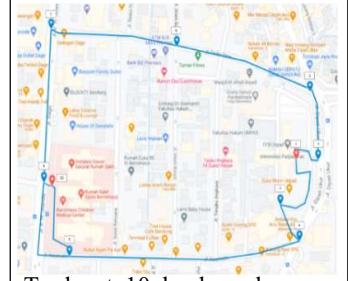
4.3.2 Pengujian Hasil Algoritma A Star

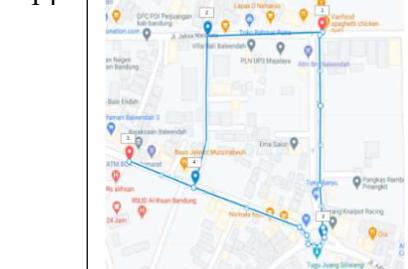
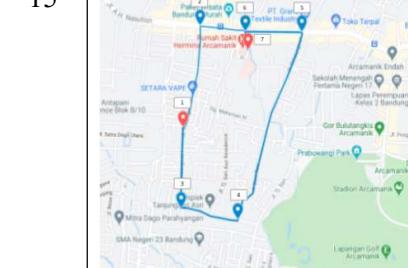
Table 25 Hasil pengujian a star

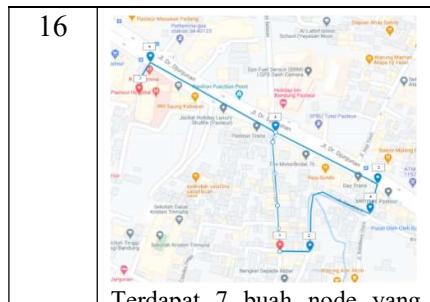
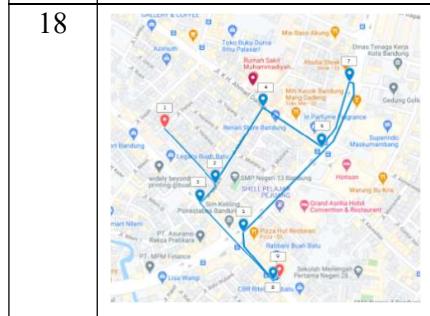
Pengujian	Node	Hasil Aplikasi	Hasil Perhitungan	Hasil Uji
1			Jln.Kebon Kawung (lokasi awal) v0-v1-v5 RS =.Mata Cicendo (lokasi tujuan)	Sesuai rujukan pada lampiran
2			Jln.Ranca Bentang (lokasi awal) v0-v1-v2-v3-v5 RS.Paru Dr H Rotinsulu (lokasi tujuan)	Sesuai rujukan pada lampiran
3			SMAN 4 Bandung (lokasi awal) v0-v1-v5 RS.Santosa Hospital (lokasi tujuan)	Sesuai rujukan pada lampiran
Pengujian	Node	Hasil Aplikasi	Hasil Perhitungan	Hasil Uji

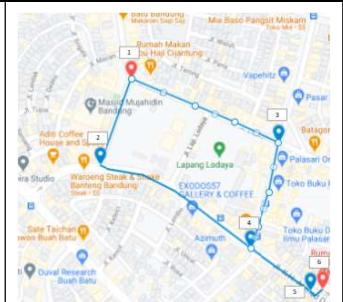
4	 <p>Terdapat 8 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>  <p>Jln.Ciheulang (lokasi awal) v0-v1-v2-v3-v5 RS.Gigi dan Mulut FKG Unpad (lokasi tujuan)</p>	Sesuai rujukan pada lampiran		
5	 <p>Terdapat 9 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>  <p>Jln.Boscha (lokasi awal) v0-v1-v3-v4-v8 RSUP Hasan Sadikin (lokasi tujuan)</p>	Sesuai rujukan pada lampiran		
6	 <p>Terdapat 8 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>  <p>Jln.Lamping (lokasi awal) v0-v1-v2-v3-v4-v5-v6 RS Advent Bandung (lokasi tujuan)</p>	Sesuai rujukan pada lampiran		
Pengujian	Node	Hasil Aplikasi	Hasil Perhitungan	Sesuai rujukan pada lampiran

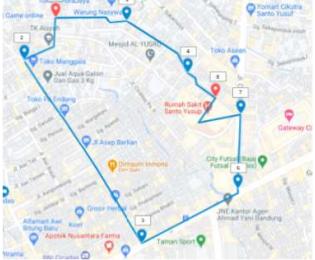
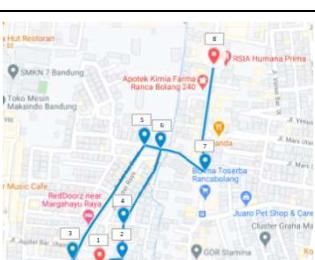
7		 <p>Rute Utama :</p> <p>Jarak Tempuh Dari : Jl. Venus Bar, No.41, Manjalihepa, Kec. Rancasari, Kota Bandung, Jawa Barat 40286, Indonesia</p> <p>Menuju : Jl. Sediarno-Hatta No.644, Manjalihepa, Kec. Rancasari, Kota Bandung, Jawa Barat 40286, Indonesia</p> <p>Total Jarak : 0.44 KM</p> <p>Dengan Durasi : 6 menit</p>	Jln.Venus Bar (lokasi awal) v0-v1-v2-v7-v8 RS.Al Islam Bandung (lokasi tujuan)	Sesuai rujukan pada lampiran
8		 <p>Rute Utama :</p> <p>Jarak Tempuh Dari : Jl. Babakan Tarogong No.24, Suki Asih, Kec. Bojongloa Kaler, Kota Bandung, Jawa Barat 40231, Indonesia</p> <p>Menuju : Jl. Nyengseret Utara No.292, Situsaeur, Kec. Bojongloa Kidul, Kota Bandung, Jawa Barat 40234, Indonesia</p> <p>Total Jarak : 0.83 KM</p> <p>Dengan Durasi : 10 min</p> <td data-bbox="1057 783 1341 1227"> Alfamart Babakan (lokasi awal) v0-v1-v2-v4-v5-v6-v7-v9 RS Immanuel Bandung (lokasi tujuan) </td> <td data-bbox="1057 1227 1341 1275">Sesuai rujukan pada lampiran</td>	Alfamart Babakan (lokasi awal) v0-v1-v2-v4-v5-v6-v7-v9 RS Immanuel Bandung (lokasi tujuan)	Sesuai rujukan pada lampiran
9		 <p>Rute Utama :</p> <p>Jarak Tempuh Dari : Gg. Panyilukan No.43, Kopo, Kec. Bojongloa Kaler, Kota Bandung, Jawa Barat 40233, Indonesia</p> <p>Menuju : Jl. Raya Kopo No.311, Kopo, Kec. Bojongloa Kaler, Kota Bandung, Jawa Barat 40233, Indonesia</p> <p>Total Jarak : 0.45 KM</p> <p>Dengan Durasi : 5 min</p> <td data-bbox="1057 1233 1341 1691"> Gg.Lapang 4 (lokasi awal) v0-v1-v2-v3-v6 RS.Khusus Ibu dan Anak (lokasi tujuan) </td> <td data-bbox="1057 1691 1341 1740">Sesuai rujukan pada lampiran</td>	Gg.Lapang 4 (lokasi awal) v0-v1-v2-v3-v6 RS.Khusus Ibu dan Anak (lokasi tujuan)	Sesuai rujukan pada lampiran
Pengujian	Node	Hasil Aplikasi	Hasil Perhitungan	Hasil Uji

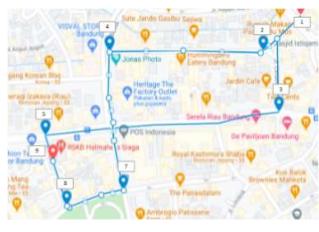
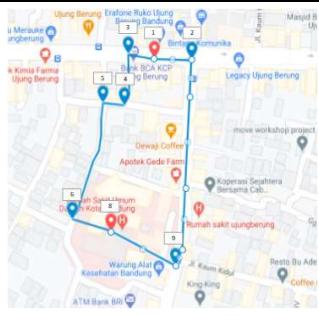
Pengujian	Node	Hasil Aplikasi	Hasil Perhitungan	Hasil Uji
10	 <p>Terdapat 6 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>	 <p>Jln. Kebon Tujuh (lokasi awal) v0-v1-v2-v4-v5 RS. Lanud Dr Salamun (lokasi tujuan)</p>	Sesuai rujukan pada lampiran	
11	 <p>Terdapat 12 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>	 <p>Jln. Caringin (lokasi awal) v0-v1-v6-v7-v8-v9-v11 RS. Santosa Hospital Kopo (lokasi tujuan)</p>	Sesuai rujukan pada lampiran	
12	 <p>Terdapat 10 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>	 <p>Unpad (lokasi awal) v0-v1-v7-v8-v9 RS. St. Borromeus (lokasi tujuan)</p>	Sesuai rujukan pada lampiran	

Pengujian	Node	Hasil Aplikasi	Hasil Perhitungan	Hasil Uji
13	 <p>Terdapat 11 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>	 <p>Rute Utama :</p> <p>Jarak Tempuh Dari : Jl. Mustang Blok 2 No.10, Sukawarna, Kec. Sukajadi, Kota Bandung, Jawa Barat 40164, Indonesia</p> <p>Menuju : Jl. Surya Sumantri No.53 B, Sukawarna, Kec. Sukajadi, Kota Bandung, Jawa Barat 40164, Indonesia</p> <p>Total Jarak : 1.17 KM</p> <p>Dengan Durasi : 16 min</p>	Jln. Mustang (lokasi awal) v0-v1-v6-v7-v8-v9-v4-v5-v10 RS.Gigi dan Mulut Maranatha (lokasi tujuan)	Sesuai rujukan pada lampiran
14	 <p>Terdapat 5 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>	 <p>Rute Utama :</p> <p>Jarak Tempuh Dari : Jl. Jakarta Naranatha No.2-3, Baleendah, Kec. Baleendah, Bandung, Jawa Barat 40375, Indonesia</p> <p>Menuju : Jl. Ki Astramanggala No.11, Baleendah, Kec. Baleendah, Bandung, Jawa Barat 40375, Indonesia</p> <p>Total Jarak : 0.44 KM</p> <p>Dengan Durasi : 5 min</p>	Jln.Jaksa Naranatha (lokasi awal) v0-v1-v3-v4 RSUD Al Ihsan Prov Jabar (lokasi tujuan)	Sesuai rujukan pada lampiran
15	 <p>Terdapat 7 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>	 <p>Rute Utama :</p> <p>Jarak Tempuh Dari : Jl. Sindangsari No.72, Antapani Wetan, Kec. Antapani, Kota Bandung, Jawa Barat 40291, Indonesia</p> <p>Menuju : Jl. A.H. Nasution No.50, Antapani Wetan, Kec. Antapani, Kota Bandung, Jawa Barat 40291, Indonesia</p> <p>Total Jarak : 0.77 KM</p> <p>Dengan Durasi : 10 min</p>	Jln.Sindang Sari 3 (lokasi awal) v0-v1-v5-v6 RS Hermina Arcamanik (lokasi tujuan)	Sesuai rujukan pada lampiran

Pengujian	Node	Hasil Aplikasi	Hasil Perhitungan	Hasil Uji
16	 <p>Terdapat 7 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p> 	Jln.Cipedes hilir (lokasi awal) v0-v4-v5-v6 RS Hermina Pasteur (lokasi tujuan)	Sesuai rujukan pada lampiran	
17	 <p>Terdapat 7 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS.</p> 	Jln.Pasir Kaliki (lokasi awal) v0-v2-v6 RS.Kebonjati (lokasi tujuan)	Sesuai rujukan pada lampiran	
18	 <p>Terdapat 9 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS.</p> 	Jln.Penyu (lokasi awal) v0-v1-v2-v7-v8 RS.Mata Bandung Eyes Center (lokasi tujuan)	Sesuai rujukan pada lampiran	

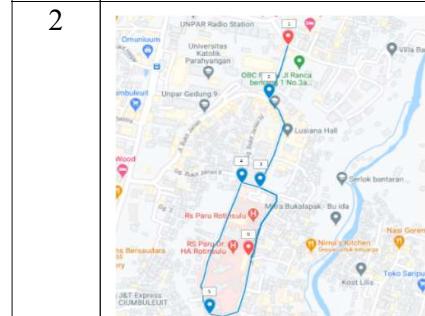
Pengujian	Node	Hasil Aplikasi	Hasil Perhitungan	Hasil Uji
19	 <p>Terdapat 6 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS.</p>	 <p>Rute Utama :</p> <p>Jarak Tempuh Dari : Jl. DR. Rum No.12, Pasir Kaliki, Kec. Cicendo, Kota Bandung, Jawa Barat 40171, Indonesia</p> <p>Menuju : Jl. Dr. Cipto No.1, Pasir Kaliki, Kec Cicendo, Kota Bandung, Jawa Barat 40171, Indonesia</p> <p>Total Jarak : 0.34 KM</p> <p>Dengan Durasi : 4 menit</p>	<p>Jln.Dr Rum (lokasi awal) v0-v1-v4-v5 RS.Melinda 2 (lokasi tujuan)</p>	Sesuai rujukan pada lampiran
20	 <p>Terdapat 6 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>	 <p>Rute Utama :</p> <p>Jarak Tempuh Dari : Jl. Sancang No.2, Malabang, Kec. Lengkong, Kota Bandung, Jawa Barat 40262, Indonesia</p> <p>Menuju : Jl. Banteng No.53, Turangga, Kec. Lengkong, Kota Bandung, Jawa Barat 40264, Indonesia</p> <p>Total Jarak : 0.72 KM</p> <p>Dengan Durasi : 9 menit</p>	<p>Jln.Lodaya (lokasi awal) v0-v1-v3-v4-v5 RS.Muhammadiyah Bandung (lokasi tujuan)</p>	Sesuai rujukan pada lampiran
21	 <p>Terdapat 7 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>	 <p>Rute Utama :</p> <p>Jarak Tempuh Dari : Jl. Tudung Sari, Kec. Andir, Kota Bandung, Jawa Barat 40114, Indonesia</p> <p>Menuju : Jl. Rajawali Barat No.38, Maleber, Kec. Andir, Kota Bandung, Jawa Barat 40114, Indonesia</p> <p>Total Jarak : 0.65 KM</p> <p>Dengan Durasi : 8 min</p>	<p>Jln,Tudung Sari (lokasi awal) v0-v1-v2-v4-v5-v6 RS.Rajawali (lokasi tujuan)</p>	Sesuai rujukan pada lampiran

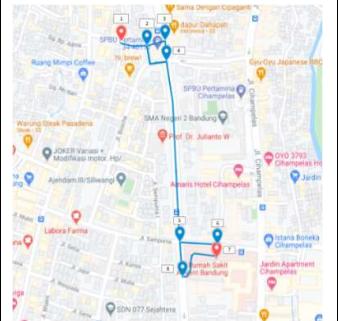
Pengujian	Node	Hasil Aplikasi	Hasil Perhitungan	Hasil Uji
22			Jln. Neundet (lokasi awal) v0-v2-v6-v7 RS St Yusup (lokasi tujuan)	Sesuai rujukan pada lampiran
23			Jln. Kiara Condong (lokasi awal) v0-v1-v2-v3-v4-v6 RS. Graha Bunda (lokasi tujuan)	Sesuai rujukan pada lampiran
24			Jln. Jupiter (lokasi awal) v0-v1-v3-v4-v6-v7 RS. Humana Prima (lokasi tujuan)	Sesuai rujukan pada lampiran

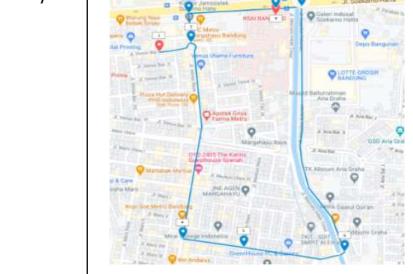
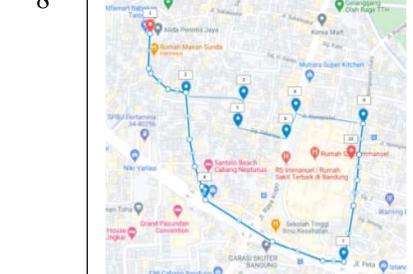
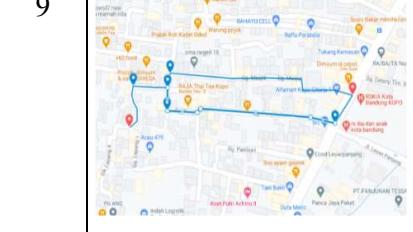
25	 <p>Terdapat 6 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>  <p>Jln.Ambon (lokasi awal) v0-v1-v3-v4-v5 RS.Limijati (lokasi tujuan)</p>	Sesuai rujukan pada lampiran
26	 <p>Terdapat 7 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>  <p>Jln.Cimanuk (lokasi awal) v0-v2-v6-v7 RSK.Bedah Halmahera Siaga (lokasi tujuan)</p>	Sesuai rujukan pada lampiran
27	 <p>Terdapat 8 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>  <p>Jln. AH Nasution (lokasi awal) v0-v1-v4-v5-v6-v7 RSUD Kota Bandung (lokasi tujuan)</p>	Sesuai rujukan pada lampiran

4.3.3 Pengujian Hasil Algoritma Floyd Warshall

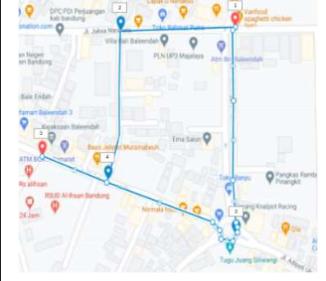
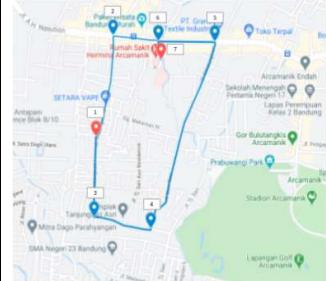
Table 26 Pengujian hasil floyd warshall

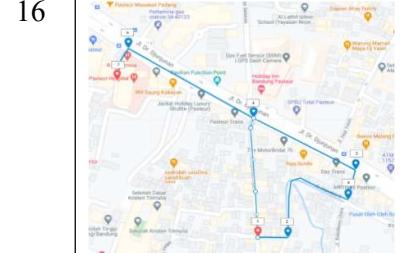
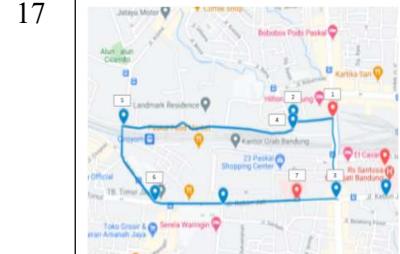
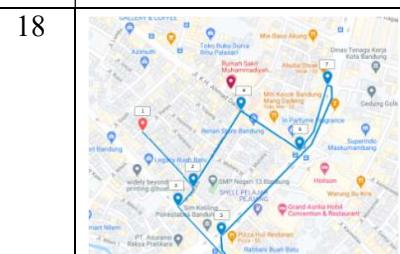
Pengujian	Node	Hasil Aplikasi	Hasil Perhitungan n	Hasil Uji
1			Jln.Kebon Kawung (lokasi awal) v0-v2-v3-v4 RS =.Mata Cicendo (lokasi tujuan)	Sesuai rujukan pada lampiran
2			Jln.Ranca Bentang (lokasi awal) v0-v1-v2-v3-v5 RS.Paru Dr H Rotinsulu (lokasi tujuan)	Sesuai rujukan pada lampiran
3			SMAN 4 Bandung (lokasi awal) v0-v2-v3-v4 RS.Santosa Hospital (lokasi tujuan)	Sesuai rujukan pada lampiran

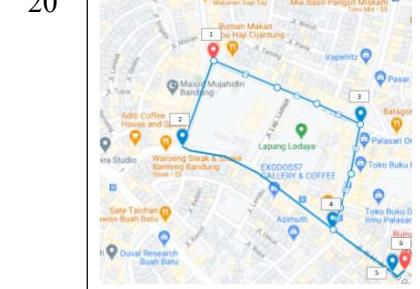
Pengujian	Node	Hasil Aplikasi	Hasil Perhitungan	Hasil Uji
4	 <p>Terdapat 8 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>	 <p>Rute Utama : Jarak Tempuh Dari : Jl. Ciheulang Baru I No.9A, Sekeloa, Kecamatan Coblong, Kota Bandung, Jawa Barat 40134, Indonesia Menuju : Jl. Sekeloa Sel. I No.32, Lebakgede, Kecamatan Coblong, Kota Bandung, Jawa Barat 40132, Indonesia Total Jarak : 0.52 KM Dengan Durasi : 2 min</p>	Jln.Ciheulang (lokasi awal) v0-v1-v2-v3-v5 RS.Gigi dan Mulut FKG Unpad (lokasi tujuan)	Sesuai rujukan pada lampiran
5	 <p>Terdapat 9 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>	 <p>Rute Utama : Jarak Tempuh Dari : Jl. Boscha No.1, Pasteur, Kec. Sukajadi, Kota Bandung, Jawa Barat 40161, Indonesia Menuju : Jl. Pasteur No.38, Pasteur, Kec. Sukajadi, Kota Bandung, Jawa Barat 40161, Indonesia Total Jarak : 0.97 KM Dengan Durasi : 3 min</p>	Jln.Boscha (lokasi awal) v0-v2-v7-v6-v5-v8 RSUP Hasan Sadikin (lokasi tujuan)	Sesuai rujukan pada lampiran
6	 <p>Terdapat 8 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>	 <p>Rute Utama : Jarak Tempuh Dari : Jl. Lamping No.16, Pasteur, Kec. Sukajadi, Kota Bandung, Jawa Barat 40161, Indonesia Menuju : Jl. Champelas No.161, Cipaganti, Kecamatan Coblong, Kota Bandung, Jawa Barat 40131, Indonesia Total Jarak : 0.74 KM Dengan Durasi : 3 minit</p>	Jln.Lamping (lokasi awal) v0-v1-v2-v3-v4-v5-v6 RS Advent Bandung (lokasi tujuan)	Sesuai rujukan pada lampiran

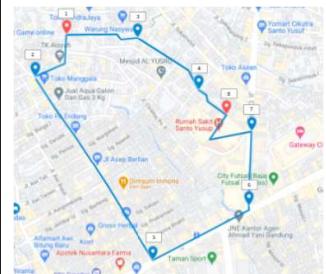
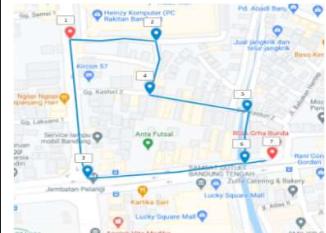
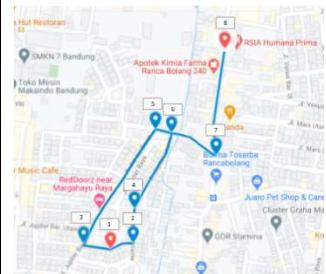
Pengujian	Node	Hasil Aplikasi	Hasil Perhitungan	Hasil Uji
7			Jln.Venus Bar (lokasi awal) v0-v1-v3-v4-v5-v6-v7-v8 RS.Al Islam Bandung (lokasi tujuan)	Sesuai rujukan pada lampiran
8			Alfamart Babakan (lokasi awal) v0-v1-v3-v8-v9 RS Immanuel Bandung (lokasi tujuan)	Sesuai rujukan pada lampiran
9			Gg.Lapang 4 (lokasi awal) v0-v1-v2-v4-v5-v6 RS.Khusus Ibu dan Anak (lokasi tujuan)	Sesuai rujukan pada lampiran

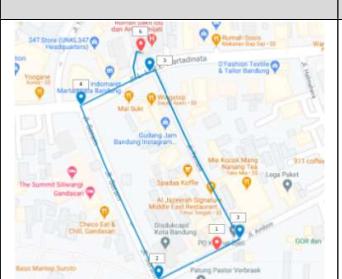
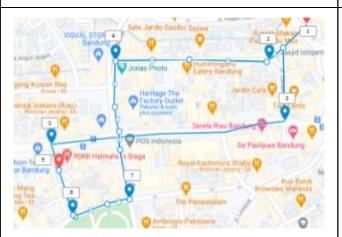
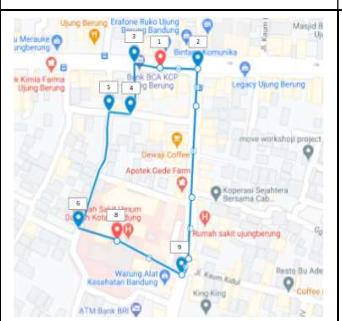
Pengujian	Node	Hasil Aplikasi	Hasil Perhitungan	Hasil Uji
10	 <p>Terdapat 6 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>	 <p>11.46 11:46 11:46 ← Rute Algoritma Floyd Wars...</p> <p>Rute Utama :</p> <p>Jarak Tempuh Dari : Jl. Kebon Tujuh (lokasi awal) v0-v1-v3-v2-v4-v5</p> <p>Menuju : RS. Lanud Dr Salamun (lokasi tujuan)</p> <p>Total Jarak : 0.59 KM</p> <p>Dengan Durasi : 3 min</p>	Jln. Kebon Tujuh (lokasi awal) v0-v1-v3-v2-v4-v5 RS. Lanud Dr Salamun (lokasi tujuan)	Sesuai rujukan pada lampiran
11	 <p>Terdapat 12 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>	 <p>11.50 11:50 11:50 ← Rute Algoritma Floyd Wars...</p> <p>Rute Utama :</p> <p>Jarak Tempuh Dari : Jl. Caringin No.101, Babakan Ciparay, Kec. Babakan Ciparay, Kota Bandung, Jawa Barat 40223, Indonesia</p> <p>Menuju : Jl. RS Santosa, Ciranjang, Kec. Babakan Ciparay, Kota Bandung, Jawa Barat 40227, Indonesia</p> <p>Total Jarak : 1.88 KM</p> <p>Dengan Durasi : 8 menit</p>	Jln. Caringin (lokasi awal) v0-v2-v3-v4-v5-v8-v9-v10-v11 RS. Santosa Hospital Kopo (lokasi tujuan)	Sesuai rujukan pada lampiran
12	 <p>Terdapat 10 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>	 <p>11.54 11:54 11:54 ← Rute Algoritma Floyd Wars...</p> <p>Rute Utama :</p> <p>Jarak Tempuh Dari : Jl. Dipati Ukur No.35, Lebakge, Kecamatan Coblong, Kota Bandung, Jawa Barat 40132, Indonesia</p> <p>Menuju : Jl. H. Juanda No 100, Lebakge, Kecamatan Coblong, Bandung, Jawa Barat 40132, Indonesia</p> <p>Total Jarak : 0.89 KM</p> <p>Dengan Durasi : 3 menit</p>	Unpad (lokasi awal) v0-v1-v7-v8-v9 RS. St. Borromeus (lokasi tujuan)	Sesuai rujukan pada lampiran

Pengujian	Node	Hasil Aplikasi	Hasil Perhitungan	Hasil Uji
13	 <p>Terdapat 11 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>		Jln. Mustang (lokasi awal) v0-v2-v3-v4- v5-v10-v11 RS.Gigi dan Mulut Maranatha (lokasi tujuan)	Sesuai rujukan pada lampiran
14	 <p>Terdapat 5 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>		Jln.Jaksa Naranatha (lokasi awal) v0-v2-v3-v4 RSUD Al Ihsan Prov Jabar (lokasi tujuan)	Sesuai rujukan pada lampiran
15	 <p>Terdapat 7 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>		Jln.Sindang Sari 3 (lokasi awal) v0-v2-v3-v4- v5-v6 RS Hermina Arcamanik (lokasi tujuan)	Sesuai rujukan pada lampiran

Pengujian	Node	Hasil Aplikasi	Hasil Perhitungan	Hasil Uji
16	 <p>Terdapat 7 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>	 <p>Rute Utama :</p> <p>Jarak Tempuh Dari : Jl. Cipedes Heger No.19a, Pajajaran, Kec. Cicendo, Kota Bandung, Jawa Barat 40173, Indonesia</p> <p>Menuju : Jl. Dr. Djunjunan No.11a, Pajajaran, Kec. Cicendo, Kota Bandung, Jawa Barat 40173, Indonesia</p> <p>Total Jarak : 0.48 KM</p> <p>Dengan Durasi : 2 menit</p>	Jln.Cipedes hilir (lokasi awal) v0-v4-v5-v6 RS Hermina Pasteur (lokasi tujuan)	Sesuai rujukan pada lampiran
17	 <p>Terdapat 7 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS.</p>	 <p>Rute Utama :</p> <p>Jarak Tempuh Dari : Jl. Pasir Kaliki No.39, Arjuna, Kec. Cicendo, Kota Bandung, Jawa Barat 40172, Indonesia</p> <p>Menuju : Jl. Kebon Jati No.152, Kb. Jeruk, Kec. Andir, Kota Bandung, Jawa Barat 40181, Indonesia</p> <p>Total Jarak : 2.05 KM</p> <p>Dengan Durasi : 8 menit</p>	Jln.Pasir Kaliki (lokasi awal) v0-v1-v3-v4-v5 RS.Kebonjati (lokasi tujuan)	Sesuai rujukan pada lampiran
18	 <p>Terdapat 9 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS.</p>	 <p>Rute Utama :</p> <p>Jarak Tempuh Dari : Jl. Penyu No.56, Turangga, Kec. Lengkong, Kota Bandung, Jawa Barat 40264, Indonesia</p> <p>Menuju : Jl. Buah Batu No.147, Turangga, Kec. Lengkong, Kota Bandung, Jawa Barat 40265, Indonesia</p> <p>Total Jarak : 1.36 KM</p> <p>Dengan Durasi : 6 menit</p>	Jln.Penyu (lokasi awal) v0-v1-v3-v4-v5-v6-v7-v8 RS.Mata Bandung Eyes Center (lokasi tujuan)	Sesuai rujukan pada lampiran

Pengujian	Node	Hasil Aplikasi	Hasil Perhitungan	Hasil Uji
19	 <p>Terdapat 6 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS.</p>		Jln.Dr Rum (lokasi awal) v0-v2-v3-v5 RS.Melinda 2 (lokasi tujuan)	Sesuai rujukan pada lampiran
20	 <p>Terdapat 6 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>		Jln.Lodaya (lokasi awal) v0-v1-v3-v4-v5 RS.Muhammadayah Bandung (lokasi tujuan)	Sesuai rujukan pada lampiran
21	 <p>Terdapat 7 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>		Jln.Tudung Sari (lokasi awal) v0-v1-v2-v4-v5-v6 RS.Rajawali (lokasi tujuan)	Sesuai rujukan pada lampiran

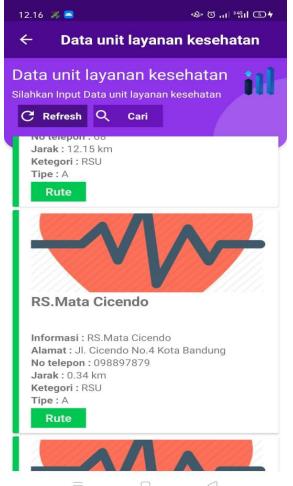
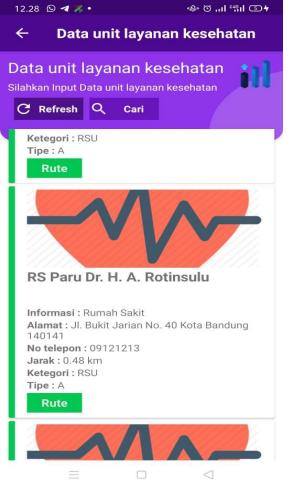
Pengujian	Node	Hasil Aplikasi	Hasil Perhitungan	Hasil Uji
22	 <p>Terdapat 8 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>		Jln.Neundet (lokasi awal) v0-v1-v3-v4-v5-v7 RS St Yusup (lokasi tujuan)	Sesuai rujukan pada lampiran
23	 <p>Terdapat 7 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>		Jln.Kiara Condong (lokasi awal) v0-v5-v4-v6 RS.Graha Bunda (lokasi tujuan)	Sesuai rujukan pada lampiran
24	 <p>Terdapat 8 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>		Jln.Yupiter (lokasi awal) v0-v1-v3-v4-v6-v7 RS.Humana Prima (lokasi tujuan)	Sesuai rujukan pada lampiran

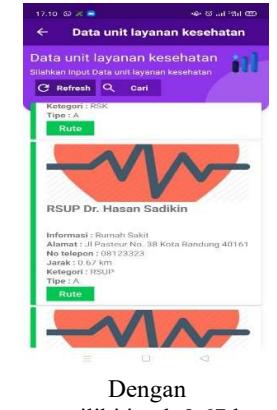
Pengujian	Node	Hasil Aplikasi	Hasil Perhitungan	Hasil Uji
25	 <p>Terdapat 6 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>	 <p>Rute Utama : Jarak Tempuh Dari : Jl. Ambon No.3, Citarum, Kec. Bandung Wetan, Kota Bandung, Jawa Barat 40115, Indonesia Menuju : Jl. L. L. R.E. Martadinata No.39, Citarum, Kec. Bandung Wetan, Kota Bandung, Jawa Barat 40115, Indonesia Total Jarak : 0.43 KM Dengan Durasi : 2 min</p>	Jln.Ambon (lokasi awal) v0-v2-v4-v5 RS.Limijati (lokasi tujuan)	Sesuai rujukan pada lampiran
26	 <p>Terdapat 7 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>	 <p>Rute Utama : Jarak Tempuh Dari : Jl. Cimanuk No.16, Citarum, Kec. Bandung Wetan, Kota Bandung, Jawa Barat 40115, Indonesia Menuju : Jl. L. L. R.E. Martadinata No.28, Citarum, Kec. Bandung Wetan, Kota Bandung, Jawa Barat 40115, Indonesia Total Jarak : 0.83 KM Dengan Durasi : 3 menit</p>	Jln.Cimanuk (lokasi awal) v0-v2-v6-v7 RSK.Bedah Halmahera Siaga (lokasi tujuan)	Sesuai rujukan pada lampiran
27	 <p>Terdapat 8 buah node yang ditandai sebagai persimpangan menuju lokasi tujuan dan node awal yang dideteksi oleh GPS</p>	 <p>Rute Utama : Jarak Tempuh Dari : Jl. Raya Ujungberung - Cigending No.82, Pakemitan, Kec. Ujungberung, Kota Bandung, Jawa Barat 40611, Indonesia Menuju : Jl. Rumah Sakit No.22, Pakemitan, Cinembo, Kota Bandung, Jawa Barat 45474, Indonesia Total Jarak : 0.28 KM Dengan Durasi : 2 menit</p>	Jln. AH Nasution (lokasi awal) v0-v2-v3-v7 RSUD Kota Bandung (lokasi tujuan)	Sesuai rujukan pada lampiran

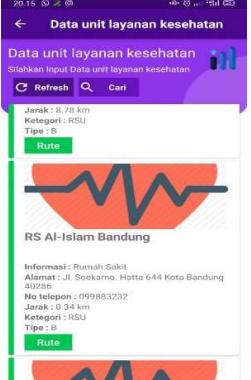
4.3.4 Pengujian Hasil Algoritma Floyd Warshall

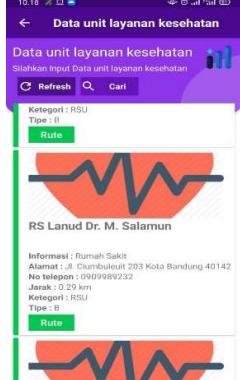
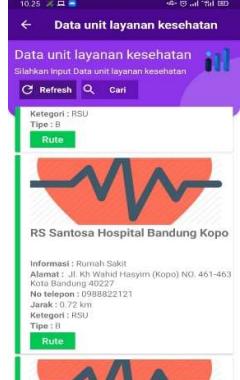
Pengujian dilakukan pada 27 titik kordinat awal dan 27 tujuan layanan darurat, sistem akan mengirim parameter koordinat bumi (*Latitude* dan *Longitude* berdasarkan lokasi perangkat *user* ke web service lalu melakukan perhitungan dan menampilkan data lokasi layanan darurat..

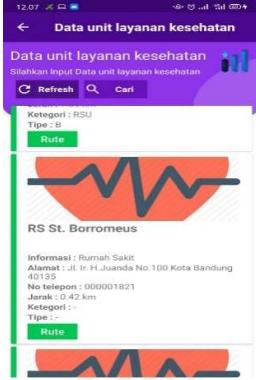
Table 27 Hasil perhitungan menampilkan rumah sakit

Uji	Titik Kordinat Awal	Titik Kordinat Tujuan	Hasil Uji
1.	-6.91261, 107.60314	-6,90956, 107.604773	 <p>Dengan memiliki jarak 0,34 km</p>
2.	-6.87453, 107,60708	-6,878753, 107,60605	 <p>Dengan memiliki jarak 0,48 km</p>

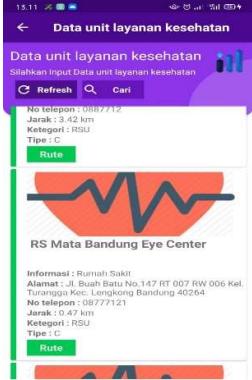
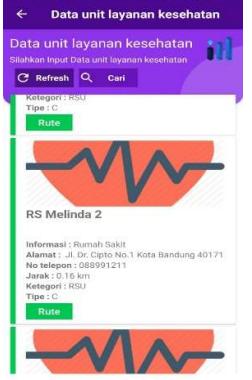
Uji	Titik Kordinat Awal	Titik Kordinat Tujuan	Hasil Uji
3.	-6.91937, 107.59839	-6.91617, 107.60023	 <p>Dengan memiliki jarak 0,52 km</p>
4.	-6.88678, 107.61968	-6.89028, 107.6192	 <p>Dengan memiliki jarak 0,39 km</p>
5.	-6.89068, 107.60072	-6.89627, 107.59872	 <p>Dengan memiliki jarak 0,67 km</p>

Uji	Titik Kordinat Awal	Titik Kordinat Tujuan	Hasil Uji
6.	-6.88745, 107.60122	-6.8918, 107,60325	 <p>Dengan memiliki jarak 0,54 km</p>
7.	-6.93999, 107.66628	-6.938875, 107,66917	 <p>Dengan memiliki jarak 0,34 km</p>
8.	-6.93266, 107.59269	-6.9354, 107.59726	 <p>Dengan memiliki jarak 0,6 km</p>

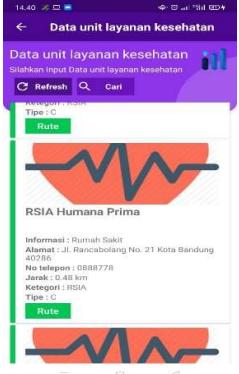
Uji	Titik Kordinat Awal	Titik Kordinat Tujuan	Hasil Uji
9.	-6.9435, 107.58815	-6.9431, 107.59127	 <p>Dengan memiliki jarak 0,36 km</p>
10.	-6.86184, 107.60349	-6.86405, 107.60518	 <p>Dengan memiliki jarak 0,29 km</p>
11.	-6.9466, 107.58362	-6.95246, 107.58612	 <p>Dengan memiliki jarak 0,72 km</p>

Uji	Titik Kordinat Awal	Titik Kordinat Tujuan	Hasil Uji
12.	-6.89339,107.61747	-6.099378, 107.61316	 <p>Dengan memiliki jarak 0,42 km</p>
13.	-6.8878,107.57722	-6.88682,107.58087	 <p>Dengan memiliki jarak 0,42 km</p>
14.	-7.00594, 107.62624	-7.00733, 107.62387	 <p>Dengan memiliki jarak 0,3 km</p>

Uji	Titik Kordinat Awal	Titik Kordinat Tujuan	Hasil Uji
15.	-6.90798, 107.6642	-6,90488, 107,66697	 <p>Dengan memiliki jarak 0,44 km</p>
16.	-6.89815, 107,59069	-6.91632, 107.58863	 <p>Dengan memiliki jarak 0,32 km</p>
17.	-6.91322, 107.59799	-6,91649, 107.59333	 <p>Dengan memiliki jarak 0,36 km</p>

Uji	Titik Kordinat Awal	Titik Kordinat Tujuan	Hasil Uji
18.	-6.93443,107.62113	-6.93784,107.62377	 <p>Dengan memiliki jarak 0,47 km</p>
19.	-6.90492, 107.59886	-6.9066, 107.59819	 <p>Dengan memiliki jarak 0,16 km</p>
20.	-6.92979,107.61975	-6.93351,107.62316	 <p>Dengan memiliki jarak 0,56 km</p>

Uji	Titik Kordinat Awal	Titik Kordinat Tujuan	Hasil Uji
21.	-6.90811, 107.57388	-6.91292, 107.57345	 <p>Dengan memiliki jarak 0,51 km</p>
22.	-6.90451, 107.6397	-6.90628, 107.64326	 <p>Dengan memiliki jarak 0,45 km</p>
23.	-6.9119, 107.64352	-6.91337, 107.64557	 <p>Dengan memiliki jarak 0,56 km</p>

Uji	Titik Kordinat Awal	Titik Kordinat Tujuan	Hasil Uji
24.	-6.94444, 107.66167	-6.94075,107.66377	 <p>Dengan memiliki jarak 0,48 km</p>
25.	-6.90857, 107.61439	-6.90636, 107.61354	 <p>Dengan memiliki jarak 0,26 km</p>
26.	-6.9039,107.62086	-6.90683,107.6155	 <p>Dengan memiliki jarak 0,65 km</p>

Uji	Titik Kordinat Awal	Titik Kordinat Tujuan	Hasil Uji
27.	-6.9141,107.69904	-6,91566,107.69862	 <p>Dengan memiliki jarak 0,18 km</p>

4.3.5 Pengujian Pengurutan Rumah Sakit

Pada tahap pengujian ini dilakukan pengujian guna untuk mengetahui tingkat keberhasilan sistem dalam proses pengurutan pencarian rumah sakit terdekat dari lokasi *user* menuju lokasi tujuan rumah sakit yang diperlukan. Lalu pengujian dilakukan sebanyak 5 kali dari lokasi yang berbeda. Hal ini dilakukan untuk mengetahui rumah sakit terdekat dari lokasi terkini *user*.

Table 28 Pengujian pengurutan rumah sakit

12	Lokasi	Titik User	Banyak Percobaan	Urutan Sesuai	Urutan Tidak Sesuai	Keberhasilan	Keterangan
1	Sekitar Jalan Kebon Kawung , Kota Bandung Jawa Barat , Indonesia	Lat:-6,91261 Long: 107,60314	5	5	-	100 %	Target; RS.Santosa Hospital Hasil: RS.Santosa Hospital
		Lat: -6,912302 Long:107,598255				100 %	Target; RS.Santosa Hospital
		Lat: -6,91251 Long:107,602877				100 %	Hasil: RS.Santosa Hospital
		Lat: -6,912417 Long: 107,600669				100 %	Target; RS.Mata Cicendo Hasil: RS.Santosa Hospital
		Lat: -6,906966 Long:107,601919				100 %	Target; RS.Santosa Hospital Hasil: RS.Melinda 2
		RATA - RATA				100 %	Hasil: RS.Melinda 2
2	Sekitar Jalan Cipaganti Kota Bandung Jawa Barat Indonesia	Lat: -6,890634 Long: 107,602436	5	5	-	100%	Target: RS Advent Bandung Hasil: RS Advent Bandung
		Lat: -6,885530 Long: 107,601786				100%	Target: RS Advent Bandung Hasil: RS Advent Bandung
		Lat: -6,882630 Long:107,600556				100%	Target: RS.Paru Dr H.A Rotinsulu Hasil: RS.Paru Dr H.A Rotinsulu.
		Lat: -6,904382 Long: 107,604221				100%	Target: RS. Mata Cicendo Hasil: RS. Mata Cicendo
		Lat: -6,874197 Long: 107,595859				100%	Target: RS.Paru Dr H.A Rotinsulu Hasil: RS.Paru Dr H.A Rotinsulu

Dalam proses pencarian rumah sakit terdekat dari lokasi terkini *user* dan berdasarkan jarak terdekat ,sistem dapat mengurutkan data jarak jika semua data tersebut dari lokasi *user* menuju rumah sakit telah didapatkan atau diterima, dengan guna mempermudah dalam pemilihan tujuan, selain itu data tersebut akan berhenti berubah pada layar apabila semua data tersebut selesai mengurutkan. Oleh sebab itu tingkat keberhasilan dalam percobaan pengujian yang dilakukan mencapai 100%.

4.3.6 Pengujian Hasil Jarak

Dilakukan pengujian jarak dengan membandingkan jarak yang dihasilkan oleh *google maps* dan jarak yang dihasilkan oleh program yang dibangun.

Table 29 Hasil jarak a star dan floydwarshall

No	Lokasi User Titik Awal	Tempat Tujuan Titik Akhir	Jarak Perhitungan Algoritma Floydwarsall	Jarak Perhitungan Algoritma Astar
			Jarak	Jarak
1.	Jln.Kebon Kawung	RS .Mata Cicendo	2,49 km	0,50 km
2.	Jln.Ranca Bentang	RS.Paru Dr H Rotinsulu	0,54 km	0,54 km
3.	SMAN 4 Bandung	RS.Santosa Hospital	1,95 km	0,72 km
4.	Jln.Ciheulang	RS.Gigi dan Mulut FKG Unpad	0,52 km	0,52 km
5.	Jln.Boscha	RSUP Hasan Sadikin	0,97 km	0,85 km
6.	Jln.Lamping	RS Advent Bandung	0,74 km	0,68 km
7.	Jln.Venus Bar	RS.Al Islam Bandung	2,02 km	0,44 km
8.	Alfamart Babakan	RS Immanuel Bandung	1,16 km	0,83 km
9.	Gg.Lapang 4	RS.Khusus Ibu dan Anak	0,41 km	0,45 km
10.	Jln.Kebon Tujuh	RS. Lanud Dr Salamun	0,59 km	0,34 km
11.	Jln.Caringin	RS. Santosa Hospital Kopo	1,88 km	1,20 km
12.	Unpad	RS. St Boromeus	0,89 km	0,82 km
13.	Jln. Mustang	RS.Gigi dan Mulut Maranatha	1,50 km	1,17 km
14.	Jln.Jaksa Naranatha	RSUD Al Ihsan Prov Jabar	0,55 km	0,44 km
15.	Jln.Sindang Sari 3	RS Hermina Arcamanik	1,90 km	0,77 km
16.	Jln.Cipedes hilir	RS Hermina Pasteur	0,48 km	0,48 km
17.	Jln.Pasir Kaliki	RS.Kebonjati	2,05 km	0,55 km
18.	Jln.Penyu	RS.Mata Bandung Eyes Center	1,36 km	0,55 km
19.	Jln.Dr Rum	RS.Melinda 2	0,38 km	0,34 km
20.	Jln.Lodaya	RS.Muhammadyah Bandung	0,72 km	0,72 km
21.	Jln.Tudung Sari	RS.Rajawali	0,65 km	0,65 km
22.	Jln.Neundet	RS St Yusup	1,20 km	0,51 km
23.	Jln.Kiara Condong	RS.Graha Bunda	0,39 km	0,36 km
24.	Jln.Yupiter	RS.Humana Prima	0,62 km	0,62 km
25.	Jln.Ambon	RS.Limijati	0,43 km	0,33 km
26.	Jln.Cimanuk	RSK.Bedah Halmahera Siaga	0,83 km	0,83 km
27.	Jln. AH Nasution	RSUD Kota Bandung	0,28 km	0,22 km

Table 30 Hasil jarak a star dan google maps

No	Lokasi User Titik Awal	Tempat Tujuan Titik Akhir	Jarak Perhitungan A star	JarakPerhitungan Pada Google Maps
			Jarak	Jarak
1.	Jln.Kebon Kawung	RS .Mata Cicendo	0,50 km	2,3 km
2.	Jln.Ranca Bentang	RS.Paru Dr H Rotinsulu	0,54 km	0,55 km
3.	SMAN 4 Bandung	RS.Santosa Hospital	0,72 km	1,9 km
4.	Jln.Ciheulang	RS.Gigi dan Mulut FKG Unpad	0,52 km	0,55 km
5.	Jln.Boscha	RSUP Hasan Sadikin	0,85 km	0,95 km
6.	Jln.Lamping	RS Advent Bandung	0,68 km	0,75 km
7.	Jln.Venus Bar	RS.Al Islam Bandung	0,44 km	2,0 km
8.	Alfamart Babakan	RS Immanuel Bandung	0,83 km	1,1 km
9.	Gg.Lapang 4	RS.Khusus Ibu dan Anak	0,45 km	0,35 km
10.	Jln.Kebon Tujuh	RS. Lanud Dr Salamun	0,34 km	0,6 km
11.	Jln.Caringin	RS. Santosa Hospital Kopo	1,20 km	1,8 km
12.	Unpad	RS. St Boromeus	0,82 km	1,0 km
13.	Jln. Mustang	RS.Gigi dan Mulut Maranatha	1,17 km	1,5 km
14.	Jln.Jaksa Naranatha	RSUD Al Ihsan Prov Jabar	0,44 km	0,55 km
15.	Jln.Sindang Sari 3	RS Hermina Arcamanik	0,77 km	1,9 km
16.	Jln.Cipedes hilir	RS Hermina Pasteur	0,48 km	0,55 km
17.	Jln.Pasir Kaliki	RS.Kebonjati	0,55 km	2,1 km
18.	Jln.Penyu	RS.Mata Bandung Eyes Center	0,55 km	1,4 km
19.	Jln.Dr Rum	RS.Melinda 2	0,34 km	0,4 km
No.	Lokasi User Titik Awal	Tempat Tujuan Titik Akhir	Jarak Perhitungan A star	JarakPerhitungan Pada Google Maps
			Jarak	Jarak
20.	Jln.Lodaya	RS.Muhammadyah Bandung	0,72 km	0,7 km
21.	Jln.Tudung Sari	RS.Rajawali	0,65 km	0,65 km
22.	Jln.Neundet	RS St Yusup	0,51 km	1,2 km
23.	Jln.Kiara Condong	RS.Graha Bunda	0,56 km	0,4 km
24.	Jln.Yupiter	RS.Humana Prima	0,48 km	0,6 km
25.	Jln.Ambon	RS.Limijati	0,26 km	0,6 km
26.	Jln.Cimanuk	RSK.Bedah Halmahera Siaga	0,65 km	0,9 km
27.	Jln. AH Nasution	RSUD Kota Bandung	0,20 km	0,28 km
		Rata – rata selisih		0,65

Dari hasil tabel diatas dilakukan perhitungan keakurasian *haversine* dengan dari hasil secara manual yang diambil gunakan *google maps* yang akan menjadi acuan tingkat ke akurasian dari segi jarak yang dihasilkan. Untuk menghitung presentase error menggunakan galat relatif dan dikonversi kedalam presentase seperti berikut :

$$e = \frac{|a - a^*|}{a} \times 100\%$$

dimana a adalah jarak sebenarnya dari *google maps* dan a^* adalah jarak yang dihasilkan perhitungan aplikasi.

Sedangkan untuk menghitung presentase hasil perhitungan

$$\text{Keakurasi} = 100 \% - e$$

Diketahui merupakan nilai yang didapatkan dari presentase error.

Sehingga hasil yang diperoleh dari hasil tingkat keakurasian jarak a star dengan secara manual gunakan *google maps* memiliki tingkat akurasi sebesar 99,56 % dengan rata rata selisih jarak 0,65.

Table 31 Hasil jarak a Floyd warshall.

No	Lokasi User Titik Awal	Tempat Tujuan Titik Akhir	Jarak Perhitungan Floyd Warshall	JarakPerhitungan Pada Google Maps
			Jarak	Jarak
1.	Jln.Kebon Kawung	RS .Mata Cicendo	2,49 km	2,3 km
2.	Jln.Ranca Bentang	RS.Paru Dr H Rotinsulu	0,54 km	0,55 km
3.	SMAN 4 Bandung	RS.Santosa Hospital	1,95 km	1,9 km
4.	Jln.Ciheulang	RS.Gigi dan Mulut FKG Unpad	0,52 km	0,55 km
5.	Jln.Boscha	RSUP Hasan Sadikin	0,97 km	0,95 km
6.	Jln.Lamping	RS Advent Bandung	0,74 km	0,75 km
7.	Jln.Venus Bar	RS.Al Islam Bandung	2,02 km	2,0 km
8.	Alfamart Babakan	RS Immanuel Bandung	1,16 km	1,1 km
9.	Gg.Lapang 4	RS.Khusus Ibu dan Anak	0,41 km	0,35 km
10.	Jln.Kebon Tujuh	RS. Lanud Dr Salamun	0,59 km	0,6 km
11.	Jln.Caringin	RS. Santosa Hospital Kopo	1,88 km	1,8 km
12.	Unpad	RS. St Boromeus	0,89 km	1,0 km

13.	Jln. Mustang	RS.Gigi dan Mulut Maranatha	1,50 km	1,5 km
14.	Jln.Jaksa Naranatha	RSUD Al Ihsan Prov Jabar	0,55 km	0,55 km
15.	Jln.Sindang Sari 3	RS Hermina Arcamanik	1,90 km	1,9 km
16.	Jln.Cipedes hilir	RS Hermina Pasteur	0,48 km	0,55 km
17.	Jln.Pasir Kaliki	RS.Kebonjati	2,05 km	2,1 km
18.	Jln.Penyu	RS.Mata Bandung Eyes Center	1,36 km	1,4 km
19.	Jln.Dr Rum	RS.Melinda 2	0,38 km	0,4 km
20.	Jln.Lodaya	RS.Muhammadyah Bandung	0,72 km	0,7 km
21.	Jln,Tudung Sari	RS.Rajawali	0,65 km	0,65 km
22.	Jln.Neundet	RS St Yusup	1,20 km	1,2 km
23.	Jln.Kiara Condong	RS.Graha Bunda	0,39 km	0,4 km
24.	Jln.Yupiter	RS.Humana Prima	0,62 km	0,6 km
25.	Jln.Ambon	RS.Limijati	0,43 km	0,6 km
26.	Jln.Cimanuk	RSK.Bedah Halmahera Siaga	0,83 km	0,9 km
27.	Jln. AH Nasution	RSUD Kota Bandung	0,28 km	0,28 km
	Rata – rata selisih		1,37	

Dari tabel dan perhitungan pengujian diatas didapatkan ke akurasian jarak a star dengan hasil secara manual gunakan *google maps* memiliki tingkat akurasi 99,44 % dengan rata rata selisih 1,37 km.

Berdasarkan hasil pengujian yang dilakukan dengan manual atau sistem terhadap metode algoritma yang dipakai dalam menentukan rute terpendek bahwa. Algoritma a star lebih dekat dari segi jarak dan pemilihan rute, diawali mendapatkan posisi awal menuju posisi akhir, lalu mencari nilai *heuristic*, dari nilai heuristic yang diperoleh kemudian dicari nilai $f(n)$ setelah $f(n)$ ditemukan selanjutnya melakukan cek nilai $f(n)$ dan memasukkan nilai $f(n)$ yang memiliki bobot terkecil kedalam *closed list* dilakukan berulang ulang sampai titik *goal tercapai*. Dengan kata lain algoritma a star memperhitungkan beberapa kemungkinan lintasan yang ada, seperti jika ada dua jalan yang panjang menuju arah *Goal* namun ada jalan lain dari posisi menuju *Goal* maka jalan itulah yang

akan dipilih dan jalan lain akan dihiraukan serta nilai bobot terkecil menjadi faktor pengaruh dalam penentuan rute.

Sedangkan algoritma floyd warshall pada saat pemilihan rute berbeda karena dalam proses perhitungannya melakukan pemecahan dengan memandang semua solusi yang akan diperoleh sebagai keputusan yang saling terkait karena solusi tersebut berasal dari solusi sebelumnya dan melakukan semua pengecekan dengan melakukan perulang-ulang sampai iterasi terakhir selesai oleh sebab nilai bobot akan mengalami perubahan setiap iterasi yang dilakukan setelah itu jalan atau rute akan dipilih.

Selain itu keberhasilan pengujian sistem yang sangat di pengaruhi oleh jaringan internet yang tersedia. Karena pada saat internet tidak stabil dapat memungkinkan data rumah sakit dan pengguna tidak terambil oleh sistem sehingga terjadinya kesalahan lalu data tersebut tidak muncul, dan juga berpengaruh pada ketepatan GPS dalam mengambil atau membaca lokasi pengguna yang nantinya akan mempengaruhi dalam kalkulasi jarak dari posisi pengguna ke lokasi tujuan layanan gawat darurat rumah sakit. Disamping itu juga akan berpengaruh terhadap rute yang dipilih.

BAB V

PENUTUP

5.1 Kesimpulan

Penelitian ini bertujuan untuk membantu dan mempermudah pengguna dalam pencarian rute terpendek dari lokasi *user* menuju lokasi tujuan rumah sakit. Berdasarkan hasil dari serangkaian penelitian yang telah dilakukan didapatkan sebuah analisis yang dijadikan poin kesimpulan diantaranya sebagai berikut :

1. Dalam proses pencarian rumah sakit terdekat dari lokasi terkini *user* dan berdasarkan jarak sistem dapat mengurutkan data jarak jika semua data tersebut dari lokasi *user* menuju rumah sakit telah didapatkan, sistem mencari dan mengurutkan jarak terdekat berdasarkan jarak dari hasil percobaan yang dilakukan tingkat keberhasilan dalam percobaan pengujian yang dilakukan mencapai 100%.
2. Dari penelitian yang telah dilakukan , dapat disimpulkan bahwa algoritma Floyd Warshall dan algoritma A Star dapat digunakan untuk menentukan rute terpendek dari asal menuju lokasi tujuan. Hasil dari salah satu pengujian studi kasus diperoleh rute terpendek menggunakan Floyd warshall dari lokasi terkini *user* menuju rumah sakit tujuan adalah v0-v2-v3-4-v5 dengan jarak 2,49 km dengan melalui 5 simpul, sedangkan a star adalah v0-v1-v5 dengan jarak 0,50 km dengan melalui 3 simpul. Disamping itu hasil ke akurasian berdasarkan jarak dari algoritma dengan secara manual yang didapatkan *google maps*, diperoleh algoritma floydwarsall 99,44 % dengan selisih rata-rata 1,37 sedangkan a star 99,56 % dengan selisih rata-rata 0,65. Dengan demikian dapat disimpulkan bahwa algoritma a star lebih efektif dan cocok dalam menentukan rute terpendek dilihat dari akurasi yang dihasilkan dengan *google maps* lebih besar, dan rata-rata selisih lebih kecil dibandingkan dengan floyd warshall. Hasil dari setiap perhitungan jarak yang berbeda disebabkan oleh konsep setiap metodenya, *a star* dengan menerapkan fungsi *heuristic* dan *floyd warshall* dengan mengitung semua iterasi, selain itu rute yang dihasilkan dapat berbeda dan dipengaruhi oleh bobot terkecil setiap metode.

DAFTAR PUSTAKA

- (Farid dan Yulanda 2017) *Analisa Algoritma Haversine Formula untuk pencarian rute terdekat rumah sakit.* (n.d.).
- Anam, S., Studi Matematika, P., & Matematika, J. (2016). *Konferensi Nasional Penelitian Matematika dan Pembelajarannya (KNPMP I) 873 Universitas Muhammadiyah Surakarta. 12.*
- Anugrah, F., Nuzulita, N., & Syawli, A. (2017). *SIMULASI ALGORITMA A* DAN DIJKSTRA PADA WAN.* 12(2).
- Azan Cahyadi, M., Arif Bambang, M. P., & Widhiarso, W. (n.d.). “BACTERIA DEFENSE.” *Julyxxxx, x, No.x, 1–5.*
- Eka Yulia, W. R., Istiadi, D., & Roqib, A. (2015). *PENCARIAN SPBU TERDEKAT DAN PENENTUAN JARAK TERPENDEK MENGGUNAKAN ALGORITMA DIJKSTRA (STUDI KASUS DI KABUPATEN JEMBER)* (Vol. 4, Issue 1).
- Rute, P., Menggunakan, T., Dijkstra, A., Kemacetan, M., Lintas, L., Purwokerto, D., & Rifanti, U. M. (2017). PEMILIHAN RUTE TERBAIK MENGGUNAKAN ALGORITMA DIJKSTRA UNTUK MENGURANGI KEMACETAN LALU LINTAS DI PURWOKERTO (BEST ROUTE SELECTION USE DIJKSTRA ALGORITHM TO REDUCE TRAFFIC CONGESTION IN PURWOKERTO). In *Jurnal Matematika dan Pendidikan Matematika* (Vol. 2, Issue 2).
- Satria Dahni, Y. (2017). *Sistem Informasi Penentuan Jalur Terpendek Bagi Pengantar Surat Menggunakan Algoritma Semut.* 2(2).
- Syafiq, A., Prastyo, R., & Listyorini, T. (2016). PEMANFAATAN GOOGLE MAPS API UNTUK PENCARIAN JALUR LOKASI SPBU TERDEKAT DI KOTA JEPARA & KUDUS DENGAN TEKNOLOGI NODE-JS. *Zuniar Rizqi Prastyo.*
- Syukriyah, Y., Falahah,), & Solihin, H. (2016). *PENERAPAN ALGORITMA A* (STAR) UNTUK MENCARI RUTE TERCEPAT DENGAN HAMBATAN.*