

**APLIKASI PENCARIAN RUTE TERDEKAT TEMPAT DOKTER
PRAKTEK FASILITAS BPJS MENGGUNAKAN
METODE DIJKSTRA**

**Oleh
LUVIANI
T3118037**

SKRIPSI

**Untuk Memenuhi Salah Satu Syarat Ujian
Guna Memperoleh Gelar Sarjana**



**PROGRAM SARJANA
TEKNIK INFORMATIKA
UNIVERSITAS IHSAN GORONTALO
GORONTALO
2023**

PERSETUJUAN SKRIPSI

APLIKASI PENCARIAN RUTE TERDEKAT TEMPAT DOKTER PRAKTEK FASILITAS BPJS MENGGUNAKAN METODE *DJIKSTRA*

Oleh
LUVIANI
T3118037

SKRIPSI

Untuk memenuhi salah satu syarat ujian guna memperoleh gelar Sarjana
Program Studi Teknik Informatika,
ini telah disetujui oleh Tim Pembimbing

Gorontalo, Desember 2023

Pembimbing I



Haditsah Annur, M.Kom
NIDN. 0908058403

Pembimbing II



Budy Santoso, S.Kom, M.Eng
NIDN. 0908048403

PENGESAHAN SKRIPSI

APLIKASI PENCARIAN RUTE TERDEKAT TEMPAT DOKTER PRAKTEK FASILITAS BPJS MENGGUNAKAN METODE *DJIKSTRA*

Oleh
LUVIANI
T3118037

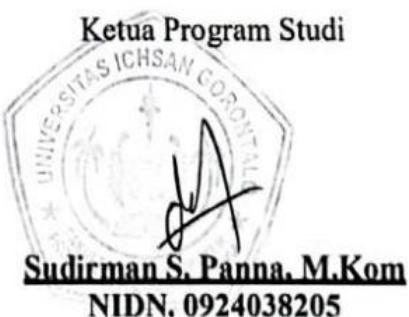
Diperiksa Oleh Panitia Ujian Strata (S1)
Universitas Ichsan Gorontalo

1. Ketua Penguji
Amiruddin, M.Kom, MCF
2. Anggota
Sunarto Taliki, M.Kom
3. Anggota
Kartika Chandra Pelangi, M.Kom
4. Anggota
Haditsah Annur, M.Kom
5. Anggota
Budy Santoso, S.Kom, M.Eng



.....
.....
.....
.....
.....

Mengetahui



PERNYATAAN SKRIPSI

Dengan isi saya menyatakan bahwa:

1. Karya tulis saya (Skripsi) ini adalah asli dan belum pernah di ajukan untuk mendapatkan gelar akademik (Sarjana) baik di Universitas Ichsan Gorontalo maupun di peguruan tinggi lainnya.
2. Karya tulis (Skripsi) saya ini adalah murni gagasan, rumusan dan penelitian saya sendiri, tanpa bantuan pihak lain, kecuali arahan dari Tim Pembimbing.
3. Dalam karya tulis (Skripsi) saya ini tidak dapat karya atau pendapat yang telah di publikasikan orang lain, kecuali secara tertulis dicantumkan sebagai acuan dalam naskah/sitasi dalam naskah dan di cantumkan pula dalam daftar pustaka.
4. Pernyataan ini saya buat dengan sesungguhnya dan apabila dikemudian hari terdapat penyimpangan dan tidak kebenaran dalam pernyataan ini, maka saya tidak bersedia menerima sanksi akademik berupa pencabutan gelar yang telah di peroleh karena karya tulis ini, serta sanksi lainnya sesuai dengan norma yang berlaku di Universitas Ichsan Gorontalo

Gorontalo, Desember 2023

Yang Membuat Pernyataan



Luviani

ABSTRACT

LUVIANI. T3118037. APPLICATION FOR FINDING THE NEAREST ROUTE TO DOCTOR PRACTICES FACILITIES OF SOCIAL INSURANCE ADMINISTRATION ORGANIZATION USING DIJKSTRA METHOD

Searching for the fastest route while traveling is necessary besides finding the destination. The reason for finding the fastest route is to streamline the trip and save travel costs. Another problem in making an effective trip is a city search carried out by someone who knows, someone who spreads hospitals, health centers, clinics, and pharmacies that serve the Social Insurance Administration Organization in Gorontalo City/Regency, and knows several nearby places. In the implementation of the Social Insurance Administration Organization in Gorontalo City/Regency branch, there are still obstacles faced by the community. The lack of information regarding the distribution of health facilities and information makes people confused when choosing and locating which health facility they will visit for treatment using a card from the Social Insurance Administration Organization. By designing this information system, people can map the health facilities that serve the Social Insurance Administration Organization. The designation of this system uses the Dijkstra algorithm to find the fastest route to the health facilities of Social Insurance Administration Organization to obtain a research result of a system that can find the fastest route to the health facilities of the Social Insurance Administration Organization. It is hoped that it can help the public in getting the fastest way to health facilities of the Social Insurance Administration Organization in Gorontalo City/Regency.

Keywords: *fastest route search system, health facilities, Social Insurance Administration Organization, Dijkstra algorithm*



ABSTRAK

LUVIANI. T3118037. APLIKASI PENCARIAN RUTE TERDEKAT TEMPAT DOKTER PRAKTEK FASILITAS BPJS MENGGUNAKAN METODE DIJKSTRA.

Pencarian rute terdekat saat melakukan perjalanan merupakan hal yang perlu dilakukan selain menemukan kota tujuan. Alasan pencarian rute terdekat adalah meringkas perjalanan dan menghemat biaya perjalanan. Persoalan lain dalam melakukan perjalanan yang efektif adalah penelusuran kota yang dilakukan oleh seorang mengetahui, dimana seorang penyebaran tempat dokter praktek yang melayani BPJS di Kabupaten Kota Gorontalo tersebut, mengetahui beberapa tempat terdekat. Pada penyelenggaran BPJS cabang Kabupaten Kota Gorontalo masih terdapat kendala yang dihadapi oleh masyarakat. Kurangnya informasi mengenai persebaran fasilitas kesehatan dan informasi mengenai fasilitas kesehatan itu sendiri, membuat masyarakat kebingungan pada saat memilih dan mencari letak posisi fasilitas kesehatan mana yang akan mereka kunjungi untuk berobat menggunakan kartu BPJS. Dengan dirancangnya sistem informasi ini kita dapat memetakan fasilitas kesehatan yang melayani BPJS. Pembuatan Sistem ini menggunakan algoritma Dijkstra untuk mencari rute terdekat tempat dokter praktek fasilitas kesehatan bpjs sehingga mendapatkan hasil penelitian sebuah sistem yang dapat mencari rute terdekat tempat dokter praktek fasilitas bpjs dan di harapkan dapat membantu masyarakat dalam mencari tempat dokter praktek fasilitas bpjs terdekat di Kabupaten Kota Gorontalo.

Kata kunci: Sistem Pencarian Rute Terdekat, Dokter Praktek Fasilitas BPJS, Algoritma Dijkstra.



KATA PENGANTAR

Alhamdulillah, puji syukur kehadiran Allah SWT atas segala limpahan taufiq, rahmat hidayah, serta inayah-Nya, sehingga penulis dapat menyelesaikan skripsi yang berjudul "***Aplikasi Pencarian Rute Terdekat Tempat Dokter Praktek Fasilitas BPJS Menggunakan Metode Dijkstrak***". Skripsi ini disusun sebagai salah satu syarat untuk menyelesaikan studi pada program S1 di Jurusan Teknik Informatika, Fakultas Ilmu Komputer, Universitas Ichsan Gorontalo.

Penulis menyadari sepenuhnya bahwa skripsi ini tidak mungkin terwujud tanpa bantuan dan dorongan dari berbagai pihak. Untuk itu dengan segala keikhlasan dan kerendahan hati, penulis mengucapkan banyak terimakasih kepada:

1. Dr. Juriko Abdussamad, M.Si, selaku ketua Yayasan Pengembangan Ilmu Pengetahuan dan Teknologi (YPIPT) Ichsan Gorontalo;
2. Dr. Abdul Gaffar La Tjokke, M.Si, selaku Rektor Universitas Ichsan Gorontalo;
3. Irvan Abraham Salihi, S.Kom., M.Kom, Selaku Dekan Fakultas Ilmu Komputer Universitas Ichsan Gorontalo;
4. Sudirman Melangi, M.Kom, selaku Wakil Dekan I Bidang Akademik dan Bidang Kemahasiswaan Fakultas Ilmu Komputer Universitas Ichsan Gorontalo;
5. Irma Surya Kumala Idris, M.Kom, selaku Wakil Dekan II Bidang Administrasi Umum dan Keuangan Fakultas Ilmu Komputer Universitas Ichsan Gorontalo;
6. Sudirman S. Panna, S.Kom, M.Kom, selaku Ketua Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Ichsan Gorontalo;
7. Haditsah Annur, M.Kom, selaku Pembimbing I, yang selalu membantu atau membimbing penulis untuk menyelesaikan usulan penelitian.
8. Budy Santoso, S.Kom, M.Eng, selaku Pembimbing II, yang selalu membantu atau membimbing penulis untuk menyelesaikan usulan penelitian.

9. Bapak dan Ibu Dosen Universitas Ichsan Gorontalo yang telah mendidik dan mengajarkan berbagai disiplin ilmu kepada penulis;
10. Teristimewa untuk orang tua saya, Ayahanda Ramin Matiro(alm) dan ibunda Sumiati yang telah memberikan kasih sayangnya, perhatian dan dukungan moril maupun material tanpa henti bagi penulis serta menenangkan dan menguatkan penulis untuk terus melangkah meraih mimpi-mimpi akan masa depan dalam doa-doa kebaikan. Orang tua motivator terbesar saya dan kepada tante serta paman saya, dan kepada saudara tercinta, Rifalgi dan Ronaldi terima kasih buat segalanya.
11. Kepada ibu Ivana F. Umboh selaku Kepaa Bagian SDMUK yang telah membantu penelitian saya.
12. Kepada Sahabat saya Multia Yusuf, Terima kasih sudah banyak membantu dalam segala hal.
13. Rekan-rekan seperjuangan yang telah banyak memberikan bantuan dan dukungan sangat besar kepada saya
14. Kepada semua pihak yang ikut membantudalam penyelesaian usulan penelitian ini yang tak sempat penulis sebutkan satu-persatu.

Penulis mengucapkan terima kasih banyak kepada semua pihak yang telah membantu dalam menyusun skripsi ini sehingga skripsi ini dapat terselesaikan. Penulis mengharapkan saran dan kritik sehingga skripsi ini dapat memberikan manfaat bagi bidang pendidikan serta bisa dikembangkan lagi lebih lanjut.

Gorontalo, Desember 2023

Luviani

DAFTAR ISI

HALAMAN PERSETUJUAN	iii
HALAMAN PENGESAHAN.....	iv
HALAMAN PERNYATAAN.....	v
ABSTRACT	vi
ABSTRAK	iii
KATA PENGANTAR.....	iii
DAFTAR ISI.....	v
DAFTAR TABLE.....	viii
DAFTAR GAMBAR.....	ix
BAB I.....	1
PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Identifikasi Masalah.....	3
1.3 Rumusan Masalah.....	4
1.4 Tujuan Penelitian.....	4
1.5 Manfaat Penelitian.....	4
1.5.1 Manfaat Teoretis	4
1.5.1 Manfaat Praktis	4
BAB II	5
LANDASAN TEORI.....	5
2.1 Tinjauan Studi.....	5
2.2 Tinjauan Pustaka.....	7
2.2.2 Fasilitas Kesehatan BPJS	7
2.2.3 Android	8
2.2.4 Google Maps Api.....	8
2.2.6 Penerapan Algorithma Dijkstra.....	10
2.2.7 Database.....	12
2.2.8 Analisis Sistem	12
2.2.9 Android Studio.....	17
2.2.11 Pengujian Sistem.....	17
2.2.11.1 White Box Testing	17
2.2.11.2 Black Box Texting	17

2.2.11.3	User Acceptance Testing.....	18
2.2.12	Perancangan Sistem	18
2.2.13	Implementasi sistem.....	18
BAB III.....		20
METODE PENELITIAN		20
3.1	Jenis, Metode, Subjek, Objek, Waktu, dan Lokasi Penelitian.....	20
3.2	Pengumpulan Data.....	20
3.3	Pengembangan Sistem.....	21
3.1.1	Analisis Sistem	22
3.1.2	Desain Sistem	22
3.1.3	Konstuksi Sistem	23
BAB IV		24
HASIL PENELITIAN		24
4.1	Hasil Pengumpulan Data	24
4.2	Penerapan Metode	25
4.3	Hasil Pembangunan Sistem	26
4.3.1	Use Case Diagram	26
4.3.2	Activity Diagram Login.....	27
4.3.3	Activity Diagram Tambah Data.....	28
4.3.4	Sequence Diagram Proses Cari Rute	29
4.4	Arsitektur Sistem.....	29
4.5	Interface Design	30
4.5.1	Mekanisme User.....	30
4.5.2	Mekanisme navigasi home	30
4.5.3	Mekanisme <i>Login</i>	30
4.5.4	Mekanisme Input Data Rute Jalan.....	31
4.5.5	Mekanisme Output Website	31
4.6	Data Desain	32
4.6.1	Struktur Data	32
4.7	Pengujian Sistem	35
4.7.1	Pengujian White Box.....	35
4.7.2	Flowchart.....	37
4.7.3	Flowchart.....	38
4.7.4	Perhitungan CC Pada Pengujian White Box	39

4.7.5 Pengujian BlackBox	40
BAB V.....	42
HASIL DAN PEMBAHASAN	42
5.1 Pembahasan Sistem	42
5.1.1 Tampilan Login	42
5.1.2 Tampilan Beranda	43
5.1.3 Tampilan Kategori Node	43
5.1.4 Tampilan Data Node.....	44
5.1.5 Tampilan Data Graph	45
5.1.6 Tampilan Pencarian Rute Algoritma <i>Dijkstra</i>	46
BAB VI.....	47
PENUTUP	47
6.1 Kesimpulan.....	47
6.2 Saran	47
DAFTAR PUSTAKA	48
LAMPIRAN.....	51

DAFTAR TABEL

Tabel 1.1 :	Daftar Dokter Praktek Kota Gorontalo	2
Table 2.1 :	Tinjauan Studi	5
Table 2.2 :	Simbol dari Use Case Diagram	13
Table 2.3 :	Simbol Class Diagram.....	14
Table 2.4 :	Simbol <i>Sequence</i> Diargam	16
Table 2.5 :	Simbol Diargam <i>Activity</i>	17
Tabel 4.1:	Hasil Pengumpulan Data	24
Tabel 4.2:	Pencarian rute terpendek dengan Algoritma <i>Dijkstra</i>	26
Tabel 4.3:	Mekanisme User	31
Tabel 4.4:	Tabel Jalan.....	33
Tabel 4.5:	Tabel Lokasi	33
Tabel 4.6:	Tabel Relasi.....	34
Tabel 4.7:	Tabel Rute	34
Tabel 4.8:	Tabel User	35
Tabel 4.9:	Basis Path	40
Tabel 4.10:	Hasil Pengujian BlackBox.....	41

DAFTAR GAMBAR

Gambar 2.1: Menentukan Titik Awal	10
Gambar 2.2: Bobot Terendah terhubung dengan A	10
Gambar 2.3: Bobot Terendah Terhubung dengan C	11
Gambar 2.4: Bobot Terendah Terhubung dengan D	11
Gambar 2.5: Hasil Akhir Jalur Terpendek.....	11
Gambar 3.1: Sistem yang Diusulkan	21
Gambar 4.1: Graf <i>Node</i>	26
Gambar 4.2: <i>Use Case Diagram</i>	27
Gambar 4.3: <i>Activity Diagram Login</i>	28
Gambar 4.4: <i>Activity Diagram Tambah Data</i>	29
Gambar 4.5: <i>Sequence Diagram Cari Rute</i>	30
Gambar 4.6: Mekanisme Navigasi Home User	31
Gambar 4.7: Mekanisme <i>Login</i>	31
Gambar 4.8: Mekanisme <i>Input</i> data jalan	32
Gambar 4.9: Mekanisme <i>Output</i>	32
Gambar 4.10 <i>Flowchart</i>	38
Gambar 4.11 <i>Flowgraph</i>	39
Gambar 5.1: Tampilan <i>Login</i>	43
Gambar 5.2: Tampilan Beranda	43
Gambar 5.3: Tampilan Kategori <i>Node</i>	44
Gambar 5.4: Tampilan Data <i>Node</i>	44
Gambar 5.5: Tampilan Data <i>Graph</i>	45
Gambar 5.6: Tampilan Pencarian Rute Algoritma <i>Dijkstra</i>	46

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pencarian rute terpendek saat melakukan perjalanan merupakan hal yang perlu dilakukan selain menemukan kota tujuan. Alasan pencarian rute terpendek adalah meringkas perjalanan dan menghemat biaya perjalanan. Persoalan lain dalam melakukan perjalanan yang efektif adalah penelusuran kota yang dilakukan oleh seorang mengetahui, dimana seorang penyebaran rumah sakit, puskesmas, klinik dan apotik yang melayani BPJS di Kabupaten Kota Gorontalo tersebut, mengetahui beberapa tempat terdekat, sehingga dia hanya akan sekali singgah di tempat tersebut sampai tempat terakhir yang akan dituju tercapai hingga kembali ke tempat asal [1].

Sistem informasi jika di tinjau lebih lanjut dapat di manfaatkan di berbagai bidang sehingga dapat membantu masyarakat salah satu contohnya pada badan penyelenggaraan jaminan social. Sistem informasi dapat di manfaatkan sebagai pemetaan fasilitas kesehatan yang melayani BPJS (Badan Penyelenggara Jaminan Sosial) [2]. Pada penyelenggaran BPJS cabang Kabupaten Kota Gorontalo masih terdapat kendala yang dihadapi oleh masyarakat. Kurangnya informasi mengenai persebaran fasilitas kesehatan dan informasi mengenai fasilitas kesehatan itu sendiri, membuat masyarakat kebingungan pada saat memilih dan mencari letak posisi fasilitas kesehatan mana yang akan mereka kunjungi untuk berobat menggunakan kartu BPJS, dikarenakan tidak semua fasilitas kesehatan yang berada di Kabupaten Kota Gorontalo melayani jasa untuk peserta BPJS [3].

Table1.1 : Daftar Dokter Praktek Kota Gorontalo

NO	NAMA	TELPON	ALAMAT
1	DR. HERNI A. PAKAYA	62-8124416894	jl. hb jassin (cx agus salim)
2	DR. NURHAYATI LAHAY	62-81356341340	DIAPOTEK ERA SEHAT GORONTALO,J
3	DR. NURJANA ALINTI	62-81356064041	JL. FARID LIPUTO
4	DR. SERLY DAUD	62-0811438974	JL. GELATIK
5	DR. SILVANA SONDAKH	62-82292392871	JL.RAJA EYATO NO. 153
6	DR. SRI A. IBRAHIM, M.KES	62-81340069867	JL BERINGIN NO. 60
7	DR. SUNARTI TUAH	08114380972	JL. HOS COKROAMINOTO
8	DR. THAIB SALEH	0435-8536913	JALAN BRIGJEN PIOLA ISA
9	drg. DELIYANA KATILI	62-811434104	JL. SULTAN BOTUTIHE NO. 1
10	drg. FAISAL [DRUS	62-8124422123	JL.PROF HB. JASSIN NO. 137

Dengan dirancangnya sistem informasi ini kita dapat memetakan fasilitas kesehatan yang melayani BPJS sehingga diharapkan dapat membantu dan memfasilitasi masyarakat dalam mencari informasi letak lokasi fasilitas kesehatan tersebut. Seiring dengan perkembangan teknologi sistem informasi pemetaan ini sudah dapat diakses melalui perangkat Smartphone android yang menawarkan fleksibilitas yang besar dan memungkinkan pengguna memperoleh hasil informasi secara cepat sesuai dengan kebutuhan.

Gambaran aplikasi pencarian rute terdekat fasilitas kesehatan BPJS yaitu lokasi dari fasilitas BPJS yang terdekat dari *user* akan langsung ditampilkan, sehingga *user* hanya perlu memilih lokasi yang akan dituju dan akan menampilkan rute terdekat menuju lokasi yang dituju.

Google maps api (Application Programming Interface) adalah layanan yang disediakan oleh google untuk menggunakan google maps dalam pembuatan aplikasi yang tidak dibuat oleh *google*, sehingga *developer* lain

dapat menggunakan fasilitas dari *google maps* [4].

Algoritma dijkstra adalah algoritma yang berfungsi untuk mencari rute lokasi terdekat dari titik lokasi user ke titik lokasi tujuan sehingga pengguna dapat menuju kelokasi tujuan dengan lebih cepat [5].

Dalam jurnal yang tulis oleh Ardi Wijaya dan Edo Kurniawan, algoritma dijkstra menggunakan prinsip greedy, yaitu pada setiap langkah yang dipilih node dengan bobot terkecil, kemudian sebuah node yang sudah terpilih akan dihubungkan dengan simpul lain yang belum terpilih sehingga algoritma dijkstra dapat menampilkan rute terdekat dari titik lokasi pengguna menuju titik lokasi tujuan. Algoritma dijkstra akan digunakan setelah aplikasi menampilkan lokasi pencarian fasilitas kesehatan BPJS dengan menggunakan *google maps api (Application Programming Interface)*, kemudian algoritma dijkstra akan mencari rute lokasi terdekat dengan *user*, sehingga *user* dapat menemukan fasilitas kesehatan BPJS terdekat [6].

Berdasarkan uraian yang telah disebutkan di atas maka dari itu penulis mengajukan judul "***Aplikasi Pencarian Rute Terdekat Tempat Dokter Praktek Fasilitas Bpjs Menggunakan Metode Dijkstrak***" guna sebagai solusi dalam membantu masyarakat dalam mencari letak lokasi fasilitas kesehatan BPJS tersebut. Android yang merupakan perangkat yang mudah dan banyak digunakan oleh user menjadi salah satu pertimbangan dalam perancangan aplikasi sistem ini.

1.2 Identifikasi Masalah

Berdasarkan uraikan di atas penulis termotivasi untuk membuat aplikasi dengan adanya *smartphone* berbasis Android yang sudah banyak digunakan saat ini.

1. Masih kurangnya informasi fasilitas BPJS cabang Kabupaten Kota Gorontalo.
2. Belum ada Aplikasi Pencarian Rute Terdekat Fasilitas Kesehatan BPJS Kabupaten Kota Gorontalo berbasis android.

1.3 Rumusan Masalah

Berdasarkan permasalahan, maka dapat diambil rumusan masalah dalam penelitian ini yaitu:

1. Bagaimana merekayasa Informasi Pencarian Fasilitas Kesehatan BPJS menggunakan Metode Algoritma Dijkstra?
2. Bagaimana cara membuat suatu sistem yang dapat membantu masyarakat dalam Pencarian Fasilitas Kesehatan BPJS di Kabupaten Kota Gorontalo?

1.4 Tujuan Penelitian

1. Mengimplementasikan Informasi Pencarian Fasilitas Kesehatan BPJS menggunakan Metode Algoritma Dijkstra.
2. Membangun Sistem Pencarian Fasilitas Kesehatan BPJS di Kabupaten Kota Gorontalo.

1.5 Manfaat Penelitian

Manfaat penelitian secara teoritis yang bisa didapatkan sebagai berikut:

1.5.1 Manfaat Teoretis

1. Dapat membantu masyarakat dalam mendapatkan informasi Pencarian Rute Terdekat Fasilitas Kesehatan BPJS dengan baik di Kabupaten Kota Gorontalo.
2. Dapat menambah wawasan mahasiswa dalam memanfaatkan teknologi untuk melakukan inovasi.

1.5.1 Manfaat Praktis

1. Dengan adanya ilmu yang telah didapat pada saat melakukan penelitian dengan bertambahnya wawasan dan pengalaman diharapkan membantu penulis sebagai referensi mereka.
2. Agar bisa memenuhi salah satu tugas akhir untuk bisa lulus dalam persyaratan kelulusan pada Kampus Universitas Ichsan Gorontalo.

BAB II

LANDASAN TEORI

2.1 Tinjauan Studi

Table 2.1: Tijauan Studi

NO	PENELITI	JUDUL	TAHUN	METODE	HASIL
1	Firginia L. B.Putri, Xaverius B. N. Najoan, Yaulie D. Y. Rindengan	Sistem Informasi Pemetaan Fasilitas Kesehatan BPJS Berbasis Android di Kota Bitung	2017	RAD (Rapid Application Development)	Aplikasi ini dapat membantu masyarakat dalam mencari lokasi fasilitas kesehatan yang melayani BPJS di kota Bitung serta masyarakat juga bisa melihat informasi yang berkaitan dengan fasilitas tersebut secara cepat dan efisien.
2	Karyaman dan Ardyana Restu	Sistem Informasi Geografis (SIG) Pemetaan Fasilitas Kesehatan Penerima BPJS Berbasis Android di Kota Bandung	2018	Formula Haversine	Hasilnya menampilkan informasi klinik, telpon klinik hingga rute perjalanan serta fungsi- fungsi, Formula Haversine untuk pencarian klinik terdekat. melakukan integrasi aplikasi android dengan Google Maps

NO	PENELITI	JUDUL	TAHUN	METODE	HASIL
					menampilkan lokasi dalam bentuk peta dan rute lokasi asal ke lokasi tujuan.
3	Ardi Wijaya dan Edo Kurniawan	Implementasi Algorithma Dijkstra Dalam Pencarian Rute Terpendek Fasilitas Kesehatan Tingkat I (Studi Kasus BPJS Kesehatan Kota Bengkulu)	2019	Algorithma Dijkstra	Hasil dari penelitian ini adalah aplikasi yang menggunakan Google Maps untuk menampilkan lokasi dan rute dari titik lokasi pengguna ke lokasi fasilitas kesehatan. Kemudian menggunakan algoritma. Dijkstra untuk memaksimalkan kinerja aplikasi dalam mencari rute terdekat.
4	Licantik, , Nova Noor Kamala Sari, dan Nina Heldayanti	Sistem Informasi Geografis Fasilitas Kesehatan BPJS di Kota Palangka Raya Berbasis Android	2020	Waterfall	Hasil yang dapatkan yaitu sistem aplikasi yang dapat menampilkan peta fasilitas kesehatan BPJS secara online lokasi serta informasi detail dari fasilitas kesehatan BPJS di Palangkaraya.

NO	PENELITI	JUDUL	TAHUN	METODE	HASIL
5	Agus Salam Sitio, Poningsih, Iin Parlina, Indra Gunawan dan Sumarno	Analisis Metode Sig (Sistem Informasi Geografis) Untuk Pemetaan Klinik BPJS Dengan Pendekatan RAD (Rapid Application Developme nt)	2021	RAD (Rapid Application Developme nt)	Hasil yang di dapatkan yaitu sistem yang menampilkan

2.2 Tinjauan Pustaka

2.2.1 Pencarian Lokasi Terdekat

Saat melakukan perjalanan ke tempat tujuan sering kali kita membawa peta sebagai petunjuk jalan. Penggunaan peta dalam bentuk ini secara visual mampu menggambarkan rute yang akan ditempuh dari kota asal ke kota tujuan. Pemakain peta kertas mempunyai kendala secara visual kita harus dapat mengurutkan rute mana yang akan di tempuh, salain itu penggunaan peta jenis ini tidak memungkinkan untuk dapat memberikan suatu saran rute manayang paling efektif yang dapat di lalui, sehingga dapat menampilkan informasi dalam bentuk peta tercepat, serta simbol yang terdapat dalam unsur muka jalan yang disajikan dapat mengetahui lokasi dari bangunan yang di tuju [7].

2.2.2 Fasilitas Kesehatan BPJS

Badan Penyelenggara Jaminan Sosial (BPJS) merupakan sebuah bentuk fasilitas yang di berikan oleh pemerintah sebagai program jaminan kesehatan untuk menjamin pengobatan yang baik untuk masyarakat indonesia. Fasilitas kesehatan untuk pengguna PBJS terdiri dari kelas 1,2,dan 3. Perbedaan fasilitas kelas 1,2,3 hanyalah harga dan fasilitas di dalam ruangan rawat inap seperti di dalam kelas 1 ada tv,kursi tamu dan lain lain , sedangkan kelas lain tidakada.

Untuk pelayanan yang di berikan tetap sama walaupun berbeda kelas[8].

2.2.3 Android

Android adalah sistem operasi yang digunakan untuk *smartphone* dan *tablet*. Android berfungsi sebagai perantara *device* dengan pengguna sehingga pengguna dapat menjalankan aplikasi yang terdapat pada *device*. Android merupakan sistem operasi berbasis linux yang pertama kali dibangun oleh Android Inc kemudian dibeli oleh Google pada tahun 2005, Android berfungsi sebagai perantara *device* dengan pengguna sehingga pengguna dapat menjalankan aplikasi yang terdapat pada *device* [9].

Dengan nama besar Google dan konsep open source pada OS Android, tidak membutuhkan waktu lama bagi android untuk bersaing dan menyisihkan Mobile OS lainnya seperti Symbian, Windos Mobile, Blackberry dan iOS, Dukungan Perangkat Tambahan Android dapat memanfaatkan kamera, layar sentuh, accelerometer, magnetometers, GPS, akselerasi 2D, dan Akselerasi 3D. Multi-TouchKemampuan layaknya handset modern yang dapat menggunakan dua jari atau lebih untuk berinteraksi dengan perangkat. Lingkungan Development memiliki fitur emulator, tools, untuk debugging, profil dan kinerja memori dan plugin untuk IDE Eclipse. MarketSeperti kebanyakan handphone yang memiliki tempat penjualan aplikasi, Market pada android merupakan katalog aplikasi yang dapat di download dan di install pada handphone melalui internet[10].

2.2.4 Google Maps Api

Google maps api (Application Programming Interface) adalah layanan yang disediakan oleh google untuk menggunakan google map dalam pembuatan aplikasi yang tidak dibuat oleh google, sehingga *developer* lain dapat menggunakan fasilitas dari *google maps* untuk membangun *software* yang menggunakan *Google maps api* sebagai pendukung *software* yang akan dibuat oleh *developer* [11].

Fasilitas *Google maps api* untuk menampilkan, memanipulasi peta beserta feature lainnya seperti tampilan satelit, jalan yang disediakan oleh google untuk

menggunakan peta google dalam aplikasi yang dibangun. *Google Maps API* memungkinkan para pengembang aplikasi maupun teknologi untuk menggunakan fasilitas layanannya dan mengizinkan para pengguna layanan ini untuk memodifikasi peta dan informasi yang ada di dalamnya [12].

2.2.5 Algorithma Dijkstra

Algoritma Dijkstra merupakan algoritma yang digunakan untuk mencari lokasi terpendek berdasarkan bobot terkecil dari suatu titik ke titik lain sehingga dapat mencari lokasi terdekat dari titik lokasi pengguna ke lokasi tujuan [13].

Pada algoritma Dijkstra juga termasuk dalam algoritma untuk masalah pencarian graf yang mampu menuntaskan masalah mencari lintasan terpendek dengan satu sumber pada sebuah graf yang digunakan untuk menyelesaikan masalah lintasan terpendek dengan satu sumber pada sebuah graf yang tidak memiliki cost sisi negatif, dan menghasilkan sebuah alur lintasan terpendek. Algoritma Dijkstra ini sering digunakan untuk melakukan routing. Algoritma Dijkstra ini sendiri ditujukan untuk menemukan jalur terpendek berdasarkan bobot terkecil dari titik keberangkatan ke titik lainnya. Misalkan sebuah gedung dan monumen dijadikan titik dan jalanan dijadikan garis, maka algoritma dijkstra akan melakukan perhitungan terhadap semua garis dengan bobot terkecil dari setiap titik [14].

Rumus Algorithma Dijkstra sebagai berikut [15]:

$$V(G) = \{v_1, v_2, v_3, \dots, v_n\}$$

L = Himpunan node

V(G) yang sudah terpilih dalam jalur terpendek

D(j) = Jumlah bobot jarak terkecil dari v1 ke vj

W(i,j) = Bobot garis dari node vi ke node vj

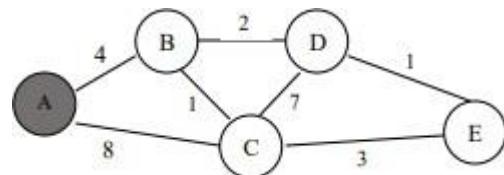
W*(1,j) = Jumlah bobot jarak terkecil dari v1 ke vj

Secara formal, algoritma Dijkstra untuk mencari jarak terpendek adalah:

1. $L = \{ \}$
 2. $V = \{ v_2, v_3, \dots, v_n \}$
 3. Untuk $i = 2, 3, \dots, n$, lakukan $D(i) = w(1,i)$
 4. Selama $v_n \in L$ lakukan :
 - Pilih node $v_k \in V - L$ dengan $D(k)$ terkecil $L = L \cup \{v_k\}$
 - Untuk setiap $v_j \in V - L$ lakukan : Jika $D(j) > D(k) + w(k,j)$ maka ganti $D(j)$ dengan $D(k) + w(k,j)$
- Untuk setiap $v_j \in V$, $w^*(1,j) = D(j)$.

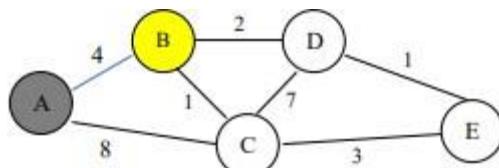
2.2.6 Penerapan Algorithma Dijkstra

Dalam penerapannya algoritma Dijkstra membutuhkan parameter awal dan akhir [16].



Gambar 2.1: Menentukan titik awal

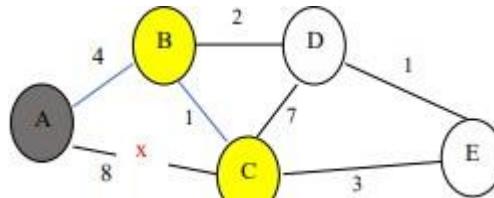
Gambar 2.1 menunjukkan titik A adalah titik awal dan titik tujuan adalah E. Algoritma Dijkstra akan mencari titik terdekat yang terhubung dengan titik A dan memiliki bobot terkecil [16].



Gambar 2.2: Bobot terendah terhubung dengan A

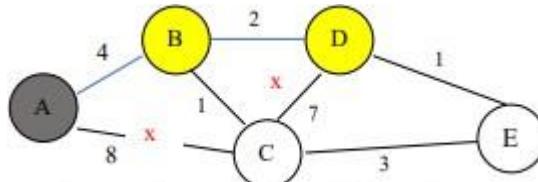
Gambar 2.2 menunjukkan bahwa titik yang memiliki bobot minimum adalah B. Selanjutnya, Algoritma Dijkstra akan membandingkan bobot dari titik yang terhubung dengan B dengan cara sebagai berikut: - Titik A-B-D = 6 - Titik A-B-C = 5 Dari perbandingan diatas maka didapatkan titik AB-C memiliki bobot

yang terkecil, sehingga algoritma Dijkstra akan melanjutkan perhitungan bobot yang terhubung dengan C. Pada tahap ini titik D tidak menjadi pilihan karena bobot yang dimiliki lebih besar dari bobot A-B-D dan A-B-C seperti gambar 2.3 [16].



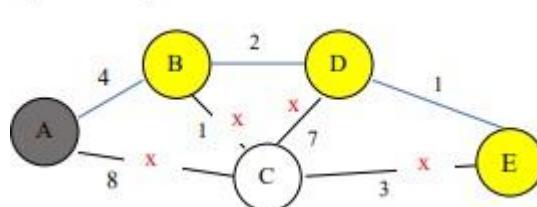
Gambar 2.3: Bobot terendah terhubung dengan C

Gambar 2.3 menunjukkan bahwa titik A-B-C-D = 12, sedangkan A-B-C-E = 8 dan sudah mencapai titik tujuan yaitu titik E, akan tetapi A-B-C-E belum bisa menjadi pilihan karena masih ada jalur lain yang memungkinkan memiliki bobot lebih rendah yaitu AB-D = 6 dan belum dibandingkan oleh algoritma [16].



Gambar 2.4: Bobot terendah terhubung dengan D

Gambar 2.4 menunjukkan bahwa titik D terhubung dengan C dan E, akan tetapi titik C tidak menjadi pilihan karena bobot yang dimiliki A-B-D-C melebihi bobot minimum A-B-C-E, sehingga algoritma Dijkstra akan langsung menghitung bobot A-B-D-E. Pada tahap ini, didapatkan nilai minimum A-B-D-E = 7 sedangkan A-B-C-E = 8, maka hasil jalur yang memiliki bobot terendah adalah A-B-D-E dapat dilihat pada Gambar 2.5 [16].



Gambar 2.5: Hasil akhir jalur terpendek

2.2.7 Database

Database merupakan kumpulan data yang tersimpan dalam tabel-tabel. Database sebagai penampung data yang digunakan sebagai data uji coba saat melakukan proses pencarian Database dapat mempermudah dalam penyimpanan data, ubah data, dan menghapus data sehingga data yang ada di dalam database dapat menghemat memori *computer* [17].

Dalam pembuatan suatu sistem informasi peran database sangat penting, karena berperan sebagai komponen utama dalam suatu sistem informasi dan merupakan komponen utama dalam menyediakan informasi. Database dapat mempertahankan kualitas informasi yang sehingga data informasi yang di berikan oleh database tidak rusak [18].

2.2.8 Analisis Sistem

2.2.8.1 Unified Modeling Language (UML)

Menyatakan bahwa *Unified Modeling Language* yaitu suatu metodologi untuk pembangunan sistem berorientasi objek dan digunakan sebagai pendukung pembangunan sistem, UML adalah bahasa standar, UML digunakan untuk memvisualisasikan, menentukan, mengonstruksi, dan mendokumentasikan artifak-artifak suatu sistem *software-intensive*. Secara umum UML merupakan suatu metode dalam pemodelan secara visual yang dipakai dalam sarana perancangan sistem yang berorientasi objek [19].

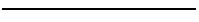
Perlu ketahui bahwa sistem tujuan dari UML yaitu mampu membantu mempermudah dalam pengembangan merekayasa perangkat lunak, Dan juga mampu Berguna sebagai blueprint, tujuannya mampu menjelaskan mengenai informasi yang lebih detail dalam perancangan berupa pada saat memulai penulisan coding suatu program UML juga sering juga dipakai untuk alat transfer ilmu dalam pengembangan aplikasi yang dari developer satu ke developer lainnya. UML punya fungsi alat sebagai penyambung penerjemah antara pengembang sistem dengan pengguna. Di sinilah manfaat pengguna bisa paham akan sistem yang baru akan dikembangkan [20].

2.2.8.2 Diagram Unified Modelling Language (UML)

2.2.8.2.1 Use Case Diagram

Menjelaskan sebuah interaksi sistem informasi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat [21]. Berikut ini tampilan dari *Use Case Diagram*:

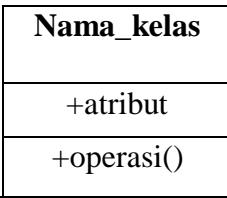
Tabel 2.3 :Simbol dari User Case Diagram [21].

SIMBOL	DESKRIPSI
<i>Use case</i> 	menampilkan pertukaran pesan antar unit dan aktor; serta ditambahkan salah satu kata kerja diawali frase nama <i>use case</i> .
<i>Aktor / actor</i> 	Merupakan gambaran orang, proses, Atau juga sistem yang mulai berkomunikasi dengan <i>use case</i> yang merupakan sistem informasi itu sendiri
<i>Asosiasi / association</i> 	Garis tanpa panah adalah interaksi komunikasi langsung aktor dengan sistem atau sistem dengan aktor
<i>Generalisasi</i> 	Garis dengan panah terbuka menampilkan interaksi aktor dengan sistem secara pasif
<i><<include>></i> 	<i>Include</i> menampilkan model pemanggilan <i>use case</i> oleh <i>use case</i> lain, contoh sederhana seperti memanggil suatu fungsi pada sebuah program.
<i><<extends>></i> 	<i>Extends</i> adalah penambahan fungsional <i>use case</i> lainnya jika kondisi terlengkapi.

2.2.8.2.2 Class Diagram

Class diagram yaitu suatu *diagram* yang bisa menampilkan hubungan antara kelas serta menjelaskan sumber detail kelas desain dalam sebuah sistem. *Class diagram* menampilkan operasi dan atribut suatu kelas yang dikoneksikan dengan objek yang terdapat dalam sistem [20]. *Class diagram* menggunakan simbol-simbol sebagai berikut:

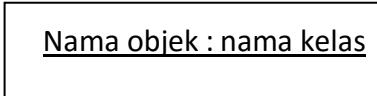
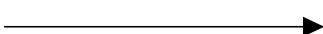
Tabel 2.4 Simbol *Class Diagram* [20]

SIMBOL	DESKRIPSI
Kelas 	Struktur kelas yang terdapat pada sistem
antar muka / <i>interface</i> 	<i>Interface</i> merupakan simbol yang sama dengan konsep yang terdapat dalam pemogramman berorientasi objek.
Asosiasi / <i>assocation</i> 	<i>Association</i> merupakan hubungan yang bermakna umum dengan kelas satu dan kelas lainnya..
Asosiasi Berarah / <i>Directed association</i> 	Asosiasi Berarah yaitu asosiasi dengan makna kelas digunakan oleh kelas yang lain.
Generalisasi 	Generalisasi adalah hubungan antar kelas umum dan khusus, realisasinya antar kelas ditambahkan dengan makna generalisasi-spesialisasi.
Aggregation 	Agregasi adalah hubungan antar kelas yang menjadi atribut untuk kelas lain.

2.2.8.2.3 Sequence Diagram

Merupakan diagram yang menampilkan aktifitas suatu objek yang terdapat dalam *use case* yang dikirim dan diterima antar objek. Berikut tampilan simbol-simbol *Sequence Diagram*:

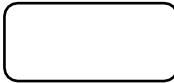
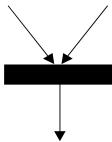
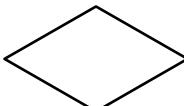
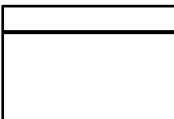
Tabel 2.5 Simbol *Sequence Diagram* [22]

SIMBOL	DESKRIPSI
<i>Aktor</i> 	Aktor Yaitu orang atau sistem eksternal lainnya yang menerima manfaat atau penggunaan sistem.
<i>Garis hidup / Lifeline</i> 	Garis hidup merupakan suatu garis yang terhubung dengan objek dan berbentuk garis putus-putus. Atau menyatakan suatu objek itu sendiri.
<i>Objek</i>  Nama objek : nama kelas	Menyatakan objek yang terlibat dalam sistem
<i>Message</i> 	merupakan simbol yang digunakan untuk mengirim pesan antar <i>class</i> lainnya.

2.2.8.2.4 Activity Diagram

Diagram aktifitas atau Diagram *Activity* yaitu menggambarkan *workflo* (aliran kerja) atau aktivitas dari sebuah sistem. Berikut tampilan simbol-simbolnya.

Tabel 2.6 Simbol *Diagram Activity* [20].

SIMBOL	DESKRIPSI
<i>Start point /Status awal</i> 	Start adalah aktivitas awal dari suatu diagram pada keadaan awal pada sistem mulai hidup
<i>End Point Status akhir</i> 	Merupakan akhir aktivitas dari suatu diagram
<i>Aktivities</i> 	Merupakan suatu aktivitas atau proses dari sebuah sistem
<i>Penggambungan / join</i> 	Merupakan suatu simbol yang menampilkan suatu aktivitas yang lebih dari satu dan digabungkan menjadi satu.
<i>Percabangan / Decision point</i> 	Merupakan simbol yang akan menampilkan tanda berupa pengambilan keputusan contoh, <i>true</i> atau <i>false</i> .
<i>Swimlane</i> 	Merupakan model pembagian <i>activity</i> diagram, serta menjadikan model ini sebagai model diagram yang bertanggung jawab terhadap aktivitas yang terjadi.

2.2.9 Android Studio

Menurut Rachmawati, Nugraha, dan Awaluddin(2017) Android Studio yaitu *Integrated Development Environment* (IDE) merupakan mengembangkan pada *platform* Android. Karena Android Studio merupakan IDE dari *Google*, Kemudian software ini secara langsung akan terintegrasi dengan *Google Maps* menggunakan API Key yang dibuat di lamanya yang disediakan dari *Google Maps* API untuk mengintegrasikan peta pada *software* sehingga peta akan secara otomatis ditampilkan di aplikasi yang dibuat [23].

2.2.10 Google Maps API

Application Programming Interface. API merupakan fungsi-fungsi bahasa pemrograman yang disediakan oleh aplikasi atau layanan agar bisa diintegrasikan dengan aplikasi yang dibuat. *Google API* adalah fungsi-fungsi pemrograman yang disediakan oleh *Google Maps* agar *Google Maps* bisa diintegrasikan kedalam aplikasi yang dibuat. Jadi, *Google API* dapat membantu mengetahui sebuah lokasi dengan akurat berdasarkan garis geografis yang ada [4]

2.2.11 Pengujian Sistem

Pengujian sistem yaitu suatu proses dalam membangun *software* yang wajib dilakukan untuk menguji sistem berfungsi sesuai keinginan. Beberapa tahap pengujian sistem sebagai berikut:

2.2.11.1 White Box Testing

Pengujian *white box* adalah metode desain *test case* dengan struktur desain procedural untuk mendapatkan *test case*. Modul yang menghasilkan *output* tidak sesuai akan dicari kesalahannya dari variabel, baris program dan parameter yang digunakan untuk diperbaiki [24].

2.2.11.2 Black Box Texting

Black box testing adalah metode yang digunakan untuk pengujian fungsional *software* tanpa melakukan pengujian terhadap kode program dan desain. Pengujian black box testing dibutuhkan oleh perusahaan untuk menguji *software* yang dibuat sudah sesuai dengan yang diinginkan oleh perusahaan. Pengujian dilakukan dengan cara mencoba *software* yang dibuat dan memasukan data ke dalam *form* yang telah disediakan [25].

2.2.11.3 User Acceptance Testing

User Acceptance Testing adalah proses menguji aplikasi melalui pengguna diluar ruang lingkup pengembang sehingga mendapatkan dokumen yang digunakan untuk bukti bahwa aplikasi sesuai dengan keinginan pengguna dan dapat memenuhi kebutuhan pengguna [26].

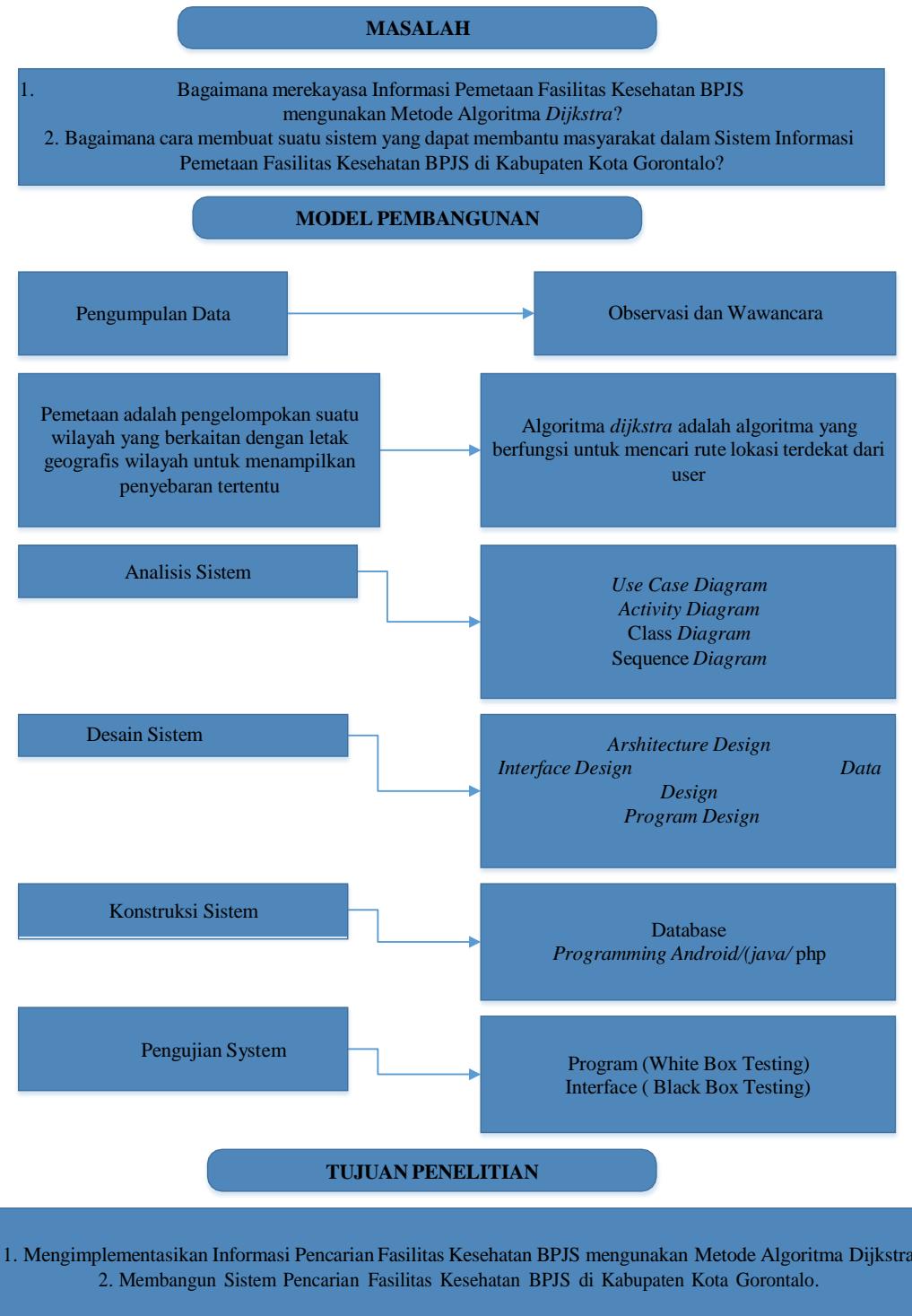
2.2.12 Perancangan Sistem

Perancangan sistem adalah unsur atau tahapan dalam pengembangan sistem komputerisasi. Perancangan sistem membutuhkan waktu yang lebih lama dalam pembangunan sistem daripada menyelesaikan masalah pada umumnya. Perancangan sistem sangat dibutuhkan dalam pengembangan sistem agar dapat menyelesaikan masalah yang terjadi padasaat melakukan pengembangan sistem sehingga sistem dapat dipastikan berjalan dengan baik [27].

2.2.13 Implementasi sistem

Implementasi sistem adalah proses yang menerapkan hasil dari perancangan sistem yang telah dibuat. Tujuan dari implementasi adalah memperoleh sistem yang sesuai dengan kebutuhan pengguna [28].

2.3 Kerangka Pikir



BAB III

METODE PENELITIAN

3.1 Jenis, Metode, Subjek, Objek, Waktu, dan Lokasi Penelitian

Dipandang dari tingkat penerapannya, maka penelitian ini merupakan penelitian terapan. Berdasarkan jenis informasi yang diolah, maka penelitian ini merupakan penelitian kuantitatif. Dilihat dari perlakuan terhadap data, maka penelitian ini merupakan penelitian konfirmatori.

Penelitian ini menggunakan metode penelitian studi kasus. Dengan demikian jenis penelitian ini adalah penelitian eksperimental.

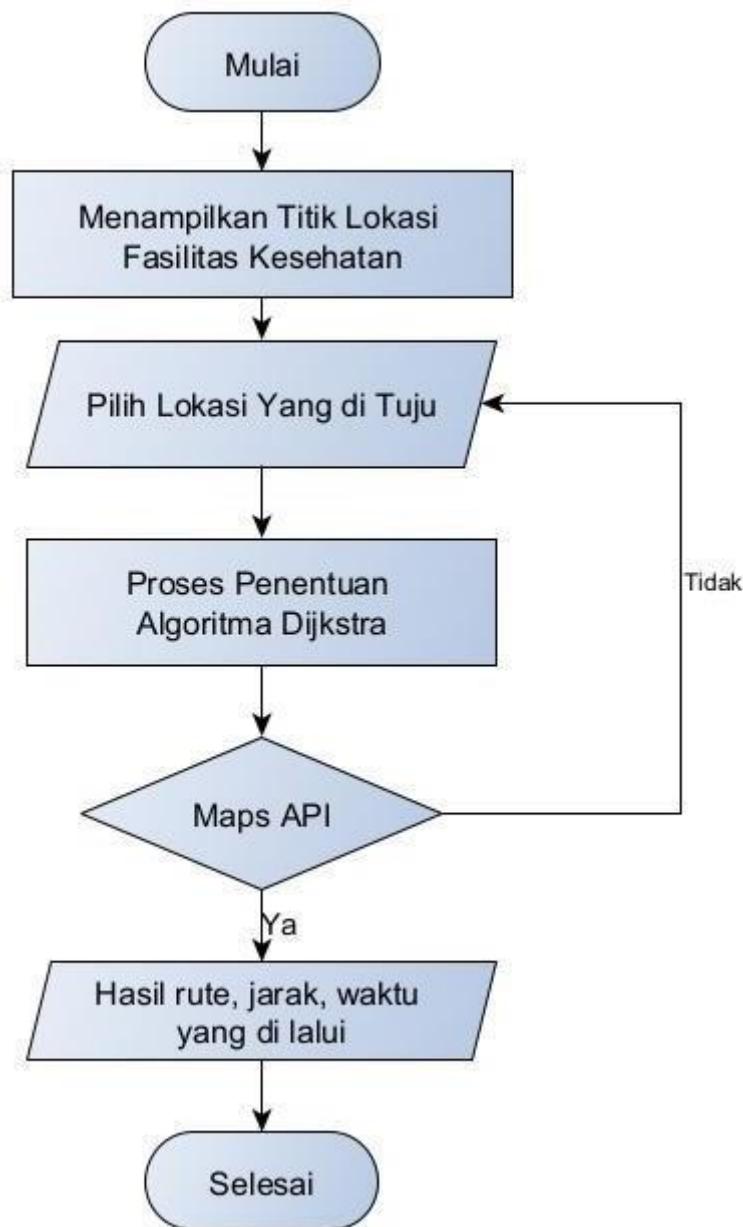
Subjek penelitian ini adalah membangun pada objek Aplikasi Pencarian Rute Terdekat Fasilitas Kesehatan BPJS Berbasis Android di Kabupaten Kota Gorontalo. Penelitian ini dimulai dari bulan Desember 2022 sampai dengan bulan Februari 2023 yang berlokasi di Dinas Kesehatan Kabupaten Kota Gorontalo.

3.2 Pengumpulan Data

Data Primer penelitian ini adalah nama fasilitas kesehatan yang melayani BPJS dan alamat dari fasilitas kesehatan yang melayani BPJS tersebut yang dikumpulkan menggunakan teknik observasi dan wawancara. Sedangkan Data Kepustakaan berasal dari penelitian (Sekunder), Adapun model teknik-teknik mencari data yang peneliti lakukan yaitu dengan pengumpulan data secara langsung dari sumber-sumber lain seperti jurnal, buku-buku pedoman yang berkaitan dengan penelitian yang dilakukan.

3.3 Pengembangan Sistem

Sistem yang diusulkan dapat digambarkan menggunakan *flowchart* dokument yang ditunjukan pada gambar 3.1 Berikut ini.



Gambar 3.1: Sistem yang Diusulkan

Peneliti memilih Algoritma dijkstra untuk memperoleh pencarian letak lokasi fasilitas kesehatan BPJS tersebut, sebagai solusi dalam membantu masyarakat dalam mencari letak lokasi fasilitas kesehatan BPJS tersebut.

3.1.1 Analisis Sistem

Analisis sistem menggunakan pendekatan berorientasi objek yang digambarkan dalam bentuk.

- a. *Funtional Modelling*, menggunakan alat bantu UML, dalam bentuk:
 - *Use Case Diagram*
 - *Actifity Diagram*
- b. *Structural Modelling*, menggunakan alat bantu UML, dalam bentuk:
 - *Class Diagram*
- c. *Behavioral Modelling*, menggunakan alat bantu UML, dalam bentuk:
 - *Sequence Diagram*

3.1.2 Desain Sistem

Desain sistem menggunakan pendekatan berorientasi objek yang digambarkan dalam bentuk.

- a. *Architektucture Design*, menggunakan alat bantu Android Studio, dalam bentuk:
 - Model jaringan dari sistem *client server*
 - Spesifikasi hardware dan sofware yang di rekomendasikan
- b. *Interface Design*, menggunakan alat bantu Android Studio, dalam bentuk:
 - Mekanisme User.
 - Mekanisme Navigasi
 - Mekanisme Input
 - Mekanisme Output
- c. *Program Desaign*, menggunakan alat bantu Android Studio, dalam bentuk:
 - *Class*
 - *Attibutes*
 - *Methot*
 - *Event*

3.1.3 Konstruksi Sistem

Pada tahap ini menerjemahkan hasil pada tahap analisis dan desain ke dalam kode-kode program komputer kemudian membangun sistemnya. Alat bantu yang digunakan pada tahap ini adalah Android Studio dengan bahasa pemrograman *Java*. Alat bantu database serta untuk perancangan *report* yang digunakan menggunakan *Google Maps Api sebagai* proses pencarian lokasi dan output dari sistem ini berupa sebuah mencari letak lokasi fasilitas kesehatan BPJS Kabupaten Kota Gorontalo sehingga dapat dijalankan oleh pengguna untuk ke tempat tujuan.

BAB IV

HASIL PENELITIAN

4.1 Hasil Pengumpulan Data

Pengumpulan data dalam penelitian ini dilakukan melalui wawancara untuk memperoleh informasi yang ingin dikumpulkan. Bentuk informasi yang diperoleh dinyatakan dalam tulisan, Metode ini digunakan untuk menemukan rute terdekat menuju fasilitas kesehatan bpjs di Kabupaten Kota Gorontalo

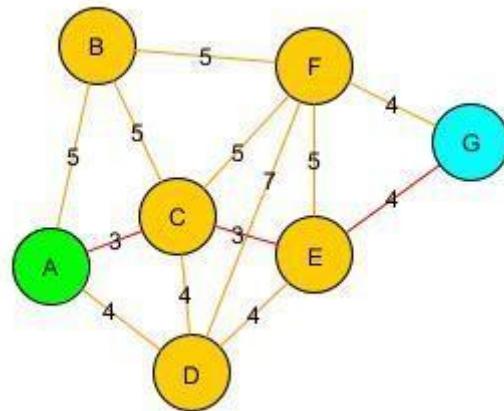
Berdasarkan hasil pengumpulan data, diperoleh data primer sebagai berikut:

Tabel 4.1: Hasil Pengumpulan Data

NO	NAMA	TELPON	ALAMAT
1	DR. HERNI A. PAKAYA	62-8124416894	jl. hb jassin (cx agus salim)
2	DR. NURHAYATI LAHAY	62-81356341340	DIAPOTEK ERA SEHAT GORONTALO,J
3	DR. NURJANA ALINTI	62-81356064041	JL. FARID LIPUTO
4	DR. SERLY DAUD	62-0811438974	JL. GELATIK
5	DR. SILVANA SONDAKH	62-82292392871	JL.RAJA EYATO NO. 153
6	DR. SRI A. IBRAHIM, M.KES	62-81340069867	JL BERINGIN NO. 60
7	DR. SUNARTI TUAH	08114380972	JL. HOS COKROAMINOTO
8	DR. THAIB SALEH	0435-8536913	JALAN BRIGJEN PIOLA ISA
9	drg. DELIYANA KATILI	62-811434104	JL. SULTAN BOTUTIHE NO. 1
10	drg. FAISAL [DRUS	62-8124422123	JL.PROF HB. JASSIN NO. 137

4.2 Penerapan Metode

Algoritma Dijkstra adalah algoritma yang berfungsi untuk mencari rute terdekat menuju titik tujuan. Ilustrasi penerapan metode Algoritma Dijkstra dari node a menuju node g ke dalam sistem ini sebagai berikut:



Gambar 4.1 Graf Node

Tabel 4.2 Pencarian rute terpendek dengan Algoritma Dijkstra

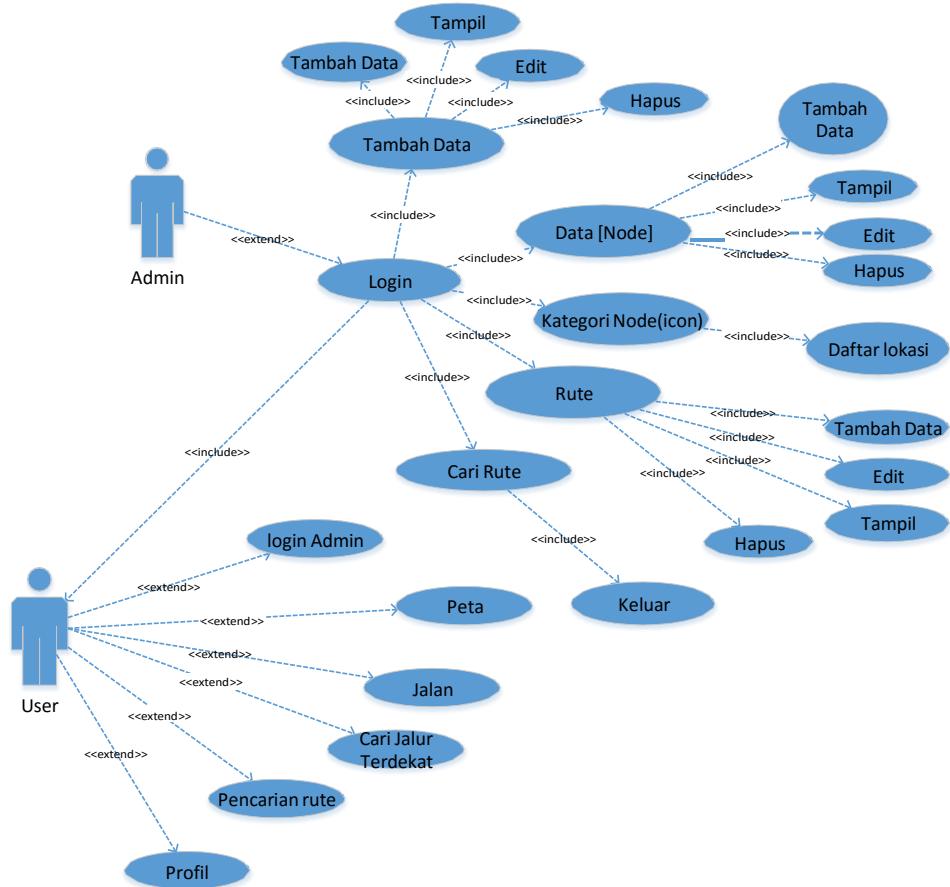
No	Node	Total Jarak
1	A – C – E – G	10
2	A – C – F – G	12
3	A – D – E – G	12
4	A – B – F – G	14
5	A – C – D – G	15
6	A – D – C – E – G	15
7	A – C – B – F – G	17
8	A – B – C – F – G	19

Berdasarkan node di atas jarak minimum dari node a menuju node g menggunakan algoritma dijkstra menghasilkan rute terpedek yaitu node A – C – E – G dengan total jarak sebanyak 10.

4.3 Hasil Pembangunan Sistem

Desain yang menjelaskan tentang diagram class untuk mempermudah pemahaman dalam menjalannya program.

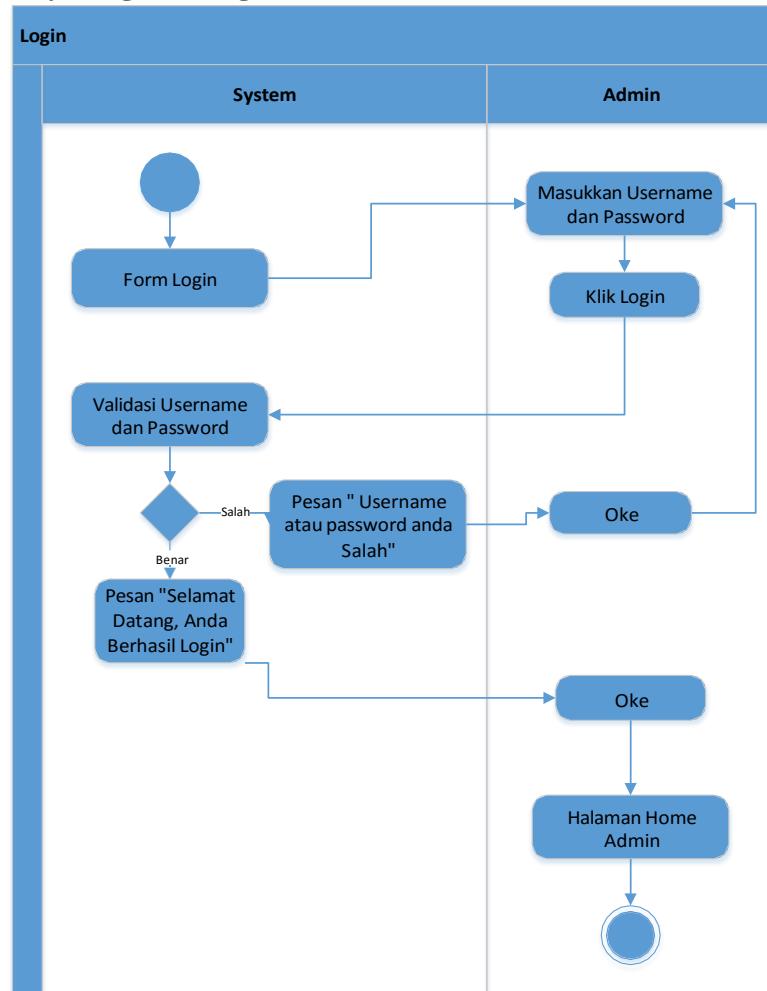
4.3.1 Use Case Diagram



Gambar 4.2 *Use Case Diagram*

Use Case Diagram menjelaskan tentang dirancang untuk menunjukkan secara umum fungsi dan tanggung jawab masing-masing aktor dalam Sistem, seperti menjelaskan fungsionalitas dari admin dan User dalam pencarian rute tercepat.

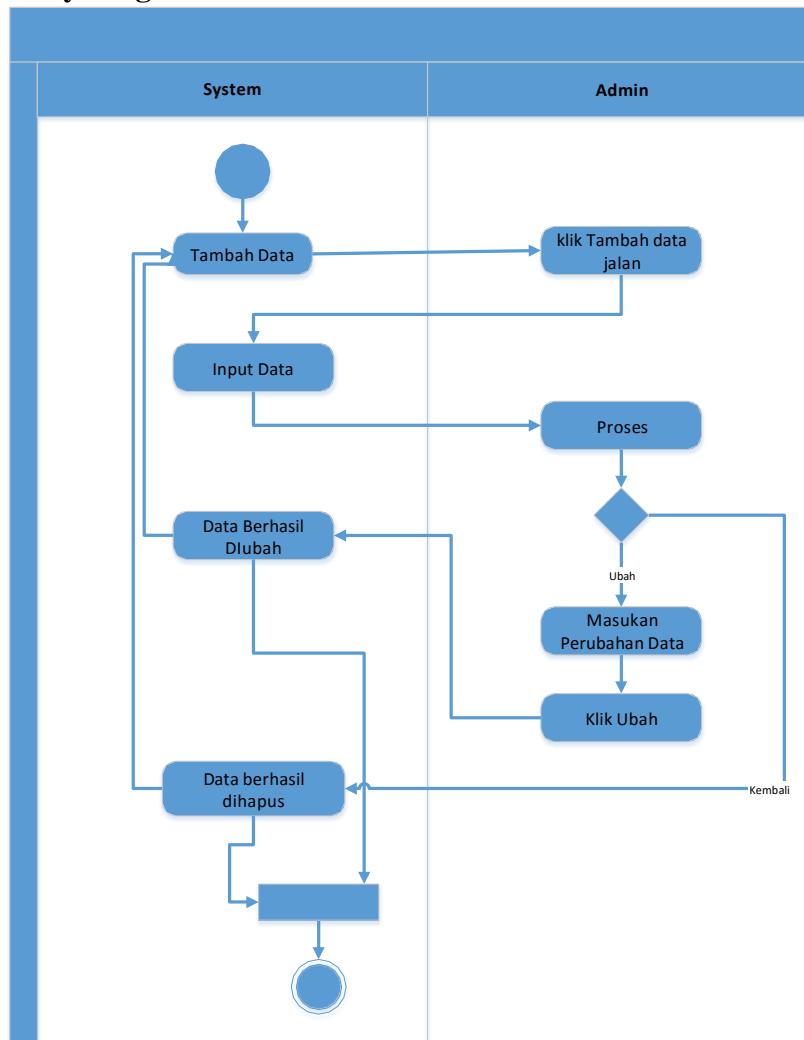
4.3.2 Activity Diagram Login



Gambar 4.3 Activity Diagram Login

Activity Diagram Login Pada halaman ini, sistem menampilkan halaman login. User lalu menekan tombol login, jika berhasil sistem menerima pesan Selamat Datang, Andah Berhasil Login, Jika tidak berhasil maka sistem akan menampilkan pesan Usernam atau password anda salah.

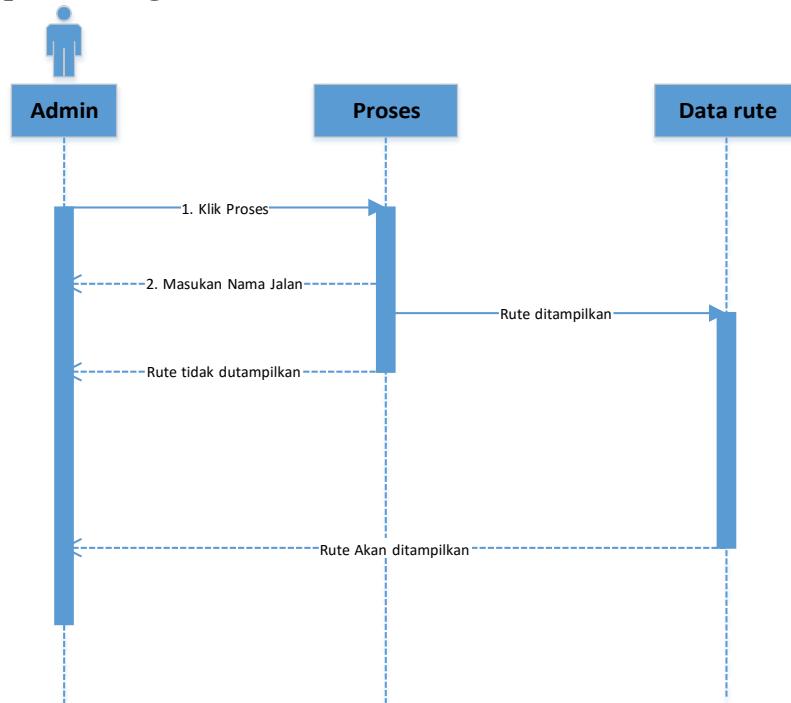
4.3.3 Activity Diagram Tambah Data



Gambar 4.4: Activity Diagram Tambah Data.

Activity Diagram Tambah Data, Untuk menginputkan atau memasukan data nama jalan, Seluruh informasi yang tersimpan di database.

4.3.4 Sequence Diagram Proses Cari Rute



Gambar 4.5 Sequence Diagram Cari Rute

Sequence Diagram Cari Rute menjelaskan bagaimana seorang aktor melakukan proses pencarian rute dalam sebuah database.

4.4 Arsitektur Sistem

Untuk kinerja sistem yang optimal, sebaiknya gunakan perangkat keras dan perangkat lunak sebagai berikut:

1. Processor : Minimal Core i3
2. RAM : 8GB
3. VGA : 16 Bit
4. Hardisk : 1 TB
5. Operating System : Windows 8 /10 /11
6. Tools : Mozilla, Chrome, Android Studio, Java

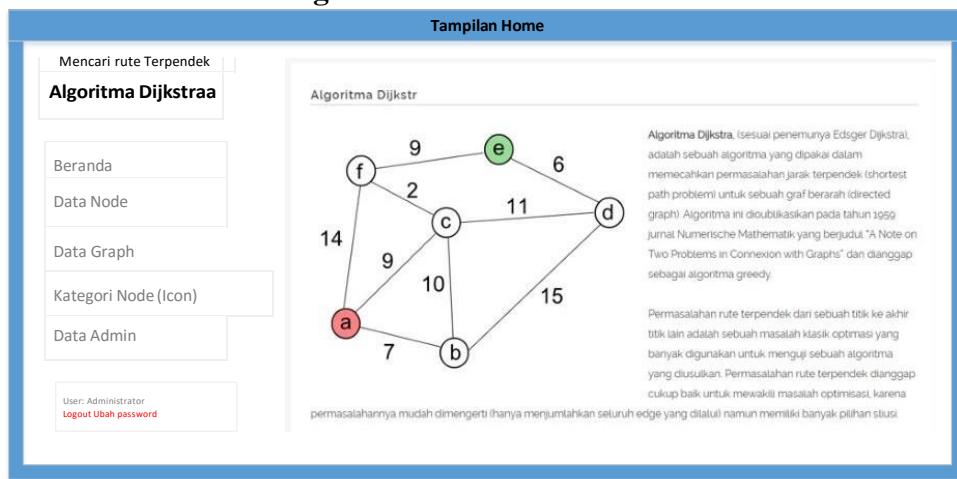
4.5 Interface Design

4.5.1 Mekanisme User

Tabel 4.3 Mekanisme User

User	Kategori	Akses Input	Akses Output
User admin	User administrator	- All	Rute All

4.5.2 Mekanisme navigasi home



Gambar 4.6: Mekanisme Navigasi Home User

4.5.3 Mekanisme Login

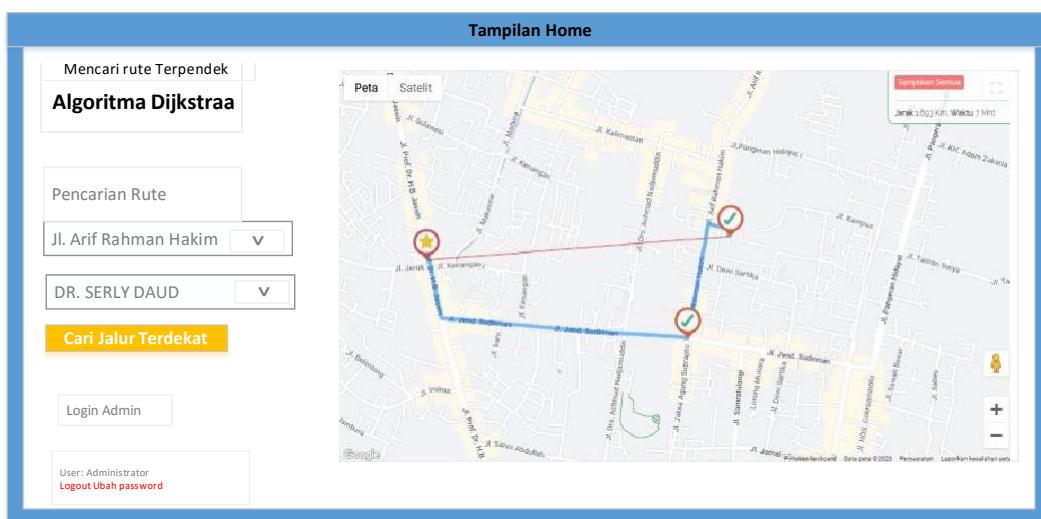
Gambar 4.7 Mekanisme Login

4.5.4 Mekanisme Input Data Rute Jalan



Gambar 4.8 Mekanisme *Input* data jalan

4.5.5 Mekanisme Output Website



Gambar 4.9 Mekanisme Output

4.6 Data Desain

4.6.1 Struktur Data

Tabel 4.4:Tabel Jalan

Nama File : tb_jalan				
Primary key : id_jalan				
Media : Hardisk				
fungsi : Menyimpan data login				
struktur data :				
No	Field Name	Type	Width	Keterangan
1.	id_jalan	Varchar	5	Nama admin
2.	nm_jalan	Varchar	255	Password admin

Tabel 4.5: Tabel Lokasi

Nama File : tb_lokasi				
Primary key : id_lokasi				
Media : Hardisk				
fungsi : Menyimpan data Lokasi				
struktur data :				
No	Field Name	Type	Width	Keterangan
1.	Id_lokasi	Varchar	5	Id lokasi
2.	Id_jalan	Varchar	5	id jalan
3.	Nm_lokasi	Varchar	255	Nama lokasi
4.	alamat	Varchar	255	Alamat
5.	No_telp	Varchar	15	Nomor kode jalan
6.	Latitude	Varchar	255	Garis Lintang
7.	Longitude	Varchar	255	Garis bujur
8.	gbr_lokasi	Varchar	255	Gambaran lokasi
9.	info	Varchar	255	Info

Tabel 4.6: Tabel Relasi

Nama File : tb_relati				
Primary key : id_lokasi				
Media : Hardisk				
fungsi : Menyimpan data Relasi				
struktur data :				
No	Field Name	Type	Width	Keterangan
1.	Id_lokasi	Varchar	5	Id lokasi
2.	Id_jalan	Varchar	5	Id jalan

Tabel 4.7: Tabel Rute

Nama File : tb_rute				
Primary key : id_rute				
Media : Hardisk				
fungsi : Menyimpan data Rute				
struktur data :				
No	Field Name	Type	Width	Keterangan
1.	Id_rute	Varchar	5	Id rute
2.	Id_lokasi	Varchar	5	Id lokasi
3.	Id_jalan1	Varchar	5	Id jalan 1
4.	Id_jalan2	Varchar	5	Id jalan 2
5.	jarak	Varchar	5	Jarak dilalui

Tabel 4.8: Tabel User

Nama File : tb_user				
Primary key : id_user				
Media : Hardisk				
fungsi : Menyimpan data User				
struktur data :				
No	Field Name	Type	Width	Keterangan
1.	Id_user	Int	5	Nomor id
2.	Username	Varchar	255	Nama user
3.	Password	Varchar	255	Password user
4.	User	Varchar	255	User
5.	Nm_lengkap	Varchar	255	Nama lengkap
6.	level	Varchar	255	level

4.7 Pengujian Sistem

4.7.1 Pengujian White Box

```

$dataNode=$this->getNodeData($lokasi_awal);..... 1
$graph = $dataNode[0];      $koordinat = $dataNode[1];..... 1
$distanse = $dataNode[3];..... 1
$distanse_from = [];..... 1
$distanse_temp = $distanse;..... 1
$vis_node=[];..... 1
$unv_node=$dataNode[2];..... 1
..... 1

$i=0;..... 1
while(!empty($unv_node)){..... 2
    if(empty($distanse_temp)){..... 3
        break;..... 4
    }
    $node=array_search(min($distanse_temp), $distanse_temp); ..... 5
    if((int)$node>0){..... 6

        $tetangga=$graph[$node];..... 7
        foreach($tetangga as $n => $dis){..... 8
            $distance=$distanse[$node]+$dis; .. 9
            if($distance<$distanse[$n]){..... 10
                $distanse[$n]=$distance;..... 11
                $distanse_temp[$n]=$distance;..... 11
                $distanse_from[$n]=$node;..... 11
            }
        }
        unset($distanse_temp[$node]);..... 12
        if (($key = array_search($node, $unv_node)) !== false) { .. 13
            unset($unv_node[$key]);..... 14
        }
        $vis_node[]=$node;..... 15

    }
    $i++;..... 16
}

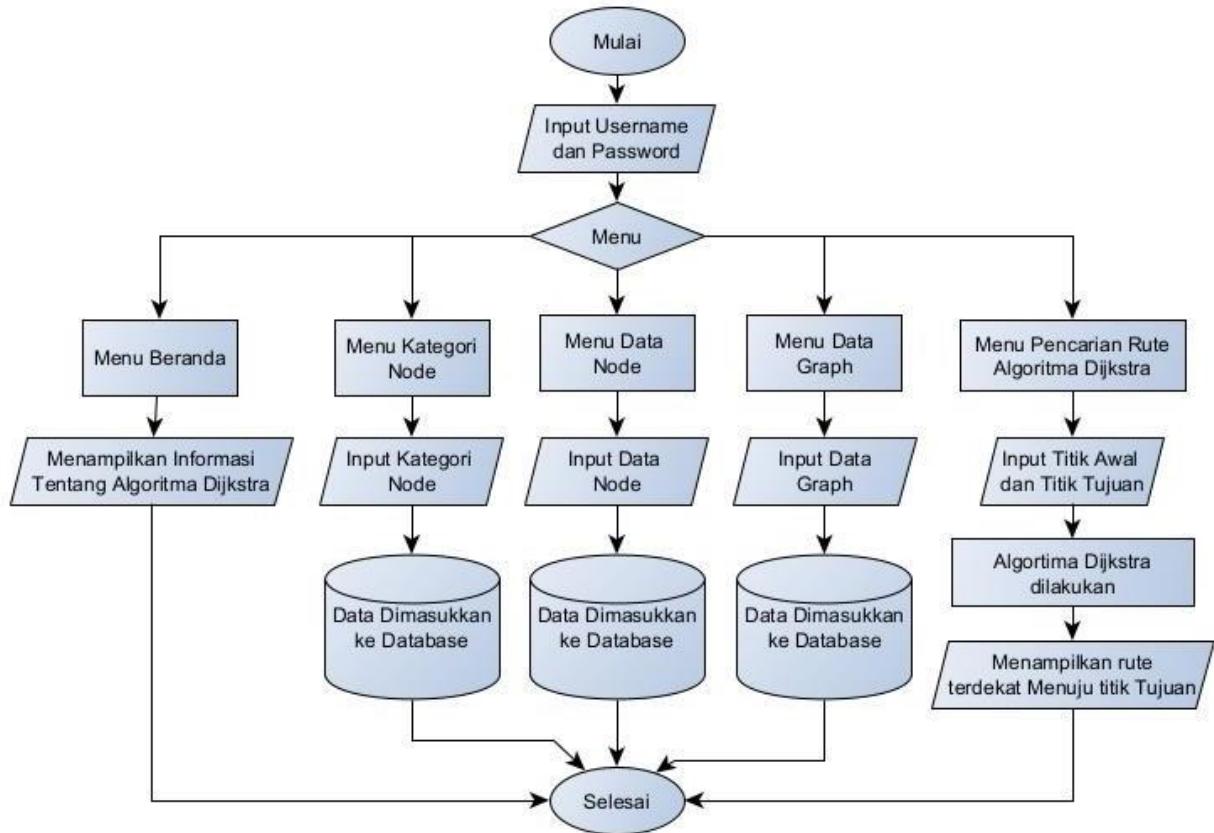
$jalur_terdekat=$this->cari_jalur($distanse, $distanse_from, $lokasi_awal,..... 17
$lokasi_tujuan);..... 17
$koordinat_jalur = []..... 17
foreach ($jalur_terdekat as $key => $node) {..... 18
    array_push($koordinat_jalur, @$koordinat[$node]);..... 19
}

```

```
}
```

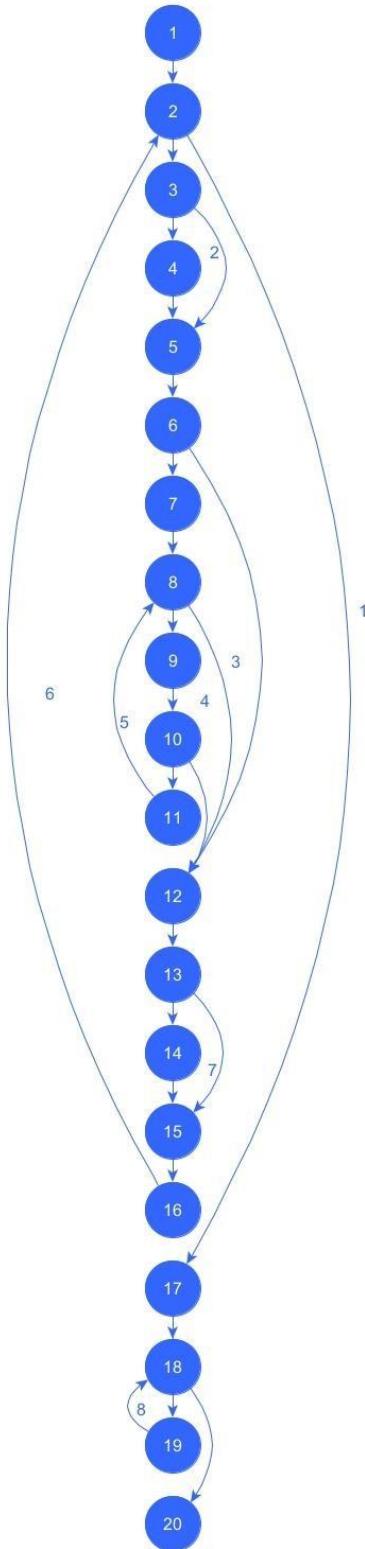
```
return [$koordinat_jalur, $jalur_terdekat];..... 20
```

4.7.2 Flowchart



Gambar 4.10: *Flowchart*

4.7.3 Flowgraph



Gambar 4.11: *Flowgraph*

4.7.4 Perhitungan CC Pada Pengujian White Box

Dari Flowgraph di atas didapatkan hasil perhitungan sebagai berikut:

Diketahui: Region (R) = 8

Predict Node (P) = 7

Node (N) = 20

Edge (E) = 26

Rumus $V(G) = E - N + 2$ dan $V(G) = P+1$

Penyelesaian: $V(G) = 26 - 20 + 2 = 8$

$V(G) = 7 + 1 = 8$

Tabel 4.9: Basis Path

No	Path	Ket
1	1-2-17-18-20	OK
2	1-2-3-4-5-6-7-8-9-10-11-8-12-13-14-15-16-2-17-18-19-18-20	OK
3	1-2-3-5-.....	OK
4	1-2-3-4-5-6-12-.....	OK
5	1-2-3-4-5-6-7-8-12-....	OK
6	1-2-3-4-5-6-7-8-9-10-12-....	OK
7	1-2-3-4-5-6-7-8-9-10-11-8-....	OK
8	1-2-3-4-5-6-7-8-9-10-11-12-13-15-....	OK

Ketika sistem dijalankan, maka terlihat bahwa semua basis path yang dihasilkan telah dieksekusi satu kali. Berdasarkan ketentuan tersebut dari segi kelayakan software, sistem ini telah memenuhi syarat.

4.7.5 Pengujian BlackBox

Pengujian blackbox merupakan pengujian yang dilakukan dengan mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak.

Tabel 4.9: Hasil Pengujian BlackBox

Nama Pengujian	Tujuan	Skenario	Hasil Yang Diharapkan	Ket
Melakukan Login Sistem	Mengetahui admin dapat login ke sistem	Memasukan username dan password	Admin dapat masuk ke sistem	OK
Membuka Tampilan Beranda	Untuk menampilkan halaman beranda	Menekan menu beranda	Sistem Dapat Menampilkan halaman beranda	OK
Membuka Kategori Node	Untuk menampilkan halaman kategori node dan menginput kategori node	Menekan menu kategori node dan menginput kategori node	Sistem dapat menampilkan halaman kategori node dan admin dapat menginput kategori node	OK
Membuka Data Node	Untuk menampilkan halaman data node dan menginput data node	Menekan menu data node dan menginput data node	Sistem dapat menampilkan halaman data node dan admin dapat menginput data node	OK
Membuka Data Graph	Untuk menampilkan halaman data graph dan menginput data graph	Menekan menu dara graph dan menginput data graph	Sistem dapat menampilkan halaman data graph dan admin dapat	OK

	menginput data graph		menginput data graph	
Menjalankan Pencarian Rute Algoritma Dijkstra	Untuk menampilkan hasil pencarian rute algortima Dijkstra	Menekan menu pencarian rute algortima Dijkstra dan memasukkan lokasi titik tujuan	Sistem dapat menampilkan hasil pencarian rute Dijkstra	OK

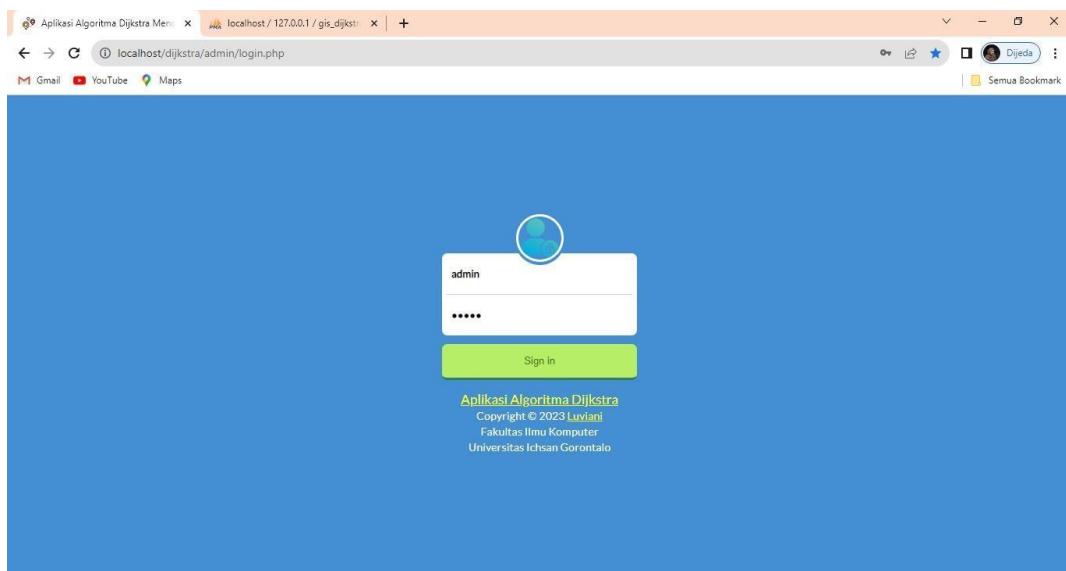
Berdasarkan hasil pengujian yang dilakukan pada sistem, terlihat bahwa semua pengujian black box yang diperoleh sudah dites satu kali. Berdasarkan ketentuan tersebut dari segi kelayakan aplikasi, maka aplikasi ini sudah memenuhi syarat.

BAB V

HASIL DAN PEMBAHASAN

5.1 Pembahasan Sistem

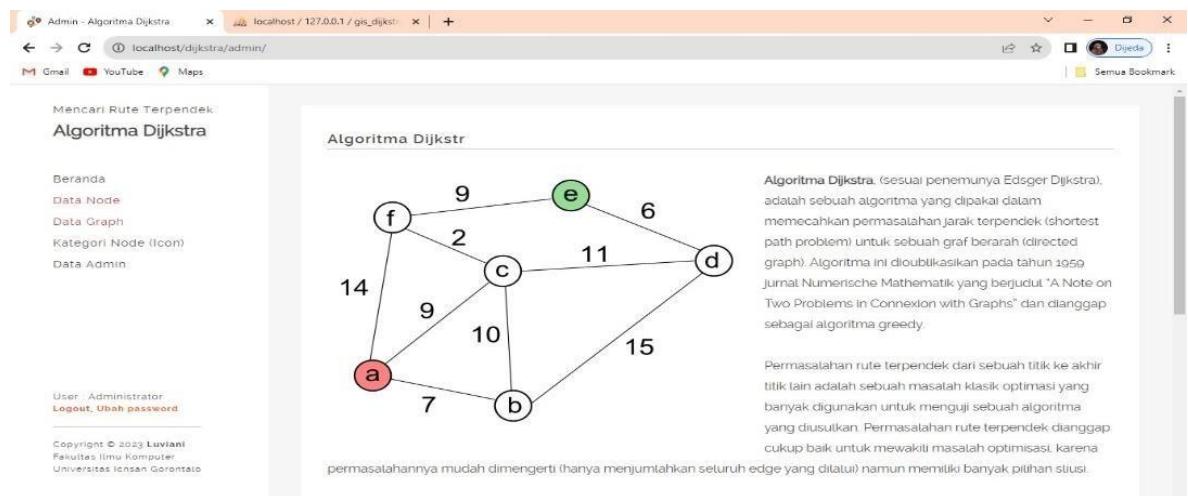
5.1.1 Tampilan Login



Gambar 5.1 Tampilan Login

Tampilan login berfungsi sebagai pengaman sistem sehingga hanya admin yang bisa masuk dan sistem menjadi aman serta bisa menambah Data, Untuk menginputkan atau memasukan data nama jalan, Seluruh informasi yang tersimpan di database..

5.1.2 Tampilan Beranda



Gambar 5.2 Tampilan Beranda

Tampilan beranda menampilkan informasi tentang algoritma Dijkstra mulai dari graph node dan pengertian dari metode algoritma *Dijkstra* dengan jelas.

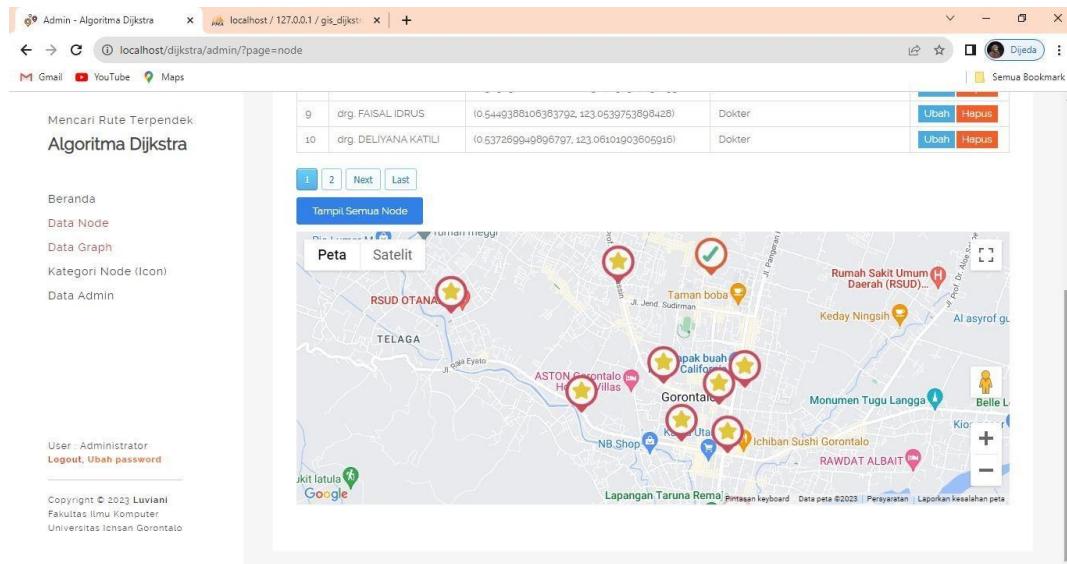
5.1.3 Tampilan Kategori Node

No.	Kategori Node	Icon	Tambah
1	Masjid		Ubah Hapus
2	Dokter		Ubah Hapus
3	Persimpangan		Ubah Hapus

Gambar 5.3 Tampilan Kategori Node

Tampilan kategori node menampilkan berbagai icon dan kategori yang terdapat dalam sistem. Untuk menginputkan Kategori node atau memasukan data nama-nama data yang terkait Seluruh informasi yang tersimpan di database.

5.1.4 Tampilan Data Node



Gambar 5.4 Tampilan Data Node

Tampilan data node menampilkan data titik lokasi yang diinput sehingga lokasinya bisa diketahui di *maps*. Untuk melihat detail data Lokasi klik data tribut klik Aksi Tampil.

5.1.5 Tampilan Data Graph

The screenshot shows a web browser window with two tabs: 'Admin - Algoritma Dijkstra' and 'localhost / 127.0.0.1 / gis_dijkstra'. The main content area is titled 'Data Graph' and displays a table of data. Below the table is a map with several location markers.

Data Graph

No.	Node 1	Node 2	Jarak	Arah	Tambah
1	DR. SERLY DAUD	DR. SRI A. IBRAHIM M.KES	2.876	Dua Arah	Ubah Hapus
2	Jln Arif Rahman Hakim	DR. SRI A. IBRAHIM M.KES	4.515	Dua Arah	Ubah Hapus
3	Jln Arif Rahman Hakim	DR. SERLY DAUD	1.693	Dua Arah	Ubah Hapus
4	Jln Arif Rahman Hakim	DR. THAIB SALEH	3.026	Satu Arah	Ubah Hapus

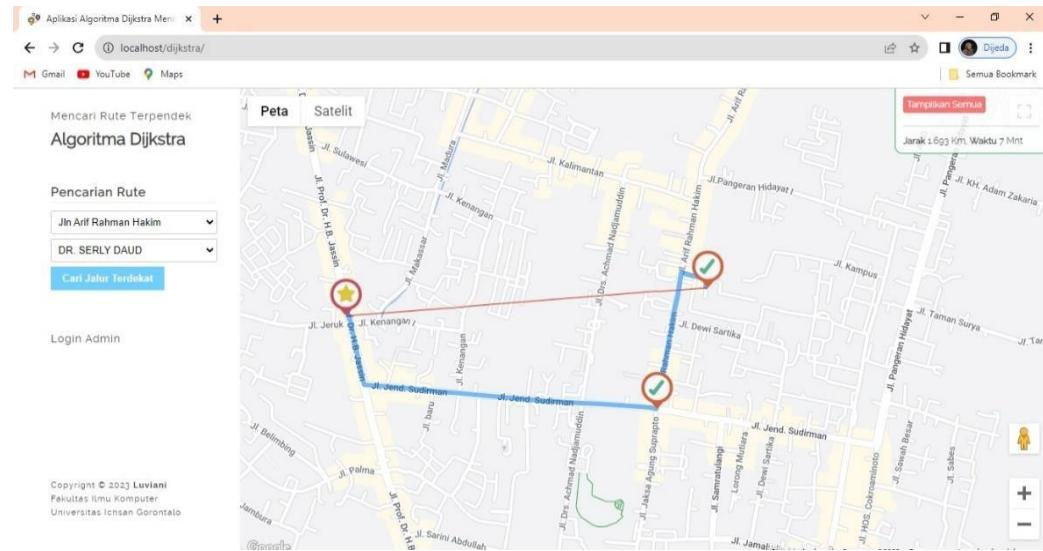
Peta **Satelite**

The map shows various locations in Gorontalo, including RSUD TOTO KABILA, TIARA Water Park, Kejaksaan Tinggi Gorontalo, RSUD OTANAH, and Benteng Otanaha. A red line connects several points on the map, indicating a route or path.

Gambar 5.5 Tampilan Data Graph

Tampilan data graph menampilkan data rute dan jarak yang didapatkan dari titik node yang ditentukan. Halaman ini digunakan untuk menampilkan hasil pencarian Jarak terdekat tempat Dokter Praktek BPJS Menggunakan Metode *Dijkstrak*. Untuk menampilkan hasil Cari tampilan Semua node klik tombol Lakukan Proses.

5.1.6 Tampilan Pencarian Rute Algoritma Dijkstra



Gambar 5.6 Tampilan Pencarian Rute Algoritma Dijkstra
 Berdasarkan node diatas jarak minimum antara node Tampilan pencarian rute terdekat menggunakan algoritma *Dijkstra* yang berdasarkan data. Titik tujuan di Jalan Arif Rahman Hakim dari arah Jl. Jeruk – Jl. H.B Jasiin- Jl. Jend sudirman Dengan jarak 1.693 km. Waktu 7 menit.

BAB VI

PENUTUP

6.1 Kesimpulan

Berdasarkan pembahasan, implementasi dan hasil pengujian yang telah dilakukan pada bab-bab sebelumnya maka dapat di ambil kesimpulan bahwa tujuan penelitian ini Untuk mengetahui Efektifitas Metode *Algoritma Dijkstra*. Dalam Pencarian rute Terdekat Tempat Dokter Praktek Fasilitas BPJS Menggunakan Metode *Dijkstra* di Kabupaten Kota Gorontalo telah tercapai yaitu:

1. Bahwa dapat diketahui cara merekayasa aplikasi pencarian rute tercepat menggunakan *Algoritma Dijkstra*.
2. Aplikasi pencarian rute tercepat menggunakan Metode *Algoritma Dijkstra* dapat digunakan untuk Pencarian rute Terdekat Tempat Dokter Praktek Fasilitas BPJS di Kabupaten Kota Gorontalo.

6.2 Saran

Adapun saran yang dapat diberikan oleh penulis berdasarkan pengamatan yang telah dilakukan antara lain:

1. Memperbaiki dan memperkaya daftar lokasi dan jalan yang digunakan oleh sistem, juga menyertakan kondisi jalan seperti ada tidaknya lampu lalu lintas, tingkat kemacetan, dan lain-lain. Sehingga proses pencarian rute dapat menghasilkan hasil yang lebih akurat dan nyata.
2. Tambahkan pencarian titik lokasi maps otomatis pada pencarian lokasi awal sehingga mempermudah pengguna dalam mencari titik lokasi maps
3. Tambahkan filter pencarian berdasarkan spesialis dokter pada sistem sehingga pengguna dapat mencari dokter sesuai kebutuhan pengguna

DAFTAR PUSTAKA

- [1] L. Licantik and Nova Noor Kamala Sari, “Sistem Informasi Geografis Fasilitas Kesehatan Bpjs Di Kota Palangka Raya Berbasis Android,” *J. Teknol. Inf. J. Keilmuan dan Apl. Bid. Tek. Inform.*, vol. 14, no. 1, pp. 30–39, 2020, doi: 10.47111/jti.v14i1.402.
- [2] F. L. B. Putri, X. B. N. Najoan, and Y. D. . Rindengan, “Sistem Informasi Pemetaan Fasilitas Kesehatan BPJS Berbasis Android di Kota Bitung,” *J. Tek. Inform.*, vol. 11, no. 1, 2017, doi: 10.35793/jti.11.1.2017.16923.
- [3] Fabiana Meijon Fadul, “Kualitas pelayanan pasien rawat jalan di rumah sakit umu daerah trikora salakan kecamatan tinangkung kabupaten banggai kepulauan provinsi sulawesi tengah,” pp. 74–83, 2019.
- [4] F. Mahdia and F. Noviyanto, “211271-Pemanfaatan-Google-Maps-API-Untuk-Pembangunan,” vol. 1, pp. 162–171, 2013.
- [5] M. C. Bunaen, H. Pratiwi, and Y. F. Riti, “Penerapan algoritma dijkstra untuk menentukan rute terpendek dari pusat kota surabaya ke tempat bersejarah,” vol. 4, no. 1, pp. 213–223, 2022.
- [6] A. Wijaya and E. Kurniawan, “Implementasi Algorithma Dijkstra Dalam Pencarian Rute Terpendek Fasilitas Kesehatan Tingkat I (Studi Kasus BPJS Kesehatan Kota Bengkulu),” *J. Media Infotama*, vol. 15, no. 2, 2019, doi: 10.37676/jmi.v15i2.871.
- [7] S. H. Wardani, T. Rismawan, and S. Bahri, “Aplikasi Klasifikasi Jenis Tumbuhan Mangrove Berdasarkan Karakteristik Morfologi Menggunakan Metode K-Nearest Neighbor (KNN) Berbasis Web,” *J. Coding, Sist. Komput. Untan*, vol. 4, no. 3, pp. 9–21, 2016.
- [8] Ika Widiastuti, “PELAYANAN BADAN PENYELENGGARA JAMINAN SOSIAL (BPJS) KESEHATAN DI JAWA BARAT,” *J. Sains dan Inform.*, vol. 4, no. 1, pp. 88–100, 2557.
- [9] E. Maiyana, “Pemanfaatan Android Dalam Perancangan Aplikasi Kumpulan Doa,” *J. Sains dan Inform.*, vol. 4, no. 1, pp. 54–65, 2018, doi: 10.22216/jsi.v4i1.3409.
- [10] J. Kuswanto and F. Radiansah, “Media Pembelajaran Berbasis Android Pada Mata Pelajaran Sistem Operasi Jaringan Kelas XI,” *J. Media Infotama*, vol. 14, no. 1, 2018, doi: 10.37676/jmi.v14i1.467.
- [11] K. Rizki and A. Adil, “Implementasi Google Maps API Berbasis Android untuk Lokasi Fasilitas Umum di Kabupaten Sumbawa,” *J. MATRIK*, vol. 17, no. 2, pp. 34–44, 2018, doi: 10.30812/matrik.v17i2.87.
- [12] T. Informatika, S. Dipanegara, K. Makassar, K. Makassar, G. M. Api, and K. Kunci, “DoubleClick : Journal of Computer and Information Technology Aplikasi Pencarian Bengkel Aktif (Aisa) | 61 DoubleClick : Journal of Computer and Information Technology 62 | Aplikasi Pencarian Bengkel Aktif (Aisa),” vol. 4, no. 2, pp. 61–69, 2021.
- [13] B. Junanda, D. Kurniadi, and Y. Huda, “Pencarian Rute Terpendek Menggunakan Algoritma Dijkstra Pada Sistem Informasi Geografis

- Pemetaan Stasiun Pengisian Bahan Bakar Umum,” *Voteteknika (Vocational Tek. Elektron. dan Inform.)*, vol. 4, no. 1, 2018, doi: 10.24036/voteteknika.v4i1.6014.
- [14] L. Anindito *et al.*, “Implementasi Algoritma Dijkstra Dalam Menentukan Rute,” *ProSANDIKA UNIKAL*, 2022.
 - [15] R. F. EKASARI, “Penerapan Algoritma Dijkstra Untuk Menemukan Lintasan Terpendek Pada Pengiriman Barang Pt Kharisma Suma Jaya Sakti,” pp. 1–12, 2017.
 - [16] P. A. Kurniawijaya, “Pencarian Pom Bensin Terdekat di Denpasar Menggunakan Algoritma Dijkstra Berbasis Web Mobile,” *J. Teknol. Inf. dan Komput.*, vol. 1, no. 1, pp. 7–19, 2016, doi: 10.36002/jutik.v1i1.18.
 - [17] E. A. W. Sanad, “Pemanfaatan Realtime Database di Platform Firebase Pada Aplikasi E-Tourism Kabupaten Nabire,” *J. Penelit. Enj.*, vol. 22, no. 1, pp. 20–26, 2019, doi: 10.25042/jpe.052018.04.
 - [18] M. Mujahidin, F. Adiandoro, E. R. Swedia, and M. Cahyanti, “Pemanfaatan Internet of Things Dalam Rancang Bangun Sistem Informasi Pengawasan Bus Pada Terminal Bus Berbasis Arduino Uno Dan Node Mcu,” *Sebatik*, vol. 24, no. 2, pp. 228–233, 2020, doi: 10.46984/sebatik.v24i2.1061.
 - [19] N. A. T. W. Farabi, Nur Ali , Andi Rosano, “Rancang Bangun Sistem Informasi Penggajian Dengan Desain Sistem Berorientasi Objek (Study Kasus : CV. Angkutan Agung),” *J. AKRAB JUARA*, vol. 3, no. 4, pp. 117–128, 2018.
 - [20] S. Mujilahwati and S. N. Fauziah, “Pemodelan Ooad Aplikasi Prediksi Harga Sembako Berbasis Android,” *Antivirus J. Ilm. Tek. Inform.*, vol. 12, no. 1, pp. 1–7, 2018, doi: 10.35457/antivirus.v12i1.430.
 - [21] B. Fitriani, T. Angraini, and Y. H. G. Putra, “Pemodelan Use Case Diagram Sistem Informasi Inventaris Laboratorium Teknik Mesin,” *Semin. Nas. Sist. Inf. dan Teknol. Inf. 2018*, pp. 626–631, 2018.
 - [22] ع. م. . و. ع. کوچکی *et al.*, “APLIKASI MOBILE PENCARIAN RUTE TERPENDEK PADA PENGIRIMAN ORDER CV. ALFA FRESH DENGAN ALGORITMA BRUTE FORCE,” *Bitkom Res.*, vol. 63, no. 2, pp. 1–3, 2018, [Online]. Available: http://forschungsunion.de/pdf/industrie_4_0_umsetzungsempfehlungen.pdf %0Ahttps://www.dfg.de/fileadmin/user_upload/import/9744_171012-KI-Gipfelpapier-online.pdf %0A<https://www.bitkom.org/sites/default/files/pdf/Presse/Anhaenge-an-PIs/2018/180607 -Bitkom>.
 - [23] M. Tinambunan and S. Sintaro, “Aplikasi Restfull Pada Sistem Informasi Geografis Pariwisata Kota Bandar Lampung,” *J. Inform. dan Rekayasa Perangkat Lunak*, vol. 2, no. 3, pp. 312–323, 2021, doi: 10.33365/jatika.v2i3.1230.
 - [24] C. T. Pratala, E. M. Asyer, I. Prayudi, and A. Saifudin, “Pengujian White Box pada Aplikasi Cash Flow Berbasis Android Menggunakan Teknik Basis Path,” *J. Inform. Univ. Pamulang*, vol. 5, no. 2, p. 111, 2020, doi: 10.32493/informatika.v5i2.4713.
 - [25] W. N. Cholifah, Y. Yulianingsih, and S. M. Sagita, “Pengujian Black Box Testing pada Aplikasi Action & Strategy Berbasis Android dengan

- Teknologi Phonegap,” *STRING (Satuan Tulisan Ris. dan Inov. Teknol.*, vol. 3, no. 2, p. 206, 2018, doi: 10.30998/string.v3i2.3048.
- [26] E. L. Hady, K. Haryono, and N. W. Rahayu, “User Acceptance Testing (UAT) pada Purwarupa Sistem Tabungan Santri (Studi Kasus : Pondok Pesantren Al-Mawaddah) User Acceptance Testing (UAT) of the Prototype of Students ’ Savings Information System (Case Study : Al-Mawaddah Islamic Boarding Scho,” *J. Ilm. Multimed. dan Komun.*, vol. 5, no. 1, pp. 1–10, 2020.
 - [27] S. Fadli and K. Imtihan, “Analisis Dan Perancangan Sistem Administrasi Dan Transaksi Berbasis Client Server,” *J. Inform. dan Rekayasa Elektron.*, vol. 1, no. 2, p. 7, 2018, doi: 10.36595/jire.v1i2.54.
 - [28] R. A. Krisdiawan, “Implementasi Model Pengembangan Sistem Gdlc Dan Algoritma Linear Congruential Generator Pada Game Puzzle,” *Nuansa Inform.*, vol. 12, no. 2, pp. 1–9, 2018, [Online]. Available: <https://journal.uniku.ac.id/index.php/ilkom/article/view/1634/1211>.

LAMPIRAN

Lampiran 1: Code Program Pencarian rute Terdekat Tempat Dokter Praktek Fasilitas BPJS di Kabupaten Kota Gorontalo.

*Code Program Algoritma Dijkstra

```
<?php
include_once 'crud.php';
include_once 'crudNode.php';
include_once 'crudGraph.php';

class mDijkstra extends crud
{
    private $_node;
    private $_graph;

    public function __construct()
    {
        parent::__construct();
        $this->_node=new crudNode();
        $this->_graph=new crudGraph();
    }

    private function getNodeData($lokasi_awal){
        $koordinat=[];
        $X=[];
        $nodes=[];
        $initial_distance=[];
        $hquery = $this->_node->getNode();
        foreach ($hquery as $k => $data) {
            $node_a = $data['id_node'];
            $nodes[]=$node_a;
            $initial_distance[$node_a] = ($node_a==$lokasi_awal) ? 0 : INF;
            $koordinat[$data['id_node']] = $data['koordinat'];

            $h_graph1 = $this->_graph->getGraphByNode_1($node_a);
            foreach ($h_graph1 as $k1 => $d_graph1) {
                $X[$data['id_node']][$d_graph1['id_node']] = $d_graph1['jarak'];
            }
        }

        $h_graph2 = $this->_graph->getGraphByNode_2($node_a);
        foreach ($h_graph2 as $k1 => $d_graph2) {
            $X[$data['id_node']][$d_graph2['id_node']] = $d_graph2['jarak'];
        }
    }
}
```

```

        }

    }

    return [$X, $koordinat, $nodes, $initial_distance];

}

public function dijkstra($lokasi_awal, $lokasi_tujuan){
    $dataNode=$this->getNodeData($lokasi_awal); //Ambil data Graph Node
    $graph = $dataNode[0];
    $koordinat = $dataNode[1];
    $distances = $dataNode[3];
    $distances_from = [];
    $distances_temp = $distances;
    //echo "<pre>"; print_r($graph); echo "</pre>";
    //echo "<pre>"; print_r($distances_temp); echo "</pre>";
    $vis_node=[];
    $unv_node=$dataNode[2];

    $i=0;
    while(!empty($unv_node)){
        if(empty($distances_temp)){
            break;
        }
        $node=array_search(min($distances_temp), $distances_temp); //Current Node
        if((int)$node>0){
            $tetangga=$graph[$node];
            foreach($tetangga as $n => $dis){
                $distance=$distances[$node]+$dis; //Hitung jarak dari lokasi awal
                if($distance<$distances[$n]){ //Update jika jarak lebih kecil dari jarak
                    sebelumnya
                    $distances[$n]=$distance;
                    $distances_temp[$n]=$distance;
                    $distances_from[$n]=$node;
                }
            }
            unset($distances_temp[$node]);
            if (($key = array_search($node, $unv_node)) !== false) { //Hapus node dari
unvisited node
                unset($unv_node[$key]);
            }
            $vis_node[]=$node;
        }
    }
}

```

```

        }
        $i++;
    }

    //print_r($distances);echo "<br>";
    //print_r($distances_from);echo "<br>";

    $jalur_terdekat=$this->cari_jalur($distances, $distances_from, $lokasi_awal,
$lokasi_tujuan);
    $koordinat_jalur = [];
    foreach ($jalur_terdekat as $key => $node) {
        array_push($koordinat_jalur, @$koordinat[$node]);
    }

    return [$koordinat_jalur, $jalur_terdekat];
}

private function cari_jalur($distances, $distances_from, $lokasi_awal, $lokasi_tujuan){
    $node=$lokasi_tujuan;
    $rute=[];
    $rute[]=$node;
    while($node!=$lokasi_awal){
        $node=$distances_from[$node];
        $rute[]=$node;
    }

    return $rute;
}

public function escapeString($value)
{
    return $this->escape_string($value);
}
}

```

PAPER NAME

SKRIPSI_T3118037_LUVIANI.docx

AUTHOR

LUVIANI luvianii69@gmail.com

WORD COUNT

6774 Words

CHARACTER COUNT

40920 Characters

PAGE COUNT

52 Pages

FILE SIZE

3.3MB

SUBMISSION DATE

Dec 13, 2023 7:26 AM GMT+8

REPORT DATE

Dec 13, 2023 7:27 AM GMT+8

● 20% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 20% Internet database
- Crossref database
- 1% Submitted Works database
- 4% Publications database
- Crossref Posted Content database

● Excluded from Similarity Report

- Bibliographic material
- Cited material
- Quoted material
- Small Matches (Less than 30 words)

● 20% Overall Similarity

Top sources found in the following databases:

- 20% Internet database
- Crossref database
- 1% Submitted Works database
- 4% Publications database
- Crossref Posted Content database

TOP SOURCES

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	jsiskom.undip.ac.id	4%
	Internet	
2	core.ac.uk	2%
	Internet	
3	repository.widyatama.ac.id	2%
	Internet	
4	jurnal.unmer.ac.id	2%
	Internet	
5	andi.ddns.net	1%
	Internet	
6	repository.upbatam.ac.id	1%
	Internet	
7	journal.unnes.ac.id	1%
	Internet	
8	ejurnal.unisan.ac.id	<1%
	Internet	

- | | | |
|----|--|--|
| 9 | eprints.polsri.ac.id | <1% |
| | Internet | |
| 10 | media.neliti.com | <1% |
| | Internet | |
| 11 | openlibrarypublications.telkomuniversity.ac.id | <1% |
| | Internet | |
| 12 | id.scribd.com | <1% |
| | Internet | |
| 13 | kc.umn.ac.id | <1% |
| | Internet | |
| 14 | katalog.ukdw.ac.id | <1% |
| | Internet | |
| 15 | scribd.com | <1% |
| | Internet | |
| 16 | sinta.unud.ac.id | <1% |
| | Internet | |



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
UNIVERSITAS ICHSAN GORONTALO
FAKULTAS ILMU KOMPUTER
UPT. PERPUSTAKAAN FAKULTAS
SK. MENDIKNAS RI NO. 84/D/0/2001
Jl. Achmad Nadjamuddin No.17 Telp(0435) 829975 Fax. (0435) 829976 Gorontalo

SURAT KETERANGAN BEBAS PUSTAKA

No : 029/Perpustakaan-Fikom/XI/2023

Perpustakaan Fakultas Ilmu Komputer (FIKOM) Universitas Ichsan Gorontalo dengan ini menerangkan bahwa :

Nama Anggota : Luviani
No. Induk : T3118037
No. Anggota : M202391

Terhitung mulai hari, tanggal : Kamis, 30 November 2023, dinyatakan telah bebas pinjam buku dan koleksi perpustakaan lainnya.

Demikian keterangan ini di buat untuk di pergunakan sebagaimana mestinya.

Gorontalo, 30 November 2023

Mengetahui,
Kepala Perpustakaan



Apriyanto Alhamad, M.Kom

NIDN : 0924048601

SURAT KETERANGAN
Nomor : 3250/X-02/1223

Yang bertanda tangan dibawah ini :

Nama : Ivana F. Umboh
Jabatan : Kepala Bagian SDMUK

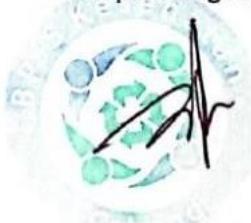
Selaku perwakilan dari BPJS Kesehatan Cabang Gorontalo, menerangkan bahwa atas nama sebagai berikut :

Nama : Luviani
NIM : Teknik Informatika
Fakultas : Universitas Ichsan Gorontalo

Telah melakukan penilitian di BPJS Kesehatan Cabang Gorontalo pada periode bulan Maret tahun 2023.

Demikian disampaikan, atas perhatian dan kerja samanya diucapkan terima kasih.

Gorontalo, 22 Desember 2023
Kepala Bagian SDMUK



Ivana F. Umboh

Kantor Cabang Gorontalo

Jl. Sultan Botutihe No. 58
Kel. Ipilo, Kec. Kota Timur, Kota Gorontalo, 96112
Telp. (0435) 823000, Fax (0435)828905
www.bpjs-kesehatan.go.id

Lampiran 5: Riwayat Hidup

Nama : Luviani
Tempat, Tanggal Lahir : kalupapi 06 september 2000
Alamat : Desa Kalupapi, kec bangkurung,
Kab banggai laut
Agama : Islam
Kewarganegaraaan : WNI
Email : luvianii69@gmail.com

Riwayat pendidikan:

Jenjang Pendidikan	Nama Sekolah	Tahun Masuk	Tahun Lulus
SD	SDN INPRES KALUPAPI	2006	2013
SMP	MTS KSM KALIPAPI	2013	2015
SMA	SMK N 1 BANGGAI	2015	2018