

# SISTEM INFORMASI LETAK GEOGRAFIS PENENTUAN JALUR TERCEPAT RUMAH SAKIT DI KOTA PALU MENGGUNAKAN ALGORITMA GREEDY BERBASIS WEB

Diana Grace<sup>1</sup>  
Mu'amar S. Tanciga<sup>2</sup>  
Nurdin Nurdin<sup>3</sup>

<sup>1,2,3</sup> Jurusan Teknik Informatika STMIK Bina Mulia Palu

## Abstrak

Permasalahan yang terjadi dikota Palu yaitu masyarakat belum mengetahui Jalur menuju rumah sakit dengan jalan tercepat. Permasalahan yang lain muncul adalah ketika pasien mengalami berbagai macam hambatan dalam menuju suatu lokasi tujuan yang mana ditempuh dengan waktu yang lama karena kurangnya pengetahuan mereka dalam menentukan jalur tercepat sehingga waktu yang di tempuh dengan singkat menjadi lebih lama. Tujuan penelitian ini adalah untuk membuat sistem informasi penentuan jalur tercepat rumah sakit di kota palu agar masyarakat tau bahwa rumah sakit apa saja yang ada di kota palu serta memberi tahu lokasi dan jarak yang ditempuh untuk sampai di rumah sakit tersebut. Penelitian ini menggunakan jenis penelitian kualitatif. Jenis dan sumber data yang dibutuhkan dalam penelitian ini adalah data primer dan data sekunder. Teknik pengumpulan data meliputi: observasi, wawancara, dan studi pustaka. Teknik pengembangan system menggunakan metode sekuensial linier (*waterfall*). Bahasa pemrograman yang digunakan PHP, *MySQL*. waktu yang dibutuhkan dalam pembuatan Sistem Informasi ini kurang lebih selama satu bulan. Dari penjelasan yang telah di sampaikan pada proses awal perancangan sampai dengan geografis pencarian rute menggunakan algoritma greedy yang dibuat berjalan sesuai dengan rancangan awal serta kesesuaian sistem informasi yang dibuat sehingga secara langsung dapat mempermudah dan mempercepat dalam mendapatkan rute menuju rumah sakit yang dipilih sehingga dapat tercipta efektifitas serta efisiensi kerja. Sistem informasi letak geografis penentuan jalur tercepat rumah sakit di kota palu menggunakan *algoritma greedy* berbasis web.

**Kata kunci** : *sistem informasi, Rumah sakit, algoritma greedy*

## 1. Pendahuluan

Pada dasarnya manusia membutuhkan waktu untuk mencapai suatu tujuan. Semakin cepat waktu yang ditempuh maka semakin pendek pula jalur yang di tempuh. Hal ini menunjukkan bahwa tingkat efisiensi waktu digunakan pada pola hidup manusia. Apalagi dengan tingkat perkembangan jaman yang begitu cepat, membuat manusia mengalami berbagai macam hambatan dalam menuju suatu lokasi tujuan. Perubahan

Teknologi membuat manusia dapat mengembangkan pola pikir mereka menjadi lebih baik dari sebelumnya. Suatu lokasi yang dulunya ditempuh dengan waktu yang lama kini mereka sudah bisa ditempuh dengan waktu yang singkat. Ini disebabkan oleh pola pikir mereka yang semakin berkembang pula. Pemanfaatan berbagai jalur untuk mencapai tujuan merupakan salah satu perkembangan pola pikir manusia.

Jalur menuju rumah sakit dengan jalan tercepat merupakan suatu bentuk kebutuhan manusia. Ini dikarenakan kondisi pasien untuk mendapat pertolongan juga harus cepat dan tepat, sehingga tidak menjadikan hal-hal yang tidak diinginkan terjadi. Letak dari rumah sakit ditiap kota berbeda-beda sehingga jika ingin menuju ke rumah sakit ada banyak pilihan yang harus dipilih, sesuai dengan kebutuhan dari pasien. Jarak dari pusat kota menuju rumah sakit memiliki perbedaan waktu tempuh. Terkadang ada orang yang mengabaikan waktu tempuh untuk menuju rumah sakit dengan jarak terdekat. Oleh karena itu penerapan algoritma sangat diperlukan dalam menyelesaikan kasus ini.

Algoritma adalah sebuah prosedur komputasi yang mentransformasikan sejumlah input menjadi sejumlah output. Sebuah algoritma dikatakan “benar (correct)” jika untuk setiap inputnya menghasilkan output yang benar pula (Prayitno, Arif, & Nurdin, Nurdin, 2017). Dalam hal ini algoritma dapat digunakan sebagai metode untuk mengetahui langkah langkah secara urut untuk mencapai tujuan. Setiap algoritma memiliki perbedaan dalam mencapai tujuan. Ada beberapa algoritma yang dapat digunakan untuk menentukan jalur tercepat. Salah satunya menggunakan algoritma Greedy.

Sistem Informasi Geografis (SIG) pada dasarnya merupakan gabungan tiga unsur pokok yaitu sistem, informasi dan geografis. Dengan melihat unsur-unsur pokoknya, maka jelas sistem informasi geografis merupakan salah satu sistem informasi dengan tambahan unsur “geografis”. Sistem Informasi Geografis diartikan sebagai sistem informasi yang digunakan untuk memasukkan, menyimpan, memanggil kembali, mengolah, menganalisis dan menghasilkan data bereferensi geografis atau data *geospasial*, untuk mendukung keputusan dalam perencanaan dan pengelolaan penggunaan lahan, sumber daya alam, lingkungan transportasi, fasilitas kota dan pelayanan umum lainnya.

## 2. Tinjauan Pustaka

### 2.1 Sistem informasi

Pengertian sistem informasi menurut beberapa ahli yaitu, Menurut Tata Sutabri (2012:6) pada buku Analisis Sistem Informasi, pada dasarnya sistem adalah sekelompok unsur yang erat hubungannya satu dengan yang lain, yang berfungsi bersama-sama untuk mencapai tujuan tertentu.

Selanjutnya Menurut McLeod (2004) dikutip oleh Yakub dalam buku Pengantar Sistem Informasi (2012:1) mendefinisikan sistem adalah Sekelompok elemen-elemen yang terintegrasi dengan tujuan yang sama untuk mencapai tujuan. Sistem juga merupakan suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, terkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk tujuan tertentu.

Sedangkan menurut Jogianto (2005: 2) pada buku Analisis dan Desain Sistem Informasi mendefinisikan sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. sistem ini menggambarkan suatu kejadian-kejadian dan kesatuan yang nyata adalah suatu objek nyata, seperti tempat, benda, dan orang-orang yang betul-betul ada dan terjadi.

### 2.2 Pengertian Sistem informasi Geografis

Sistem Informasi Geografis (SIG) pada dasarnya merupakan gabungan tiga unsur pokok yaitu sistem, informasi dan geografis. Dengan melihat unsur-unsur pokoknya, maka jelas sistem informasi geografis merupakan salah satu sistem

informasi dengan tambahan unsur “geografis”. Sistem Informasi Geografis diartikan sebagai sistem informasi yang digunakan untuk memasukkan, menyimpan, memanggil kembali, mengolah, menganalisis dan menghasilkan data bereferensi geografis atau data *geospatial*, untuk mendukung keputusan dalam perencanaan dan pengelolaan penggunaan lahan, sumber daya alam, lingkungan transportasi, fasilitas kota dan pelayanan umum lainnya.

Saat ini SIG sudah semakin berkembang dan para ahli pun mulai memiliki definisinya masing-masing, antara lain:

1. Menurut Ariyanti (2008, p310), sistem informasi merupakan berbagai macam kombinasi dari orang, perangkat keras, perangkat lunak, jaringan komunikasi dan sumber data yang menyimpan, memilah, mengubah bentuk dan mengeliminasi informasi didalam organisasi.
2. Menurut Prahasta (2005, p49) sistem informasi geografi merupakan suatu kesatuan formal yang terdiri dari berbagai sumber daya fisik dan logika yang berkenaan dengan objek-objek yang terdapat di permukaan bumi. Jadi Sistem Informasi Geografi merupakan kumpulan data geografi (spasial) dan data dokumen (non-spasial) yang terorganisir dan dapat dimanipulasi.

### 2.3 Tujuan Sistem Informasi Geografis

Untuk lingkup individu, Sistem Informasi Geografis sangat efektif untuk membantu proses pembentukan, pengembangan dan perbaikan peta mental yang sudah dimiliki oleh setiap orang yang berdampingan dengan dunia nyata.

Dalam lingkup pendidikan, Sistem Informasi Geografis dapat digunakan sebagai alat bantu utama dalam usaha meningkatkan pemahaman, pembelajaran dan pengertian mengenai ide atau konsep sebuah lokasi, ruang, kependudukan dan informasi geografis lainnya.

Dalam penelitian, Sistem Informasi Geografis dapat memberikan gambaran yang lebih lengkap dan akurat terhadap suatu masalah nyata yang terkait dengan data spasial permukaan bumi. Selain itu, Sistem Informasi Geografis juga memiliki kemampuan yang baik dalam memvisualisasikan data spasial.

Sehingga dengan menggunakan sistem ini, maka akan mempermudah dalam modifikasi warna, ukuran simbol dan bentuk yang diperlukan untuk menggambarkan unsur-unsur permukaan bumi. Pengguna juga dapat menginterpretasikan data yang didapat melalui Sistem Informasi Geografis secara manual.

### 2.4 Letak Geografis

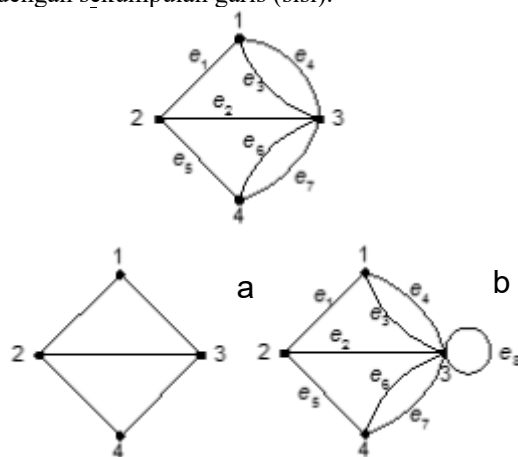
Letak geografis adalah posisi keberadaan sebuah wilayah berdasarkan letak dan bentuknya di permukaan bumi. Letak geografis biasanya dibatasi dengan berbagai fitur geografi yang ada di bumi dan nama daerah yang secara langsung bersebelahan dengan daerah tersebut. Fitur bumi yang dimaksud disini contohnya seperti benua, laut, gunung, samudera, gurun, dan lain sebagainya.

Robert Kaplan, seorang analist geopolitik dari Amerika Serikat menyebutkan bahwa letak geografis suatu wilayah (negara) secara luas akan menjadi determinan yang mempengaruhi berbagai peristiwa yang lebih daripada apa yang pernah terjadi sebelumnya. Disini jelas yang dimaksudkan adalah bahwa letak geografis sangat menentukan masa depan dari suatu negara dalam melakukan hubungan internasional.

### 2.5 Teori Graf

Graf merupakan suatu cabang ilmu yang memiliki banyak terapan. Banyak sekali struktur yang bisa direpresentasikan dengan graf, dan banyak masalah yang bisa diselesaikan dengan bantuan graf. Seringkali graf digunakan untuk merepresentasikan suatu jaringan. Misalkan jaringan jalan raya dimodelkan graf dengan kota sebagai simpul (vertex/node) dan jalan yang menghubungkan setiap kotanya sebagai sisi (edge) yang bobotnya (weight) adalah panjang dari jalan tersebut. Dalam beberapa model persoalan dimungkinkan bahwa bobot dari suatu sisi bernilai negatif. Misalkan simpul merepresentasikan kota, sisi merepresentasikan perjalanan yang memungkinkan, dan bobot dari setiap sisi adalah biaya yang dikeluarkan dalam perjalanan yang memungkinkan, dan bobot dari setiap sisi adalah jarak yang ditempuh dalam perjalanan tersebut. Graf  $G$  didefinisikan sebagai pasangan himpunan  $(V, E)$ . ditulis dengan notasi  $G = (V, E)$ , yang

dalam hal ini  $V$  adalah himpunan tidak kosong dari simpul-simpul (vertices atau node) dan  $E$  adalah himpunan sisi (edges atau arcs) yang menghubungkan sepasang simpul.  $V$  tidak boleh kosong, sedangkan  $E$  boleh kosong (Ristono dan Puryani, 2011). Simpul pada graf dapat dinomori dengan huruf, seperti  $a, b, c, \dots, v, w, \dots$ , dengan bilangan asli  $1, 2, 3, \dots$ , atau gabungan keduanya. Sedangkan sisi yang menghubungkan simpul  $u$  dengan simpul  $v$  dinyatakan dengan pasangan  $(u, v)$  atau dinyatakan dengan lambing  $e_1, e_2, \dots$ . Dengan kata lain, jika  $e$  adalah sisi yang menghubungkan simpul  $u$  dengan simpul  $v$ , maka  $e$  dapat ditulis sebagai  $e = (u, v)$ . Secara geometri graf digambarkan sebagai sekumpulan noktah (simpul) di dalam bidang dwimatra yang dihubungkan dengan sekumpulan garis (sisi).



**Gambar 1** Tiga buah graf (a) graf sederhana, (b) graf ganda, dan (c) graf semu

Graf sederhana adalah graf yang tidak mengandung gelang maupun sisi-ganda. Graf ganda adalah graf yang mengandung sisi-ganda. Graf semu adalah graf yang mengandung gelang (loop), graf semu lebih umum daripada graf ganda, karena sisi pada graf semu dapat terhubung ke dirinya sendiri. (Munir, 2005).

## 2.6 Algoritma Greedy

Adalah algoritma yang memecahkan masalah langkah demi langkah dan merupakan salah satu metode dalam masalah optimasi. Prinsip dari algoritma greedy adalah "take what you can get now" yaitu mengambil pilihan yang terbaik yang dapat diperoleh pada saat itu tanpa

memperhatikan konsekuensi kedepan. Algoritma greedy membentuk solusi langkah per langkah sebagai berikut :

1. Terdapat banyak pilihan yang perlu diekspolarasi pada setiap langkah solusi. Oleh karena itu, pada setiap langkah harus dibuat keputusan yang terbaik dalam menentukan pilhan. Keputusan yang telah diambil pada suatu langkah tidak dapat diubah lagi pada langkah selanjutnya.
2. Pendekatan yang digunakan di dalam algoritma greedy adalah membuat pilihan yang terlihat memberikan perolehan terbaik, yaitu dengan membuat pilihan optimum lokal pada setiap langkah dan diharapkan akan mendapatkan solusi optimum global.

Algoritma greedy didasarkan pada pemindahan edge per edge dan pada setiap langkah yang diambil tidak memikirkan konsekuensi ke depan, greedy tidak beroperasi secara menyeluruh terhadap semua alternatif solusi yang ada serta sebagian masalah greedy tidak selalu berhasil memberikan solusi yang benar-benar optimum tapi pasti memberikan solusi yang mendekati nilai optimum.

Algoritma greedy disusun oleh elemen-elemen sebagai berikut :

1. Himpunan Kandidat  
Himpunan ini berisi elemen-elemen yang memiliki peluang pembentuk solusi.
2. Himpunan Solusi  
Himpunan ini berisi kandidat-kandidat yang terpilih sebagai solusi persoalan. Elemennya terdiri dari elemen dalam himpunan kandidat, namun tidak semuanya dengan kata lain himpunan solusi ini adalah bagian dari himpunan kandidat.
3. Fungsi seleksi  
Fungsi yang pada setiap langkah memilih kandidat yang paling mungkin untuk menghasilkan solusi optimal. Kandidat yang sudah dipilih pada suatu langkah tidak pernah dipertimbangkan lagi pada langkah selanjutnya.
4. Fungsi kelayakan  
Fungsi yang memeriksa apakah suatu kandidat yang telah dipilih (diseleksi) dapat memberikan solusi yang layak.
5. Fungsi obyektif

Fungsi yang memaksimumkan atau meminimumkan nilai solusi. Tujuannya adalah memilih satu saja solusi terbaik dari masing-masing anggota himpunan solusi.

Contoh Kasus :

1. Penerapan Algoritma Greedy di dalam Penukaran Uang
  - Koin: 5, 4, 3, dan 1  
Uang yang ditukar = 7.  
Solusi greedy:  $7 = 5 + 1 + 1$  ( 3 koin) --> tidak optimal  
Solusi optimal:  $7 = 4 + 3$  ( 2 koin)
  - Koin: 10, 7, 1  
Uang yang ditukar = 15  
Solusi greedy:  $15 = 10 + 1 + 1 + 1 + 1 + 1$  (6 koin)  
Solusi optimal:  $15 = 7 + 7 + 1$  (hanya 3 koin)
  - Koin: 15, 10, dan 1  
Uang yang ditukar = 20  
Solusi greedy:  $20 = 15 + 1 + 1 + 1 + 1 + 1$  (6 koin)  
Solusi optimal:  $20 = 10 + 10$  (2 koin)

a. Penyelesaian dengan exhaustive search

Terdapat  $2^n$  kemungkinan solusi

(nilai-nilai  $X = \{x_1, x_2, \dots, x_n\}$  )

Untuk mengevaluasi fungsi obyektif =  $O(n)$

Kompleksitas algoritma *exhaustive search* seluruhnya =  $O(n \times 2^n)$ .

b. Penyelesaian dengan algoritma greedy

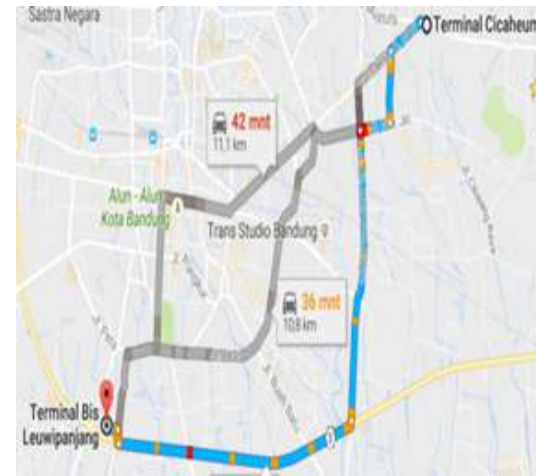
Strategi greedy : Pada setiap langkah, pilih koin dengan nilai terbesar dari himpunan koin yang tersisa.

Penyelesaian Greedy: (jika  $1 < k < 10$ , maka  $k < 10$  dan  $k < 10$ )  
(jika  $10 < k < 20$ , maka  $k < 20$  dan  $k < 20$ )  
(jika  $20 < k < 30$ , maka  $k < 30$  dan  $k < 30$ )  
(jika  $30 < k < 40$ , maka  $k < 40$  dan  $k < 40$ )  
(jika  $40 < k < 50$ , maka  $k < 50$  dan  $k < 50$ )  
(jika  $50 < k < 60$ , maka  $k < 60$  dan  $k < 60$ )  
(jika  $60 < k < 70$ , maka  $k < 70$  dan  $k < 70$ )  
(jika  $70 < k < 80$ , maka  $k < 80$  dan  $k < 80$ )  
(jika  $80 < k < 90$ , maka  $k < 90$  dan  $k < 90$ )  
(jika  $90 < k < 100$ , maka  $k < 100$  dan  $k < 100$ )  
(jika  $100 < k < 110$ , maka  $k < 110$  dan  $k < 110$ )  
(jika  $110 < k < 120$ , maka  $k < 120$  dan  $k < 120$ )  
(jika  $120 < k < 130$ , maka  $k < 130$  dan  $k < 130$ )  
(jika  $130 < k < 140$ , maka  $k < 140$  dan  $k < 140$ )  
(jika  $140 < k < 150$ , maka  $k < 150$  dan  $k < 150$ )  
(jika  $150 < k < 160$ , maka  $k < 160$  dan  $k < 160$ )  
(jika  $160 < k < 170$ , maka  $k < 170$  dan  $k < 170$ )  
(jika  $170 < k < 180$ , maka  $k < 180$  dan  $k < 180$ )  
(jika  $180 < k < 190$ , maka  $k < 190$  dan  $k < 190$ )  
(jika  $190 < k < 200$ , maka  $k < 200$  dan  $k < 200$ )  
(jika  $200 < k < 210$ , maka  $k < 210$  dan  $k < 210$ )  
(jika  $210 < k < 220$ , maka  $k < 220$  dan  $k < 220$ )  
(jika  $220 < k < 230$ , maka  $k < 230$  dan  $k < 230$ )  
(jika  $230 < k < 240$ , maka  $k < 240$  dan  $k < 240$ )  
(jika  $240 < k < 250$ , maka  $k < 250$  dan  $k < 250$ )  
(jika  $250 < k < 260$ , maka  $k < 260$  dan  $k < 260$ )  
(jika  $260 < k < 270$ , maka  $k < 270$  dan  $k < 270$ )  
(jika  $270 < k < 280$ , maka  $k < 280$  dan  $k < 280$ )  
(jika  $280 < k < 290$ , maka  $k < 290$  dan  $k < 290$ )  
(jika  $290 < k < 300$ , maka  $k < 300$  dan  $k < 300$ )  
(jika  $300 < k < 310$ , maka  $k < 310$  dan  $k < 310$ )  
(jika  $310 < k < 320$ , maka  $k < 320$  dan  $k < 320$ )  
(jika  $320 < k < 330$ , maka  $k < 330$  dan  $k < 330$ )  
(jika  $330 < k < 340$ , maka  $k < 340$  dan  $k < 340$ )  
(jika  $340 < k < 350$ , maka  $k < 350$  dan  $k < 350$ )  
(jika  $350 < k < 360$ , maka  $k < 360$  dan  $k < 360$ )  
(jika  $360 < k < 370$ , maka  $k < 370$  dan  $k < 370$ )  
(jika  $370 < k < 380$ , maka  $k < 380$  dan  $k < 380$ )  
(jika  $380 < k < 390$ , maka  $k < 390$  dan  $k < 390$ )  
(jika  $390 < k < 400$ , maka  $k < 400$  dan  $k < 400$ )  
(jika  $400 < k < 410$ , maka  $k < 410$  dan  $k < 410$ )  
(jika  $410 < k < 420$ , maka  $k < 420$  dan  $k < 420$ )  
(jika  $420 < k < 430$ , maka  $k < 430$  dan  $k < 430$ )  
(jika  $430 < k < 440$ , maka  $k < 440$  dan  $k < 440$ )  
(jika  $440 < k < 450$ , maka  $k < 450$  dan  $k < 450$ )  
(jika  $450 < k < 460$ , maka  $k < 460$  dan  $k < 460$ )  
(jika  $460 < k < 470$ , maka  $k < 470$  dan  $k < 470$ )  
(jika  $470 < k < 480$ , maka  $k < 480$  dan  $k < 480$ )  
(jika  $480 < k < 490$ , maka  $k < 490$  dan  $k < 490$ )  
(jika  $490 < k < 500$ , maka  $k < 500$  dan  $k < 500$ )  
(jika  $500 < k < 510$ , maka  $k < 510$  dan  $k < 510$ )  
(jika  $510 < k < 520$ , maka  $k < 520$  dan  $k < 520$ )  
(jika  $520 < k < 530$ , maka  $k < 530$  dan  $k < 530$ )  
(jika  $530 < k < 540$ , maka  $k < 540$  dan  $k < 540$ )  
(jika  $540 < k < 550$ , maka  $k < 550$  dan  $k < 550$ )  
(jika  $550 < k < 560$ , maka  $k < 560$  dan  $k < 560$ )  
(jika  $560 < k < 570$ , maka  $k < 570$  dan  $k < 570$ )  
(jika  $570 < k < 580$ , maka  $k < 580$  dan  $k < 580$ )  
(jika  $580 < k < 590$ , maka  $k < 590$  dan  $k < 590$ )  
(jika  $590 < k < 600$ , maka  $k < 600$  dan  $k < 600$ )  
(jika  $600 < k < 610$ , maka  $k < 610$  dan  $k < 610$ )  
(jika  $610 < k < 620$ , maka  $k < 620$  dan  $k < 620$ )  
(jika  $620 < k < 630$ , maka  $k < 630$  dan  $k < 630$ )  
(jika  $630 < k < 640$ , maka  $k < 640$  dan  $k < 640$ )  
(jika  $640 < k < 650$ , maka  $k < 650$  dan  $k < 650$ )  
(jika  $650 < k < 660$ , maka  $k < 660$  dan  $k < 660$ )  
(jika  $660 < k < 670$ , maka  $k < 670$  dan  $k < 670$ )  
(jika  $670 < k < 680$ , maka  $k < 680$  dan  $k < 680$ )  
(jika  $680 < k < 690$ , maka  $k < 690$  dan  $k < 690$ )  
(jika  $690 < k < 700$ , maka  $k < 700$  dan  $k < 700$ )  
(jika  $700 < k < 710$ , maka  $k < 710$  dan  $k < 710$ )  
(jika  $710 < k < 720$ , maka  $k < 720$  dan  $k < 720$ )  
(jika  $720 < k < 730$ , maka  $k < 730$  dan  $k < 730$ )  
(jika  $730 < k < 740$ , maka  $k < 740$  dan  $k < 740$ )  
(jika  $740 < k < 750$ , maka  $k < 750$  dan  $k < 750$ )  
(jika  $750 < k < 760$ , maka  $k < 760$  dan  $k < 760$ )  
(jika  $760 < k < 770$ , maka  $k < 770$  dan  $k < 770$ )  
(jika  $770 < k < 780$ , maka  $k < 780$  dan  $k < 780$ )  
(jika  $780 < k < 790$ , maka  $k < 790$  dan  $k < 790$ )  
(jika  $790 < k < 800$ , maka  $k < 800$  dan  $k < 800$ )  
(jika  $800 < k < 810$ , maka  $k < 810$  dan  $k < 810$ )  
(jika  $810 < k < 820$ , maka  $k < 820$  dan  $k < 820$ )  
(jika  $820 < k < 830$ , maka  $k < 830$  dan  $k < 830$ )  
(jika  $830 < k < 840$ , maka  $k < 840$  dan  $k < 840$ )  
(jika  $840 < k < 850$ , maka  $k < 850$  dan  $k < 850$ )  
(jika  $850 < k < 860$ , maka  $k < 860$  dan  $k < 860$ )  
(jika  $860 < k < 870$ , maka  $k < 870$  dan  $k < 870$ )  
(jika  $870 < k < 880$ , maka  $k < 880$  dan  $k < 880$ )  
(jika  $880 < k < 890$ , maka  $k < 890$  dan  $k < 890$ )  
(jika  $890 < k < 900$ , maka  $k < 900$  dan  $k < 900$ )  
(jika  $900 < k < 910$ , maka  $k < 910$  dan  $k < 910$ )  
(jika  $910 < k < 920$ , maka  $k < 920$  dan  $k < 920$ )  
(jika  $920 < k < 930$ , maka  $k < 930$  dan  $k < 930$ )  
(jika  $930 < k < 940$ , maka  $k < 940$  dan  $k < 940$ )  
(jika  $940 < k < 950$ , maka  $k < 950$  dan  $k < 950$ )  
(jika  $950 < k < 960$ , maka  $k < 960$  dan  $k < 960$ )  
(jika  $960 < k < 970$ , maka  $k < 970$  dan  $k < 970$ )  
(jika  $970 < k < 980$ , maka  $k < 980$  dan  $k < 980$ )  
(jika  $980 < k < 990$ , maka  $k < 990$  dan  $k < 990$ )  
(jika  $990 < k < 1000$ , maka  $k < 1000$  dan  $k < 1000$ )

Gambar 2. Penyelesaian dengan algoritma greedy

1. Penerapan Algoritma Greedy Untuk Menentukan Jalur Terpendek

Proses pencarian jalur terpendek akan penulis jelaskan pada gambar-gambar dan table-tabel di bawah ini.



Gambar 3. Peta lintasan

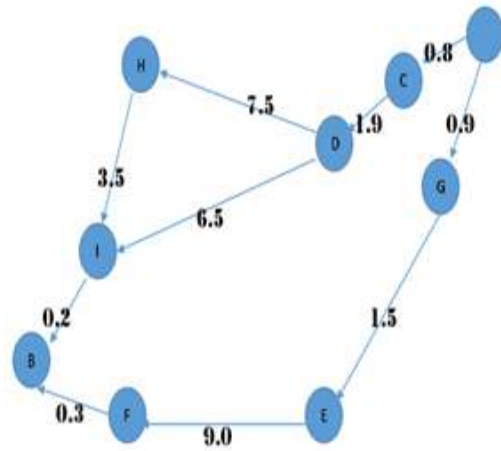
Dari peta yang ditampilkan di atas, dapat dilihat bahwa terdapat beberapa jalur dari titik A ke titik B. Sistem peta pada gambar secara otomatis telah memilih jalur terpendek (berwarna biru). Penulis akan mencoba mencari jalur terpendek juga, dengan menggunakan algoritma greedy.

Langkah pertama yang harus kita lakukan tentunya adalah memilih struktur data yang tepat untuk digunakan dalam merepresentasikan peta. Jika dilihat kembali, sebuah peta seperti pada gambar di atas pada dasarnya hanya menunjukkan titik-titik yang saling berhubungan, dengan jarak tertentu pada masing-masing titik tersebut. Misalnya, peta di atas dapat direpresentasikan dengan titik-titik penghubung seperti berikut:



**Gambar 4.** Graf Berarah

Dari gambar di atas, kita dapat melihat bagaimana sebuah peta jalur perjalanan dapat direpresentasikan dengan menggunakan graph, spesifiknya Directed Graph (graph berarah). Maka dari itu, untuk menyelesaikan permasalahan jarak terpendek ini kita akan menggunakan struktur data graph untuk merepresentasikan peta. Berikut adalah graph yang akan digunakan :



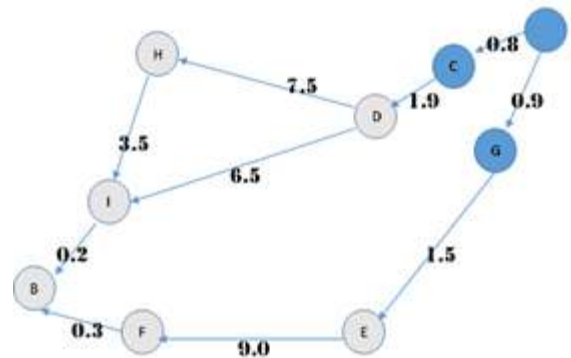
**Gambar 5.** Graf Berarah dari titik A ke titik B

Untuk mencari jarak terpendek dari A ke B, sebuah algoritma greedy akan menjalankan langkah-langkah seperti berikut:

1. Kunjungi satu titik pada graph, dan ambil seluruh titik yang dapat dikunjungi dari titik sekarang.
2. Cari local maximum ke titik selanjutnya.
3. Tandai graph sekarang sebagai graph yang telah dikunjungi, dan pindah ke local maximum yang telah ditentukan.
4. Kembali ke langkah 1 sampai titik tujuan didapatkan.

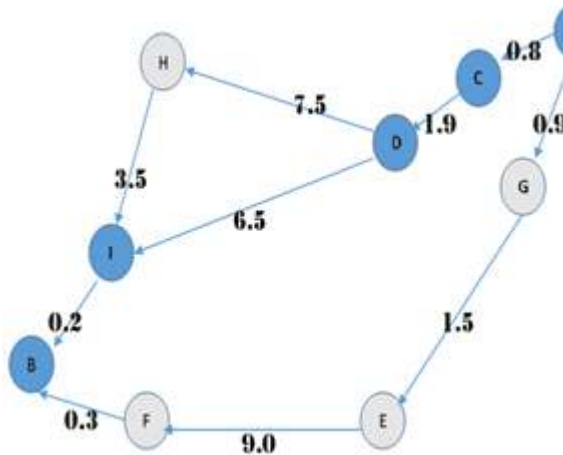
Jika mengaplikasikan langkah-langkah di atas pada graph A ke B sebelumnya maka kita akan mendapatkan pergerakan seperti berikut:

1. Mulai dari titik awal (A). Ambil seluruh titik yang dapat dikunjungi.



**Gambar 6.** Langkah Awal algoritma greedy

- a. Local maximum adalah ke C, karena jarak ke C adalah yang paling dekat.
- b. Tandai A sebagai titik yang telah dikunjungi, dan pindah ke C.
- c. Ambil seluruh titik yang dapat dikunjungi dari C.



**Gambar 7.** Langkah Pengambilan Lintasan

Dengan menggunakan algoritma greedy pada graph di atas, hasil akhir yang akan didapatkan sebagai jarak terpendek adalah A-C-D-I-B. Hasil jarak terpendek yang didapatkan ini tidak tepat dengan jarak terpendek yang di sarankan oleh aplikasi google maps yaitu (A-G-E-F-B). Walaupun jika di lihat dari jarak jika kita mengikuti lintasan berdasarkan penerapan algoritma greedy maka jaraknya lebih dekat yaitu 10.5km, sedangkan yang di sarankan oleh aplikasi google masps yaitu 11.7km. walaupun banyak pertimbangan untuk menentukan rute lintasan tersebut, salah satunya tingkat kemacetan di jalan raya, Algoritma greedy memang tidak selamanya memberikan solusi yang optimal, dikarenakan pencarian local maximum pada setiap langkahnya, tanpa memperhatikan solusi secara keseluruhan. Gambar berikut memperlihatkan bagaimana algoritma greedy dapat memberikan solusi.

### 3. METODOLOGI

#### 3.1 Jenis Penelitian

Jenis penelitian yang dilakukan adalah menggunakan metode penelitian *deskriptif kualitatif*. Menurut Maman (2002:3) penelitian deskriptif berusaha menggambarkan suatu gejala sosial. Dengan kata lain penelitian ini bertujuan untuk menggambarkan sifat sesuatu yang tengah berlangsung pada saat *study* (Nurdin, Nurdin, 2018; Nurdin, Nurdin, 2021).

Metode kualitatif ini memberikan informasi yang mutakhir hingga bermanfaat bagi perkembangan ilmu pengetahuan serta lebih banyak dapat diterapkan di berbagai masalah (Husein Umar, 1999:81).

Sedangkan penelitian ini lebih memfokuskan pada studi kasus yang merupakan penelitian yang rinci mengenai suatu objek tertentu selama kurun waktu tertentu dengan cukup mendalam dan menyeluruh. Deskriptif kualitatif yaitu penelitian yang di ajukan untuk memperoleh pembenaran atau verifikasi dalam bentuk dukungan data empiris di lapangan. (Santoso, 2005).

Menurut Vredembregt (1987: 38) studi kasus ialah suatu pendekatan yang bertujuan untuk mempertahankan keutuhan dari objek, Artinya data yang di kumpulkan dalam rangka studi kasus di pelajari sebagai suatu keseluruhan yang terintegrasi. Dimana tujuannya adalah untuk memperkembangkan pengetahuan yang mendalam mengenai objek yang bersangkutan yang berarti bahwa studi kasus harus di sifikan sebagai penelitian yang eksploratif dan deskriptif.

#### 3.2 Tipe Penelitian

Tipe penelitian ini adalah penelitian rekayasa perangkat keras dengan menggunakan metode dan *protocol* standar dalam ilmu teknik komunikasi dan teknologi informatika. Pengertian penelitian rekayasa menurut Muhammadiyah (1995:7) yaitu “penelitian rekayasa adalah penelitian yang menerapkan ilmu pengetahuan menjadi suatu rancangan guna mendapatkan kinerja sesuai dengan persyaratan yang di tentukan. Rancangan tersebut merupakan sintesis dari unsur-unsur yang dipadukan dengan ilmiah menjadi model yang memenuhi spesifikasi tertentu. penelitian di arahkan untuk memenuhi spesifikasi yang di tentukan”.

Tipe penelitian yang merupakan rancangan jaringan. Perancangan merupakan tahap persiapan

untuk merancang sistem informasi geografis. Yang menggambarkan letak terdekat rumah sakit dan memberikan jalan tercepat untuk menuju suatu titik yang ditunjuk dipeta.

### 3.3 Objek dan Waktu Penelitian

Objek penelitian adalah kantor Dinas Kesehatan dan Rumah Sakit yang ada di Kota Palu, waktu penelitian kurang lebih 2 (Dua) bulan.

### 3.4 Metode Pengembangan Sistem

Sistem informasi Letak Geografis Penentuan Jalur Tercepat Rumah Sakit Di Kota Palu Menggunakan Algoritma Greedy Berbasis Web menggunakan metode sekuensial linear (*waterfall*). Metode waterfall merupakan metode perangkat lunak yang sistematis dan sekuensial yang mulai pada tingkat dan kemajuan sistem sampai pada analisis, desain, kode, test dan pemeliharaan.

## 4. HASIL DAN PEMBAHASAN

### 4.1 Desain Sistem DFD

Pemodelan data untuk merepresentasikan aliran data/informasi digunakan *Data Flow Diagram* (DFD). Notasi pada diagram alir data dapat digunakan untuk merepresentasikan sistem atau aplikasi perangkat lunak ditingkat abstraksi diagram konteks atau DFD level 0. Selain itu DFD dikomposisi ke beberapa tingkatan yang lebih detail untuk merepresentasikan aliran-aliran data berikut fungsi-fungsi yang mentransformasikannya.

DFD adalah suatu model logika data atau proses yang dibuat untuk menggambarkan darimana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut. DFD menunjukkan hubungan antara data pada sistem dan proses pada sistem. Gambar 4.2 berikut merupakan DFD dari sistem informasi geografis pencarian rute ke rumah sakit melakukan masukan data spasial dan non-spasial ke sistem, selanjutnya di dalam sistem terjadi proses menghasilkan informasi spasial dari *Google Maps* dan interaksi dengan tabel-tabel dalam basis data sistem, kemudian dari data yang

dihasilkan proses pada sistem tersebut maka user/pengguna dapat mencari informasi dan juga sekaligus melihat informasi yang ada.

#### a. Konteks Diagram

Diagram konteks sistem informasi geografis pencarian rute ke rumah sakit seperti yang ditunjukkan pada Gambar 4.2 :

1. Menampilkan data spasial atau peta dari Google Maps.
2. Data dituliskan ke dalam basis data MySQL.

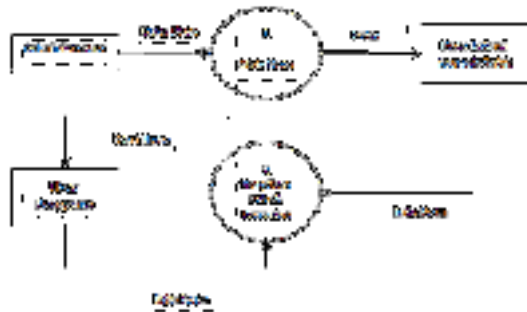


Gambar 8. Konteks Diagram SIG yang dibuat

#### b. DFD Level 1

Selanjutnya DFD Level 1 menggambarkan sistem sebagai jaringan kerja antara fungsi yang berhubungan satu dengan yang lain dengan aliran dan penyimpanan data. Dalam DFD Level ini akan terjadi penurunan level dimana level yang lebih rendah harus mampu merepresentasikan proses tersebut ke dalam spesifikasi proses yang lebih jelas. Pada gambar 4.3 DFD Level 1, administrator melakukan pengolahan data peta, selanjutnya sistem melakukan proses olah data spasial tersebut sehingga menghasilkan informasi yang disimpan pada berkas informasi peta fasilitas, dari berkas tersebut akan diproses untuk dapat ditampilkan pada sistem, dan dari hasil proses sistem tersebut dapat menghasilkan informasi peta untuk user/pengguna dan sebaliknya user/pengguna juga dapat mencari informasi peta dari sistem .

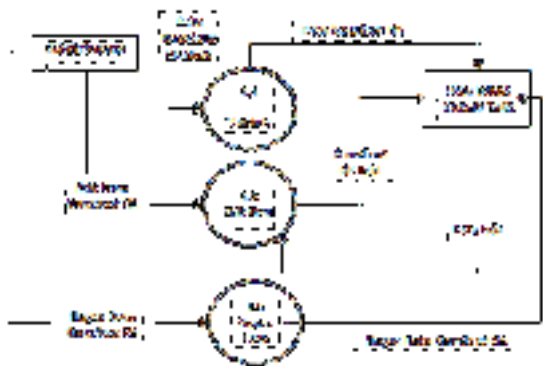




Gambar 9. DFD Level 1

c. DFD Level 2 Proses 1

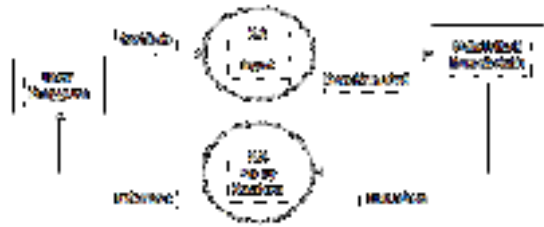
Gambar 4.4 merupakan diagram penurunan proses dari DFD level yang sebelumnya. Di sini administrator dapat melakukan proses masukan/tambah data, edit data, dan hapus data koordinat yang ada. Proses tambah data, edit data, dan hapus data mengacu ke berkas penyimpanan informasi peta.



Gambar 10. DFD Level 2 Proses 1

d. DFD Level 2 Proses 2

Gambar 5 merupakan diagram penurunan proses dari DFD level yang sebelumnya. Di sini administrator dapat melakukan proses masukan kata kunci pencarian. Proses pencarian data mengacu ke berkas penyimpanan informasi peta dan dari berkas tersebut akan diproses *query* basis data untuk dapat menghasilkan informasi peta untuk pengguna.



Gambar 11. DFD Level 2 Proses 2

#### 4.2 E-R Diagram/Conceptual Data Model

*Entity Relationship* (E-R) merupakan suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan suatu persepsi bahwa *real word* terdiri *object-object* dasar yang mempunyai hubungan atau relasi antar *object-object* tersebut.

Diagram E-R atau lebih dikenal dengan sebutan ERD atau diagram ER yang merupakan suatu model jaringan yang menggunakan susunan data yang tersimpan dalam sistem secara abstrak, untuk menggambarkan model relasional dari tabel yang digunakan dalam sistem pengolahan data. Adapun model diagram tersebut adalah :



Gambar 12. E-R Diagram (*Relasional Database*)

#### 4.3 Rancangan Tabel

Berikut akan diuraikan rancangan tabel pada sistem informasi geografi pencarian rute dengan algoritma Greedy. Rancangan tabel ini akan menguraikan nama tabel, jenis tabel, kunci primer yang dipergunakan, fungsi dari tabel itu sendiri serta atribut-atribut yang terkait didalamnya termasuk juga jenis serta lebar dari atribut tersebut. Adapun table-table yang terkait antara lain sebagai berikut ;

**Tabel 1** Properti Tabel Tempat

No.	Nama Field	Type	Size	Keterangan
1.	Id_Tempat	Int	11	Autonumber
2.	Nama_Tempat	Varchar	255	
3.	Gambar	Varchar	255	
4.	Lat	Double	9	
5.	Lng	Double	8	
6.	Lokasi	Varchar	255	
7.	Keterangan	Varchar	255	

**Tabel 2.** Properti Option

No.	Nama Field	Type	Size	Keterangan
1.	Option Name	Varchar	16	
2.	Option Value	Text		

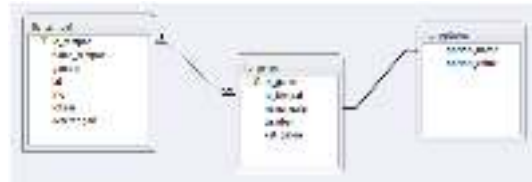
**Tabel 3.** Properti Tabel Galeri

No.	Nama Field	Type	Size	Keterangan
1.	Id_Galeri	Int	11	Autonumber
2.	Id_Tempat	Int	11	
3.	Nama_Galeri	Varchar	255	
4.	Gambar	Varchar	255	
5.	Ket_Galeri	Varchar	255	

#### 4.4 E-R Schema

Berdasarkan pada diagram E-R diatas, kemudian penulis mentransformasikan kedalam bentuk *ER-Schema* dengan ciri yaitu setiap *entity* dibentuk dalam sebuah kotak dengan nama *entity* berada diluar kotak dan atribut berada didalam kotak serta sebuah relasi dibuat dalam sebuah kotak bersama *entity*, sehingga lebih memudahkan

bagi pemakai dalam memahami struktur data yang dibuat sebagai berikut :



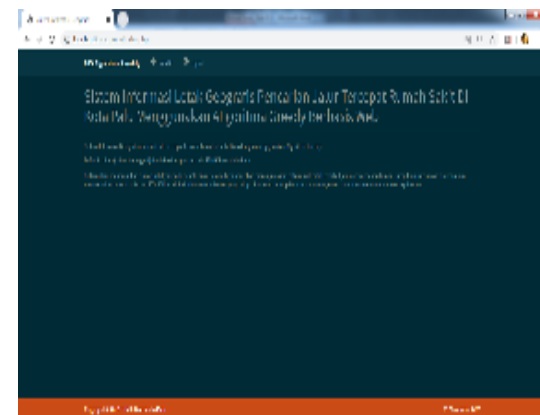
**Gambar 13.** ER Schema sistem informasi yang dibuat

#### 4.4 Laman

Berikut adalah rancangan laman yang dipergunakan oleh sistem informasi geografis mencari rute dengan algoritma Greddy,

##### 1. Form Utama

Form Utama digunakan sebagai laman pembuka dari sistem informasi geografis mencari rute dengan algoritma Greddy serta pintu masuk untuk melihat rute maupun mengolah data-data yang berkaitan dengan sistem informasi geografis mencari rute dengan algoritma Greddy.

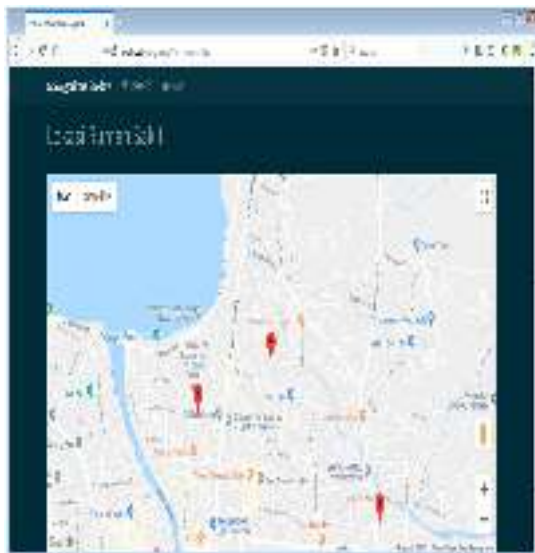


**Gambar 14.** Laman utama sistem informasi geografis mencari rute dengan algoritma Greddy yang dibuat

##### 2. Laman Pencarian Rute

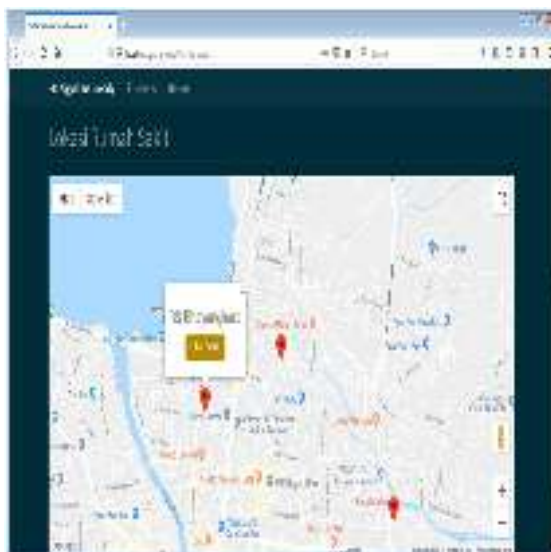
Untuk memudahkan dalam pembahasan maka penulis membuat nilai default atau nilai awal lokasi berangkat. Disini penulis menetapkan STMIK Bina Mulia Palu sebagai lokasi awal keberangkatan. Lokasi STMIK Bina Mulia Palu berada pada koordinat (-0.8867241978604428 ,

119.87571482817282). Mulai dari lokasi ini nanti aplikasi akan mencari rute ke rumah sakit yang dipilih menggunakan algoritma Greddy. Berikut tampilan laman untuk pencarian rute,



**Gambar 15.** Tampilan lokasi rumah sakit (tanda pin) yang telah di set.

Setelah muncul tampilan diatas, user diminta memilih salah satu lokasi rumah sakit (yang ditandai dengan pin).

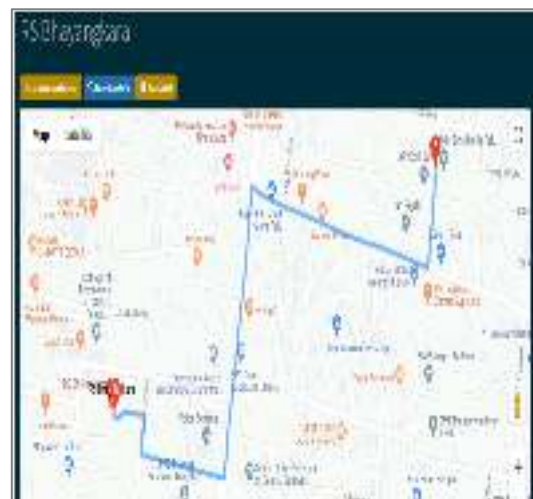


**Gambar 16.** Tampilan salah satu lokasi rumah sakit yang dipilih

Setelah dipilih akan muncul tampilan seperti gambar diatas, user diminta untuk mengklik tombol Lihat Detail.



**Gambar 17.** Tampilan rumah sakit dengan komponen pendukung Jika user telah mengklik tombol tersebut maka akan muncul beberapa tombol seperti Tampilkan Rute dan Detail rute.



**Gambar 18.** Tampilan rute dari lokasi awal ke rumah sakit yang dipilih. Jika tombol tampilan rute diklik maka akan muncul tampilan citra peta dengan rute dari lokasi awal ke rumah sakit yang dipilih.



**Gambar 19.** Tampilan detail rute dari lokasi awal ke rumah sakit yang dipilih, dan jika user mengklik detail rute maka disebelah kanan citra peta akan ditampilkan nama-nama jalan yang bisa dilalui lengkap dengan jarak dan tanda arah menuju ke lokasi rumah sakit yang dipilih.

### 3. Laman Login

Laman login berfungsi sebagai laman filter dari user / pengguna yang akan mengakses / mengoperasikan sistem informasi geografis mencari rute dengan algoritma Greddy.



**Gambar 20.** Laman Login

### 4. Laman Admin

Laman Admin digunakan sebagai laman utama admin dalam mengolah data pada sistem informasi geografis mencari rute dengan algoritma Greddy.



**Gambar 21.** Laman Admin

### 5. Laman Data Rumah Sakit dan Lokasi Awal

Laman Data Rumah Sakit dan Lokasi Awal digunakan sebagai laman untuk mengolah data rumah sakit dan lokasi awal berangkat



**Gambar 22.** Laman Data Rumah Sakit dan Lokasi Awal

#### 4.5 Implementasi Program

Pada bab ini akan dibahas tentang implementasi terhadap sistem informasi yang telah dibuat. Proses implementasi dilakukan dengan menampilkan halaman yang dibuat melalui aplikasi penampil web, seperti Mozilla Firefox, Google Chrome, Microsoft Internet Explorer atau penampil web lainnya. Untuk dapat menjalankan program tersebut terdapat hal-hal lain yang berhubungan diantaranya kebutuhan perangkat keras, perangkat lunak pendukung, serta beberapa hal yang akan diuraikan pada sub bab berikut ini.

##### 4.5.1 Kebutuhan Perangkat Keras

Berikut adalah kebutuhan perangkat keras minimum yang diperlukan. Komputer *desktop* atau laptop dengan spesifikasi sebagai berikut;

- Processor Intel Pentium dual Core 1,3 Ghz atau yang lebih tinggi
- Memory 1 Gb atau yang lebih tinggi.
- Hard drive dengan space 45 Mb.
- Display Card 512 Mb.
- Mouse dan Keyboard.
- Printer Inkjet atau laserjet.

##### 4.5.2 Kebutuhan Perangkat Lunak

Sedangkan untuk *software* atau perangkat lunaknya diperlukan spesifikasi sebagai berikut;

- Windows XP Service Pack 2 atau yang lebih tinggi yang berfungsi sebagai sistem operating.
- Aplikasi penampil web seperti Mozilla Firefox, Google Chrome, Microsoft Internet Explorer atau penampil web lainnya.
- MySql 5.0 atau yang lebih tinggi yang berfungsi sebagai database.

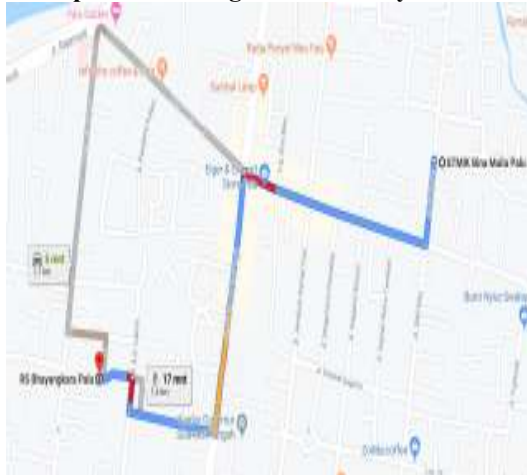
##### 4.5.3 File Database

Database MySQL mempunyai ekstensi \*.sql, file ini bisa langsung diekstrak aplikasi xampp atau aplikasi lain seperti MySQL Yog, Navicat dan lain-lain. Didalam file database ini terdiri table-table (tabel tempat, tabel galeri, serta tabel option), *view/query* serta komponen-komponen database lainnya.

##### 4.5.4 File Program

Semua file-file yang berhubungan dengan jalannya sistem informasi nilai siswa ini terdapat pada drive 'C' pada folder 'xampp\htdocs\gis'. Terdapat beberapa file pada *folder* tersebut seperti *index.php*, *koneksi.php* dan lainnya. Masing-masing file tersebut saling berkaitan, tetapi file *index.php* yang pertama kali dieksekusi oleh sistem untuk menjalankan sistem informasi ini.

#### 4.6 Implementasi Algoritma Greedy



**Gambar 23.** Peta Lintasan

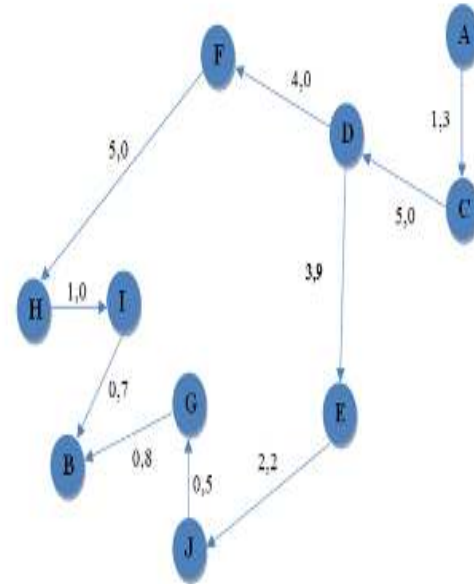
Dari peta yang ditampilkan di atas, dapat dilihat bahwa terdapat beberapa jalur dari titik A ke titik B. Pada sistem informasi yang di buat peta pada gambar secara otomatis telah memilih jalur tercepat (berwarna biru). Penulis akan mencoba merepresentasikan jalur tercepat juga, dengan menggunakan algoritma greedy.



**Gambar 24.** Graf Berarah

Dari gambar di atas, kita dapat melihat bagaimana sebuah peta jalur perjalanan dapat direpresentasikan dengan menggunakan graph,

spesifiknya Directed Graph (graph berarah). Maka dari itu, untuk menyelesaikan permasalahan jarak tercepat ini kita akan menggunakan struktur data graph untuk merepresentasikan peta. Berikut adalah graph yang akan digunakan :



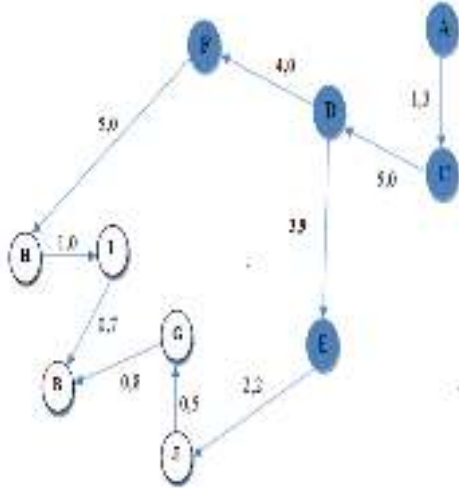
**Gambar 25.** Graf Berarah dari titik A ke titik B

Untuk mencari jarak tercepat dari A ke B, sebuah algoritma greedy akan menjalankan langkah-langkah seperti berikut:

1. Kunjungi satu titik pada graph, dan ambil seluruh titik yang dapat dikunjungi dari titik sekarang.
2. Cari local maximum ke titik selanjutnya.
3. Tandai graph sekarang sebagai graph yang telah dikunjungi, dan pindah ke local maximum yang telah ditentukan.
4. Kembali ke langkah 1 sampai titik tujuan didapatkan.

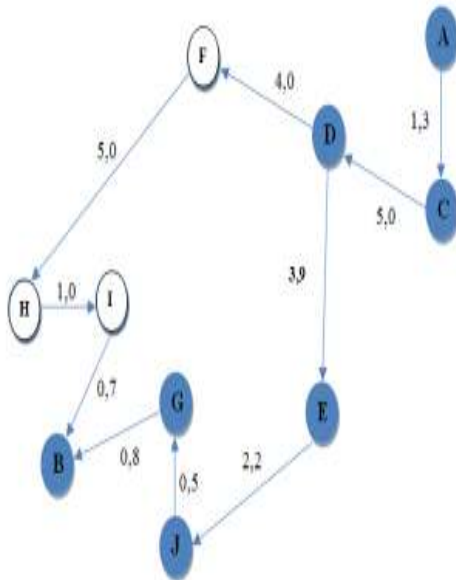
Jika mengaplikasikan langkah-langkah di atas pada graph A ke B sebelumnya maka kita akan mendapatkan pergerakan seperti berikut:

1. Mulai dari titik awal (D). Ambil seluruh titik yang dapat dikunjungi.



**Gambar 26.** Langkah awal algoritma greedy

- Local maximum adalah ke E, karena jarak ke E adalah yang paling dekat.
- Tandai D sebagai titik yang telah dikunjungi, dan pindah ke E.
- Ambil seluruh titik yang dapat dikunjungi dari E.



**Gambar 27.** Langkah Pengambilan Lintasan

Dengan menggunakan algoritma greedy pada graph di atas, hasil akhir yang akan didapatkan sebagai jarak tercepat adalah A-C-D-E-J-G-B.

#### 4.7 Uji Coba Program

Uji coba adalah sebuah proses yang harus dilakukan sebelum sistem informasi yang dibuat siap untuk diimplementasikan. Uji coba ini bertujuan untuk melihat kemungkinan-kemungkinan terjadinya kesalahan yang ditimbulkan oleh sistem informasi yang dibuat dengan penggunaan waktu dan usaha seminimal mungkin. Pada tugas akhir ini peneliti menggunakan tiga cara yaitu, meminta informan kunci untuk mencoba mengoperasikan sistem informasi yang dibuat, kemudian menjawab pertanyaan yang diajukan lewat lembar pertanyaan serta menerapkan teknik pengujian *black box* dan *white box* untuk menguji kesesuaian desain, logika dan fungsi serta kesalahan penulisan koding pada program.

a. *Black Box* adalah pengujian untuk mengetahui apakah semua fungsi perangkat lunak telah berjalan semestinya sesuai dengan kebutuhan fungsional yang telah didefinisikan. Metode ini memungkinkan perencana perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program. *Black box* dapat menemukan kesalahan dalam kategori,

- Fungsi-fungsi yang tidak benar atau hilang.
- Kesalahan *interface*.
- Kesalahan dalam struktur data atau akses basis data eksternal.
- Inisialisasi dan kesalahan terminasi.
- Validasi fungsional.
- Kesensitifan sistem terhadap nilai input tertentu.
- Batasan dari suatu data.

Untuk menguji sesuai kategori diatas, peneliti mengambil salah satu laman yaitu laman tempat. Laman ini dipilih karena didalam laman perangkat terdapat proses yang disyaratkan oleh kategori diatas. Berikut adalah sebagian *listing program* dari laman tersebut,

*Listing* program yang diblok paling atas adalah fungsi untuk menghubungkan program php dengan database (dalam contoh diatas tabel tempat). Selanjutnya pada blok dibawahnya adalah mengambil data pada tempat untuk ditampilkan. *Listing program* diatas dikatakan berhasil apabila program dapat terhubung dengan database dan mengambil data perangkat untuk ditampilkan. Ketika halaman dijalankan ternyata terlihat.



**Gambar 28.** Tampilan yang terisi data tempat bahwa pertanyaan ditampilkan tanpa terjadi kesalahan yang ditandai dengan munculnya pesan salah, sehingga uji coba ini dinilai berhasil.

Uji coba dilakukan dengan melihat logika serta proses iterasi (perulangan) yang ada pada saat pemanggilan data mata pelajaran pada laman tempat. Pada saat program dijalankan serta menampilkan laman tempat, yang terjadi adalah laman dapat menampilkan data tempat. Data-data tersebut kemudian dikomparasi dengan data yang ada dalam database. Setelah dikomparasi ternyata antara data yang ada dalam database dan data yang ditampilkan adalah sama. Hal ini menandakan bahwa logika program dan eksekusi terhadap perintah tersebut telah sesuai.



**Gambar 29.** Tampilan data tempat pada laman tempat

b. *White Box* adalah pengujian untuk memperlihatkan cara kerja dari sistem informasi secara rinci sesuai dengan spesifikasinya. Pada pengujian ini dipergunakan dua metode antara lain,

1. Metode pengujian dengan menggunakan struktur control program untuk memperoleh kasus uji.
2. Dengan menggunakan *white box* akan didapatkan kasus uji yang menjamin semua keputusan logika, menguji seluruh *loop* yang sesuai dengan batasannya, serta menguji seluruh struktur data internal yang menjamin validitas.

No	Nama	Alamat	No. Telp	No. Fax
1	PT. SINA MULIA	Bondowonegoro	031-8050000	031-8050000
2	PT. SINA MULIA	Bondowonegoro	031-8050000	031-8050000
3	PT. SINA MULIA	Bondowonegoro	031-8050000	031-8050000
4	PT. SINA MULIA	Bondowonegoro	031-8050000	031-8050000

**Gambar 30.** Data tempat pada database MySQL

Hasil ini juga didapatkan pada data tempat yang terdapat pada *database*.

#### 4.8 Hasil Analisa



Analisa data dilakukan dengan cara melihat hasil uji komponen dan logika menggunakan uji *black box* dan *white box* serta hasil pertanyaan yang diajukan pada beberapa informan kunci dalam mencoba sistem yang dibuat. Kepuasan pengguna juga dilakukan dengan menggunakan survey (Nurdin, Nurdin, 2015). Hasil survey tersebut dapat dilihat pada tabel berikut ini.

**Tabel 4.** Jawaban dari informan kunci 1

No.	Pertanyaan	Puas	Tidak Puas
1.	Apakah anda puas dengan fitur yang disajikan pada sistem informasi yang anda coba	YA	-
2.	Apakah anda puas dengan kesesuaian sistem informasi yang anda coba dengan tujuan pembuatannya	YA	-

**Tabel 5.** Jawaban dari informan kunci 2

No.	Pertanyaan	Puas	Tidak Puas
1.	Apakah anda puas dengan fitur yang disajikan pada sistem informasi yang anda coba	YA	-
2.	Apakah anda puas dengan kesesuaian sistem informasi yang anda coba dengan tujuan pembuatannya	YA	-

## 5. KESIMPULAN

Dari penjelasan yang telah disampaikan baik pada proses awal perancangan sampai dengan geografis pencarian rute menggunakan algoritma Greddy yang dibuat berjalan sesuai dengan rancangan awal serta jawaban oleh informan kunci menyatakan mereka puas dengan fitur serta kesesuaian sistem informasi yang dibuat sehingga secara langsung dapat mempermudah dan mempercepat dalam mendapatkan rute menuju rumah sakit yang dipilih sehingga dapat tercipta efektifitas serta efisiensi kerja.

## DAFTAR PUSTAKA

- Audrey Maximillian Herli. dkk (2015), Sistem Pencarian Hotel Berdasarkan Rute Perjalanan Terpendek Dengan Mempertimbangkan Daya Tarik Wisata Menggunakan Algoritma Greedy. *Journal of Information Systems Engineering and Business Intelligence*, 1(1):9-15.
- Arief, M. Rudianto. 2011. *Pemrograman Web Dinamis Menggunakan PHP dan MySQL*. Yogyakarta. Andi Offset.
- Aditya, Alan Nur. 2011. *Jago PHP & MySQL Dalam Hitungan Menit*. Jakarta: Dunia Komputer.
- Ariyani.2015. Pengertian sistem informasi geografis. Online: <http://www.ruanguru.co.id/sisteminformasi-geografis-jenismanfaat-tujuan-sumber-data-terlengkap/>. Diakses tanggal 8 November 2019.
- Efendi, F. S., Pinto, M., & Tempake, H. S. (2012). Implementasi Algoritma Greedy Untuk Melakukan Graph Coloring: Studi Kasus Peta Propinsi Jawa Timur. *Jurnal Informatika* vol 4 edisi 2 , 440-448.
- Ghofur Wibowo, Abdul dan Purwo Wicaksono, Agung (2012). Rancang Bangun Aplikasi untuk Menentukan Jalur Terpendek Rumah Sakit di Purbalingga dengan Metode Algoritma Dijkstra. *JUITA*, 2(1):21-35
- House.Cloude.2014.LocalHost.Online: <https://indcloudhostcom/mengenal-macam-macam-web-server-localhost/>. Diakses tanggal 10 November 2019

- Jenni Veronika Gnting & Ertina Sabarita Barus (2018), Aplikasi Penentuan Rute Rumah Sakit Terdekat Menggunakan Algoritma Dijkstra. *Jurnal Mantik Penusa*, 2(2):1-8
- Jogianto.2005. Analisis dan desain sistem informasi. Yogyakarta :CV ANDIOFFEST.
- Lukman, A., AR, R., & Nurhayati. (2011). Penyelesaian Travelling Salesman Problem dengan Algoritma Greedy. Makassar: Prosiding Konferensi Nasional Forum Pendidikan Tinggi Teknik Elektro Indonesia (FORTEI) 2011.
- Levitin, A., & Mukherjee, S. (2007). Introduction to the design & analysis of algorithms. Vol. 2. Pearson Addison-Wesley.
- Mahasiswa.Alumni.2016. Sistem informasi. Online:<https://www.google.co>. Landasan+teori+sistem+informasi. Diakses tanggal 7 November 2019.
- Muh. Yamin & Moh. Bandrigo Talai (2015). Aplikasi Pencarian Jalur Terpendek Pada Rumah Sakit Umum Bahteramas Menggunakan Algoritma A\* (A-STAR). *Jurnal informatika*, 9(2):1065-1078.
- Nugroho, Adi. 2006. *E-commerce. Informatika Bandung. Bandung.*
- Nugroho,Arief.2014. Pengertian bahasa pemrograman PHP. Online:[http//Batubenjo.blogspot.com/2018/01/mysql-dikembangkan-oleh-html-](http://Batubenjo.blogspot.com/2018/01/mysql-dikembangkan-oleh-html-). Diakses tanggal 07 November 2019.
- Nurdin, Nurdin. (2015). Analisis Adopsi dan Pemanfaatan Internet di Kalangan Mahasiswa Perguruan Tinggi di Kota Palu. *Jurnal Elektronik Sistem Informasi dan Komputer*, 1(1), 49-52.
- Nurdin, Nurdin. (2018). Institutional Arrangements in E-Government Implementation and Use: A Case Study From Indonesian Local Government. *International Journal of Electronic Government Research (IJEGR)*, 14(2), 44-63. doi: 10.4018/ijegr.2018040104
- Nurdin, Nurdin. (2021). A Collective Action In Indonesia Local E-Government Implementation Success. *INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH*, 10(2), 160-166.
- Nurdin, Nurdin, & Aratusa, Zana Chobita. (2020). Benchmarking level interactivity of Indonesia government university websites. *TELKOMNIKA Telecommunication, Computing, Electronics and Control*, 18(2), 853-859.
- Prayitno, Arif, & Nurdin, Nurdin. (2017). Analisa Dan Implementasi Kriptografi Pada Pesan Rahasia Menggunakan Algoritma Cipher Transposition. *Jurnal Elektronik Sistem Informasi dan Komputer*, 3(1), 1=10.
- Siska,Block. 2013. Sistem informasi letak geografis .Online: <https://www.goole.com/search?q=bab+2+p enelitian+terdahulu+sistem+informasi+geo grafis&oq>. Diakses tanggal 6 November 2019.
- Sulhan, Mohammad.2007. Pengembangan Aplikasi Berbasis Web dengan PHP & ASP. Yogyakarta: Gava Media.
- Taylor, 1975 dalam J. Moleong, Lexy. 1989. Metodologi Penelitian Kualitatif. Bandung: Remadja Karya.