

TUGAS AKHIR
SISTEM INFORMASI GEOGRAFIS MENCARI RUTE MENUJU LOKASI
TRAVEL DI KOTA PALANGKARAYA BERBASIS WEBSITE



OLEH:

NONO ARIADI
DBC 112 114

JURUSAN/PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PALANGKARAYA

2019

**“SISTEM INFORMASI GEOGRAFIS MENCARI RUTE MENUJU
LOKASI TRAVEL DI KOTA PALANGKARAYA BERBASIS WEBSITE”**

TUGAS AKHIR

Sebagai salah satu syarat untuk menyelesaikan Program Strata-1 pada Jurusan
Teknik Informatika Fakultas Teknik Universitas Palangka Raya

Oleh

NONO ARIADI
DBC 112 114

Disetujui untuk diajukan dalam Sidang Tugas Akhir

Mengetahui :

Dosen Pembimbing I,



VIKTOR HANRIANUS P., ST., MT
NIP. 19810606 200501 1 001

Dosen Pembimbing II,



WIDIATRY, ST., MT
NIP. 19820717 200312 2 002

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PALANGKA RAYA
PALANGKA RAYA
2019**

**SISTEM INFORMASI GEOGRAFIS MENCARI RUTE MENUJU LOKASI
TRAVEL DI KOTA PALANGKA RAYA BERBASIS WEBSITE**

TUGAS AKHIR

Sebagai salah satu syarat untuk menyelesaikan Program Strata-1 pada jurusan Teknik
Informatika Fakultas Teknik Universitas Palangka Raya

Oleh

NONO ARIADI
DBC 112 114

Telah dipertahankan didepan tim penguji, pada :

Hari / Tanggal : Selasa, 12 Februari 2019
Waktu : 13.15 – 14.45 WIB


1. Sherly Christina, S.Kom., M.Kom
NIP. 19810929 200604 2 001


.....(Ketua)

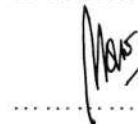
2. Viktor H. Pranatawijaya, ST., MT.
NIP. 19810606 200501 1 001


.....(Anggota)

3. Widiatry, ST., MT
NIP. 19820717 200312 2 002


.....(Anggota)

4. Nova Noor Kamala Sari, ST., M.Kom
NIP. 19890407 201504 2 004


.....(Anggota)

Mengetahui :

Fakultas Teknik
Universitas Palangka Raya
Dekan,

Jurusan/Program Studi Teknik Informatika
Fakultas Teknik Universitas Palangka Raya
Ketua Jurusan,

LEMBAR PERNYATAAN

Dengan ini Saya menyatakan dengan sebenar-benarnya bahwa dalam Tugas Akhir ini tidak terdapat karya ilmiah yang pernah diajukan untuk memperoleh gelar kesarjanaan disuatu Perguruan Tinggi, serta tidak terdapat karya ilmiah atau pendapat yang pernah ditulis atau diterbitkan orang lain, kecuali secara tertulis diacu dalam Tugas Akhir ini dan disebutkan dalam Daftar Pustaka.

Palangka Raya, 18 February 2019

NONO ARIADI
NIM. DBC 112 114

RIWAYAT PENYUSUN

Data Diri

Nama : NONO ARIADI
NIM : DBC 112 114
Fakultas : Teknik
Jurusan/Program Studi : Teknik Informatika
Jenjang : Strata 1 (S-1)
Jenis Kelamin : Laki - Laki
Tempat, Tanggal Lahir : Sampit, 10 Juni 1994
Agama : Islam
Status Dalam Keluarga : Anak Kandung
Anak Ke - : 3 (Tiga)
Alamat : Jl. Teuku Umar , Palangka Raya
No. Telepon/HP : 082255480017



Data Orang Tua

Nama Ayah : ZALDI
Pekerjaan Ayah : Petani
Nama Ibu : SITI MARDIYAH
Pekerjaan Ibu : PNS
Alamat Orang Tua : Tumbang Sangai Kec. Telaga Antang
No. Telepon/HP : 082152943210

Riwayat Pendidikan

SD : SD Negeri 1 Tumbang Sangai (Tahun Lulus 2006)
SMP : SMP PGRI Tumbang Sangai (Tahun Lulus 2009)
SMA : SMK Negeri 1 Sampit (Tahun Lulus 2012)

Palangka Raya, 19 Februari 2019

NONO ARIADI
NIM. DBC 112 114

LEMBAR PERSEMBAHAN



Yang Utama Dari Segalanya...

Sembah sujud serta syukur kepada Allah SWT. Taburan cinta dan kasih sayang-Mu telah memberikanku kekuatan, membekaliku dengan ilmu serta memperkenalkanku dengan cinta. Dari semua yang telah engkau tetapkan baik itu rencana indah yang engkau siapkan untuk masa depanku sebagai harapan kesuksesan. Atas karunia serta kemudahan yang Engkau berikan akhirnya skripsi yang sederhana ini dapat terselesaikan. Sholawat dan salam selalu terlimpahkan keharibaan Rasulullah Muhammad SAW.

Kupersembahkan karya sederhana ini kepada orang yang sangat kukasihi dan kusayangi

Dalam kesempatan ini juga saya ingin menyampaikan ucapan terima kasih yang sebesar-besarnya kepada :

1. Best support saya yaitu kedua orang tua Zaldi dan Siti Mardiyah, serta kakak-kakak dan ade saya yang telah mendukung saya dalam doa dan memberikan motivasi lebih bagi saya.
2. Semua dosen, terutama dosen pembimbing Tugas Akhir saya, Bapak Viktor H. Pranatawijaya, ST., MT. dan Ibu Widiatry, ST., MT. membahas yang selalu memberikan saya saran yang membangun demi selesainya Tugas Akhir ini, Serta Ibu Devi Karolita, S.Kom., M.Kom selaku dosen pembimbing akademik saya.
3. Dan untuk sahabat – sahabat saya yaitu rian alam, aprikardo, fauzan Abdi, Joni Teras, dan untuk semua teman terima kasih banyak telah menemani

saya dan membantu saya dalam menyelesaikan studi serta penelitian saya dan teman-teman Teknik Informatika angkatan 2012 yang tidak bisa saya sebutkan satu persatu.

KATA PENGANTAR

Puji syukur saya panjatkan kehadiran Tuhan Yang Maha Esa atas segala limpah dan Berkah – Nya saya dapat menyelesaikan laporan akhir Tugas Akhir, di kesempatan ini saya mengajukan penelitian dengan judul “Sistem Informasi Geografis Mencari Rute Menuju lokasi Travel Di Kota Palangkaraya Berbasis Website”.

Saya sadari laporan akhir yang saya buat masih jauh dari sempurna, oleh karena itu saya mengharapkan kritik dan saran yang bersifat membangun bagi siapapun yang membaca demi penyempurnaan laporan Tugas Akhir ini. Akhir kata, semoga laporan Tugas Akhir ini dapat bermanfaat bagi siapapun yang membacanya.

Palangka Raya, 29 Januari 2019

NONO ARIADI

SISTEM INFORMASI GEOGRAFIS Mencari Rute Menuju Lokasi Travel di Kota Palangkaraya Berbasis Website

Nono Ariadi (NIM. DBC 112 114)

Jurusan Teknik Informatika Fakultas Teknik Universitas Palangka Raya

Kampus Tunjung Nyaho Jl. Yos Sudarso Palangka Raya 73112

Email : nonoariadi12@gmail.com

Abstrak

Travel merupakan angkutan antar kota yang banyak di gunakan masyarakat di kota palngkaraya untuk bepergian ke luar kota. Namun banyak masyarakat yang kurang mengetahui lokasi dimana-mana letak travel yang ada di kota palangkaraya ini semua data yang mereka peroleh masih terbatas.

Dengan demikian dibutuhkan sebuah sistem yang menyediakan informasi geografis yang terkomputerisasi. Penelitian ini membahas tentang pemetaan lokasi dari travel yang diteliti, pencarian lokasi, filter data lokasi dan penentuan jalur terpendek menggunakan algoritma Dijkstra. Untuk pengembangan sistem dilakukan menggunakan metode *waterfall* dengan proses sebagai berikut: *Requirement Analysis* (analisis), *dedign*, *coding*, *Implementation* dan, *maintenance*. Sehingga diharapkan tersedianya sitem berbasis *web* sebagai wadah untuk memberikan informasi geografis letak lokasi travel yang di sajikan menggunakan *google map API*.

Pada penelitian, hasil pengujian untuk penentuan rute terpendek pada jalur tranportasi umum dapat ditentukan dengan algoritma Dijkstra. Jalur transportasi dianggap sebagai *graph* berarah dan berbobot, titik merepresentasikan sebuah tempat pemberhentian dan sisi sebagai jalur kendaraan. Pada pengujian *user* terlebih dahulu menginputkan titik awal pada peta kemudian mencari tempat tujuan pada peta pencarian tujuan, sistem akan menampilkan *output* rute terpendek awal menuju tempat tujuan.

Kata Kunci: Sistem Informasi Geografis, Travel, *Waterfall*, algoritma *dijkstra*, jalur terpendek

GEOGRAPHIC INFORMATION SYSTEMS LOOK FOR ROUTES TO TRAVEL LOCATION IN THE CITY OF PALANGKARAYA BASED ON WEBSITES

Nono Ariadi (NIM. DBC 112 114)

Department Of Information Engineering Faculty of Engineering University of Palangka Raya
Kampus Tunjung Nyaho, Jl. Yos Sudarso Palangka Raya 73112

E-mail : nonoariadi12@gmail.com

Abstract

Travel service is an intercity transportation that is widely used by people in the city of Palangkaraya to travel outside the city. But many people do not know where travel located in the city of Palangkaraya, all the data of they have is limited.

Thus a system that provides computerized geographic information is needed. This research discusses the location mapping of the travel studied, location search, location data filter and the shortest path determination using the Dijkstra Algorithm. For the development of the system carried out using the Waterfall method with the following process: *Requirement Analysis* (analysis), *design*, *coding*, *implementation* and *maintenance*. So it is expected that the availability of a web-based system as a place to provide geographic information on the location of travel is presented using the *Google map API*.

In research, the results of testing to determined the shortest route on the public transportation routes can be determined by the Dijkstra algorithm. Transportation lines are considered as directed and weighted graphs, the point represents stop place and side next to the vehicle line. In testing the user first inputs the starting point on the map then looks for the destination on the destination search map, the system will show output the shortest initial route to the destination.

Keywords: Geographic Information System, Travel, Waterfall, Dijkstra algorithm, shortest route

DAFTAR ISI

HALAMAN SAMPUL DEPAN	
KATA PENGANTAR	i
ABSTRAK	ii
DAFTAR ISI	iv
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	3
1.6 Metodologi.....	4
1.7 Sistematika Penulisan.....	5
1.8 Jadwal Pelaksanaan	7
BAB II LANDASAN TEORI	8
2.1 Sistem Informasi Geografis.....	8
2.2 Konsep <i>Real World</i>	8
2.3 Sumber Data Spasial.....	9
2.4 Sistem Informasi Geografis Berbasis <i>Web</i> dan <i>Platform</i> Google	10
2.5 <i>Database</i>	11
2.7 Sistem Basis Data	12
2.7 Metodologi <i>waterfall</i>	13
2.7.1 <i>Requirement analysis</i>	14
2.7.2 <i>Design</i>	14
2.7.3 <i>Coding</i>	14
2.7.4 <i>Implementasi dan Testing</i>	15
2.7.5 <i>maintenance</i>	15
2.8 Proses Bisnis.....	15
2.9 Metode <i>Graph</i>	28
2.10 Algoritma Sistem.....	29
2.10.1 Algoritma <i>Dijkstra</i>	29

2.11 HTML5	29
2.12 Pengenalan PHP	30
2.12.1 Menggunakan Semi Kolon	30
BAB III ANALISIS DAN DESAIN	32
3.1 Analisis	32
3.1.1 Bisnis Proses Sistem Lama	32
3.1.2 Bisnis Proses Sistem Baru.....	34
3.2 <i>Design</i>	37
3.2.1 <i>DFD (Data Flow Diagram)</i>	37
3.2.2 <i>ERD (Entitiy Relationship Diagram)</i>	45
3.2.3 <i>Desain Tabel</i>	47
3.2.4 Desain Struktur dan Interaksi Untuk Halaman Pengunjung.....	48
3.2.5 Desain Struktur dan Interkasi Untuk Halaman Admin.....	49
3.2.6 Navigation Desain	50
3.2.6.1 Halaman Admin	50
3.2.6.2 Halaman Pengunjung	51
3.3 Proses Algoritma Dijkstra	51
BAB IV IMPLEMENTASI	52
4.1 Implementasi Proses	52
4.2 Implementasi Data.....	52
4.3 Pengujian <i>Website</i>	53
4.3.1 Kegiatan Administrator didalam <i>Website</i>	53
4.3.2 Kegiatan Pengunjung didalam <i>Website</i>	58
4.4 Testing	60
4.5 Perhitungan Algoritma <i>Dijkstra</i>	62
BAB V PENUTUP	
5.1 Kesimpulan.....	68
5.2 Saran	68
DAFTAR PUSTAKA	70

DAFTAR GAMBAR

1. Gambar 1.1 Diagram Alur Metodologi Waterfall	4
2. Gambar 2.1 Tampilan konsep <i>real world</i>	9
3. Gambar 2.2 contoh <i>flowchart</i>	17
4. Gambar 2.3 Komponen – Komponen DFD Menurut Gane dan Serson	20
5. Gambar 2.4 Komponen – Komponen Entitas Luar	20
6. Gambar 2.5 Komponen Proses	21
7. Gambar 2.7 Contoh Dokumen HTML5	30
8. Gambar 2.8 Contoh Script php	31
9. Gambar 3.1 <i>Flowchart</i> Sistem Lama	33
10. Gambar 3.2 <i>Flowchart</i> Sistem Baru	36
11. Gambar 3.3 DFD Level 0 / Diagram Konteks	39
12. Gambar 3.4 DFD Level 1	42
13. Gambar 3.5 DFD level 2 Pada Travel	43
14. Gambar 3.6 DFD level 2 Pada Lokasi	44
15. Gambar 3.7 ERD(<i>Entity Relationship Diagram</i>)	46
16. Gambar 3.8 Tampilan Utama Pengunjung	48
17. Gambar 3.9 Tampilan Akun Admin	49
18. Gambar 3.10 Tampilan Rute Admin	49
19. Gambar 3.11 Tampilan Travel Admin	50
20. Gambar 3.12 desain Navigasi halaman Admin	50
21. Gambar 3.13 Desain Navigasi Halaman Pengunjung	51
22. Gambar 3.14 <i>Flowchart</i> Algoritma Dijkstra	51
23. Gambar 4.1 Belum Memasukan <i>Username</i> dan <i>password</i>	54
24. Gambar 4.2 validasi belum memasukan <i>password</i> dan <i>username</i>	54
25. Gambar 4.3 Validasi username atau password salah	54
26. Gambar 4.4 antarmuka halaman kelola Akun	55
27. Gambar 4.5 antarmuka halaman kelola rute	56
28. Gambar 4.6 antarmuka halaman kelola Travel	56
29. Gambar 4.7 validasi data berhasil pada halaman kelola Travel	57
30. Gambar 4.8 antarmuka halaman edit data kelola Travel	57
31. Gambar 4.9 validasi data berhasil di simpan pada halaman edit kelola Travel	58

32. Gambar 4.10 validasi data berhasil di hapus pada halaman edit kelola travel.....	58
33. Gambar 4.11 antarmuka halaman menu utama.....	59
34. Gambar 4.12 antarmuka halaman pencarian lokasi	59
35. Gambar 4.13 antarmuka rute menggunakan algoritma dijkstra.....	60
36. Gambar 4.14 alur graf gambar 1	62
37. Gambar 4.15 Pengujian perjalanan dari titik P ke titik J dengan algoritma dijkstra.....	65

DAFTAR TABEL

1. Tabel 1.1 Tabel jadwal pelaksanaan	7
2. Tabel 2.1 Keterangan symbol <i>flowchart</i>	17
3. Tabel 3.1 Definisi Diagram Konteks	40
4. Tabel 3.2 Tabel <i>Login</i>	47
5. Tabel 3.3 Tabel Travel	47
6. Tabel 3.4 Tabel Graph	48
7. Tabel 4.1 Tabel Testing Admin	61

BAB I

PENDAHULUAN

1.1. LATAR BELAKANG

Perkembangan yang sangat pesat saat ini membuat arus kebutuhan dalam dunia teknologi informasi turut berkembang cepat. Internet sebagai salah satu media untuk mendapatkan informasi juga semakin mudah diakses dari mana saja. Dengan berkembangnya teknologi internet, masyarakat semakin di mudahkan dalam melakukan segala macam fasilitas dan proses, salah satu contohnya adalah mencari travel di kota Palangkaraya, dimana terkadang masyarakat umum sangat sulit mencari lokas travel pada saat mereka ingin bepergian saat liburan tiba, khusus masyarakat yang berasal dari luar kota Palangkaraya.

Pencarian rute terpendek merupakan suatu masalah yang paling banyak dibahas dan dipelajari sejak akhir tahun 1950. Pencarian rute terpendek ini telah diterapkan diberbagai bidang untuk mengoptimasi kinerja suatu sistem baik untuk meminimalkan biaya ataupun mempercepat jalannya suatu proses. Pencarian rute terpendek termasuk dalam salah satu persoalan dalam teori graf yang berarti meminimalisasi bobot suatu lintasan dalam graf.

Metode yang dapat digunakan untuk menyelesaikan permasalahan pencarian jalur terpendek salah satunya adalah algoritma Dijkstra. Algoritma Dijkstra merupakan salah satu algoritma yang efektif dalam memberikan lintasan terpendek dari suatu lokasi ke lokasi yang lain. Prinsip dari algoritma Dijkstra adalah dengan pencarian dua lintasan yang paling kecil.

Kota palangkaraya adalah Ibu kota provinsi Kalimantan Tengah, di kota inilah pusat dari pemerintahan provinsi Kalimantan Tengah. Kota palangkaraya

merupakan kota yang berkembang dan kebanyakan penduduknya warga pendatang dari kota-kota lain di seluruh Indonesia. Model transportasi yang paling menonjol di kota Palangkaraya adalah travel yang digunakan untuk melakukan perjalanan keluar kota.

Berdasarkan latar belakang di atas pada proyek akhir ini dibuat sebuah program untuk memberikan solusi yaitu pencarian lokasi travel di kota Palangkaraya yang diakses lewat web. Dengan demikian sistem informasi geografis ini akan menampilkan semua lokasi travel di kota Palangkaraya, diharapkan hasil dari sistem informasi geografis lokasi travel di kota Palangkaraya dapat diakses melalui internet dengan menggunakan web.

Kemajuan teknologi Informasi mengenai geografis semakin dibutuhkan oleh banyak kalangan masyarakat misalnya informasi jarak antar daerah, lokasi, fasilitas dan banyak informasi lainnya. Informasi tersebut diperlukan pengguna untuk berbagai keperluan seperti penelitian, pengembangan, perancangan wilayah serta manajemen sumber daya alam. Karena adanya geografis ini dapat membantu penyajian suatu peta yang lebih interaktif, dimana pengguna dapat mengakses informasi geografis yang lengkap hanya dengan menggunakan komputer, web-browser dan jaringan internet. Maka untuk mendapatkan informasi itu semua diperlukan geographical information system (GIS) atau yang dikenal dengan sistem informasi geografis (SIG).

Oleh sebab itu, penulis merancang suatu sistem untuk menyajikan informasi geografis. Dari uraian di atas penulis mengangkat judul "***Sistem Informasi Geografis Mencari Rute Menuju lokasi Travel Di Kota Palangkaraya Berbasis Website***".

1.2 Rumusan Masalah

1. Bagaimana membangun sebuah sistem informasi geografis untuk mencari rute lokasi Travel di Kota Palangkaraya berbasis web ?
2. Bagaimana mencari lokasi Travel di Kota Palangkaraya secara cepat dan menampilkan rute ?
3. Bagaimana menampilkan titik-titik lokasi travel yang ada di kota Palangkaraya ?

1.3 Batasan Masalah

Berdasarkan identifikasi masalah di atas, agar pembahasan dapat lebih terfokus, maka permasalahan dalam penelitian ini dibatasi pada **“Sistem Informasi Geografis Mencari Rute Menuju lokasi Travel Di Kota Palangkaraya Berbasis Website.”** adalah :

1. Rute yang ditampilkan adalah rute menuju lokasi Travel yang ada di kota Palangkaraya.
2. Rute terpendek hanya untuk angkutan kota dan hanya jalan besar saja.
3. Sistem ini menggunakan algoritma dijkstra untuk memecahkan masalah lintasan terpendek dan perangkat lunak yang digunakan adalah PHP sebagai bahasa pemrograman, Google Maps untuk menampilkan peta, dan Mysql sebagai database.

1.4 Tujuan Penelitian

1. Untuk merancang dan membuat sebuah sistem informasi geografis mencari lokasi Travel di Kota Palangkaraya berbasis *web*.
2. Untuk membangun sistem informasi geografis berbasis web

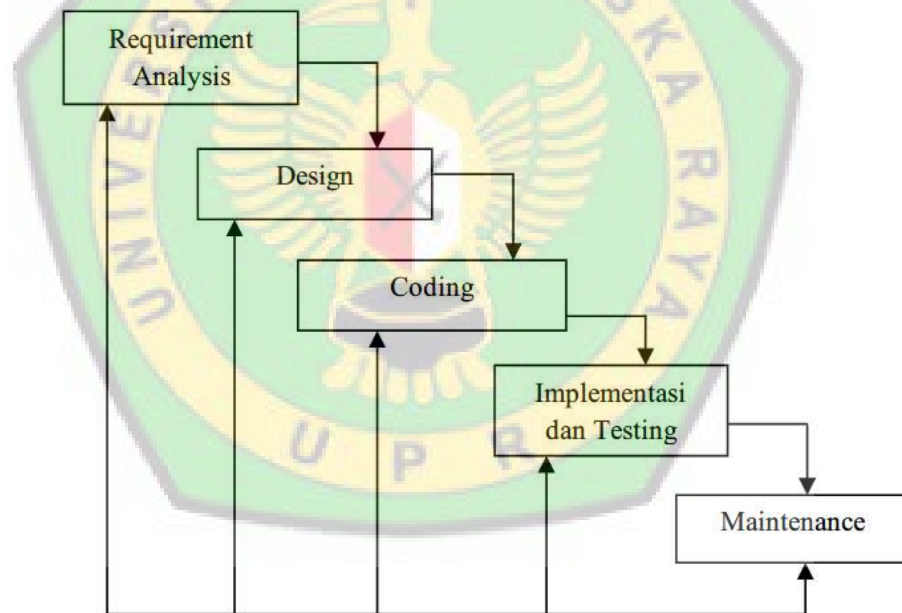
3. Membantu dalam hal pencarian informasi lokasi Travel yang berada di Kota Palangkaraya

1.5 Manfaat Penelitian

Manfaat penelitian yang ingin penulis peroleh melalui penelitian ini adalah untuk membantu masyarakat umum mengetahui informasi lokasi Travel yang ada di kota palangkaraya secara akurat.

1.6 METODOLOGI

Metode yang digunakan untuk Sistem Informasi Geografis Mencari Rute Menuju Lokasi Travel Di Kota Palangkaraya Berbasis Web adalah Waterfall (2011, Sommerville), berikut ini tahapan:



Gambar 1.1 Diagram Alur Metodologi Waterfall

Sumber : Sommerville,2011;30

Berikut ini langkah-langkah untuk mengembangkan sistem seperti yang ditentukan dalam Waterfall.

1. Requirement Analysis (analisis)

Pada bagian ini berisi rancangan dasar program yang akan dibuat sesuai dengan sasaran . Kemudian dari sistem lama dibuat rekomendasi sistem baru. Pada tahap analisis ini juga dilakukan dengan DFD (Data Flow Diagram) sebagai rancangan software yang akan dibuat.

2. Design

Pada tahapan ini desain yang akan di gunakan, yaitu : Desain interface aplikasi ini di buat disesuaikan dengan pengguna

3. Coding

Desain yang telah dibuat akan diubah kedalam bentuk yang dapat dimengerti oleh sistem komputer, yaitu kedalam bahasa pemrograman melalui proses *coding*. Tahap ini merupakan implementasi dari tahap desain yang secara teknis akan dikerjakan oleh *programmer*.

4. Implementasi Dan Testing

Dalam melakukan proses coding juga dilakukan proses testing untuk menguji dan melihat kesalahan yang ada pada program maupun fungsi dari sistem, testing yang digunakan yaitu menggunakan blackbox.

5. maintenance

Maintenance (pemeliharaan) sangat diperlukan, termasuk di dalamnya adalah pengembangan, karena aplikasi sistem informasi yang dibuat tidak selamanya hanya seperti itu. Ketika dijalankan mungkin saja masih ada kesalahan kecil yang tidak ditemukan sebelumnya, atau ada penambahan fitur-fitur yang belum ada pada aplikasi tersebut.

1.7 Sistematika Penulisan

BAB I PENDAHULUAN

Bab ini berisikan Latar Belakang, Rumusan Masalah, Batasan Masalah, Tujuan Penelitian, Manfaat Penelitian, Metodologi, Sistematika Penulisan dan Rencana Kegiatan.

BAB II LANDASAN TEORI

Bab ini berisikan Landasan Teori yang mendasari pembuatan Sistem Informasi Geografis Mencari Rute Menuju lokasi Travel Di Kota Palangkaraya Berbasis Website

BAB III ANALISIS DAN DESIGN

Bab ini berisikan analisis dan design untuk Sistem Informasi Geografis Lokasi Travel Di Kota Palangkaraya Berbasis Web

BAB IV IMPELEMENTASI

Bab ini berisikan implementasi analisis dan design Tugas Akhir Sistem Informasi Geografis Mencari Rute Menuju lokasi Travel Di Kota Palangkaraya ke bahasa pemrograman, serta pengujian sistem.

BAB V PENUTUP

Bab ini berisikan Kesimpulan dan Saran.

1.8 Jadwal Kegiatan

Adapun jadwal penyusunan Tugas Kerja ini dapat dilihat dalam tabel

1.1 berikut:

Tabel 1.1 Jadwal kegiatan

No.	Kegiatan	Bulan															
		Bulan I				Bulan II				Bulan III				Bulan IV			
		I	II	III	IV	I	II	III	IV	I	II	III	IV	I	II	III	IV
1	Analisis		■	■	■												
2	Desain																
3	Coding dan Testing					■	■	■	■			■					
4	Implementasi													■	■		
5	Pengujian Program												■	■	■		
6	Seminar															■	■

BAB II

LANDASAN TEORI

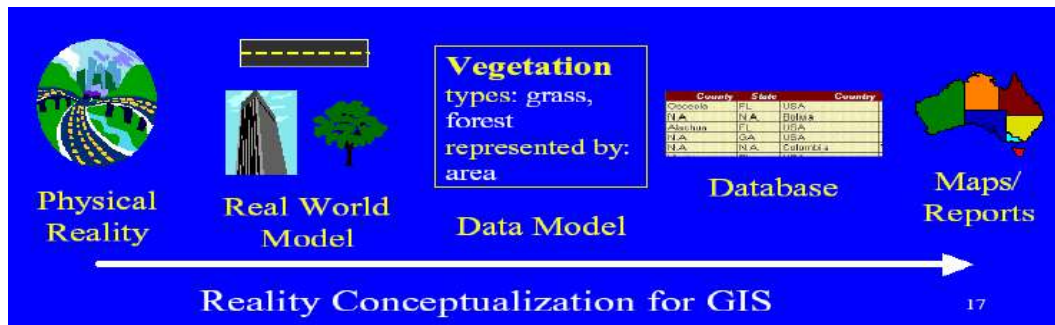
2.1 Sistem Informasi Geografis

Sistem Informasi Geografis (SIG) atau *Geographic Information System* (GIS) adalah sebuah sistem yang didesain untuk menangkap, menyimpan, memanipulasi, menganalisa, mengatur dan menampilkan seluruh jenis data geografis. Akronim GIS terkadang dipakai sebagai istilah untuk *geographical information science* atau *geospatial information studies* yang merupakan ilmu studi atau pekerjaan yang berhubungan dengan *Geographic Information System*. Dalam artian sederhana sistem informasi geografis dapat kita simpulkan sebagai gabungan kartografi, analisis statistik dan teknologi sistem basis data (*database*).

SIG tidak lepas dari data spasial, yang merupakan sebuah data yang mengacu pada posisi, obyek dan hubungan di antaranya dalam ruang bumi. Data spasial merupakan salah satu item informasi dimana di dalamnya terdapat informasi mengenai bumi termasuk permukaan bumi, di bawah permukaan bumi, perairan, kelautan, dan bawah atmosfer.

2.2 Konsep *Real World*

Konsep *Real World* merupakan sebuah cara bagaimana SIG mengubah realitas fisik sebuah dunia menggunakan model menjadi sebuah sistem informasi geografis yang dapat disimpan, dimanipulasi, diproses dan direpresentasikan.



Gambar 2.1 Tampilan konsep *real world*
(Edy Irwansyah, 2013:3)

Konsep *real world* memiliki beberapa tahapan sebagai berikut:

1. *Physical Reality*

Merupakan tahapan dimana menganalisa dunia nyata yang akan dibuat menjadi SIG.

2. *Real World Model*

Tahapan mengubah objek – objek yang ada di dunia nyata menjadi model.

3. *Data Model*

Tahapan yang mengubah model – model objek dunia nyata menjadi sebuah tipe data.

4. *Database*

Meyimpan keseluruhan data model ke dalam sistem basis data.

5. *Maps/Reports*

Merupakan hasil akhir dunia nyata yang telah dikonversi menjadi sebuah sistem informasi geografis.

2.3 Sumber Data Spasial

Sebagaimana telah kita ketahui, SIG membutuhkan masukan data yang bersifat spasial maupun deskriptif. Beberapa sumber data tersebut antara lain adalah:

1. Data dari sistem Penginderaan Jauh (antara lain citra satelit, foto udara, dsb.)
Data Penginderaan Jauh dapat dikatakan sebagai sumber data yang terpenting bagi SIG karena ketersediaannya secara berskala. Dengan adanya bermacam-macam satelit di ruang angkasa dengan spesifikasinya masing-masing, kita bisa menerima berbagai jenis citra satelit untuk beragam tujuan pemakaian. Data ini biasanya direpresentasikan dalam format *raster*.
2. Data GPS. Teknologi GPS memberikan terobosan penting dalam menyediakan data bagi SIG. Keakuratan pengukuran GPS semakin tinggi dengan berkembangnya teknologi. Data ini biasanya direpresentasikan dalam format vektor.

2.4. SIG Berbasis Web dan Platform Google

Sistem Informasi Geografi adalah sebuah sistem berbasis komputer yang digunakan untuk menyimpan dan “memanipulasi” informasi-informasi geografis (Arronof, 1989). SIG berbasis Web atau juga sering disebut dengan WebGIS atau InternetGIS, didefinisikan sebagai suatu jaringan (*network*) berbasis layanan informasi geografis yang memanfaatkan internet baik menggunakan jaringan kabel (*wired*) maupun tanpa kabel (*wireless*) untuk mengakses informasi geografis maupun sebagai tools guna melakukan spatial analisis (Ren Peng.Z and Hsing Tsou.M, 2003).

Google merupakan sebuah perusahaan multinasional Amerika Serikat yang bergerak dalam pengembangan teknologi berbasis internet dan produk. *Google* didirikan pada tahun 1996 oleh Larry Page dan Sergey Brin yang saat ini telah mengembangkan beberapa aplikasi berbasis internet dan produk yang digunakan secara luas seperti *Google search engine*, *Google Mail* (Email), *Google*

Talk (Jejaring sosial), *Google Chrome* (Mesin penjelajah), *Android* (Sistem operasi), *Google earth* dan *Google Maps* .

Google Maps merupakan bentuk layanan dari *Google* yang menawarkan teknologi pemetaan terkini yang dapat digunakan untuk berbagai macam kebutuhan. *Google Maps* mempunyai *platform opensource* sehingga dapat digunakan dengan bebas namun harus mematuhi syarat yang telah ditetapkan. *Google Maps* juga memberikan kebebasan kepada pengembang untuk mengembangkan teknologi pemetaan yang berbasis *Google Maps*, sehingga dapat memperkaya fitur yang sebelumnya ada pada *Google Maps*. Untuk pengembangan ini, *Google* mempunyai 2 pilihan *platform*, yaitu *opensource platform* (gratis) dan *Enterprise Platform* (berbayar). Dalam hal ini pengembangan *platform Google Maps* menggunakan sebuah bahasa pemrograman yang dinamakan dengan *Maps API Java Script programming*.

2.5 Database

Basis data dapat dipahami sebagai suatu kumpulan data terhubung (*interrelated data*) yang disimpan secara bersama – sama pada suatu media, tanpa mengatap suatu sama lain atau tidak perlu suatu kerangkapan data (kalaupun ada maka kerangkapan data tersebut harus seminimal mungkin dan terkontrol [*controlled redundancy*]), data disimpan dengan cara - cara tertentu sehingga mudah digunakan/atau ditampilkan kembali; data dapat digunakan oleh satu atau lebih program – program aplikasi secara optimal; data disimpan tanpa mengalami ketergantungan dengan program yang akan menggunakannya, data disimpan sedemikian rupa sehingga proses penambahan, pengambilan, dan modifikasi data dapat dilakukan dengan mudah dan terkontrol.

Berdasarkan definisi tersebut maka suatu basis data mempunyai beberapa kriteria penting yang harus dipenuhi yaitu(Martin, 1975):

1. Berorientasi pada data dan bukan berorientasi pada program(*program oriented*) yang akan menggunakannya.
2. Data dalam basis data dapat berkembang dengan mudah, baik volume maupun strukturnya.
3. Data yang dapat memenuhi kebutuhan system – system baru secara mudah.
4. Data dapat digunakan dengan cara yang berbeda – beda.
5. Kerangkapan data(*data redundancy*) minimal.

Sebagai bahan pertimbangan, berikut definisi *database* menurut beberapa pakar lainnya. (C.J Date,1995) mendefinisikan basis data sebagai beberapa kumpulan data yang akan tetap tersimpan, digunakan oleh system – system aplikasi yang diberikan oleh organisasi. J.L. Whitten & L.D. Bentkey (1998) mendefinisikan basis data sebagai sekumpulan data dalam file yang saling terhubung, *record* dalam file harus mengizinkan adanya kerelasian (dapat dibayangkan sebagai pointer) ke *recor –record* lain dalam file yang lain. (Raghu Ramakrishan, 2000) mendefinisikan *database* sebagai sekumpulan data yang saling berhubungan yang menjelaskan aktivitas – aktivitas pada sebuah organisasi.

2.6 Sistem Basis Data

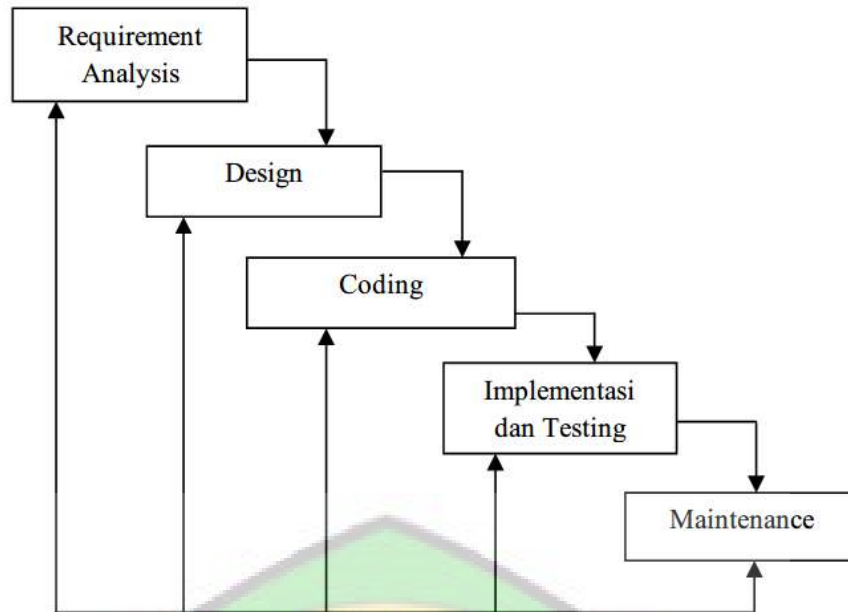
Sistem basis data dapat diartikan sebagai sekumpulan basis data dalam suatu sistem yang mungkin tidak ada hubungan satu sama lain tetapi secara keseluruhan mempunyai hubungan sebagai sebuah system dengan didukung oleh komponen lainnya. Isitilah basis data juga dapat didefinisikan sebagai sekumpulan

subsistem yang terdiri atas basis data dengan para pemakai yang menggunakan basis data secara bersama – sama, personal – personal yang merancang dan mengelola basis data, teknik –teknik untuk merancang dan mengelola basis data, serta sistem komputer untuk mendukungnya(Maartin ,1975). Dari definisi tersebut dapat disimpulkan bahwa sistem basis data mempunyai beberapa elemen penting, yaitu(Stutanta, 2004):

1. Basis data sebagai inti dari sistem basis data.
2. Perangkat lunak (*software*) untuk perancangan dan pengelolaan basis data.
3. Perangkat keras (*hardware*) sebagai pendukung operasi pengolahan data Manusia (*brainware*) yang mempunyai peran penting dalam sistem tersebut, yaitu sebagai pemakai atau para spesialis informasi yang mempunyai fungsi sebagai perancang atau pengelola.

2.7 Metodologi Waterfall

Menurut Ian Sommerville (2011, p30-31), tahapan utama dari *waterfall model* langsung mencerminkan aktifitas pengembangan dasar. Terdapat 5 tahapan pada *waterfall model*, yaitu *requirement analysis and definition, system and software design, implementation and unit testing, integration and system testing, dan operation and maintenance.*



2.7.1. *Requirement Analysis*

Merupakan tahapan penetapan fitur, kendala dan tujuan sistem melalui konsultasi dengan pengguna sistem. Semua hal tersebut akan ditetapkan secara rinci dan berfungsi sebagai spesifikasi system.

2.7.2. *Design*

Dalam tahapan ini akan dibentuk suatu arsitektur sistem berdasarkan persyaratan yang telah ditetapkan. Dan juga mengidentifikasi dan menggambarkan abstraksi dasar sistem perangkat lunak dan hubungan-hubungannya.

2.7.3. *Coding*

Dalam tahapan ini, hasil dari desain perangkat lunak akan direalisasikan sebagai satu set program atau unit program. Setiap unit akan diuji apakah sudah memenuhi spesifikasinya.

2.7.4. Implementasi dan testing

Dalam tahapan ini, setiap unit program akan diintegrasikan satu sama lain dan diuji sebagai satu sistem yang utuh untuk memastikan sistem sudah memenuhi persyaratan yang ada. Setelah itu sistem akan dikirim ke pengguna sistem.

2.7.5. Maintenance

Dalam tahapan ini, sistem diinstal dan mulai digunakan. Selain itu juga memperbaiki *error* yang tidak ditemukan pada tahap pembuatan. Dalam tahap ini juga dilakukan pengembangan sistem seperti penambahan fitur dan fungsi baru.

2.8. PROSES BISNIS

Adalah kumpulan dari proses dan berisi kumpulan aktifitas (tasks) yang saling berelasi satu sama lain untuk menghasilkan suatu keluaran yang mendukung pada tujuan dan sasaran strategis dari sistem. Masing - masing memerlukan serangkaian tugas dan memiliki sumber daya yang berbeda dalam bisnis tersebut. setiap proses memiliki pelanggan terbatas yang menerima hasil akhir. proses bisnis mengharuskan kelompok organisasi yang berbeda berpartisipasi dalam “tugas-tugas yang dihubungkan secara logis” yang menentukan proses. Komponen bisnis proses adalah sebagai berikut.

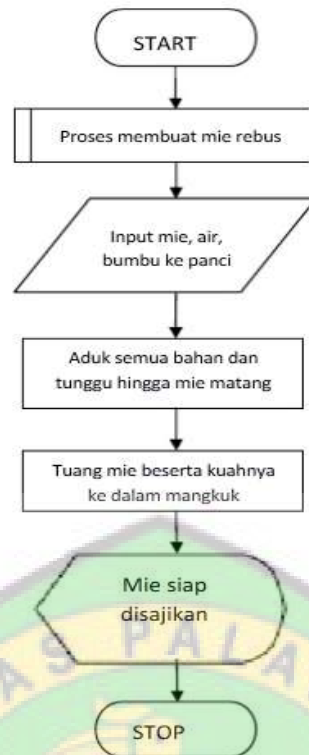
1. Skenario

Adalah suatu cerita atau narasi, yang mudah dimengerti untuk membuat aplikasi menjadi lebih hidup. Cerita yang baik harus spesifik dan diceritakan seperti pada dunia nyata. Skenario harus spesifik, menarik dan

atraktif. Pada skenario kalimat harus mengandung rumus dasar yaitu SPOK(Subjek Predikat Objek dan Keterangan).

2. *Flowchart*

Flowchart merupakan gambar atau bagan yang memperlihatkan urutan dan hubungan antar proses beserta instruksinya. Gambaran ini dinyatakan dengan simbol. Dengan demikian setiap simbol menggambarkan proses tertentu. Sedangkan hubungan antar proses digambarkan dengan garis penghubung. *Flowchart* ini merupakan langkah awal pembuatan program. Dengan adanya *flowchart* urutan poses kegiatan menjadi lebih jelas. Jika ada penambahan proses maka dapat dilakukan lebih mudah. Setelah *flowchart* selesai disusun, selanjutnya pemrogram (*programmer*) menerjemahkannya ke bentuk program dengan bahasa pemrograman. Dalam pembuatan *flowchart* tidak ada rumus atau patokan yang bersifat mutlak. Karena *flowchart* merupakan gambaran hasil pemikiran dalam menganalisa suatu masalah dengan komputer. Sehingga *flowchart* yang dihasilkan dapat bervariasi antara satu pemrogram dengan pemrogram lainnya.





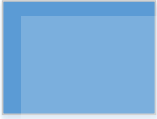

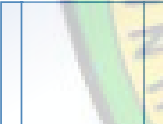
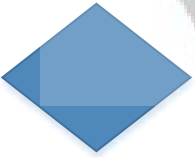

Gambar 2.2 contoh flowchart


(Whitten, Jeffery L, 2004:64)

Adapun tabel menunjukkan keterangan simbol *flowchart* berikut:

Tabel 2.1 Keterangan simbol *flowchart*

Simbol	Nama	Fungsi
	<i>Terminator</i>	Menjelaskan sebagai permulaan/akhir program.

Simbol	Nama	Fungsi
	<i>Garis Alir (Flow Line)</i>	Menjelaskan arah aliran program
	<i>Preparation</i>	Menjelaskan proses inisialisasi/ pembelian harga awal.
	<i>Process</i>	Menjelaskan proses perhitungan /proses pengolahan data
	<i>Input/Output Data</i>	Menjelaskan proses <i>input/output</i> data, parameter, informasi.
	<i>Predefined process (Sub Program)</i>	Menjelaskan permulaan sub program /proses menjalankan sub program.
	<i>Decision</i>	Menjelaskan perbandingan pernyataan, penyelesaian data yang memberikan pilihan untuk langkah selanjutnya.
	<i>One Page Connector</i>	Menjelaskan penghubung bagian-bagian <i>flowchart</i> yang berada pada satu halaman.

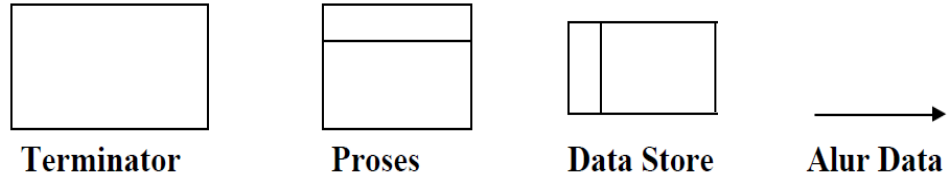
Simbol	Nama	Fungsi
	<i>Off Page Connector</i>	Menjelaskan penghubung bagian-bagian <i>flowchart</i> yang berada pada halaman berbeda.

3. DFD(*Data Flow Diagram*)

Data Flow Diagram (DFD) adalah alat pembuatan model yang memungkinkan profesional sistem untuk menggambarkan sistem sebagai suatu jaringan proses fungsional yang dihubungkan satu sama lain dengan alur data, baik secara manual maupun komputerisasi. DFD ini sering disebut juga dengan nama Bubble chart, Bubble diagram, model proses, diagram alur kerja, atau model fungsi. DFD adalah salah satu alat pembuatan model yang sering digunakan, khususnya bila fungsi-fungsi sistem merupakan bagian yang lebih penting dan kompleks dari pada data yang dimanipulasi oleh sistem. Dengan kata lain, DFD adalah alat pembuatan model yang memberikan penekanan hanya pada fungsi sistem. DFD ini merupakan alat perancangan sistem yang berorientasi pada alur data dengan konsep dekomposisi dapat digunakan untuk penggambaran analisa maupun rancangan sistem yang mudah dikomunikasikan oleh profesional sistem kepada pemakai maupun pembuat program.

Komponen – Komponen *Data flow Diagram*

1. Menurut Gane dan Serson



Gambar 2.3 Komponen – Komponen DFD Menurut Gane dan Serson

1. Komponen Terminator/Entitas Luar

Terminator mewakili entitas eksternal yang berkomunikasi dengan sistem yang sedang dikembangkan. Biasanya terminator dikenal dengan nama entitas luar (*external entity*). Terdapat dua jenis terminator :

1. Terminator Sumber (*source*) : merupakan terminator yang menjadi sumber.
2. Terminator Tujuan (*sink*) : merupakan terminator yang menjadi tujuan data / informasi system.



Gambar 2.4 Komponen – Komponen Entitas Luar

Terminator dapat berupa orang, sekelompok orang, organisasi, departemen di dalam organisasi, atau perusahaan yang sama tetapi di luar kendali sistem yang sedang dibuat modelnya. Terminator dapat juga berupa departemen, divisi atau sistem di luar sistem yang berkomunikasi dengan sistem yang sedang dikembangkan. Komponen terminator ini perlu diberi nama sesuai dengan dunia luar yang berkomunikasi dengan sistem

yang sedang dibuat modelnya, dan biasanya menggunakan kata benda, misalnya Bagian Penjualan, Dosen, Mahasiswa. Ada tiga hal penting yang harus diingat tentang terminator:

- a. Terminator merupakan bagian/lingkungan luar sistem. Alur data yang menghubungkan terminator dengan berbagai proses sistem, menunjukkan hubungan sistem dengan dunia luar.
- b. Profesional Sistem Tidak berhak mengubah isi atau cara kerja organisasi atau prosedur yang berkaitan dengan terminator
- c. Hubungan yang ada antar terminator yang satu dengan yang lain tidak digambarkan pada DFD.

2. Komponen Proses

Komponen proses menggambarkan bagian dari sistem yang mentransformasikan input menjadi output. Proses diberi nama untuk menjelaskan proses/kegiatan apa yang



Gambar 2.5 Komponen Proses

sedang/akan dilaksanakan. Pemberian nama proses dilakukan dengan menggunakan kata kerja transitif (kata kerja yang membutuhkan obyek), seperti Menghitung Gaji, Mencetak KRS, Menghitung Jumlah SKS. Ada beberapa hal yang perlu diperhatikan tentang proses :

- a. Proses harus memiliki input dan output.

- b. Proses dapat dihubungkan dengan komponen terminator, data store atau proses melalui alur data.
- c. Sistem/bagian/divisi/departemen yang sedang dianalisis oleh profesional sistem digambarkan dengan komponen proses.

3. Penggambaran DFD

Tidak ada aturan baku untuk menggambarkan DFD. Tapi dari berbagai referensi yang ada, secara garis besar langkah untuk membuat DFD adalah:

1. Identifikasi terlebih dahulu semua entitas luar yang terlibat di sistem.
2. Identifikasi semua input dan output yang terlibat dengan entitas luar.
3. Buat Diagram Konteks (*diagram context*)

Diagram ini adalah diagram level tertinggi dari DFD yang menggambarkan hubungan sistem dengan lingkungan luarnya.

Caranya :

- a. Tentukan nama sistemnya.
 - b. Tentukan batasan sistemnya.
 - c. Tentukan terminator apa saja yang ada dalam sistem.
 - d. Tentukan apa yang diterima/diberikan terminator dari/ke sistem.
 - e. Gambarkan diagram konteks.
4. Buat Diagram *Level Zero*

Diagram ini adalah dekomposisi dari diagram konteks. Caranya:

- a. Tentukan proses utama yang ada pada sistem.
- b. Tentukan apa yang diberikan/diterima masing-masing proses ke/dari sistem sambil memperhatikan konsep keseimbangan (alur data yang keluar/masuk dari suatu level harus sama dengan alur data yang masuk/keluar pada level berikutnya).
- c. Apabila diperlukan, munculkan data store (master) sebagai sumber maupun tujuan alur data.
- d. Gambarkan diagram level zero.

5. Buat Diagram Level Satu

Diagram ini merupakan dekomposisi dari diagram level *zero*.

Caranya:

- a. Tentukan proses yang lebih kecil (sub-proses) dari proses utama yang ada di level *zero*.
- b. Tentukan apa yang diberikan/diterima masing-masing sub-proses ke/dari sistem dan perhatikan konsep keseimbangan.
- c. Apabila diperlukan, munculkan data store (transaksi) sebagai sumber maupun tujuan alur data.
- d. Gambarkan DFD level Satu

6. DFD Level Dua, Tiga, ...

Diagram ini merupakan dekomposisi dari level sebelumnya.

Proses dekomposisi dilakukan sampai dengan proses siap dituangkan ke dalam program. Aturan yang digunakan sama dengan level satu.

4. ERD(Entitiy Relationship Diagram)

Merupakan suatu model data yang dikembangkan berdasarkan objek.ER_M digunakan untuk menjelaskan hubungan antara data dalam basis data kepada pengguna secara logik. Sebuah diagram ER_D tersusun atas tiga komponen , yaitu entitas atribut, dan kerelasian antar entititas. Secara garis besar, entitas merupakan objek dasar yang terlibat dalam system. Atribut berperan sebagai penjelas entitas , sedangkan kerelasian menunjukan hubungan yang terjadi diantara dua entititas(Silberschatz,dkk,2002).

1. Entitas

Entitas merupakan objek – objek dasar yang terkait dalam system. Objek dasar dapat berupa orang, benda, atau hal yang keterangannya perlu disimpan didalam basis data. Untuk menggambarkan sebuah entitas digunakan aturan sebagai berikut (Sutanta,2004):

- a. Entitas dinyatakan dengan symbol persegi panjang
- b. Nama entitias dituliskan di dalam symbol persegi panjang
- c. Nama entitas berupa kata benda, tunggal
- d. Naman entitas dapat mungkin menggunkan nama yang mudah dipahami dan dapat menyatakan maknanya dengan jelas.

Seringkali nama entitas dapat tersusun atas lebih dari satu kata. Untuk memnuhi aturan penggambaran tersebut diatas maka sering digunakan tanda _(garis bawah) yang dimaksudkan untuk menyatakan bahwa beberapa kata tersebut dinggap sebagai kata

tunggal. **Catatan:** Penentuan entitas dalam suatu system perlu dilakukan dengan cermat dan hati –hati. Tidak semua orang,benda atau hal dapat disebut entitas. Hanya orang, benda, dan hal yang terkait dengan system dan keterangannya perlu disimpan dalam basis data yang dapat disebut entitas.

2. Kerelasiaan Antar Entitas

Mendefinisikan antara dua buah entitas. Kerelasiaan adalah kejadian atau interaksi yang terjadi diantara dua buah entitas yang keterangannya perlu disimpan dalam basis data(Martin,1975). Kejadian atau interaksi yang tidak perlu disimpan dalam basis data (Sekalipun benar – benar terjadi) bukan termasuk kerelasiaan. Aturan penggambaran kerelasiaan antar entitas adalah sebagai berikut(Sutanta,2004).

- a. Kerelasiaan dinyatakan dengan symbol belah ketupat
- b. Nama kerelasiaan dituliskan di dalam symbol belah ketupat
- c. Kerelasiaan menggambarkan atau menghubungkan dua entitas
- d. Nama kerelasiaan berupa kata kerja aktif (diawali dengan awalan me-), tunggal.
- e. Nama kerelasiaan sedapat mungkin menggunakan nama yang mudah dipahami dan dapat menyatakan maknanya dengan jelas.

3. Jenis Kerelasiaan Antar Entitas(*Relationship*)

Kerelasiaan antar entitas dikelompokkan dalam tiga jenis, yaitu (Silberschatz,2001):

a. Kerelasiaan jenis 1- ke- 1/satu ke satu(*one to one*)

Kerelasiaan jenis ini terjadi jika kejadian atau interaksi di antara dua entitas yang berhubungan hanya memungkinkan terjadi sebuah kejadian atau transaksi dua entitas.

b. Kerelasiaan jenis n- ke -1/banyak ke satu(*many to one*)

Kerelasiaan jenis ini terjadi jika kejadian atau transaksi di antara dua entitas yang berhubungan hanya terjadi satu kali dalam entitas pertama dan dapat terjadi lebih dari satu kali kejadian atau transaksi pada entitas kedua. Secara lebih teknis, jika nilai yang digunakan sebagai penghubung pada entitas pertama dimungkinkan muncul lebih dari satu kali pada entitas kedua yang berhubungan.

c. Kerelasiaan jenis n- ke -n/banyak ke banyak(*many to many*)

Kerelasiaan jenis ini terjadi jika kejadian atau transaksi di antara dua entitas yang berhubungan memungkinkan terjadi lebih dari satu kali dalam entitas pertama dan entitas kedua. Secara lebih teknis, jika nilai yang digunakan sebagai penghubung pada entitas pertama dimungkinkan muncul lebih dari satu kali, baik pada entitas pertama maupun pada entitas kedua yang saling berhubungan, dan sebaliknya.

4. Kunci Relasi

Kunci relasi diperlukan dalam rangka pengaksesan data dari dalam relasi atau untuk menyusun kerelasiaan antar relasi. Kunci relasi merupakan satu atau gabungan atribut yang bersifat unik yang dapat

digunakan untuk mengidentifikasi/membedakan setiap *record* dalam realasi. Berdasarkan macamnya, kunci relasi terdiri atas:

a. Kunci kandidat

Kunci kandidat adalah satu atau gabungan minimal atribut yang bersifat unik yang dapat digunakan untuk mengidentifikasi/membedakan setiap *record* dalam relasi. Dalam setiap sebuah kunci kandidat.

b. Kunci primer

Kunci primer adalah bagian dari CK yang tidak dipilih/digunakan sebagai PK. Dalam setiap relasi tidak harus mempunyai PK dan jumlahnya hanya satu buah. PK harus unik dan tidak boleh *null*.

c. Kunci alternative

Kunci alternative adalah bagian dari CK yang tidak dipilih/digunakan sebagai PK. Dalam setiap relasi tidak harus mempunyai PK. Hal ini tergantung pada jumlah CK yang ada. Jika jumlah dalam sebuah realasi lebih dari satu maka salah satu akan digunakan sebagai PK dan satu yang lainnya menjadi AK. Tetapi apabila sebuah relasi hanya memiliki sebuah CK maka ia akan digunakan sebagai PK dan tidak ada lagi AK.

d. Kunci penghubung

Kunci penghubung atau sering pula disebut sering pula disebut sebagai kunci tamu atau kunci asing adalah sebuah gabungan sembarang atribut menjadi PK dalam realasi lain yang

mempunyai hubungan secara logic. PK tidak harus dimiliki dalam sebuah relasi. Jika PK muncul dalam sebuah relasi maka PK tersebut akan menunjukkan adanya kerelasian antar relasi dalam basis data.

5. Atribut

Merupakan keterangan –keterangan yang terkait pada sebuah entitas yang perlu disimpan dalam basis data. Atribut berfungsi sebagai penjelas pada sebuah entitas(Sutanta,2004):

- a. Atribut dinyatakan dengan simbol elips
- b. Nama atribut dituliskan didalam symbol elips
- c. Nama atribut berupa kata benda, tunggal
- d. Nama atribut sedapat mungkin menggunakan nama yang mudah dipahami dan dapat menyatakan maknanya dengan jelas.
- e. Atribut dihubungnkan dengan entitas yang bersesuaian dengan menggunakan sebuah garis (sebagiannya menggunakan garis lurus , namun dalam kondisi yang tidak memungkinkan dapat juga tidak menggunakan garis lurus).

2.9 Metode Graph

Graf adalah suatu kumpulan simpul (*nodes*) yang dihubungkan satu sama lain melalui sisi atau busur (*edges*) (Zakaria, 2006). Secara informal, suatu graf adalah himpunan benda-benda yang disebut *verteks* atau *node* yang terhubung oleh *edge edge*. Biasanya graf digambarkan sebagai kumpulan titik-titik (melambangkan *verteks*) yang dihubungkan oleh garis-garis (melambangkan *edge-edge*).

2.10 Algoritma Sistem

2.10.1 Algoritma Dijkstra

Algoritma Dijkstra merupakan salah satu varian dari algoritma *Greedy*, yaitu salah satu bentuk algoritma populer dalam pemecahan persoalan yang terkait dengan masalah optimasi. Sifatnya sederhana dan lempang (*straight forward*). Sesuai dengan artinya yang secara harafiah berarti tamak atau rakus (namun tidak dalam konteks negatif), algoritma *Greedy* ini hanya memikirkan solusi terbaik yang akan diambil pada setiap langkah tanpa memikirkan konsekuensi ke depan.

Intinya algoritma *Greedy* ini berupaya membuat pilihan nilai optimum lokal pada setiap langkah dan berharap agar nilai optimum local ini mengarah kepada nilai optimum global. Penggunaan strategi *Greedy* pada algoritma Dijkstra adalah pada setiap langkah, ambil sisi berbobot minimum yang menghubungkan sebuah simpul yang sudah terpilih dengan sebuah simpul lain yang belum terpilih. Lintasan dari simpul asal ke simpul yang baru haruslah merupakan lintasan yang terpendek di antara semua lintasannya ke simpul simpul yang belum terpilih (Novandi, 2007).

2.11. HTML 5

HTML(*Hypertext Markup language*) dikembangkan pertama kali pada tahun 1989 oleh Tim Barnes-Lee. Bahasa skrip ini memungkinkan penyajian informasi dalam bentuk teks dengan tautan yang bisa mengaitkan berbagai server. Aplikasi – aplikasi *web* yang kita nikmati sekarang berkat HTML tersebut.

HTML5 (angka 5 dan HTML tidak dipisahkan oleh spasi) dikembangkan oleh badan lain yaitu WHATMG (*Web Hypertext Application Technology*

Working Group). Spesifikasi HTML5 belum final, tetapi diyakini oleh banyak orang akan menjadi standar yang diterima dimasa depan. Berikut contoh dokumen yang berbasis HTML5.

```
<!DOCTYPE HTML>
<html>
  <head>
    <title> Tes HTML5 </title>
  </head>
  <body>
    Selamat belajar HTML5
  </body>
</html>
```

Gambar 2.7 Contoh Dokumen HTML5

(Abdul Kadir ,2013:4)

2.12 Pengenalan PHP

PHP merupakan bahasa pemrograman yang disebut sebagai bahasa pemrograman *scripting*, dalam arti PHP merupakan bahasa pemrograman yang ditempelkan/*embedded* pada bahasa atau aplikasi lain. Sebagai contoh, PHP ditempelkan ke dalam *script* HTML, yang merupakan bahasa ibu untuk *world wide web* (Stendy B. Sakur,2010). Berikut ini contoh hal yang berhubungan dengan cara meletakkan dan menuliskan *script* PHP.

2.12.1 Menggunakan Semi Kolon

Setiap akhir pernyataan untuk program PHP harus menggunakan tanda semi kolon yaitu titik koma “;”. Hal ini menunjukkan bahwa baris di atasnya tidak ada hubungan dengan baris berikutnya dalam proses interpreter. Perhatikan *listing* berikut:

```
<?php  
Echo "Hello World";  
Echo "<br> Saya dari Script php </br>";  
?>  
Output :  
Hello World  
Saya dari Script php
```

Gambar 2.8 Contoh Script php

(Stendy B. Sakur,2010:8)

Sekalipun PHP tidak memberikan aturan yang tegas akan hal ini namun sebaiknya Anda tetap menggunakan pengkodean untuk akhir dari pernyataan ini agar script mudah dipahami dan mencegah dari terjadinya kesalahan interpreter.



BAB III

ANALISIS DAN DESAIN

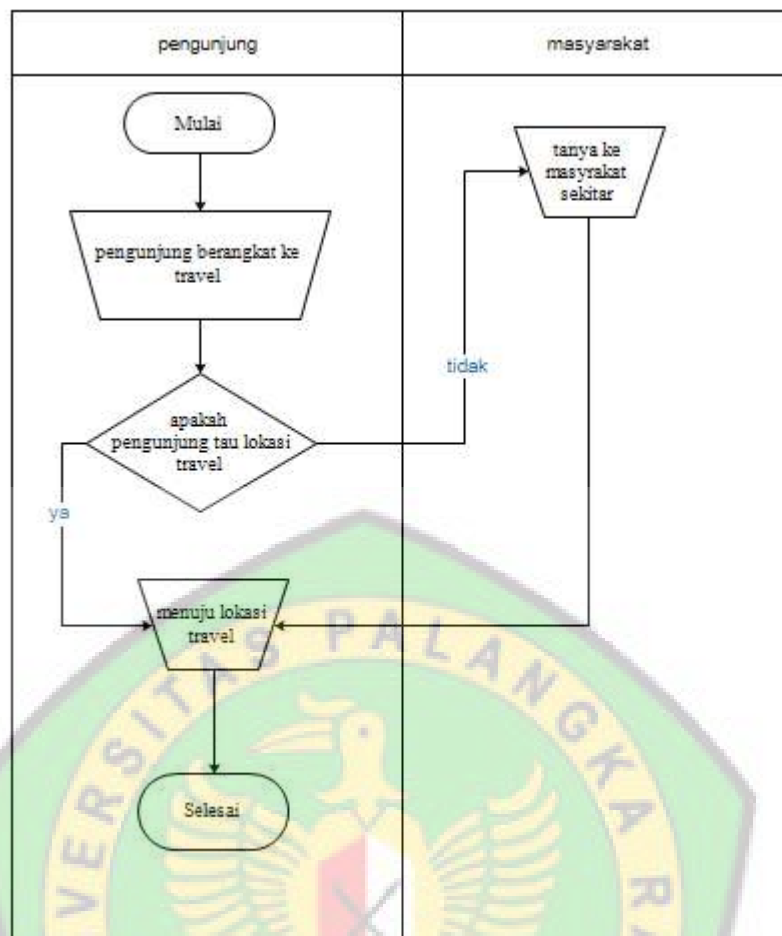
3.1 Analisis

Pada *website* sistem informasi geografis mencari rute menuju lokasi travel di kota palangkarya, terdapat 2 pengguna yang terlibat didalam sistem yaitu administrator dan pengunjung *web*. Kegiatan Admin di dalam *website* yaitu *Login* ke dalam *website*, mengelola jalur, mengelola lokasi travel, mengelola map. Sedangkan kegiatan pengunjung didalam *website*, Melihat map, mencari lokasi travel, melihat informasi rute yang akan dituju.

3.1.1 Bisnis Proses Sistem Lama

1. Pengunjung berangkat ke tempat tujuan.
2. Pengunjung mencari travel yang hendak dituju.
3. Jika pengunjung tidak tau tempat travel yang dituju maka pengunjung bertanya pada masyarakat sekitar, jika pengunjung sudah tau tempat yang dituju, Pengunjung langsung ke tempat tujuan.
4. Masyarakat memberitahu alamat kepada pengunjung.
5. Jika travel yang dituju masih tutup, pengunjung harus menunggu buka terlebih dahulu dan, jika tidak pengunjung pulang.

Untuk memperjelas alur kegiatan sistem pada deskripsi diatas, dibawah ini merupakan *flowchart* dari sistem lama tersebut.



Gambar 3.1 Flowchart Sistem Lama

Kesimpulan Sistem Lama

1. Kelemahan Sistem Lama

- a. Tidak ada gambabaran data spasial.
- b. Pengunjung mencari sendiri lokasi travel yang akan dituju.

2. Rekomendasi Sistem Baru

Dari kelemahan – kelemahan yang terdapat pada sistem lama diatas, didapatkan usulan rekomendasi sistem baru untuk merancang Sistem Informasi Geografis Mencari Rute Menuju lokasi Travel Di Kota Palangkaraya. Munculnya ide untuk mengusulkan rekomendasi sistem baru ini adalah supaya informasi tidak hanya banyak diketahui oleh masyarakat yang

tinggal di palangkaraya melainkan juga harus diketahui oleh masyarakat luas.

Adapun rekomendasi fitur yang terdapat pada website ini antara lain:

Fitur Administrator

a. Login Admin

pada bagian ini hanya digunakan oleh administrator, *login* admin berfungsi untuk membuat administrator bisa masuk kedalam *website* dan nantinya admin akan mengelola fitur – fitur yang ada didalam *website*, baik itu menambah, mengubah dan menghapus fitur yang ada pada *website*.

b. Logout Admin

Merupakan fungsi untuk Admin dapat keluar dari *website*.

a. Gmap

Memuat sistem untuk menambah, mengedit dan menghapus data jalur yang di kelola oleh admin.

c. Travel

Memuat sistem untuk menambah, mengedit dan menghapus data travel yang di kelola oleh admin.

Fitur Halaman Pengunjung

a. Beranda

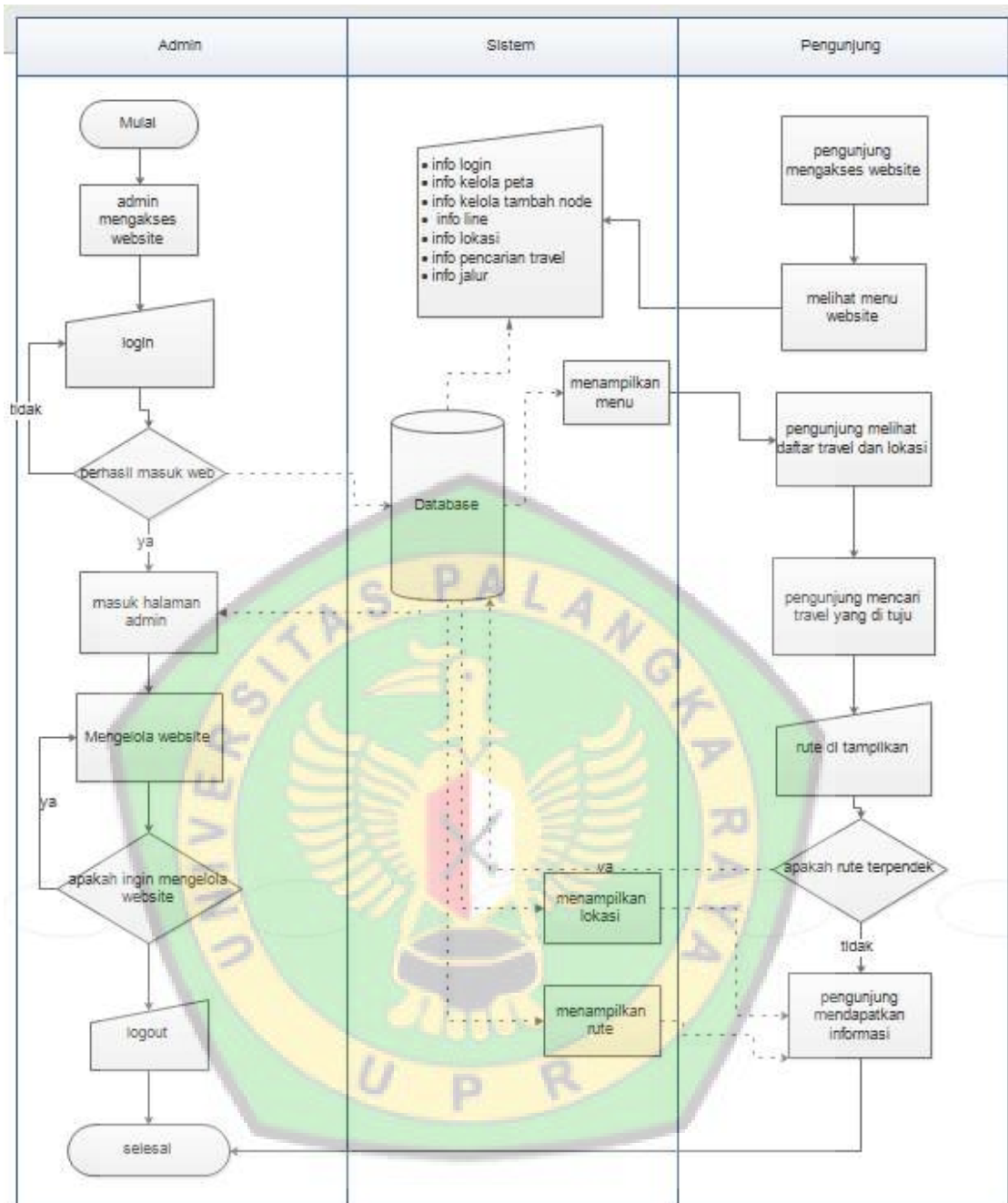
Berisi content map dan fitur pencarian, daftar nama-nama travel yang ada serta lokasi travel.

3.1.2 Bisnis Proses Sistem Baru

1. Admin mengakses *website*

2. Admin login kedalam *website*
3. Admin mengelola informasi yang terdapat di dalam website.
4. Pengunjung mengakses *website*
5. Pengunjung melihat menu yang terdapat pada *website*.
6. Pengunjung mencari lokasi travel yang ingin dituju.
7. Pengunjung melihat daftar nama travel yang terdapat pada menu *map*.
8. Pengunjung mencari rute dengan mengklik posisi awal pada peta.
9. Pengunjung mencari travel yang akan di tuju.
10. Sistem menampilkan informasi jalur pada peta.
11. Pengunjung mendapatkan Informasi tempat yang akan dituju.
12. Jika ada informasi yang masih di cari pengujung tetap di dalam *website*, jika tidak ada maka pengujung keluar.
13. Jika tidak ada informasi yang di perbaharui Admin logout, jika ada Admin tetap di salam sistem.
14. Admin keluar dari *Website*.

Dibawah ini merupakan *flowchart* dari sistem baru, untuk memperjelas proses yang dilakukan:



Gambar 3.2 Flowchar Sistem Baru

Kesimpulan Sistem Baru

1. Pengguna yang terlibat didalam sistem

- a. Adminp
- b. Pengunjung

2. Kegiatan Admin

- a. Login.
- b. Kelola Peta.
- c. Kelola travel
- d. Kelola menu membuat jalur.
- e. Logout.

3. Kegiatan Pengunjung

- a. Melihat menu beranda.
- b. Melihat daftar travel
- c. Melihat *map*.
- d. Mencari lokasi yang dituju.
- e. Melihat jalur yang ingin tujuan

3.2 Design

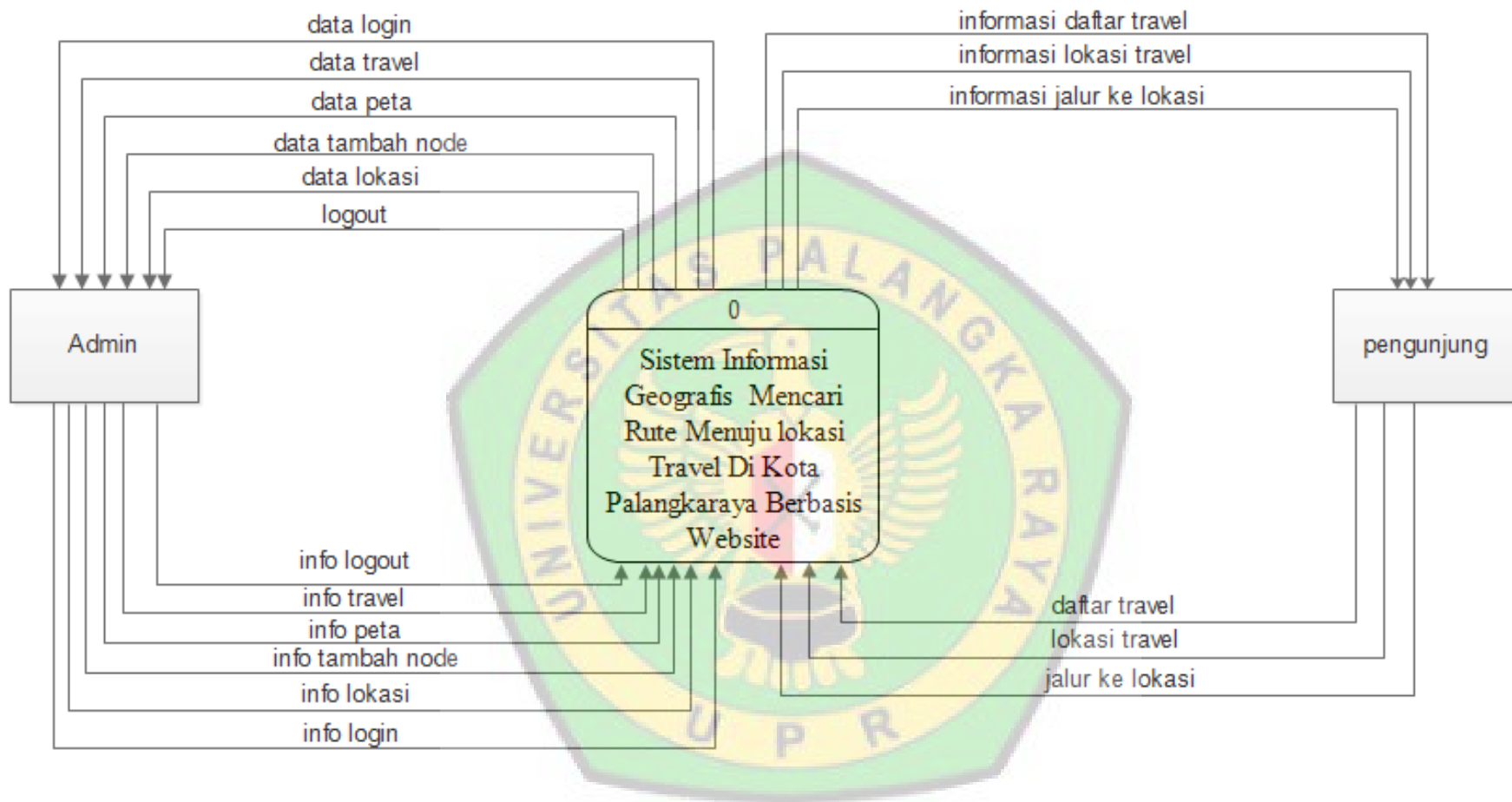
3.2.1 DFD (*Data flow Diagram*)

Pada sistem arsitektur yang digunakan pada *website* “Sistem Informasi Geografis Mencari Rute Menuju lokasi Travel Di Kota Palangkaraya Berbasis Website” menggunakan *Data Flow Diagram* (DFD) adalah alat pembuatan model yang memungkinkan profesional sistem untuk menggambarkan sistem sebagai suatu jaringan proses fungsional yang dihubungkan satu sama lain dengan alur data, baik secara manual maupun komputerisasi. Dibawah merupakan proses visualisasi menggunakan DFD:

1. Diagram konteks / DFD 0

Diagram ini adalah diagram level tertinggi dari DFD yang menggambarkan hubungan sistem dengan lingkungan luarnya. Berikut penggambaran Diagram Konteks pada *website*.





Gambar 3.3 DFD Level 0 / Diagram Konteks

2. Tabel Definisi Diagram Konteks

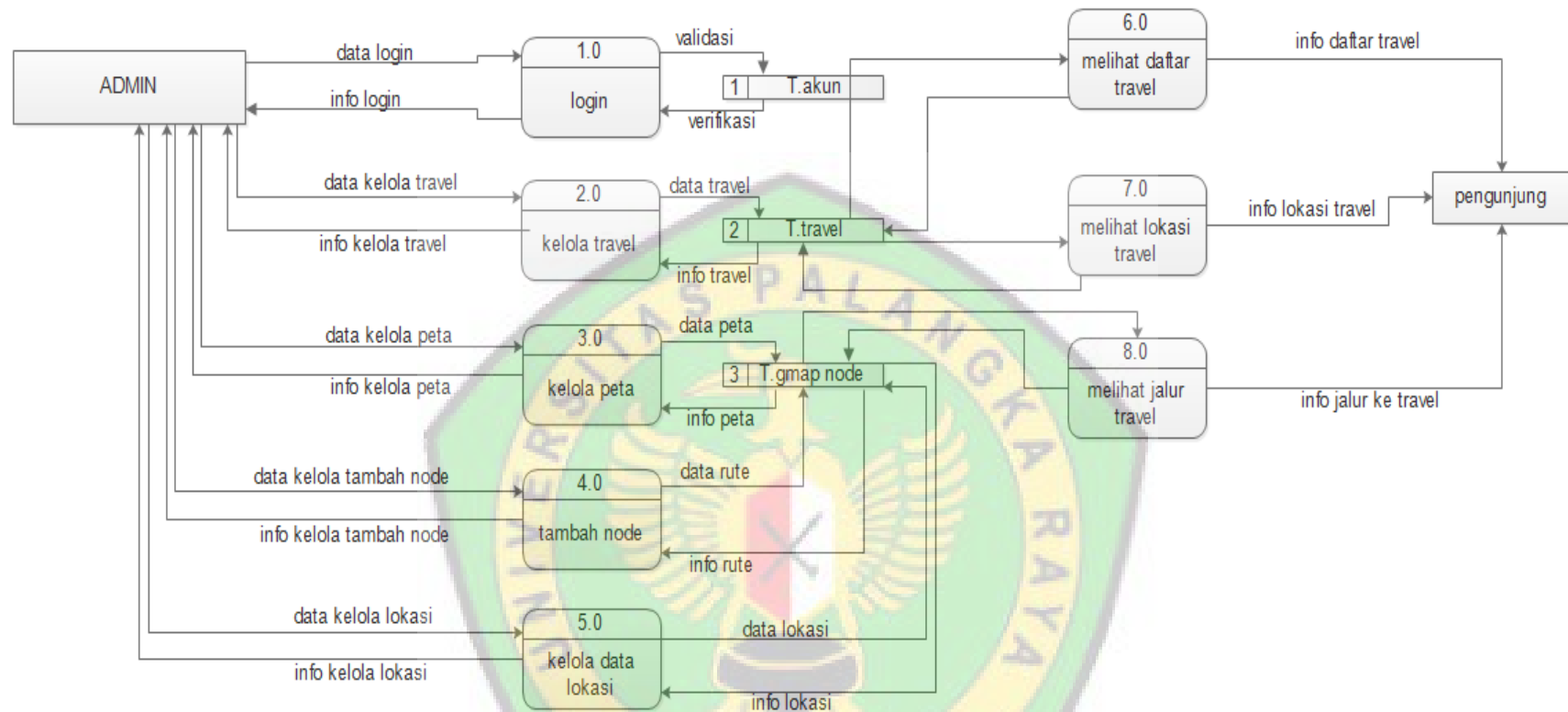
Tabel 3.1 Definisi Diagram Konteks

No	Entitas Luar	Keterangan
1	Admin	<p>1.Penjelasan</p> <p>Merupakan orang yang bertugas untuk mengelola semua fitur – fitur yang ada pada website.</p> <p>2.Input</p> <ul style="list-style-type: none">• Data login.• Data kelola travel.• Data kelola peta.• Data tambah <i>node</i>.• Data lokasi• Logout. <p>3.Output</p> <ul style="list-style-type: none">• Info login.• Info kelola daftar travel• Info kelola peta.• Info kelola tambah node.• line.• Info lokasi• Info logout

No	Entitias Luar	Keterangan
2	Pengunjung	<p>1.Penjelasan</p> <p>Merupakan orang yang sedang mengakses <i>website</i>, dan mencari informasi yang diperlukan pada fitur yang sudah disediakan di<i>website</i>.</p> <p>1. Input</p> <ul style="list-style-type: none"> • Melihat jalur menuju lokasi. • Mencari informasi lokasi. • Melihat daftar travel <p>1.output</p> <ul style="list-style-type: none"> • Info jalur travel. • Info pencarian lokasi. • Informasi daftar travel.

3. DFD Level 1

Pada bagian ini merupakan proses untuk mevisualisasi dari penguraian dekomposisi diagram konteks. Diagram ini menggambarkan proses aliran data input/output dari sebuah sistem yang dibangun. Dibawah ini merupakan penggambaran DFD level 1:

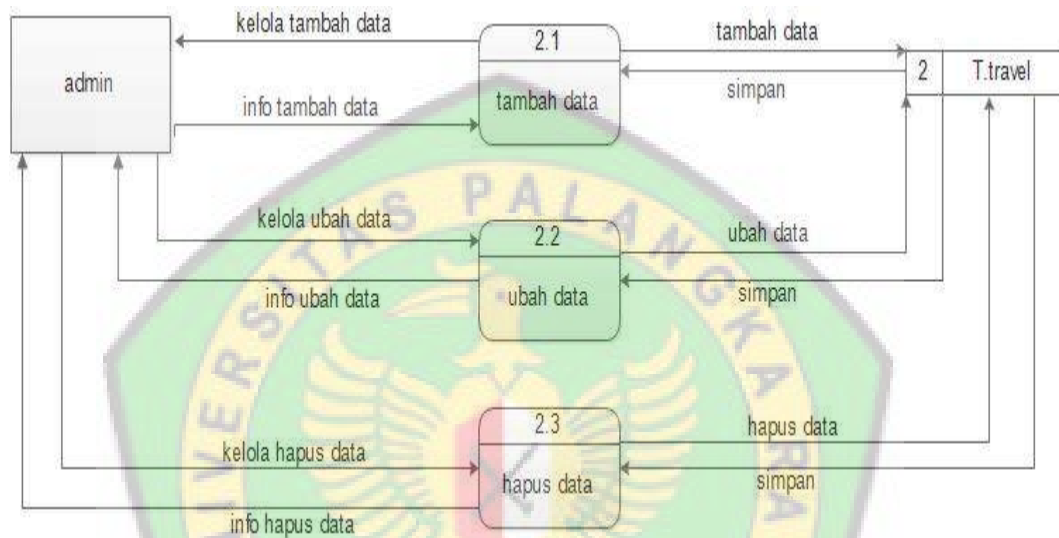


Gambar 3.4 DFD Level 1

4. DFD Level 2

Pada bagian ini merupakan proses penggambaran yang dijabarkan dari diagram DFD level 1 sebelumnya. Berikut merupakan penggambaran dari DFD level 2:

a. DFD level 2 untuk daftar travel

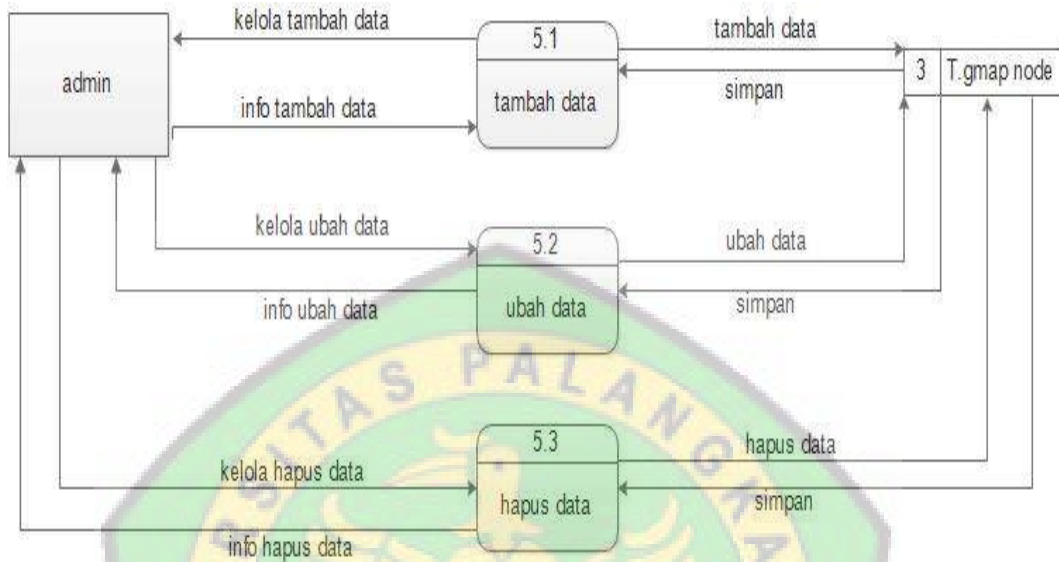


Gambar 3.5 DFD level 2 Pada Travel

Pada gambar DFD level 2 untuk menu beranda diatas terdapat 3 proses yang dilakukan untuk mengelola tabel “travel” yaitu tambah data, ubah data dan hapus data. Untuk proses tambah data mempunyai inputan kelola tambah data dan output dari proses tersebut adalah data beranda setelah data tersebut sudah di tambah nantinya data tersebut akan disimpan dan output terakhir dari proses ini adalah informasi tambah data. Ubah data admin mempunyai inputan ubah data dan output dari proses tersebut adalah data setelah data tersebut sudah diedit nantinya data tersebut akan disimpan dan output terakhir dari proses ini adalah informasi ubah data.

Dan terakhir untuk proses hapus data mempunyai inputan hapus data dan output dari proses tersebut adalah info hapus data.

b. DFD level 2 untuk lokasi travel



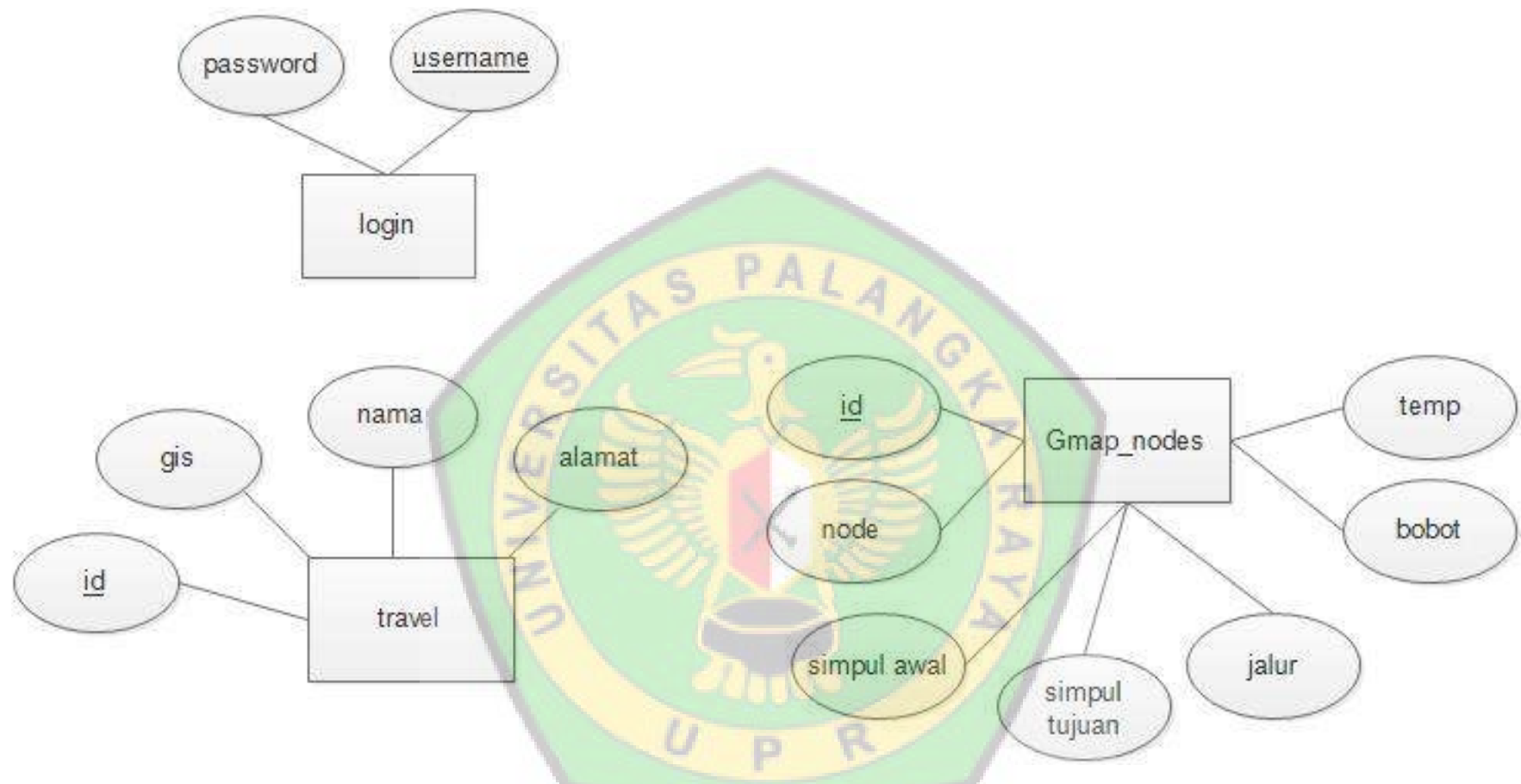
Gambar 3.6 DFD level 2 Pada Lokasi

Pada gambar DFD level 2 untuk menu kategori diatas terdapat 3 proses yang dilakukan untuk mengelola tabel “gmap node” yaitu tambah data, ubah data dan hapus data. Untuk proses tambah data mempunyai inputan kelola tambah data dan output dari proses tersebut adalah data lokasi setelah data tersebut sudah di tambah nantinya data tersebut akan disimpan dan output terakhir dari proses ini adalah informasi tambah data. Proses ubah data admin mempunyai inputan ubah data dan output dari proses tersebut adalah data setelah data tersebut sudah diedit nantinya data tersebut akan disimpan dan output terakhir dari proses ini adalah informasi ubah data. Dan terakhir untuk proses hapus data mempunyai inputan hapus data dan output dari proses tersebut adalah info hapus data.

3.2.2. ERD(*Entity Relationship Diagram*)

Entity relationship diagram (ERD) adalah suatu bentuk perencanaan database secara konsep fisik yang nantinya akan dipakai sebagai kerangka kerja dan pedoman dari struktur penyimpanan data. Dibawah ini merupakan desain dari ERD pada *website* Sistem Informasi Geografis Mencari Rute Menuju lokasi Travel Di Kota Palangkaraya Berbasis.





Gambar 3.7 ERD(Entity Relationship Diagram)

3.2.3. Desain Tabel

Bagian ini merupakan kamus data dari setiap tabel yang telah dirancang didalam ERD (*Entitiy Relationship Diagram*). Terdapat 8 tabel pada *website* yang nantinya desain tabel akan dirancang menggunakan DBMS MySQL dan PHP(*Hypertext Preprocesor*). Kamus data berfungsi untuk mendefinisikan suatu daftar data elemen yang terorganisir dengan definisi yang tetap dan sesuai dengan sistem, sehingga user dan analis sistem mempunyai pengertian yang sama tentang input, output dan komponen data store.

1. Tabel *Login*

Tabel 3.2 Tabel *Login*

No	Nama Field	Tipe Data	Lebar	Keterangan
1	Username	varchar	8	Primary key
2	Password	varchar	50	Null

2. Tabel *Travel*

Tabel 3.3 Tabel *Travel*

No	Nama Field	Tipe Data	Lebar	Keterangan
1	Id	int	11	Primary key
2	gis	varchar	150	Null
3	nama	text	-	Null
4	alamat	varchar	255	Null

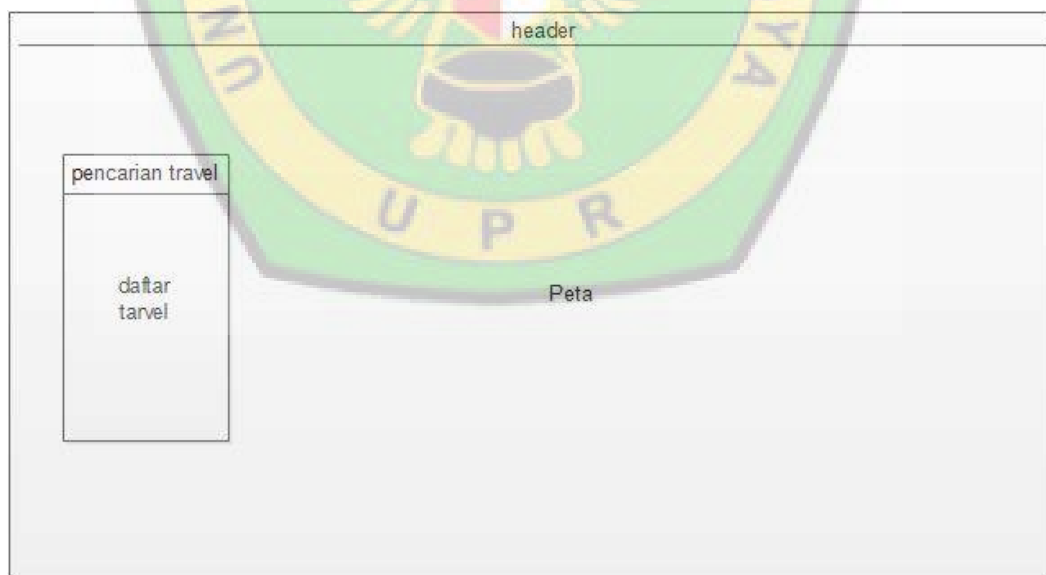
3. Tabel *Gmap_nodes*

4. Tabel 3.4 Tabel *Graph*

No	Nama Field	Tipe Data	Lebar	Keterangan
1	id	int	11	Primary key
2	node	varchar	20	Null
3	Simpul_awal	varchar	10	Null
4	Simpul_tujuan	varchar	10	Null
5	jalur	text	-	Null
6	bobot	double	-	Null
7	temp	enum	'N','Y'	Null

3.2.4. Desain Struktur dan Interaksi Sistem untuk Halaman Pengunjung

1. Tampilan Menu Utama



Gambar 3.8 Tampilan Utama Pengunjung

3.2.5. Desain Struktur dan Interaksi Sistem untuk Halaman Admin

1. Halaman Akun

header

Akun

Gmap

Travel

username

password

level

Tambah data

reset

Gambar 3.9 Tampilan Akun Admin

2. Halaman Kelola Rute

header

Akun

Gmap

Travel

Rute

Tools

Node

Line

Save

Clear

Peta

Gambar 3.10 Tampilan Rute Admin

3. Halaman Kelola Travel

header

Akun
Gmap
Travel

Nama Travel

Lokasi Peta

Alamat

Tambah data reset

Gambar 3.11 Tampilan Travel Admin

3.2.6 Navigation Desain

3.2.6.1 Halaman Admin



Gambar 3.12 Desain Navigasi Halaman Admin

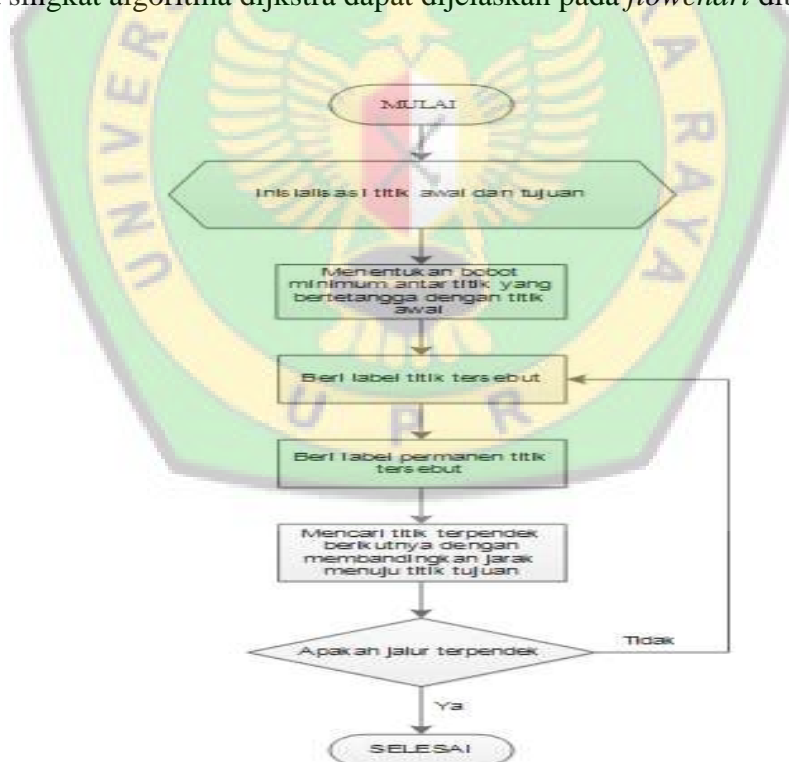
3.2.6.2 Halaman Pengunjung



Gambar 3.13 Desain Navigasi Halaman Pengunjung

3.3 Proses Algoritma Dijkstra

Secara singkat algoritma dijkstra dapat dijelaskan pada *flowchart* dibawah ini:



Gambar 3.14 *flowchart* algoritma dijkstra

BAB IV

IMPLEMENTASI DAN TESTING

4.1 IMPLEMENTASI PROSES

Implementasi proses adalah pengembangan dari perancangan proses. Penjelasan untuk implementasi proses ini menggunakan *Data Flow Diagram Level I* yang sebelumnya sudah dirancang pada BAB III. Proses – proses yang penting dalam *website* ini adalah sebagai berikut:

- a. Admin melakukan login kedalam *website* pada halaman *login* dengan memasukkan *username* dan *password*, yang dimiliki oleh admin yang mengelola *website*.
- b. Di halaman utama admin, ada beberapa kegiatan yang dapat dilakukan oleh admin tersebut. Aktivitas tersebut adalah kelola akun, kelola rute, dan kelola travel.
- c. Untuk pengunjung melihat fitur-fitur yang disediakan *website* yaitu, melihat menu utama untuk mengetahui letak lokasi travel yang ada di kota palangkaraya.

4.2 IMPLEMENTASI DATA

Data diimplementasikan dengan menggunakan *database phpmyadmin*, dimana *phpmyadmin* adalah suatu aplikasi yang dibuat menggunakan bahasa pemrograman php yang ditujukan untuk pengelolaan basis data MySQL melalui internet. Fungsi *phpmyadmin* adalah untuk merancang database. Implementasi data dibagi menjadi dua, yaitu untuk *server* basis data dan untuk basis data *website*. basis data yang ada di *website* adalah replikasi dari tabel - tabel yang ada di *server* basis data.

4.3 PENGUJIAN WEBSITE

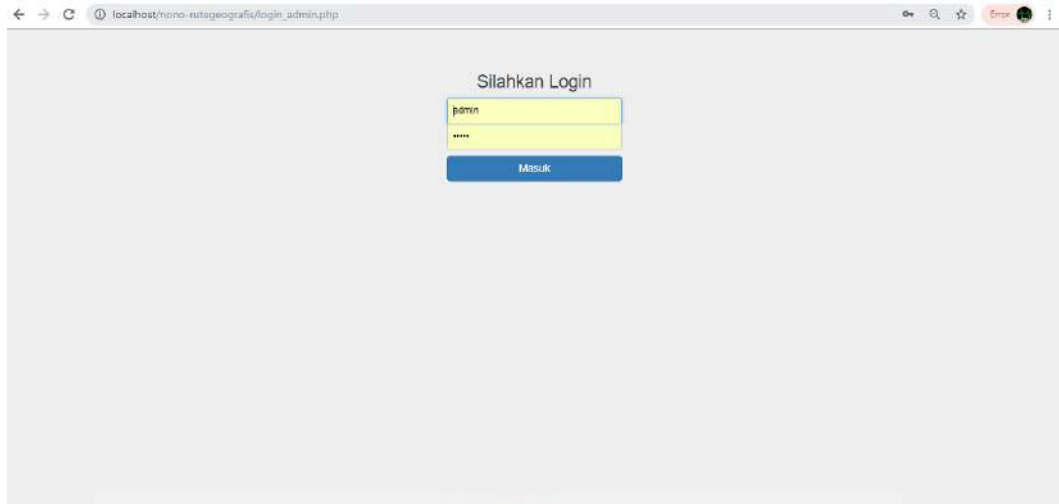
Pengujian website “Sistem Informasi Geografis Mencari Rute Menuju lokasi Travel Di Kota Palangkaraya Berbasis Website” dilakukan dengan dengan *black box testing* yang didalamnya termasuk *browser testing*. Metode uji coba menggunakan *blackbox testing* yang memfokuskan pada keperluan fungsional dari suatu aplikasi. Karena itu uji coba *blackbox* memungkinkan pengembang *software* untuk membuat himpunan kondisi input yang akan melatih seluruh syarat – syarat fungsional suatu program. Untuk pengujian *browser* ada 2 tahap yang akan di jelaskan pertama kegiatan Administrator di dalam *website* dan kedua implementasi antarmuka halaman pengunjung. Dan terakhir akan dijelaskan tentang *blackbox testing*.

4.3.1 KEGIATAN ADMINISTRATOR DI DALAM WEBSITE

Admin mempunyai hak akses penuh untuk mengelola seluruh data yang ada di dalam *website*. Untuk melakukan proses pengelolaan terhadap konten-konten yang ada di dalam *website*, terlebih dahulu admin harus melakukan proses login ke dalam halaman admin dengan memasukkan *username* dan *password*, setelah masuk kedalam *web* Admin mengelola semua fitur yang ada di dalam sistem.

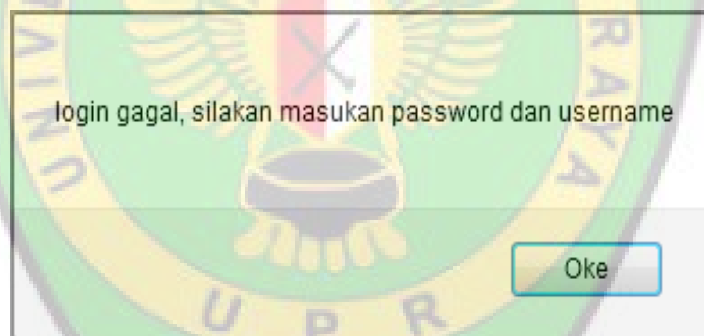
1. Antar Muka Halaman Login Admin

Dibawah ini merupakan tampilan halaman login sebelum Admin memasukan *username* dan *password*. Fungsi Admin memasukan *username* dan *password* adalah untuk bisa mengakses dan mengelola fitur – fitur yang terdapat pada *website*.



Gambar 4.1 Belum Memasukan *Username* dan *password*

Gambar diatas merupakan proses untuk meinputkan *username* dan *password* ke kolom inputan, jika Admin belum menginputkan maka akan muncul pesan "login gagal, silakan masukan *password* dan *username*". Berikut tampilan dari peringatan tersebut.



Gambar 4.2 validasi belum memasukan *password* dan *username*

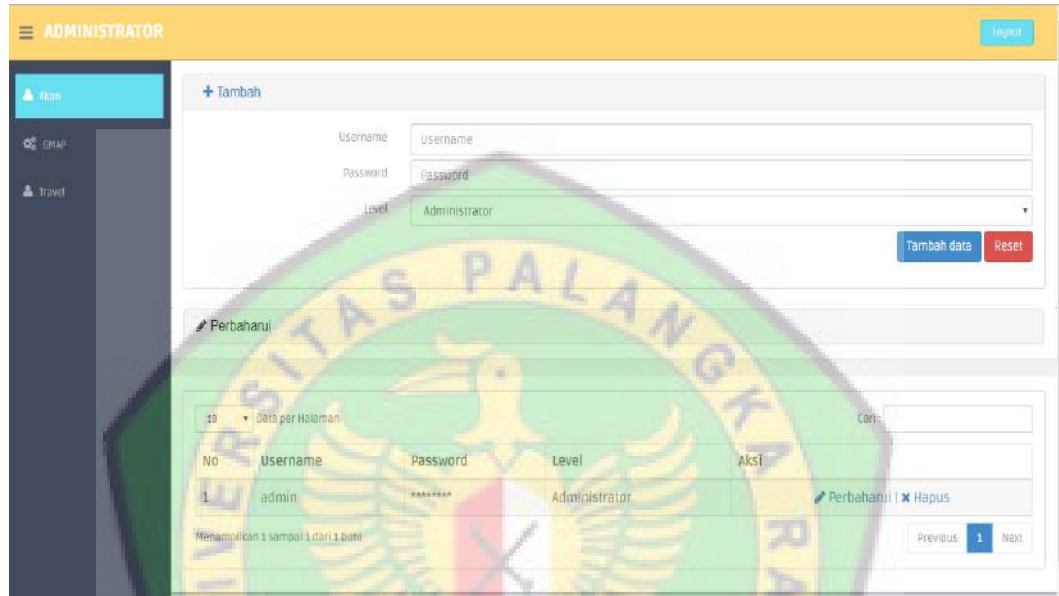
Dan apabila user menginputkan *username* atau *password* yang tidak tepat maka sistem akan membuat pesan "Maaf, Username atau password Salah". Berikut tampilan *capture* program.



Gambar 4.3 Validasi *username* atau *password* salah

2. Antarmuka Halaman Kelola Akun

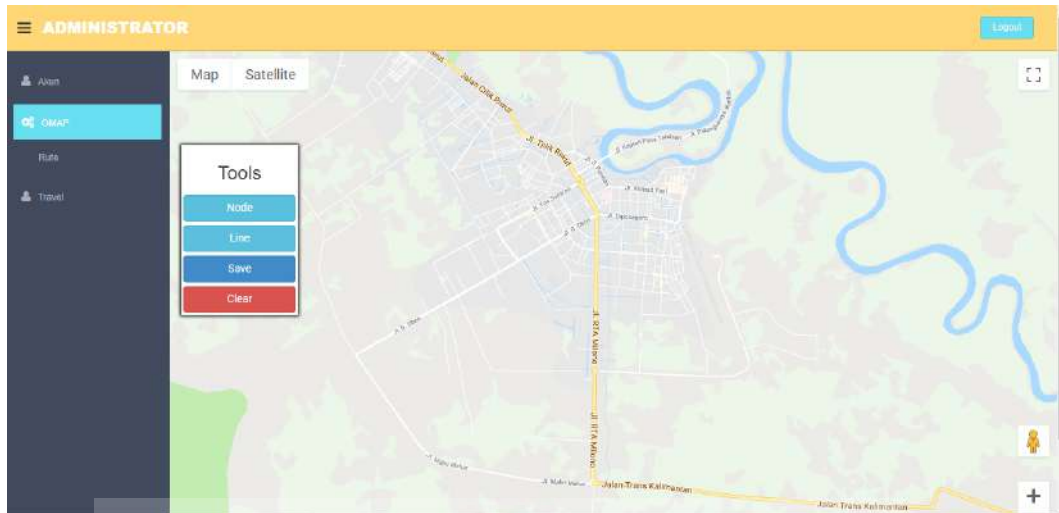
Tampilan dibawah ini merupakan implementasi dari halaman kelola akun, pada bagian halaman kelola akun ini admin dapat menambah , mengubah atau menghapus data yang. Berikut tampilan dari halaman kelola akun.



Gambar 4.4 antarmuka halaman kelola akun

3. Antarmuka Halaman Kelola Rute

Pada halaman ini mempunyai fungsi mengelola atau membuat graf untuk menuju lokasi travel dan menambahkan node pada peta. Berikut ini tampilan halaman kelola rute.

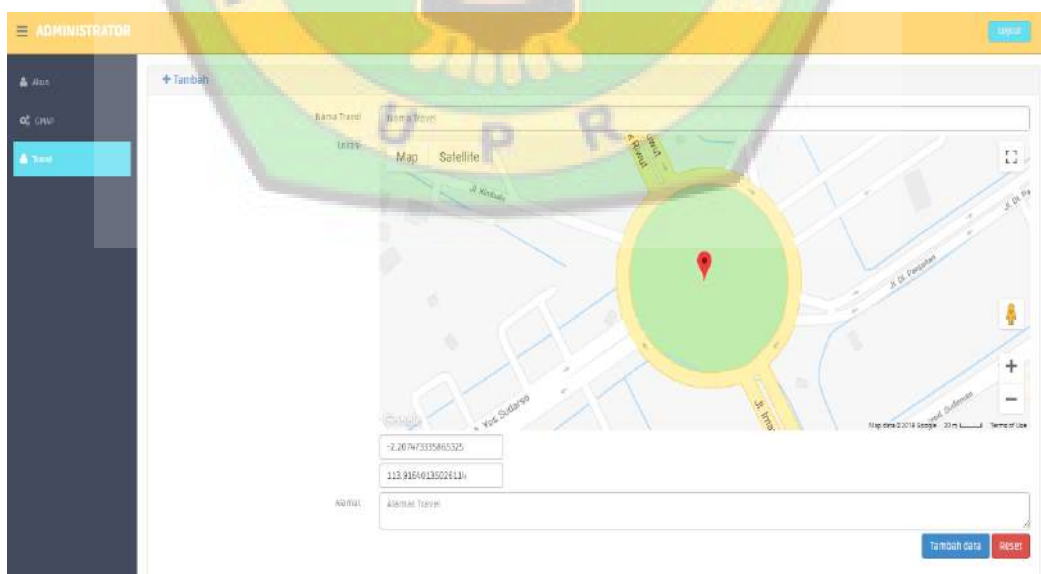


Gambar 4.5 antarmuka halaman kelola rute

4. Antarmuka Halaman Kelola travel

Selanjutnya untuk halaman kelola travel. Halaman ini mempunyai fungsi menampilkan nama lokasinya. Pada halaman ini admin dapat menambah, mengedit dan juga menghapus data yang diperlukan. Berikut tampilan dari halaman tambah data.

a. Antarmuka Halaman Tambah Data Pada Kelola Peta



Gambar 4.6 antarmuka halaman kelola travel

Pada halaman kelola peta jika admin belum mengisi data pada *field*, maka pada setiap kolom akan muncul pesan “isi isian ini”. Berikut tampilan dari capture program.

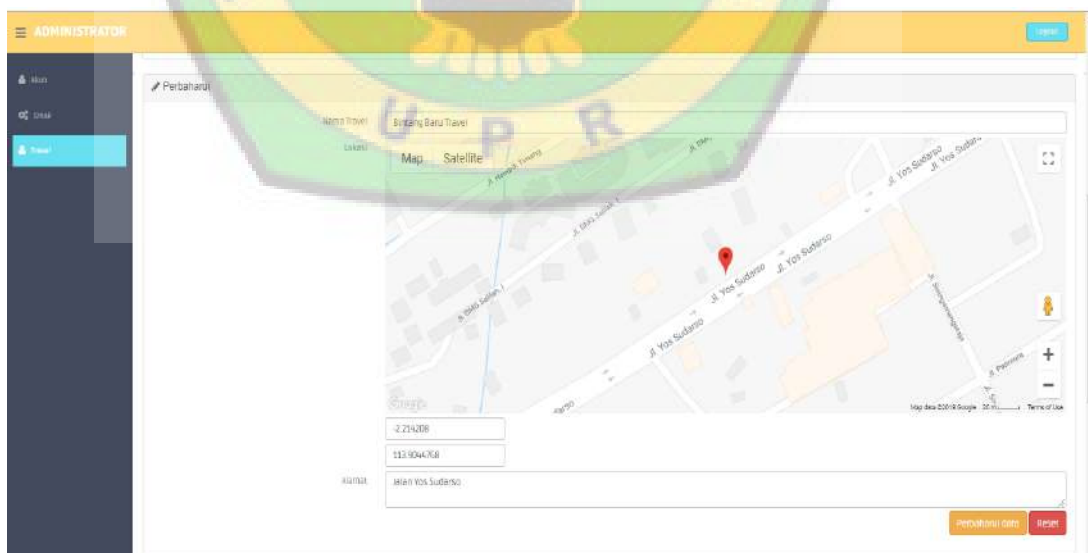
Apabila administrator sudah menginputkan semua data kedalam kolom inputan data, dan setelah admin menekan tombol simpan maka akan muncul pemberitahuan bahwa data berhasil disimpan kedalam *database*. Berikut tampilan dari capture program.

Travel telah ditambahkan.

Gambar 4.7 validasi data berhasil pada halaman kelola travel

b. Antarmuka Halaman Edit Data Pada Kelola Peta

dibawah ini merupakan antarmuka untuk admin mengubah data, jika ada data yang ingin diubah atau diperbaiki. Berikut tampilan dari halaman edit data pada menu kelola peta.



Gambar 4.8 antarmuka halaman edit data kelola Travel

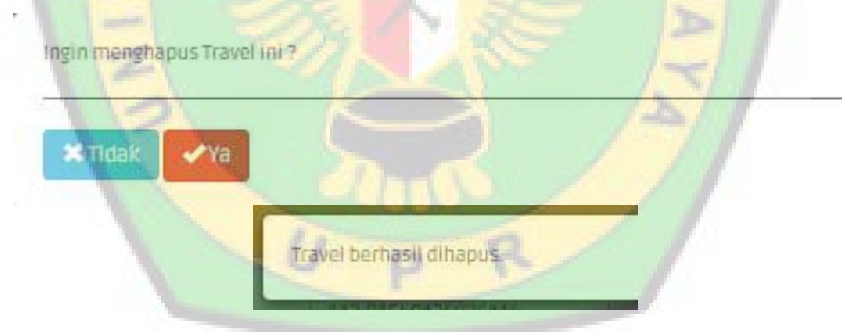
sama seperti halnya tambah data, jika admin belum lengkap mengisi data maka akan muncul pesan “isi isian ini”, dan setelah admin sudah melengkapi seluruh data yang akan di simpan maka setelah admin menekan tombol edit maka data akan langsung tersimpan. Berikut tampilan dari capture program.



Gambar 4.9 validasi data berhasil di simpan pada halaman kelola travel

c. Hapus Data Pada Kelola Peta

Pada menu kelola peta selain menambah dan mengedit data, admin juga bisa menghapus data jika pada data yang akan ditampilkan nantinya tidak di perlukan lagi. Berikut tampilan pesan jika admin sudah menghapus data.



Gambar 4.10 validasi data berhasil di hapus pada halaman kelola travel

4.3.2 KEGIATAN PENGUNJUNG DI DALAM WEBSITE

1. Implementasi Halaman Beranda

Menu utama merupakan menu yang berisi tentang aktivitas yang akan di lakukan pengunjung yang ada di website. Pada meunu ini juga terdapat fitur googlemap yang berisi sistem informasi geografis letak travel yang ada di kota Palangka Raya.



Gambar 4.11 antarmuka halaman menu utama

4. Implementasi Pencarian Lokasi

Halaman dibawah ini mengimplementasikan ketika pengunjung ingin mencari lokasi travel dengan nama jalan atau pun dengan nama travel. Berikut ini tampilan dari capture program.



Gambar 4.12 antarmuka halaman pencarian lokasi

1. Login Admin

Pengujian Halaman untuk login admin dapat dilihat pada table 4.1 berikut. Tanda (√) merupakan tanda yang berarti halaman yang di uji berhasil.

Tabel 4.1 Blackbox testing Login Admin

No	Kondisi Awal	Hal yang dilakukan	Kondisi Akhir	Hasil Test
1	Admin melakukan login	Mengisi username yang salah dan password benar	Login gagal	√
2	Admin melakukan login kembali	Mengisi username yang benar dan password salah	Login gagal	√
3	Admin melakukan login kembali	Mengisi username yang benar dan password benar	Login berhasil dan masuk ke halaman utama admin (dashboard)	√

2. Mengelola Admin

Pengujian Halaman untuk mengelola Admin dapat dilihat pada table 4.2 berikut. Tanda (√) merupakan tanda yang berarti halaman yang di uji berhasil

Tabel 4.2 Blackbox testing Mengelola Hubungi Kami

No	Kondisi Awal	Hal Yang Dilakukan	Kondisi Akhir	Hasil Test
1	Admin mengklik Travel menu admin	Admin dibawa ke halaman mengelola admin	Menampilkan daftar Admin	√

No	Kondisi Awal	Hal Yang Dilakukan	Kondisi Akhir	Hasil Test
2	Admin melakukan hapus data Travel	Admin mengklik tombol hapus untuk menghapus Travel	Admin berhasil menghapus	√

3. Edit Admin

Pengujian Halaman edit admin dapat dilihat pada table 4.3 berikut. Tanda (√) merupakan tanda yang berarti halaman yang di uji berhasil

Tabel 4.3 Blackbox testing edit admin

No	Kondisi Awal	Hal yang dilakukan	Kondisi Akhir	Hasil Test
1	Admin mengklik edit admin pada menu admin	Admin dibawa ke halaman edit admin	Menampilkan halaman edit admin	√
2	Admin berada di halaman edit admin	Admin melakukan perubahan data admin dan mengklik simpan	Data admin berhasil diubah	√

4. Keluar/logout

Pengujian Halaman untuk logout dapat dilihat pada table 4.4 berikut.

Tanda (√) merupakan tanda yang berarti halaman yang di uji berhasil

Tabel 4.4 Blackbox testing Keluar/ Logout

No	Kondisi Awal	Hal yang dilakukan	Kondisi Akhir	Hasil Test
1	Admin mengklik Logout pada menu footer	Admin/user dibawa ke halaman logout	Admin/user kembali ke halaman awal	√

5. Halaman *Travel*

Pengujian halaman *Travel* dapat dilihat pada table 4.5 berikut. Tanda (√) merupakan tanda yang berarti halaman yang di uji berhasil

Tabel 4.5 Blackbox testing halaman *travel*

No	<i>Kondisi Awal</i>	Hal yang dilakukan	Kondisi Akhir	Hasil Test
1	pengunjung mengklik <i>Travel</i> pada menu	pengunjung dibawa ke halaman <i>travel</i>	Menampilkan halaman <i>travel</i>	√

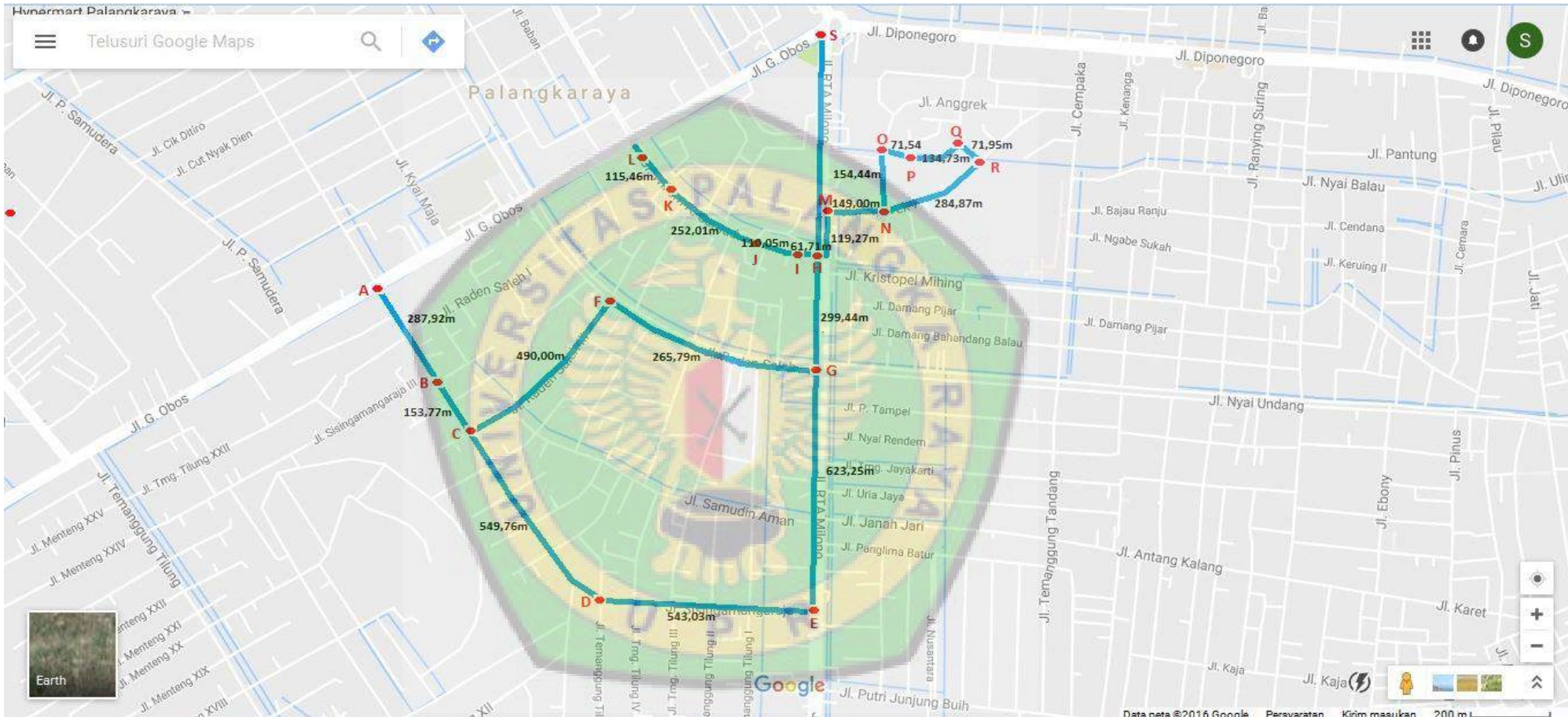
6. Halaman Pencarian Jarak Terpendek (Algoritma Dijkstra)

Pengujian halaman pencarian dapat dilihat pada table 4.6 berikut. Tanda (√) merupakan tanda yang berarti halaman yang di uji berhasil

Tabel 5.6 Blackbox testing halaman visi

No	<i>Kondisi Awal</i>	Hal yang dilakukan	Kondisi Akhir	Hasil Test
1	pengunjung mengklik menu pencarian	User dibawa ke halaman pencarian	Menampilkan halaman pencarian	√
2	Pengunjung mengisi lokasi awal	User dibawa ke halaman gmap	Kondisi lokasi saat ini	√
3	Pengunjung memilih lokasi akhir	User dibawa ke halaman gmap	Lokasi pilihan	√

4.5. Perhitungan Manual Algoritma Dijkstra



4.14 Alur Graph gambar 1

Gambar diatas merupakan merupakan sektsa alur *graph*, yang dibutuhkan untuk menjadi acuan untuk perhitungan manual.Dibawah ini merupakan perhitungan manual dari algoritma Dijkstra.

1. Perjalanan dari P ke titik J

Dari pemodelan graf diatas dapat kita lakukan perhitungan bahwa perjalanan dari titik P ke titik J melalui dua simpul yaitu:

$$P - O = 71,54$$

$$P - Q = 134,54$$

Dari dua nilai diatas simpul O yang terkecil $P - O = 71,54$. Kemudian dilanjutkan dengan perbandingan jalur :

$$P - O - N = 71,54 + 154,44 = 225,98$$

$$P - Q - R - N = 134,73 + 71,95 + 284,87$$

Ternyata dari perbandingan diatas jalur $P - O - N$ merupakan yang terkecil sehingga dilanjutkan untuk membandingkan jalur :

$$P - O - N - M = 71,54 + 154,44 + 149,00 = 374,98$$

$$P - Q - R - N - M = 134,73 + 71,95 + 284,87 + 149,00 = 640,55$$

Dan pada tahap pebandingan jalur $P - O - N - M$ merupakan yang terkecil yaitu 374,98.

Setelah itu dilanjutkan dengan membandingka jalur :

$$P - O - N - M - H = 71,54 + 154,44 + 149,00 + 119,27 = 494,25$$

$$P - Q - R - N - M - H = 134,73 + 71,95 + 284,87 + 149,00 + 119,27 = 759,82$$

Setelah dilakukan perbandingan dari kedua jalur diatas ternyata jalur P - O - N - M - H memiliki bobot yang lebih kecil yaitu 494,25. Kemudian dilanjutkan dengan membandingkan jalur :

$$P - O - N - M - H - I = 71,54 + 154,44 + 149,00 + 119,27 + 61,71 = 555,96$$

$$P - Q - R - N - M - H - I = 134,73 + 71,95 + 284,87 + 149,00 + 119,27 + 61,71 = 821,53$$

Dari perbandingan kedua jalur diatas $555,96 < 821$ sehingga jalur P - O - N - M - H - I merupakan jalur yang mempunyai bobot terkecil. Kemudian perbandingan dengan titik tujuan yaitu titik J :

$$P - O - N - M - H - I - J = 71,54 + 154,44 + 149,00 + 119,27 + 61,71 + 110,05 = 666,01$$

$$P - Q - R - N - M - H - I - J = 134,73 + 71,95 + 284,87 + 149,00 + 119,27 + 61,71 + 110,05 = 931,58$$

Sehingga dapat disimpulkan jalur terpendek dari titik P ke titik J adalah 666,01 dengan jalur perjalanan P - O - N - M - H - I - J.



Gambar 4.15 Pengujian perjalanan dari titik P ke titik J dengan algoritma dijkstra

2. Perjalanan dari titik P ke titik K

Dari pemodelan graf diatas dapat kita lakukan perhitungan bahwa perjalanan dari titik P ke titik K melalui dua simpul yaitu:

$$P - O = 71,54$$

$$P - Q = 134,54$$

Dari dua nilai diatas simpul O yang terkecil $P - O = 71,54$. Kemudian dilanjutkan dengan perbandingan jalur :

$$P - O - N = 71,54 + 154,44 = 225,98$$

$$P - Q - R - N = 134,73 + 71,95 + 284,87$$

Ternyata dari perbandingan diatas jalur $P - O - N$ merupakan yang terkecil sehingga dilanjutkan untuk membandingkan jalur :

$$P - O - N - M = 71,54 + 154,44 + 149,00 = 374,98$$

$$P - Q - R - N - M = 134,73 + 71,95 + 284,87 + 149,00 = 640,55$$

Dan pada tahap perbandingan jalur P – O – N – M merupakan yang terkecil yaitu 374,98.

Setelah itu dilanjutkan dengan membandingkan jalur :

$$P - O - N - M - H = 71,54 + 154,44 + 149,00 + 119,27 = 494,25$$

$$P - Q - R - N - M - H = 134,73 + 71,95 + 284,87 + 149,00 + 119,27 = 759,82$$

Setelah dilakukan perbandingan dari kedua jalur diatas ternyata jalur P – O – N – M – H memiliki bobot yang lebih kecil yaitu 494,25. Kemudian dilanjutkan dengan membandingkan jalur :

$$P - O - N - M - H - I = 71,54 + 154,44 + 149,00 + 119,27 + 61,71 = 555,96$$

$$P - Q - R - N - M - H - I = 134,73 + 71,95 + 284,87 + 149,00 + 119,27 + 61,71 = 821,53$$

Dari perbandingan kedua jalur diatas $555,96 < 821$ sehingga jalur P – O – N – M – H – I merupakan jalur yang mempunyai bobot terkecil. Kemudian perbandingan jalur :

$$P - O - N - M - H - I - J = 71,54 + 154,44 + 149,00 + 119,27 + 61,71 + 110,05 = 666,01$$

$$P - Q - R - N - M - H - I - J = 134,73 + 71,95 + 284,87 + 149,00 + 119,27 + 61,71 + 110,05 = 931,58$$

Dari perbandingan kedua jalur diatas $666,01 < 931,58$ sehingga jalur P – O – N – M – H – I – J merupakan yang terkecil. Kemudian perbandingan dengan titik tujuan yaitu titik K :

$$P - O - N - M - H - I - J - K = 71,54 + 154,44 + 149,00 + 119,27 + 61,71 + 110,05 + 252,01 = 918,02$$

$$P - Q - R - N - M - H - I - J - K = 134,73 + 71,95 + 284,87 + 149,00 + 119,27 + 61,71 + 110,05 + 252,01 = 1183,59$$

Sehingga dapat disimpulkan jalur terpendek dari titik P ke titik K adalah 918,02 dengan jalur perjalanan P – O – N – M – H – I – J – K.



Gambar 4.16 Pengujian perjalanan dari titik P ke titik K dengan algoritma dijkstra

BAB V

PENUTUP

5.1 Kesimpulan

Setelah melalui tahap implementasi dan pengujian sistem dilakukan maka dapat diambil beberapa kesimpulan yaitu:

1. Sistem dibuat dimulai dari proses analisis sistem kemudian dilanjutkan perancangan menggunakan metode *Waterfall (2011, Sommerville)*, setelah melalui proses perancangan dilanjutkan ke proses implementasi menggunakan bahasa pemrograman php, *javascript* dengan menggunakan *database mysql* yang ada pada *database server*.
2. Aplikasi ini menemukan rute terpendek objek lokasi travel di Kota Palangka Raya menggunakan algoritma Dijkstra.
3. Aplikasi dapat melakukan pencarian lokasi sesuai kata kunci yang dimasukan dan memfilter lokasi sesuai kategori lokasi.
4. Aplikasi ini dapat menampilkan titik-titik lokasi travel menggunakan google map.

5.2 Saran

1. Hasil pencarian rute terpendek atau tercepat yang lebih efektif dapat diperoleh dengan melakukan penambahan parameter selain jarak tempuh, misalnya dengan menghitung waktu tempuh, menghitung biaya perjalanan, peraturan lalu lintas dan lain sebagainya.
2. Masih ada beberapa algoritma pencarian rute terpendek lainnya jika pembaca ingin membandingkan.

3. Perlu dilakukan pengembangan lebih lanjut untuk proses kecepatan akses peta dan penampilan visualisasi peta serta aplikasi juga bisa di jadikan aplikasi android.
4. Jika pada penelitian ini lokasi hanya pada kota Palangka Raya, pembaca bisa mengembangkan pemetaan mencakup wilayah yang lebih luas misalnya untuk seluruh wilayah Kalimantan Tengah.



DAFTAR PUSTAKA

- Abdul Kadir.2013. Panduan Untu Mempelajari Pengembangan Rich Internet Application Melibatkan CSS, Javascript, dan PHP. Yogyakarta: Penerbit Andi.
- Aronoff.S. 1989. *Geographics Information System: A Management Perspective*. WDL: Publication. Ottawa.
- Date, C.J. 1995. *An introduction To Database Systems. Massachusetts: Adisson Wesley Publishing, Co., Inc.*
- Edy Irwansyah. 2013. *Sistem Informasi Geografis: Prinsip Dasar dan Pengembangan Aplkasi. Yogyakarta: Digi Books.*
- Fadhli, M.2013. Aplikasi LBS(*Location Base Service*) BTS(*Base Tower Station*) CV.2K dengan Menerapkan Algoritma Floyd – Warshall.Riau Pekanbaru: Fakultas Sains dan Teknologi Universitas Islam Negeri Sultan Kasim Riau Pekanbaru.
- Martin, J. 1975. *Computer Database Organization, parth I & I. New Jersey: Prentice-Hall, Inc.*
- Munir, Rinaldi. 2006. *Algoritma Pemrograman Dalam Bahasa Pascal dan C*. Bandung: Informatika Bandung.
- Novandi, Raden Aprian Diaz. 2007. *Perbandingan Algoritma Dijkstra dan Algoritma Floyd-Warshall dalam Penentuan Lintasan Terpendek (Single Pair Shortest Path)*. Bandung: Institut Teknologi Bandung.

Raghu. Ramakrishnan. 2000. Database Management Systems. USA: The McGraw-Hill Companies, Inc.

Ramadhan.fahmi.2009. Algoritma Bellman-Ford dan Floyd-Warshall. <https://fahmiramadhan.wordpress.com/2009/03/25/algoritma-bellman-ford-dan-floyd-warshall/>. 17 November 2016.

Ren Peng, Zhong Hsiang Tsou, Ming. 2003. *Internet GIS: Distributed Geographic Information Service for the Internet and Wireless Networks*, John Wiley & Sons.

Silberschatz, A. dkk. 2002. *Database System Concepts, 4th Edition*. New York :McGraw-Hill.

Simarmata, Janner.Paryudi, dan Iman. 2005. *Basis Data*. Yogyakarta: Penerbit Andi.

Stendy B. Sakur.2010. PHP 5 Pemograman Berorientasi Objek.Yogyakarta:Penerbit Andi.

Sutanta, E., 2004. *Sistem Basis data*. Yogyakarta: Penerbit Graha Ilmu.

Sven, C.,Florian, D., Peter, D.,Maristella, M.2009. *Engineering Web Applications*. London – New York: Springer.

S, Rosa A. dan M. Shalahuddin.2014. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*.Bandung:Informatika Bandung.

Whitten, J.L.,Bentley, L.D. 1998. *Systems Analysis and Design Methods*, 4th edition. New York: Irwin/McGraw-Hill International Co.

Whitten, J.L., Bentley, L.D. 2004. *Systems Analysis and Design Methods*.
Indianapolis: McGraw-Hill Education.

