

**NASKAH PUBLIKASI**

**IMPLEMENTASI ALGORITMA DIJKSTRA UNTUK  
MENENTUKAN RUTE BENGKEL TERDEKAT DI KECAMATAN  
MLATI YOGYAKARTA**

Program Studi Teknik Informatika



Disusun oleh:

**Nabilah Tanjach Muna**

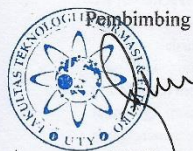
**5150411109**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI DAN ELEKTRO  
UNIVERSITAS TEKNOLOGI YOGYAKARTA  
2019**

NASKAH PUBLIKASI

IMPLEMENTASI ALGORITMA DIJKSTRA UNTUK  
MENENTUKAN RUTE BENGKEL TERDEKAT DI KECAMATAN  
MLATI YOGYAKARTA

Disusun oleh:  
NABILAH TANJACH MUNA  
5150411109



Pembimbing

Tri Widodo, S.T., M.Kom.

Tanggal: 24 Agustus 2019

# IMPLEMENTASI ALGORITMA DIJKSTRA UNTUK MENENTUKAN RUTE BENGKEL TERDEKAT DI KECAMATAN MLATI YOGYAKARTA

**Nabilah Tanjach Muna, Tri Widodo<sup>2</sup>**

<sup>1</sup>Program Studi Teknik Informatika, Fakultas Teknologi Informasi & Elektro

<sup>2</sup>Program Studi Pendidikan Teknologi Informasi, Fakultas Pendidikan

Universitas Teknologi Yogyakarta  
Jl. Ringroad Utara Jombor Sleman Yogyakarta  
Email: [nabilatm97@gmail.com](mailto:nabilatm97@gmail.com)

## ABSTRAK

*Perkembangan ilmu pengetahuan dan teknologi informasi sekarang ini sangat cepat, kebutuhan manusia akan informasi memacu pesatnya perkembangan teknologi informasi dan komunikasi, sejalan dengan perkembangan zaman penggunaan teknologi semakin meningkat dan mempunyai peran yang sangat penting dalam memberikan informasi yang kita butuhkan. Informasi mengenai rute bengkel motor terdekat sangat penting ketika kita dalam keadaan darurat di jalan misalnya ban bocor, mesin mati, dll informasi mengenai rute bengkel terdekat sangat penting untuk membantu masyarakat ketika dalam keadaan darurat di jalan. Namun penyedia informasi mengenai detail bengkel terdekat sangat minim sehingga masih kesulitan dalam menemukan lokasi bengkel terdekat.*

*Implementasi Algoritma Dijkstra Untuk Menentukan Rute Bengkel Terdekat Di Kecamatan Mlati Yogyakarta meliputi informasi dan rute bengkel terdekat. Sehingga Implementasi Algoritma Dijkstra Untuk Menentukan Rute Bengkel Terdekat Di Kecamatan Mlati Yogyakarta dapat memudahkan masyarakat dalam mencari informasi dan rute bengkel terdekat yang ada di kecamatan Mlati, Yogyakarta.*

**Kata kunci :** *Implementasi, Algoritma Dijkstra, Bengkel, Mlati Yogyakarta*

---

## 1. PENDAHULUAN

### 1.1 Latar Belakang

Seiring dengan berkembang pesatnya teknologi informasi seperti perangkat keras (Hardware) maupun perangkat lunak (Software) dan teknologi informasi lainnya telah membantu para penggunanya dalam mencari serta memperoleh informasi yang dibutuhkan. Pemanfaatan teknologi informasi saat ini sangatlah sentral dalam berbagai hal diantaranya memberikan informasi mengenai rute bengkel terdekat. Dengan pemanfaatan

teknologi dapat memudahkan dalam pencarian informasi yang kita butuhkan.

Bengkel merupakan tempat yang menaungi aktifitas dan fungsinya untuk memperbaiki kerusakan dan memberikan service mesin ataupun segala hal yang berhubungan dengan motor atau mobil. Maraknya pengguna kendaraan bermotor roda dua (motor/sepedamotor) di Kecamatan Mlati Yogyakarta, sebagian besar kendaraan yang digunakan tersebut memerlukan pemeliharaan dan perbaikan. Di samping itu, jika kendaraan tersebut terjadi kerusakan di jalan misalnya ban bocor, kerusakan mesin ketika di jalan, maupun ingin

melakukan perawatan, pengguna kendaraan tersebut membutuhkan informasi mengenai rute bengkel terdekat untuk melakukan pemeliharaan dan perbaikan.

Susahnya mencari rute terdekat menuju lokasi bengkel motor ketika kendaraan terjadi kerusakan di jalan pada pengguna kendaraan bermotor yang bukan warga asli Kecamatan Mlati Yogyakarta memerlukan informasi akurat agar dapat mempermudah pengguna kendaraan bermotor mengetahui rute terdekat menuju bengkel tersebut. Untuk menentukan rute terdekat menuju bengkel yang akan ditempuh atau didatangi pada saat kendaraan bermotor mogok/ bermasalah haruslah mencari seseorang yang terdekat untuk menanyakan rute terdekat menuju bengkel. Namun hal ini sering kali tidak dapat membantu secara maksimal karena kebanyakan warga yang bukan asli dari Kecamatan Mlati Yogyakarta akan kebingungan harus menempuh rute terdekat mana agar sistematis dan waktu menjadi efisien.

Berdasarkan uraian latar belakang permasalahan diatas, untuk mengatasi permasalahan pencarian rute bengkel terdekat di Kecamatan Mlati Yogyakarta, maka diperlukan suatu sistem pencarian rute bengkel terdekat. Sistem ini dibangun menggunakan algoritma Dijkstra. Algoritma ini adalah suatu algoritma untuk menentukan lintasan terpendek pada graf berarah. Algoritma ini pada penentuan jalur yang akan dipilih, akan dianalisis bobot dari node yang belum terpilih lalu dipilih node yang bobot terkecil. Algoritma Dijkstra akan berhenti jika semua node sudah terpilih dan dengan algoritma Dijkstra dapat menemukan jarak terpendek dari seluruh node, tidak hanya dari node asal ke node tujuan tertentu saja. Tujuan dari sistem ini nantinya dapat membantu untuk menentukan rute terpendek menuju bengkel di Kecamatan Mlati Yogyakarta.

### 1.2 Batasan Masalah

Batasan masalah dalam penelitian ini adalah:

- a. Sistem dibangun dengan menggunakan pemrograman web.
- b. Sistem mencakup wilayah Kecamatan Mlati Yogyakarta.
- c. Sistem hanya memberikan informasi mengenai rute menuju bengkel.
- d. Sistem menampilkan lokasi dan informasi mengenai bengkel motor.
- e. Rating bengkel hanya dapat diberikan oleh pengendara.

- f. Admin dan pemilik bengkel dapat melihat rating bengkel.
- g. Pembuatan peta bengkel Kecamatan Mlati Yogyakarta mengambil peta dari API Google Maps.
- h. Sistem menggunakan Algoritma Dijkstra untuk mencari rute terpendek
- i. Fitur tidak dilengkapi dengan navigasi menuju ke bengkel.
- j. Pemberian lokasi awal ditentukan oleh pengguna.

### 1.3 Tujuan penelitian

Tujuan dari penelitian tugas akhir ini adalah untuk menghasilkan suatu sistem yang memberikan informasi mengenai rute bengkel terdekat sehingga memudahkan masyarakat Kecamatan Mlati Yogyakarta untuk menentukan rute bengkel terdekat.

## 2. KAJIAN PUSTAKA DAN TEORI

### 2.1 Landasan Teori

[3] Penelitian Terdahulu membahas mengenai bagaimana membuat aplikasi digital untuk pencarian rute Bus Trans Semarang Menggunakan Algoritma Dijkstra, dimana algoritma Dijkstra digunakan untuk menentukan rute dan lokasi perpindahan koridor atau transfer point. Google Maps Api digunakan sebagai data spasial, sedangkan data non spasial berupa informasi detail shelter dan koridor. Selain itu juga dapat melihat jadwal, melihat rute, penentuan rute dan lokasi shelter.

[5] Penelitian terdahulu membahas mengenai bagaimana menghasilkan alternatif rute yang efektif dengan memperhatikan jenis arah, tingkat kemacetan, lebar jalan, kondisi jalan, volume jalan, jarak, jumlah simpangan, kondisi jalan yang rusak, jenis arah jalan (searah maupun dua arah) dengan menggunakan algoritma Dijkstra yang dapat menghasilkan rute yang efektif bagi pengiriman barang.

[12] Dalam penelitian ini membahas bagaimana mendata sekaligus menentukan rute atau jarak kiriman yang sudah disorting ke tujuan masing-masing dengan menggunakan metode Dijkstra untuk melakukan pencarian jalur terpendek. Dengan adanya aplikasi penentuan jarak terpendek maka dapat memudahkan pegawai kurir dalam hal waktu dan jarak kiriman.

[6] Penelitian terdahulu membahas bagaimana menentukan Rute Rumah Sakit Terdekat Menggunakan Algoritma Dijkstra di Kota Kisaran dengan tujuan membantu mobilitas sehari-hari

menjadi lebih efektif dan efisien dengan cara menampilkan rute terdekat. Metode algoritma dijkstra digunakan dalam menentukan rute terdekat. Algoritma ini bertujuan untuk menemukan jalur terpendek berdasarkan bobot terkecil dari satu titik ke titik lainnya.

## 2.2 Algoritma Dijkstra

[11]Algoritma Dijkstra adalah sebuah algoritma untuk menentukan lintasan terpendek dalam graf tak berarah (atau graf berarah) berbobot tanpa mengenumerasi secara eksplisit semua lintasan yang mungkin. Algoritma ini didasarkan pada sebuah metode yang dikenal sebagai pemrograman dinamik.

[8]Algoritma Dijkstra merupakan algoritma yang digunakan untuk menentukan jarak terpendek dari suatu vertex ke vertex yang lainnya pada suatu graph berbobot, jarak antar vertex adalah nilai bobot dari setiap edge pada graph. Suatu bobot harus bernilai positif (bobot  $\geq 0$ ). Algoritma dijktra ditemukan oleh Edger Wybe Dijkstra. Algoritma Dijkstra dikenal juga sebagai algoritma greedy yaitu algoritma yang penyelesaian masalah dengan mencari nilai maksimum. Cara kerja algoritma Dijkstra dalam pencarian jalur terpendek adalah perhitungan vertex terdekatnya, kemudian ke vertex yang kedua, dan seterusnya.

## 2.3 Bengkel

[7]Bengkel merupakan jasa yang melayani konsumennya dalam memperbaiki kendaraan atau hanya sekedar service rutin kendaraan yang digunakan sehari-hari.

[18]Bengkel adalah tempat (bangunan atau ruangan) untuk perawatan/pemeliharaan, memperbaiki, modifikasi alat dan mesin, tempat pembuatan bagian mesin dan perakitan mesin.

## 2.4 Website

[1]Website atau disingkat web dapat diartikan sekumpulan halaman yang terdiri atas beberapa laman yang berisi informasi dalam bentuk data digital, baik berupa teks, gambar, video, audio dan animasi lainnya yang disediakan melalui jalur koneksi internet.

[20]Website adalah sekumpulan halaman yang menampilkan konten atau sesuatu yang bisa diakses atau dibuka apabila kita mengakses internet.

## 2.5 HTML(*HiperText Markup Language*)

[19]*Hypertext Markup Language* (HTML) adalah sebuah Bahasa untuk menampilkan konten di Web. HTML sendiri adalah Bahasa pemrograman yang bebas, artinya tidak dimiliki oleh siapa pun, pengembangannya dilakukan oleh banyak orang di banyak negara dan bisa dikatakan sebagai sebuah Bahasa yang dikembangkan Bersama-sama secara global.

[1]HTML singkatan dari *Hypertext Markup Language* berperan dalam membuat struktur website. Layout website ditentukan bagaimana struktur skrip HTML ditulis.

## 2.6 PHP (*Hypertext Processor*)

[1]PHP singkatan dari *Hypertext Preprocessor* yang merupakan server-side programming yaitu Bahasa pemrograman yang diproses di sisi server. Fungsi utama PHP dalam membangun website adalah untuk melakukan pengelolaan data pada database. Data website akan dimasukkan ke database, diedit, dihapus dan ditampilkan pada website yang diatur oleh PHP.

[2]PHP singkatan dari *Hypertext Proprocessor* yaitu Bahasa pemrograman web server-side yang bersifat open source. PHP merupakan script yang terintegrasi dengan HTML dan berada pada server (*server side HTML embedded scripting*). PHP dalah script yang digunakan untuk membuat halaman website yang dinamis. Semua script PHP dieksekusi pada server di mana script tersebut dijalankan.

## 2.7 CSS (*Cascading Style Sheets*)

[14]CSS adalah Bahasa yang dapat digunakan untuk mendefinisikan bagaimana suatu Bahasa markup ditampilkan pada suatu media dimana Bahasa markup ini salah satunya adalah HTML. Atau dengan kata lain bahwa CSS merupakan kumpulan kode yang digunakan untuk mendesain halaman website agar lebih menarik dilihat. Dengan CSS bisa mengubah desain dari teks, warna, gambar dan latar belakang dari hampir semua tag HTML.

[1]CSS singkatan dari *Cascading Style Sheets* yaitu skrip yang digunakan untuk mengatur desain website. Walaupun HTML mempunyai kemampuan untuk mengatur tampilan website namun kemampuannya sangat terbatas. Fungsi CSS adalah memberikan pengaturan yang lebih lengkap agar struktur website yang dibuat dengan HTML terlihat lebih rapi dan indah.

## 2.8 Javascript

[13]Javascript adalah sekumpulan perintah khusus yang digunakan untuk membuat sebuah halaman web yang lebih responsive dan interaktif. Javascript merupakan Bahasa script yang dicantumkan pada sebuah halaman web dan dijalankan pada penjelajah web (web browser).

[19]JavaScript ditinjau dari jenisnya adalah Bahasa jenis *scripting*. Artinya perlu menyetikkan kodenya secara langsung dan dieksekusi langsung dari kode, dan tidak dikompilasi dulu untuk dijadikan file executable.

## 2.9 Google Maps API

[15]API (*Application Program Interface*) adalah sekumpulan perintah, fungsi, serta *protocol* yang dapat digunakan oleh *programmer* saat membangun perangkat lunak untuk sistem operasi tertentu. Google Maps API adalah sebuah jasa peta *globe* virtual gratis dan online yang disediakan oleh Google. Google Map API merupakan aplikasi *Interface* yang dapat diakses menggunakan *JavaScript* agar Google Map dapat ditampilkan pada halaman web yang sedang kita bangun. Untuk dapat mengakses Google Maps, kita harus melakukan pendaftaran API Key terlebih dahulu dengan data pendaftaran berupa nama domain web yang kita bangun.

[4]Google Maps API adalah layanan (*Service*) yang diberikan oleh Google kepada para pengguna untuk memanfaatkan Google Map dalam mengembangkan aplikasi.

## 2.10 Database

[10]Database adalah gabungan file data yang dibentuk dengan hubungan/relasi yang logis dan dapat diungkapkan dengan catatan serta bersifat independen.

[21]Database adalah sekumpulan dari bermacam-macam tipe record yang memiliki hubungan antar record.

## 2.11 DBMS (*Database Management Sistem*)

[21]*Database Management Sistem* (DBMS) merupakan paket program (*Software*) yang dibuat agar memudahkan dan mengefisienkan pemasukan, pengeditan, penghapusan dan pengambilan informasi terhadap database. *Software* yang tergolong kedalam DBMS anatar lain,

*Microsoft SQL*, *MySQL*, *Oracle*, *MS.Access* dan lain-lain.

[9]*Database Management System* (DBMS) adalah sekumpulan program yang digunakan untuk mendefinisikan, mengatur administrasi, dan memproses basis data dan aplikasi-aplikasi yang terkait dengan basis data.

## 2.12 ERD (*Entity Relationship Diagram*)

[17]*Entity Relationship Diagram* (ERD) merupakan suatu model data yang dikembangkan berdasarkan objek. ERD digunakan untuk menjelaskan hubungan antar data dalam basisdata kepada pengguna secara logis.

## 2.13 Kardinalitas Relasi

[17]Relasi adalah hubungan yang terjadi pada suatu tabel dengan lainnya yang mempresentasikan hubungan antar objek di dunia nyata dan berfungsi untuk mengatur operasi suatu database. Hubungan yang dapat dibentuk dapat mencakup 3 macam hubungan.

## 2.14 DFD (*Data Flow Diagram*)

[16]*Data Flow Diagram* (DFD) adalah suatu *network* yang menggambarkan suatu sistem automata atau komputerisasi, manualisasi, atau gabungan dari keduanya

## 3. METODE PENELITIAN

### 3.1 Analisis

Pada tahap analisis sistem ini penulis akan melakukan analisis mengenai kebutuhan yang dibutuhkan untuk membangun metode Algoritma Dijkstra pada penentuan rute bengkel terdekat. Analisis berupa kebutuhan fungsional yang berisi proses-proses yang ada dalam sistem. Kebutuhan fungsional dapat dilihat pada tabel 3.1.

**Tabel 1.** Kebutuhan Fungsional

No.	Kebutuhan	Deskripsi
1	Melihat posisi pengendara dan bengkel	Pengendara dapat melihat informasi posisi pengendara dan keberadaan bengkel.
2	Melihat lokasi bengkel	Pengendara dapat melihat informasi lokasi-lokasi bengkel
3	Mencari rute bengkel terdekat	Pengguna dapat mencari rute bengkel

	(Algoritma Dijkstra)	terdekat berdasarkan lokasi yang ditentukan oleh pengguna.
--	----------------------	--

Dalam masalah menentukan bengkel terdekat pengguna menggunakan metode yaitu: Algoritma Dijkstra dimana, pengambilan data mengenai bengkel didapat dari wawancara kepada pemilik bengkel tersebut. Kemudian data tersebut akan diolah dengan algoritma dijksta kemudian menghasilkan rute terdekat menuju bengkel tujuan pengendara.

### 3.2 Desain

Sistem ini dapat diakses oleh pengendara, pemilik bengkel dan admin. Pada desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak, representasi antarmuka dan prosedur pengkodean. Pada tahap desain, kegiatan yang dilakukan adalah merancang seperti apa sistem akan dibuat dan bagaimana proses kerja dari sistem. Pada tahap ini dibuat sebuah rancangan desain interface dari aplikasi yang akan dibuat yang menghasilkan sebuah input, proses dan output. Pada tahap input terdapat master data yaitu data admin, data pengendara, data persediaan, data bengkel dan data rating bengkel. Selain itu pada tahap proses pengendara dapat melakukan proses pencarian rute bengkel terdekat sedangkan pemilik bengkel dapat mendaftarkan bengkel pada sistem tersebut dan pada tahap output menghasilkan sebuah laporan data bengkel dan rating bengkel. Pada sistem ini terdapat 8 tabel basisdata yaitu tabel admin, tabel pengendara, tabel data persediaan, tabel bengkel, tabel rating, tabel jalur dan tabel graph.

### 3.3 Implementasi

Implementasi sistem merupakan kegiatan penerapan sistem yang sudah dibangun dalam kegiatan pendataan dan pengolahan data sesuai dengan rancangan sistem yang telah ditetapkan sebelumnya. Sistem ini akan diimplementasikan menggunakan Bahasa pemrograman HTML, PHP, Javascript dan database MySQL. HTML digunakan sebagai tempat menulis halaman web, PHP sebagai inti dari pemrograman web yang digunakan untuk proses perhitungan dan proses menghubungkan

antara sistem dengan database. Javascript yang digunakan dalam sistem ini berbentuk fungsi yang sudah dibuat, fungsinya bernama JQuery yang akan dipanggil oleh sistem.

### 3.4 Pengujian Sistem

Pengujian dilakukan dengan black box testing, yaitu pengujian atau testing dengan cara melihat alur kinerja dan output program yang dihasilkan. Dengan tujuan untuk mengetahui fungsi-fungsi yang diharapkan seperti output dihasilkan secara benar dari input, dan database diakses serta diupdate secara benar dan mengujinya apakah akan menjalankan fungsi-fungsi tersebut secara tepat. Uji coba dilakukan dengan mencoba fungsi menu apakah sudah sesuai harapan.

### 3.5 Pemeliharaan Sistem

Pemeliharaan sistem merupakan tahap terakhir dalam metode *waterfall*. Perangkat lunak yang sudah jadi, dijalankan serta dilakukan pemeliharaan. Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya. Selain itu, perbaikan implementasi *unit* sistem dan peningkatan jasa sistem sebagai kebutuhan baru.

## 4. ANALISA DAN PERANCANGAN SISTEM

### 4.1 Analisa Sistem

Tahap ini merupakan tahap yang sangat penting, karena kesalahan dalam tahap ini menyebabkan kesalahan pada tahap selanjutnya. Mengidentifikasi masalah merupakan langkah pertama yang dilakukan dalam tahap analisa sistem. Masalah dapat didefinisikan sebagai suatu pertanyaan yang diinginkan untuk dipecahkan. Setelah dapat mengidentifikasi masalah tersebut maka selanjutnya dapat mengidentifikasi titik-titik penyebab masalah dan dapat dimulai penguraian titik-titik masalah tersebut.

Dalam hal ini susahnya mencari rute bengkel motor terdekat ketika kendaraan terjadi kerusakan di jalan pada pengguna kendaraan bermotor yang bukan warga asli Kecamatan Mlati Yogyakarta memerlukan informasi akurat agar dapat mempermudah pengguna kendaraan bermotor mengetahui rute terdekat menuju bengkel tersebut. Untuk menentukan letak bengkel yang akan ditempuh pada saat kendaraan bermotor mogok/bermasalah haruslah mencari seorang yang terdekat untuk menanyakan lokasi bengkel terdekat,

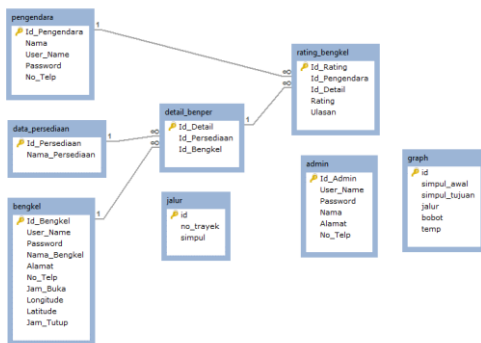
Namun hal ini sering kali tidak dapat membantu secara maksimal karena kebanyakan yang bukan warga asli dari Kecamatan Mlati Yogyakarta akan kebingungan harus menempuh rute mana agar sistematis dan waktu menjadi lebih efisien.

Sistem yang dibangun merupakan sistem pencarian rute bengkel terdekat menggunakan Algoritma Dijkstra yang dapat menentukan rute bengkel terdekat berdasarkan jarak terdekat. Dalam perancangan ini terdapat tiga pengguna yaitu Pembuat web, Pengendara/User Pengguna dan Pemilik bengkel. Pembuat Web dapat mengontrol semua data pengguna kendaraan maupun Pemilik Bengkel. Pengendara/User Pengguna bisa melihat atau menerima informasi yang disediakan pada website mengenai bengkel terdekat dan bisa memberikan rating. Sebagai Pemilik Bengkel, dapat melakukan login untuk mendaftarkan bengkel dan dapat melihat rating yang diberikan oleh Pengendara/user pengguna.

#### 4.2 Rancangan Sistem

Untuk lebih memahami aliran sistem yang berjalan, maka diperlukan suatu gambaran sistem dan arus data yang digunakan mulai dari proses input hingga proses output yang akan dibutuhkan oleh pengguna. Berikut ini adalah gambaran sistem dan arus data yang dimodelkan kedalam DFD level 0 (Diagram Konteks), Diagram Jenjang, DFD level 1, dan DFD level 2.

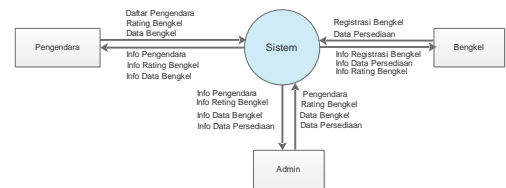
a. Diagram relasi antar tabel menggambarkan primary key dan foreign key yang terdapat pada setiap tabel dan kemudian digunakan untuk merelasikan antar tabel yang memiliki kaitan pada input dan proses. Diagram relasi antar tabel digambarkan pada gambar 1 seperti dibawah ini:



Gambar 1 Relasi Tabel

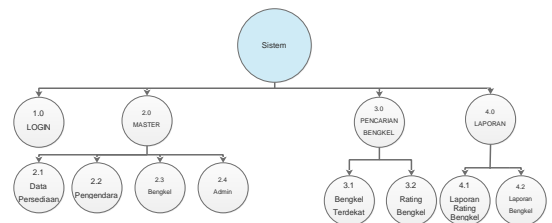
b. Diagram konteks merupakan suatu diagram yang digunakan untuk menggambarkan ruang

lingkup sistem. Dalam diagram konteks ini menunjukkan bahwa pengendara dapat melakukan registrasi, melihat data begkel dan memberikan rating. Bengkel dapat melakukan registrasi data bengkel dan dapat melihat rating yang sudah diberikan oleh user pengguna. Admin dapat melakukan semua proses yang ada pada sistem yaitu pengendara, rating bengkel, bengkel dan data persediaan. Rancangan sistem digambarkan pada diagram konteks seperti berikut:



Gambar 2 Diagram Konteks

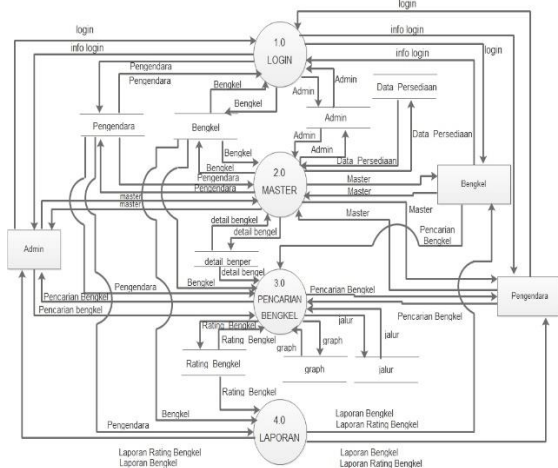
c. Diagram Diagram jenjang merupakan suatu diagram yang menggambarkan struktur dari sistem yang berupa suatu bagan jenjang yang menggambarkan semua proses yang ada pada sistem. Rancangan sistem yang dibuat dapat digambarkan dalam diagram jenjang sebagai berikut:



Gambar 3 Diagram Jenjang

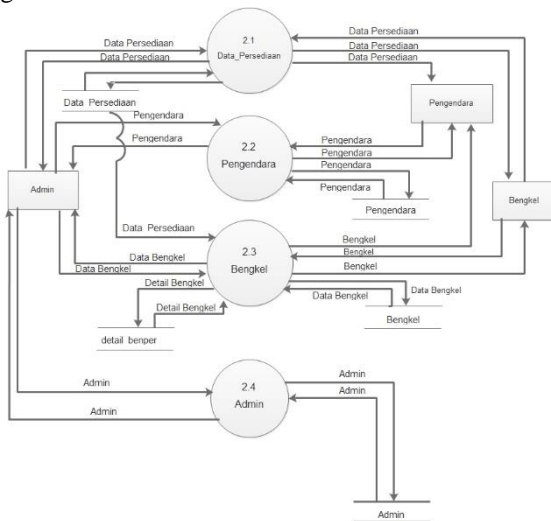
d. *Data Flow Diagram (DFD) Level 1* menjelaskan tentang gambaran proses yang lebih rinci dari diagram konteks dan menjelaskan proses sebuah penyimpanan. *Data Flow Diagram (DFD) Level 1* pada Implementasi Algoritma Dijkstra Untuk Menentukan Rute Bengkel Terdekat DI Kecamatan Mlati Yogyakarta dapat dilihat pada Gambar 4.





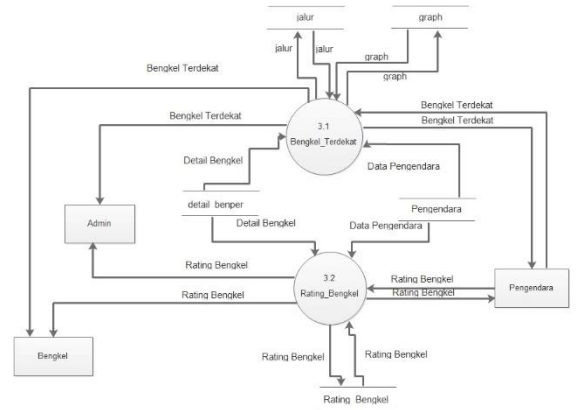
**Gambar 4 DFD Level 1**

e. DFD (*Data Flow Diagram*) level 2 proses 2 ini menjelaskan aliran data admin, pengendara dan bengkel. Dibagian ini terdapat 5 proses yaitu data\_persediaan, Pengendara, bengkel dan admin. Untuk lebih jelasnya dapat dilihat pada gambar 5 dibawah ini:



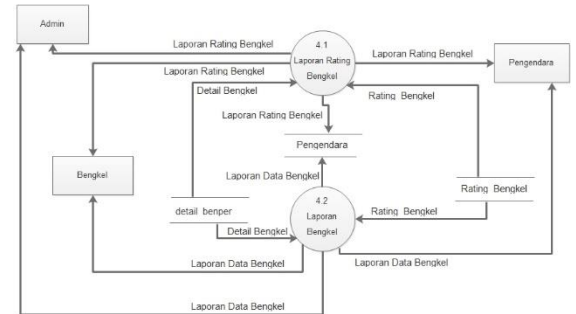
**Gambar 5 DFD Level 2 Proses 2**

f. DFD (*Data Flow Diagram*) level 2 proses 3 ini menjelaskan aliran data admin, pengendara dan bengkel. Dibagian ini terdapat 2 proses yaitu rating\_bengkel dan bengkel\_terdekat yang dihasilkan dari tabel Bengkel, Rating\_Bengkel, graph, jalur dan pengendara. Untuk lebih jelasnya dapat dilihat pada gambar 6



**Gambar 6 DFD Level 2 Proses 3**

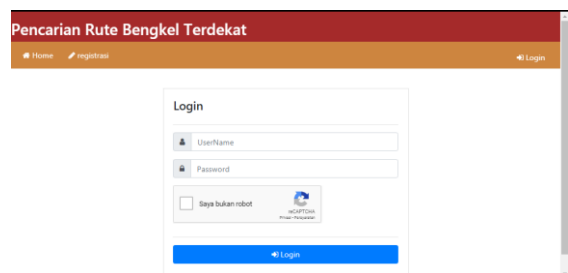
g. DFD (*Data Flow Diagram*) level 2 proses 4 ini terdapat 2 proses laporan yaitu Laporan bengkel dan laporan rating bengkel. Dapat dilihat pada gambar 7 dibawah ini.



**Gambar 7 DFD Level 2 Proses 4**

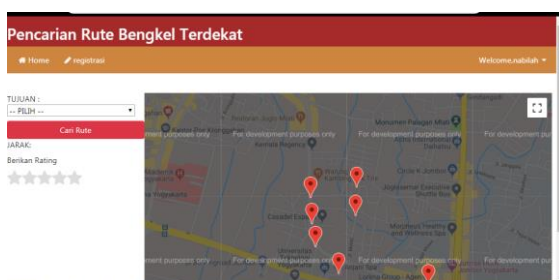
## 5. IMPLEMENTASI SISTEM

Impelementasi sistem merupakan tahapan lanjutan dari tahapan perancangan sistem. Implementasi sistem bertujuan untuk menterjemahkan keperluan perangkat lunak kedalam bentuk sebenarnya yang dimengerti oleh komputer. Dalam tahap implementasi ini, akan dijelaskan mengenai perangkat keras (Hardware) dan perangkat lunak (Software) yang digunakan dalam membangun sistem, flowchat dan data yang digunakan untuk pengolahan dari hasil pemrograman perancangan tampilan interface (antarmuka).



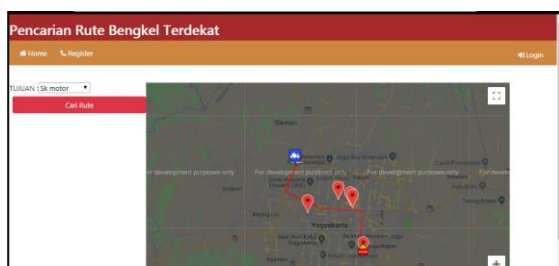
**Gambar 8 Halaman Login**

Halaman ini merupakan halaman login untuk admin, pemilik bengkel dan pengendara. Halaman ini menampilkan field untuk mengisi username dan password. Admin dapat melakukan login pada halaman ini untuk mengelola data admin, data pengendara, data bengkel, data persediaan dan data rating. Untuk pengendara dapat mengelola data rating, seperti memberikan rating kepada bengkel yang sudah pengendara cari. Selain itu pemilik bengkel dapat mengelola data bengkel, mendaftarkan bengkel tersebut dan melihat rating yang sudah diberikan oleh pengendara. Tampilan halaman login ditunjukkan oleh gambar 8



**Gambar 9 Halaman Utama**

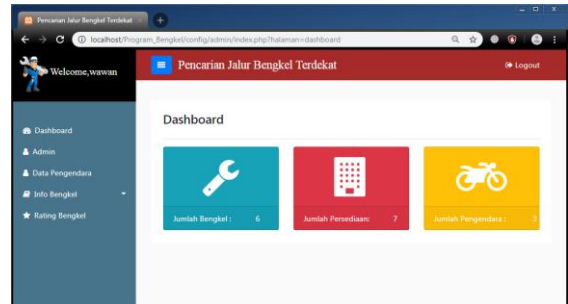
Gambar 9 menunjukkan pencarian lokasi pengendara secara otomatis. Halaman ini juga menampilkan semua bengkel yang sudah didaftarkan (titik berwarna merah). Selain itu pengendara juga dapat memberikan rating kepada bengkel. Pemilik bengkel juga bisa mendaftarkan bengkel dan melihat rating yang sudah diberikan oleh pengendara. Setelah itu akan muncul lokasi pengendara saat ini dan memilih bengkel yang akan dituju setelah itu sistem akan menampilkan rute terdekat.



**Gambar 10 Pencarian Rute Bengkel Terdekat**

Gambar 10 menunjukkan halaman pencarian rute terdekat pengendara memilih bengkel tujuan terlebih dahulu kemudian sistem akan mencari rute terdekat di antara node-node yang sudah ada dengan menggunakan Algoritma Dijkstra. Pada halaman ini titik warna merah merupakan daftar

lokasi bengkel, sedangkan tanda panah warna merah merupakan petunjuk lokasi bengkel yang dipilih oleh pengendara dan gambar warna biru merupakan posisi pengendara.



**Gambar 11 Halaman Utama Admin**

Gambar 11 menunjukkan tampilan halaman untuk admin yang muncul setelah admin melakukan registrasi (Login). Halaman ini mempunyai layanan menambah, merubah, menghapus dan melakukan pencarian data pengendara, data persediaan data bengkel. Selain itu juga dapat melihat data rating yang sudah diberikan pengendara terhadap bengkel tersebut.

## 6. PENUTUP

### 6.1 Kesimpulan

Penelitian yang dilakukan ini menghasilkan sebuah analisis, perancangan dan implementasi berupa Implementasi Algoritma Dijkstra Untuk Menentukan Rute Bengkel Terdekat. Adapun kesimpulan yang didapat dari adalah:

- Sistem ini dapat berjalan dengan baik setelah dilakukan pengujian
- Sistem ini berhasil menemukan rute tercepat dari lokasi yang ditentukan oleh pengendara menuju lokasi bengkel yang sudah dipilih, dengan menggunakan algoritma Dijkstra sebagai metode pencarian rute terdekat
- Sistem ini berhasil memberikan informasi kepada pengguna mengenai info bengkel

### 6.2 Saran

Perancangan dan pengembangan sistem yang telah dilakukan ini rasanya masih jauh dari kata sempurna, untuk penelitian selanjutnya terdapat beberapa saran penulis yang dapat digunakan untuk mengembangkan sistem kedepannya agar lebih baik. Saran-saran tersebut adalah:

- pada sistem pencarian rute bengkel terdekat diharapkan dapat dibuat lebih banyak lagi data bengkel.
- Seiring dengan sistem informasi yang semakin

berkembang, diharapkan untuk kedepannya dapat dikembangkan dalam versi mobile.

#### UCAPAN PERSEMBAHAN

Naskah Publikasi ini dapat diselesaikan tidak lepas dari segala bantuan, bimbingan, dorongan dan doa dari berbagai pihak, yang pada kesempatan ini penulis ingin menyampaikan ucapan terima kasih kepada:

1. Bapak Dr. Bambang Moertono Setiawan, M.M., CA., Akt. Selaku Rektor Universitas Teknologi Yogyakarta.
2. Bapak Sutarman, S.Kom., M.Kom., Ph.D. Selaku Dekan Fakultas Teknologi Informasi dan Elektro Universitas Teknologi Yogyakarta
3. Ibu Dr. Enny Itje Sela, S.Si., M.Kom selaku Ketua program studi Teknik Informatika Universitas Teknologi Yogyakarta
4. Bapak Tri Widodo, S.T., M.Kom selaku dosen pembimbing
5. Kepada kedua orang tua yang telah memberikan do'a serta dukungannya.

#### DAFTAR PUSTAKA

- [1]Abdulloh, R. (2016), bTrik Mudah Membuat CMS Website Dari Nol, Jakarta: PT Elex Media Komputindo.
- [2]Anhar (2010), Panduan Menguasai PHP & MYSQL Secara Otodidak, Jakarta: Mediakita.
- [3]Ardana, D. dan Saputra, R. (2016), Penerapan Algoritma Dijkstra pada Aplikasi Pencarian Rute Bus Trans Semarang, Seminar Nasional Ilmu Komputer, 4(13), 299–306.
- [4]Binarso, Y.A., Sarwoko, E.A. dan Bahtiar, N. (2012), Pembuatan Sistem Informasi Alumni Berbasis Web Pada Program Studi Teknik Informatika Universitas Diponegoro, Journal Of Informatic And Technology, 1(1), 72–84.
- [5]Budihartono, E. (2016), Penerapan Algoritma Dijkstra Untuk Sistem Pendukung Keputusan Bagi Penentuan Jalur Terpendek Pengiriman Paket Barang Travel, journal of information technology and computer science, 29(176), 69–78.
- [6]Ginting, J.V. dan Barus, E.S. (2018), Aplikasi Penentuan Rute Rumah Sakit Terdekat Menggunakan Algoritma Dijkstra, Jurnal Mantik Panusa, 2(2), 1–8.
- [7]Handoyo, I.G.S.R. dan Setiawan, P.Y. (2015), Pengaruh Karakteristik Demografi Dan Ketidakpuasan Terhadap Perilaku Mengeluh Konsumen Pada Bengkel Service Sepeda Motor Yamaha Diponegoro Denpasar Bali, E-jurnal Manajemen Unud, 4(10), 3317–3345.
- [8]Harahap, M.K. dan Khairina, N. (2017), Pencarian Jalur Terpendek dengan Algoritma Dijkstra, Jurnal dan Penelitian Teknik Informatika, 2(15), 18–23.
- [9]Hidayat, Zarman, W. dan Pamungkas, T. (2013), Implementasi Algoritma Kompresi LZW Pada Database Server, Jurnal Ilmiah Komputer dan Informatika, 2(1).
- [10]Lubis, A. (2016), Basis Data Dasar, Yogyakarta: CV Budi Utama.
- [11]Marsudi (2016), Teori Graf, Malang: UB Press.
- [12]Parapat, M.N., Kusbianto, D. dan Rahmad, C. (2017), Rancang Bangun Aplikasi Pencarian Rute Terpendek Jasa Kirim Barang Berbasis Mobile Dengan Metode Algoritma Dijkstra, jurnal Informatika Polinema, 3(3), 15–19.
- [13]Ramadhani, S. (2014), Sistem Informasi Pelayanan (BPJS) Berbasis Web, Jurnal TeknikA, 6(2).
- [14]Rerug, R.R. (2018), Pemrograman Web Dasar, Yogyakarta: Deepublish.
- [15]Rizki, K. dan Adil, A. (2018), Implementasi Google Maps Api Berbasis Android Untuk Lokasi Fasilitas Umum Di Kabupaten Sumbawa, Jurnal Matrik, 17(2).
- [16]Rosa (2013), Rekayasa Perangkat Lunak dan Berorientasi Objek, Bandung: Informatika.
- [17]Sutanta, E. (2011), Basis Data dalam Tinjauan Konseptual, Yogyakarta: C.V ANDi OFFSET.
- [18]Welim, Y.Y., Wisjhjudji, T.W. dan Firmansyah, R. (2015), Pengembangan Sistem Informasi Service Kendaraan Pada Bengkel KFMP, Jurnal Sistris, 6(1), 17–26.
- [19]Winarno, Ediy, Zaki, A. dan Community, S. (2015), Grafik Dan Animasi Web Dengan HTML 5, Jakarta: PT Elex Media Komputindo.
- [20]Winoto, P. dan Tjendrowasono, T.I. (2012), Pembuatan Website Profil Sekolah Dasar Negeri 03 Kalisoro, , 4(1), 50–55.
- [21]Yanto, R. (2016), Manajemen Basis Data Menggunakan MySQL, Yogyakarta: CV Budi Utama.