



JEPIN

(Jurnal Edukasi dan Penelitian Informatika)

ISSN(e): 2548-9364 / ISSN(p) : 2460-0741

Vol. 6
No. 1
April 2020

Analisis Sistem Jalur Terpendek Menggunakan Algoritma *Dijkstra* dan Evaluasi *Usability*

Yulvia Nora Marlim^{#1}, Deny Jollyta^{#2}, Fandri Saputra^{*3}

[#]Teknik Informatika, Sekolah Tinggi Ilmu Komputer Pelita Indonesia
Jalan Ahmad Yani No 78-88

¹yulvia.nora@lecturer.pelitaindonesia.ac.id

²deny.jollyta@lecturer.pelitaindonesia.ac.id

^{*}Sistem Informasi, Sekolah Tinggi Ilmu Komputer Pelita Indonesia
Jalan Ahmad Yani No 78-88

³fandri.saputra@student.pelitaindonesia.ac.id

Abstrak— Tingginya motivasi masyarakat pengguna kendaraan roda empat pribadi, menyebabkan semakin meningkat pula kebutuhan terhadap bengkel mobil. Usaha bengkel ini tumbuh dan berkembang pada titik-titik strategis hamper di semua kota-kotabesar di Indonesia. seperti Pekanbaru yang memiliki luas $\pm 632,26 \text{ km}^2$. Akibatnya timbul kesulitan dalam mengetahui letak/ alamat bengkel yang hendak dituju secara cepat dari posisi terdekat. Penelitian ini bertujuan untuk menentukan jalur terpendek menuju lokasi bengkel mobil menggunakan *algoritma Dijkstra* melalui aplikasi berbasis web. Hal ini dapat mempermudah masyarakat ketika mengalami kerusakan mobil dimanapun. *Algoritma Dijkstra* bekerja menggunakan *graph* dengan prinsip *greedy* yaitu mencari nilai minimum setiap simpul yang dilalui dengan teknik penelusuran menggunakan *Best First Search (BFS)*. *BFS* yaitu dengan menelusuri simpul yang tertinggi (awal) kemudian penelusuran ke simpul dibawahnya. Aplikasi selanjutnya diuji dengan evaluasi *usability*. 5 komponen *usability* yang digunakan adalah *learnability*, *efficiency*, *memorability*, *error*, dan *satisfaction* yang disusun menggunakan kuisioner. Respondennya terdiri dari masyarakat umum yang dipilih secara acak yang terdiri dari 30 responden. Uji *usability* menggunakan skala *likert* yang terdiri dari 14 pertanyaan, dengan teknik perhitungan menggunakan rata-rata sederhana. Hasil dari pengujian menunjukkan bahwa nilai *usability* dari aplikasi jalur terpendek adalah 78,5%, hal ini menunjukkan bahwa aplikasi jalur terpendek bernilai baik. Artinya bahwa responden merasa puas dengan adanya aplikasi ini dan terbantu dalam menentukan bengkel terdekat.

Kata kunci— Jalur Terpendek, Algoritma *Dijkstra*, *Usability*, *Best First Search (BFS)*.

I. PENDAHULUAN

Seiring dengan tingginya tingkat kebutuhan masyarakat terhadap kendaraan pribadi, sehingga meningkat juga kebutuhan terhadap bengkel mobil. Jika mobil mereka mengalami kerusakan, mereka akan

mencari bengkel mobil yang sesuai dengan kebutuhan, dan tentunya sebagian besar dari masyarakat ingin cepat sampai ke tujuan dengan memilih alternative jalan terdekat atau jalan pintas. Dengan alasan untuk menghemat waktu, biaya (bahan bakar) dan tenaga.

Luas wilayah kota Pekanbaru adalah 632,26 KM^2 . Dengan luas tersebut masyarakat kesulitan mencari lokasi / alamat bengkel yang dicari secara cepat. Berdasarkan wawancara dari salah satu pegawai SAMSAT kota Pekanbaru. Setiap tahun pembelian mobil baru, Rata-rata peningkatannya kurang lebih 13 % - 15 %, dilihat dari penerbitan STNK dan BPKB baru.

Data yang diambil adalah bengkel mobil dalam skala menengah ke atas. Peneliti juga melakukan observasi ke lapangan dengan mewawancarai calon konsumen umum dan pemilik bengkel. Hasil dari wawancara tersebut dapat ditarik kesimpulan bahwa calon konsumen mengalami kesulitan untuk mengetahui alamat bengkel mobil yang mereka inginkan walaupun sudah ada kartu nama. Disebabkan sebagian besar calon konsumen kurang mengetahui rute-rute dikota Pekanbaru, serta dikota Pekanbaru banyak jalan persimpangan dan jalan tikus (jalan kecil) dan juga terdapat beberapa lokasi bengkel masuk kedalam dan jauh dari jalan besar. Apalagi untuk masyarakat yang baru berkunjung ke kota Pekanbaru. Terkadang calon konsumen sering berhenti dijalan untuk menanyakan alamat ke penduduk sekitar, oleh sebab itu calon konsumen memerlukan biaya tambahan dan waktu tambahan untuk mencari lokasi bengkel yang dituju.

Dalam mencari lokasi bengkel menggunakan metode algoritma *dijkstra* untuk menentukan rute terpendek. Algoritma *dijkstra* merupakan salah satu algoritma *greedy* dengan bobot terendah [1]. Algoritma *dijkstra* merupakan salah satu bentuk algoritma yang banyak digunakan dalam menentukan rute terdekat [2]. Prinsipnya adalah mencari dua atau lebih lintasan dengan nilai bobot terkecil [3]. Syaratnya adalah bobotnya harus bernilai positif. jika

bobotnya bernilai negatif maka proses perhitungannya tidak bisa dilakukan. Penelitian yang dilakukan oleh [4] tentang pencarian SPBU terdekat menjelaskan bahwa algoritma *dijkstra* merekomendasikan jarak tempuh terdekat untuk menghemat penggunaan bahan bakar kendaraan.

Melihat kebutuhan dari pengunjung yang menginginkan sebuah sistem mempunyai *user interface*. Maka aplikasi jalur terpendek ini akan diuji. Pengujiannya berdasarkan *human computer interaction* (interaksi manusia dan computer). *Human Computer Interaction* memiliki prinsip *usability* ada 3 syarat untuk memenuhi prinsip *usability*, yaitu sistem itu harus *useful* yaitu website sesuai dengan kebutuhan pengunjung, *usable* yaitu mudah dipelajari dan digunakan [5]. *User Freindly* yaitu sistem ini mempunyai tampilan yang menarik, dan pengunjung senang dan nyaman menggunakannya sehingga nantinya sistem ini banyak diakses oleh pengunjung. Dengan kata lain prinsip *usability* atau penayagunaan ini juga berhubungan dengan tingkat kepuasan pengunjung [6]. *Usability* juga mengacu kepada bagaimana user memahami dan menggunakan produk untuk memperoleh tujuan dan menilai tingkat kepuasan dari pengguna [7]. Penilaian dari *usability* oleh pengunjung akan menjadi evaluasi untuk memperbaiki atau mengembangkan sistem menjadi lebih baik. Komponen *usability* yang digunakan pada penelitian ini adalah *learnability*, *effeciensci*, *memorability*, *error*, dan *saticfation*.

Berdasarkan uraian permasalahan diatas, dibutuhkanlah sebuah sistem informasi yang akan menjadi petunjuk jalan bagi pengunjung dengan menampilkan rute terpendek ketempat bengkel yang dituju. Diharapkan sistem yang dibuat memiliki prinsip *usability* yang sesuai dengan kebutuhan pengunjung. Diharapkan dengan adanya sistem informasi pencarian bengkel mobil dengan rute terpendek dapat membantuk calon konsumen/pengunjung cepat sampai di tempat tujuan dan menghemat waktu, biaya, dan tenaga.

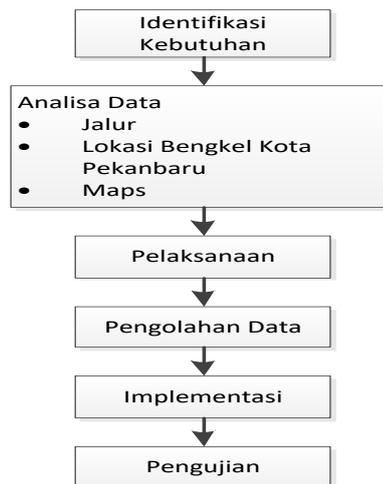
II. PERANCANGAN SISTEM

Pada penelitian ini metode penelitian gambarkan dalam bentuk kerangka penelitian. Dapat dilihat pada gambar 1 yang terdiri dari 5 tahap yaitu 1) Identifikasi kebutuhan, 2) Analisa data, 3) pelaksanaan, 4) Pengolahan data, 5) Implementasi, 6) Pengujian.

Pada tahap identifikasi kebutuhan langkah yang diterapkan adalah dengan turun langsung kelapangan. Mewawancari beberapa masyarakat dan pemilik bengkel. Kemudian merumuskan masalah yang didapat dari hasil wawancara. Serta menentukan tujuan dari penelitian.

Analisa data, pertama mengumpulkan studi literatur dengan cara mencari beberapa referensi baik itu berupa artikel, buku, majalah maupun artikel ilmiah. Kemudian dilakukan pengumpulan data, yaitu data bengkel di kota Pekanbaru dan menentukan titik koordinat (*Latitude* dan

Longitude) sebanyak 20 bengkel. Berikut titik koordinat bengkel dapat dilihat pada tabel 1.



Gambar. 1 Kerangka penelitian

TABEL I
TITIK KOORDINAT BENGKEL

Bengkel	Latitude	Longitude
B1	0.484655	101.394401
B2	0.482272	101.415623
B3	0.474658	101.424192
B4	0.490201	101.418129
B5	0.496981	101.418730
B...
B...
B20	0.494729	101.496617

Pengolahan Data dan perancangan sistem. Data yang didapat lalu diolah dan dilakukan perhitungannya dengan menggunakan algoritma *dijkstra*. Yang kemudian dirancang sebuah sistem untuk diimplementasikan. Kemudian dilakukan tahap pengujian. Pengujian sistem menggunakan evaluasi *usability*, dengan cara menyebarkan angket ke 30 orang responden, yang terdiri dari pegawai negeri sipil, ibu rumah tangga, mahasiswa, karyawan swasta.dan sopir

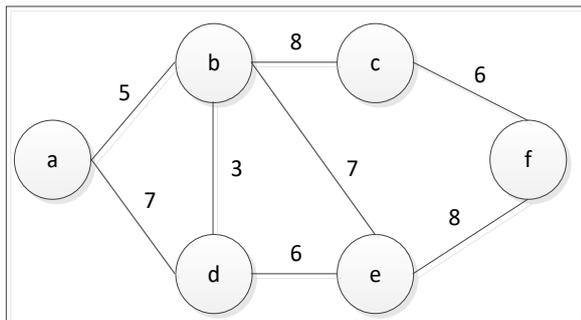
A. Jalur Terpendek

Jalur terpendek adalah lintasan yang dilalui dari simpul awal ke simpul tujuan dengan jalur terdekat dan tercepat, untuk mencari jalur tependek menggunakan *graph*. Bobotnya diambil berdasarkan jarak tempuh dan waktu tempuh atau disebut *weighted graph* [8]. Bobot pada *graph* merupakan jarak yang ditempuh, untuk menentukan jalur tependek menuju bengkel mobil. Untuk menentukan jalur terpendek menggunakan algoritma *dijkstra* dimana bobot yang digunakan hanya yang bernilai positif, jika bobotnya bernilai negatif maka tidak bisa.

B. Algoritma Dijkstra

Algoritma ini merupakan salah satu dari banyak algoritma yang banyak digunakan untuk memecahkan

permasalahan optimasi yang sering juga disebut sebagai algoritma *greedy* [1]. Cara yang diterapkan pada algoritma ini adalah untuk memecahkan permasalahan jalur terpendek adalah dengan mencari nilai atau bobot minimum dapat setiap node dalam sebuah *graph* [2]. Algoritma *dijkstra* dalam bentuk *graph*. Pada gambar 2 dapat dilihat contoh dari *graph* algoritma *dijkstra*. Pada gambar 2 merupakan contoh kasus dimana simpul a merupakan simpul awal atau bisa diilustrasikan sebagai alamat awal dan simpul f merupakan simpul alamat atau tujuan. Untuk mengetahui jalur/rute terpendek dari simpul a menuju f maka dicari bobot dengan nilai terkecil menggunakan algoritma *dijkstra*. Lebih lengkapnya tahapan dari algoritma *dijkstra* yang dilakukan adalah [1].



Gambar. 2 *Graph* algoritma *dijkstra*

Langkah pertama tentukan simpul awal atau sumber dan simpul tujuan. Gambar 2 terlihat simpul awal adalah a dan simpul tujuan adalah f. Kedua beri bobot (nilai) untuk semua simpul, cara menentukan bobot berdasarkan jarak tempuh antara simpul. Simpul awal bobotnya 0. Selanjutnya langkah ketiga dari simpul awal yaitu simpul a, di lihat simpul yang terhubung dengan a adalah simpul b dan d, dari simpul b dan d, dilihat bobot yang bernilai terkecil, yaitu a-b bernilai 5. Tahap keempat tetapkan bahwa simpul (b) terpilih, dan pada simpul yang dipilih (b) dapat dilihat ada node yang terhubung yaitu d dan c., a-b-c bernilai $5+8 = 13$, a-b-d bernilai $5+3 = 8$, a-b-e bernilai $5+7=12$, dari hasil yang didapat, terlihat nilai terkecil dilalui oleh simpul a-b-d bernilai 8. Selanjutnya lakukan hal yang sama sehingga simpul akhir atau tujuan di temukan.

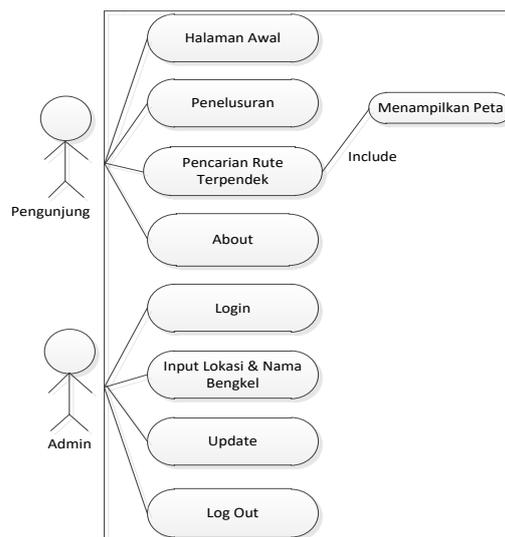
Penelusuran sirkuit dari algoritma *dijkstra* dilakukan dengan teknik *Best First Search* (BFS). Teknik BFS dilakukan dengan menelusuri titik (simpul) awal. Dikarenakan simpul awal terletak pada tingkat tertinggi kemudian penelusuran akan dilakukan ke titik (simpul) dibawahnya [8]. *Best First Search* merupakan kombinasi dari teknik *Depth First Search* dan *Breadth First Search* [9]. Pada algoritma *dijkstra* titik (simpul) yang akan ditelusuri adalah simpul yang memiliki bobot terkecil [1].

C. *Sistem Informasi Geografis (SIG)*

SIG dapat membantu dalam pengambilan keputusan, karena SIG lebih mudah dan cepat dalam mengolah data spasial [10]. SIG banyak digunakan dalam menentukan masalah-masalah jalur terpendek yang dikolaborasi dengan algoritma *dijkstra*, beberapa contohnya masalah menemukan objek wisata, peta rumah sakit, dan jalur-jalur untuk angkutan umum trans metro [11]. Sistem informasi geografis dapat mengatasi masalah-masalah [12]. Pertama memberikan ilustrasi dengan melakukan komputasi. Kedua menentukan tempat tertentu yang akan dituju. Dan Ketiga data dan informasi diorganisasikan.

D. *Rekayasa Kebutuhan Fungsional*

Kebutuhan fungsional adalah layanan-layanan yang harus disediakan oleh sistem untuk memenuhi kebutuhan dari *user*. Kebutuhan fungsional menggunakan *use case* diagram [2]. Dilihat pada gambar 3. Pada gambar 3 merupakan bentuk *use case* diagram dari Aplikasi jalur/rute terpendek. *Use case* adalah menggambarkan interaksi beberapa aktor didalam sebuah sistem. *Case* tergambaran didalam *use case*. Dapat dijelaskan pada gambar 2 *use case* sistem hanya di gunakan atau dipakai oleh 2 *user* yaitu admin dan pengunjung. Admin bertugas untuk mengupdate informasi tentang bengkel. Pengunjung dapat melihat informasi tentang bengkel berupa alamat bengkel, jasa-jasa yang disediakan, bengkel yang terdekat dengan pengunjung serta informasi rute atau jalur terpendek menuju bengkel yang diinginkan.

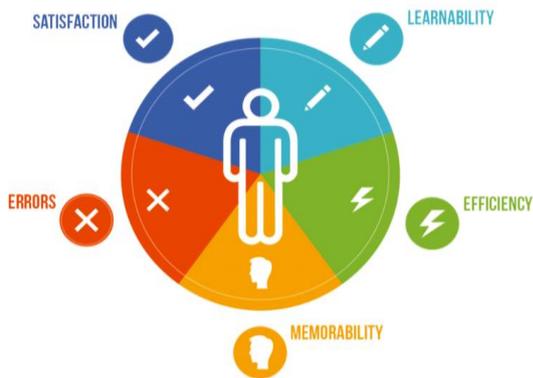


Gambar. 3 *Use case diagram*

E. *Rekayasa Kebutuhan Non Fungsional*

Kebutuhan non fungsional adalah keterbatasan yang dimiliki oleh sistem. Pada penelitian ini kebutuhan non fungsional yang di teliti adalah *usability*. *Usability* merupakan analisa kualitatif yang digunakan untuk menentukan kepuasan dai pemakai, *Usability* maksudnya agar sistem yang di gunakan dapat dipakai dengan baik. Pengujian *usability* dilakukan sebelum *website* ini di

terapkan secara *online*. *usability* terdiri dari 5 komponen dapat dilihat pada gambar 4.



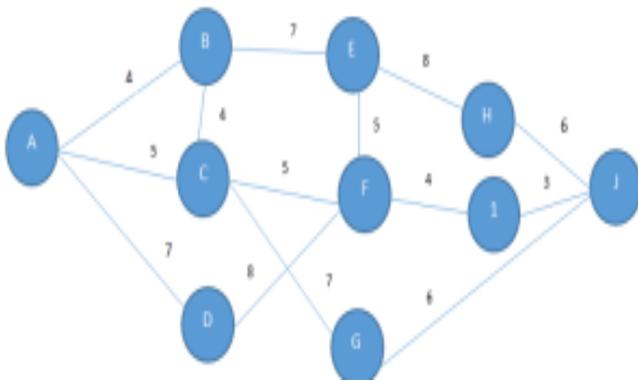
Gambar 4. Komponen *usability*

III. HASIL DAN PEMBAHASAN

Komponen *usability* yang digunakan adalah pertama *learnability* maksudnya sistem mudah dioperasikan dan dipahami. Kedua *memorability* adalah sistem mudah dipelajari, sehingga jika tidak digunakan dalam jangka lama, *user* masih bisa dengan mudah mengoperasikan. Ketiga *efficiency* merujuk kepada seberapa cepat sistem dapat membantu *user*. Keempat *error* maksudnya sistem memiliki tingkat *error* minimal. Kelima *satisfaction* maksudnya *user* puas menggunakannya dan merasa terbantu dengan adanya sistem ini.

A. Analisa Dengan Algoritma Dijkstra

Gambar 5 adalah contoh kasus analisis algoritma *dijkstra* dalam bentuk *graph*. Contoh kasus yang diambil dari Jalan Ahmad Yani No 78 menuju bengkel Mobil 3 R Jala SM. Amin Pekanbaru sehingga didapatlah bentuk *graph* seperti gambar 5.



Gambar 5. *Graph* menuju bengkel 3 R

Keterangan :

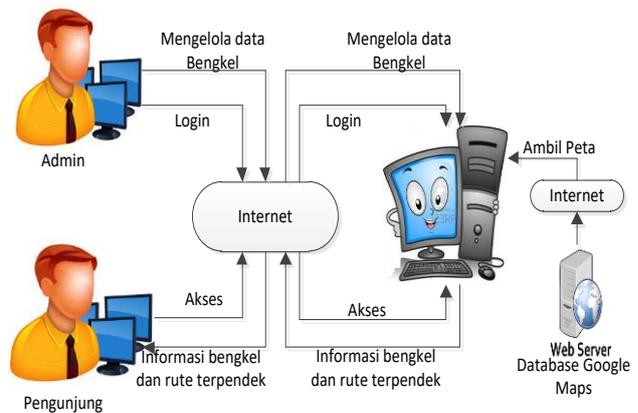
- 1). Jalan Ahmad Yani
- 2). Jalan Sudirman
- 3). Jalan Nangka

- 4). Jalan Jendral
- 5). Jalan Arifin Ahmad
- 6). Jalan Soekarno Hatta
- 7). Jalan Nangka Ujung
- 8). Jalan Hr Soebrantas
- 9). Jalan Rajawali Sakti
- 10). Jalan SM Amin Bengkel 3 R

Simpul A merupakan titik awal keberangkatan yaitu dari jalan Ahmad Yani menuju simpul J, yaitu simpul tujuan bengkel 3R Pekanbaru jalan SM Amin no 77. Kemudian beri bobot untuk semua simpul, untuk simpul awal yaitu A bobot dinilai dengan 0. Langkah selanjutnya dari simpul awal keberangkatan simpul A yang terhubung yaitu simpul B, C, D dari ketiga simpul yang dipilih jumlah bobot terkecil yaitu A-B dengan bobot $0+4=4$. Ditetapkan simpul B yang terpilih. Selanjutnya dilihat simpul yang terhubung dengan simpul B yaitu simpul C dan E, dan pilih bobot yang terkecil $A-B-C=0+4+4=8$, $A-B-E=0+4+7=11$, maka ditetapkan simpul A-B-C yang terpilih dengan nilai 8, kemudian lakukan langkah yang sama, untuk menuju simpul A ke J yang dilalui adalah simpul A-B-C-F-I-J = $0+4+4+5+4+3=20$. Cara perhitungan ini berlaku untuk semua jalur menuju bengkel mobil. sampai simpul tujuan didapati. Dari langkah diatas maka didapat hasil untuk menuju simpul A ke J yang dilalui adalah simpul A-B-C-F-I-J = $0+4+4+5+4+3=20$. Cara perhitungan ini berlaku untuk semua jalur menuju bengkel mobil.

B. Gambaran Umum Aplikasi

Gambar 6 merupakan gambaran umum dari sistem jalur terpendek. Pada gambar 6 dapat dilihat dimana pengunjung dapat mengetahui informasi tentang bengkel dan informasi rute atau jalur terpendek menuju jasa bengkel yang Pengunjung inginkan dan bengkel yang terdekat dengan posisi pengunjung. Admin bertugas mengelola data data bengkel.



Gambar. 6 Diagram arsitektur

C. Implementasi Halaman Utama

Pada halaman utama terdiri dari *home*, dan sistem pencarian, serta informasi tentang bengkel dapat dilihat pada gambar 7 berikut.



Gambar. 7 Halaman utama

D. Implementasi Halaman Sistem Pencarian

Pada implementasi halaman sistem pencarian terdapat menu rute dan info bengkel. Dapat dilihat pada gambar 8.



Gambar. 8 Halaman sistem pencarian

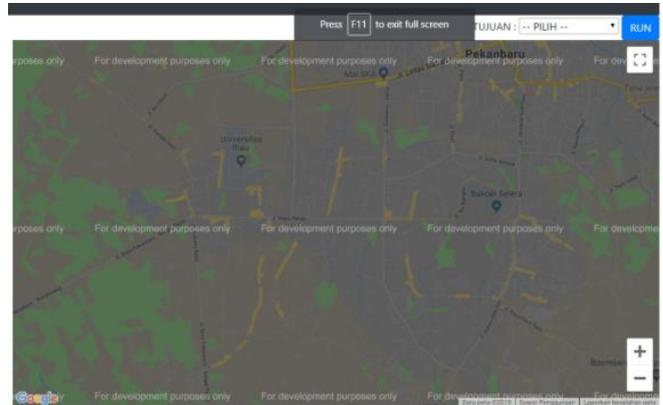
E. Implementasi Halaman Penelusuran

Implementasi pada halaman jalur dapat dilihat pada gambar 9 yang menampilkan jalur/rute menuju bengkel, dengan menuliskan tujuan/alamat bengkel. Yang dapat dilihat pada gambar 9 dibawah ini.

F. Pengujian Usability

Usability adalah pengujian yang dilakukan untuk mengetahui seberapa mudah sistem di operasikan [13]. Pada penelitian [14] tingkat efisiensi dan dan efektifitas *usability* diukur menggunakan *user's success rate* atau tingkat keberhasilan pengguna ditentukan presentasi tugas yang benar oleh pengguna. Pengujian ini dilakukan dengan cara menyebarkan angket ke responden. Responden terdiri dari masyarakat umum dipilih secara acak yang berpropersi sebagai supir, karyawan swasta, ibu rumah

tangga, pegawai negeri sipil. Pertanyaan yang berikan disesuaikan dengan kebutuhan *usability* yaitu, berdasarkan *learnability*, *memorability*, *efficiency*, *error*, dan *satisfaction*. Penyebaran angket diberikan untuk 30 orang responden. Kuisioner yang di sebarakan dalam bentuk skala *likert*. Dilihat pada table 2, tabel 2 merupakan interval dari pengujian *usability* dengan menggunakan *skala likert*.



Gambar. 9 Halaman penelusuran

TABEL III
INTERVAL SKALA LIKERT

Sangat tidak setuju	1
Kurang Setuju	2
Cukup Setuju	3
Baik	4
Sangat Baik	5

Kuisioner terdiri dari 14 pertanyaan. Sebelum mengisi kuisioner, responden terlebih dahulu disuruh mencoba *website*. Sesuai dengan panduan, kemudian dipersilahkan untuk mengisi kuisioner.

Kuisioner yang telah diisi, dikumpulkan datanya, kemudian dilakukan pengolahan data. Dengan cara sederhana yaitu menghitung rata-rata pada setiap komponen di *usability*. Yaitu *learnability*, *efficiency*, *memorability*, *error*, dan *satisfaction*. Pada komponen *error* Pertanyaan berupa pertanyaan negatif yang artinya setiap skor yang diisikan nantinya merupakan kebalikannya. Langkah pertama menggunakan rumus (1). Rumus (1) digunakan untuk menentukan nilai awal pada setiap komponen *usability*

$$Rumus = Pn/T \tag{1}$$

T = Jumlah Responden

Pn = Skor Skala Likert

Sehingga didapat hasil seperti tabel 3. Tabel 3 adalah nilai awal *usability* per komponen yang didapat dari total skor angket yang diisi oleh responden dibagi jumlah responden.

TABEL III
NILAI AWAL *USABILITY* PER KOMPONEN

<i>Usability</i>	Nilai
<i>Learnability</i>	4.08
<i>Effeciency</i>	4.18
<i>Memorability</i>	4.17
<i>Error</i>	4.22
<i>Saticfaction</i>	4.03

Selanjutnya mencari nilai persentase untuk masing masing komponen *usability* menggunakan rumus (2).

$$KomponenUsability = \frac{JumlahSkor}{SkorMaksimum} * 100\% \quad (2)$$

Pada rumus (2) jumlah skor didapat dari jumlah skor yang diisi oleh semua responden pada setiap pertanyaan, misalnya pada pertanyaan pertama, skor 5 diisi oleh 5 responden, skor 4 diisi oleh 15 responden, skor 3 diisi oleh 8 responden dan skor 2 diisi oleh 2 responden. jika dibuat dalam bentuk table dapat dilihat pada tabel 4.

TABEL IV
JUMLAH SKOR

Skala Likert	Jumlah responden
5	5
4	15
3	8
2	2
1	0

Untuk mencari skor maksimum didapat dari jumlah responden dikali skor tertinggi yaitu $30 * 5 = 150$

Sehingga dengan menggunakan rumus (2) untuk mencari nilai persentase didapatlah hasil dapat dilihat pada tabel V. Tabel V merupakan nilai tiap komponen *usability* dalam bentuk persentase.

TABEL V
PERSENTASE *USABILITY*

Komponen	Persentase
<i>Learnability</i>	77,3 %
<i>Effeciency</i>	74,7%
<i>Memorability</i>	82 %
<i>Error</i>	77,8 %
<i>Saticfaction</i>	80,9 %

Pada Tabel 5 dijelaskan bahwa *learnability* memiliki persentase 77,3%, *effeciency* 74,7%, *memorability* 82% *error* 77,8% dan *saticfaction* 80,9 %. Dapat diartikan bahwa nilai perkomponen *usability* diatas bernilai >=baik. Setelah hasil perkomponen *usability* didapat, yang

kemudian diukur tingkat penilaian *usability* perkomponen berdasarkan skala *likert* dengan pembagian persentase maksimal 100 % [15]. Dilihat pada tabel VI. Tabel VI merupakan tabel interval persentase penilaian.

TABEL VV
INTERVAL PERSENTASE PENILAIAN

Persentase	Penilaian
81 % - 100%	Sangat Baik
61 % - 80 %	Baik
41 % - 60 %	Cukup
21% - 40 %	Kurang Baik
0 % - 20 %	Sangat Kurang Baik

Setelah didapat hasil presentase masing-masing komponen. Langkah selanjutnya adalah mencari nilai *usability* dari aplikasi jalur terpendek dengan menggunakan rumus (3). Dilihat pada rumus (3) untuk mencari nilai persentase *usability* dari aplikasi jalur terpendek ini yaitu dengan menjumlahkan nilai semua komponen dibagi total komponen.

$$Usability\% = \frac{TotalPersentaseKomponen}{5} \quad (3)$$

Sehingga dengan menggunakan rumus (3), maka didapat nilai akhir dari *usability website* pencarian bengkel mobil dengan jalur terpendek adalah 78,5 %. Di lihat dari tabel 6 persentase *usability* menggunakan skala *likert* maka 78,5 dikategorikan dengan penilaian baik.

IV. KESIMPULAN

Pada penelitian ini algoritma *dijkstra* dapat diterapkan pada website pencarian rute terpendek yang mudah diakses dan digunakan oleh pengguna. Keunggulan dari sistem ini dapat terlihat dari pengujian non fungsional *usability* yang melibatkan 5 komponen, yaitu *learnability* dengan nilai 77,3 % kategori penilaian baik, *Effeciency* dengan nilai 74,7 % dengan penilaian baik. *Memorability* persentase 82 % dengan kategori penilaian sangat baik. *Error* 77,8 % dengan kategori penilaian baik. Dan *Saticfaction* dengan nilai persentase 80,9 %. Dapat disimpulkan bahwa nilai *usability* yang diuji dengan 30 responden, untuk seluruh komponen adalah 78,5 %, artinya adalah bahwa *website* pencarian bengkel mobil dengan rute terpendek dengan nilai baik. Dengan interval 61 % - 80 %. Sehingga dapat dikatakan bahwa website ini *user friendly* yaitu pengunjung merasa senang menggunakannya, *useful* yaitu website ini bermanfaat untuk yang menggunakan dan *useble* artinya mudah dipelajari.

UCAPAN TERIMA KASIH

Terima Kasih penulis ucapkan kepada pihak Institut Bisnis dan Teknologi Pelita Indonesia yang telah memberikan dana untuk publikasi artikel ini.

REFERENSI

- [1] D. Ardana and R. Saputra, "Penerapan Algoritma Dijkstra pada Aplikasi Pencarian Rute Bus Trans Semarang," no. Snik, pp. 299–306, 2016.
- [2] D. A. Syamsuddin, muhammad yusuf, Musliman Hanifah A, Harnoni, "Implementasi Algoritma Dijkstra Dalam Menemukan Jarak Terdekat Dari Lokasi Pengguna Ke Tanaman Yang Di Tuju Berbasis Android (Studi Kasus di Kebun Raya Purwodadi)," vol. 1, no. August, 2017.
- [3] D. T. Salaki, "Penentuan Lintasan Terpendek Dari Fmipa Ke Rektorat Dan Fakultas Lain Di Unsrat Manado Menggunakan Algoritma Dijkstra," *J. Ilm. Sains*, vol. 11, no. 1, p. 73, 2011.
- [4] R. A. Eka, yulia windi, Istiadi Dwiretno, "Pencarian SPBU Terdekat dan Penentuan Jarak Terpendek Menggunakan Algoritma Dijkstra (Studi Kasus di Kabupaten Jember)," *Nas. Tek. Elektro*, vol. 4, no. 1, pp. 89–93, 2015.
- [5] D. A. Sudarman, *Interaksi Manusia dan Komputer*. Yogyakarta: Penerbit Andi, 2007.
- [6] B. R. Pudjoatmodjo, "Tes Kegunaan (Usability Testing) Pada Aplikasi Kepegawaian Dengan Menggunakan System Usability Scale (Studi Kasus : Dinas Pertanian Kabupaten Bandung)," 2016, pp. 6–7.
- [7] Purwani Istiana, "Evaluasi Usability Situs Web Perpustakaan," *Visi Pustaka*, vol. 13, no. 3, pp. 5–10, 2011.
- [8] A. Juniansyah, "Aplikasi Penentuan Rute Terpendek Untuk Bagian Pemasaran Produk Roti Surya dengan Metode Best First Search," vol. 12, no. 1, pp. 31–40, 2016.
- [9] R. Apriandi, T. Rismawan, D. M. Midyanti, and J. S. Komputer, "Penerapan Metode Best First Search (BFS) Untuk Pencarian Lokasi SPBU Terdekat Menggunakan Arduino Berbasis Android," vol. 06, no. 1, pp. 1–11, 2018.
- [10] D. Manongga *et al.*, "Sistem Informasi Geografis Untuk Perjalanan Wisata di Kota Semarang," *Informatika*, vol. 10, pp. 1–9, 2009.
- [11] A. Gusmão and S. H. Pramono, "Sistem Informasi Geografis Pariwisata Berbasis Web Dan Pencarian Jalur Terpendek Dengan Algoritma Dijkstra," vol. 7, no. 2, pp. 125–130, 2013.
- [12] L. J. E. Dewi, "Pencarian Rute Terpendek Tempat Wisata di Bali Dengan Menggunakan Algoritma Dijkstra," 2010, vol. 2010, no. Snati, pp. 2008–2011.
- [13] D. R. Rahadi, "Pengukuran Usability Sistem Menggunakan Use Questionnaire Pada Aplikasi Android," vol. 6, no. 1, pp. 661–671, 2014.
- [14] Y. Nurhadryani, S. K. Sianturi, I. Hermadi, and H. Khotimah, "Pengujian Usability untuk Meningkatkan Antarmuka Aplikasi Mobile," *J. Ilmu Komput. dan Agri-Informatika*, vol. 2, no. 2, p. 83, 2013.
- [15] E. Susilo, B. Soedijono WA, and H. Al Fatta, "Evaluasi Aplikasi Mobile SSP (Secure System Of Payment) Menggunakan Prinsip Usability," *Semin. Nas. Teknol. Inf. dan Multimed. 2017*, vol. 2.6, pp. 7–12, 2017.