

Implementasi Algoritma Shortest Path Pada Aplikasi Pencarian Rute Distribusi Kurban

Aridhanyati Arifin¹, Dzaky Muhammad T.

Jurusan Teknik Informatika Fakultas Teknologi Industri

Universitas Islam Indonesia

Yogyakarta

aridhanyati@uii.ac.id¹

Abstrak— Ketepatan waktu pengiriman merupakan aspek yang sangat penting diperhatikan dalam proses distribusi paket kurban. Lamanya waktu pengangkutan paket kurban akan menimbulkan stres pada hewan yang diangkut dan memengaruhi kualitas daging yang diangkut. Masalah pengangkutan juga berdampak pada besar/kecilnya biaya transportasi yang akan ditanggung oleh lembaga penyalur. Oleh karena itu, dibutuhkan solusi penentuan rute terpendek dalam pengiriman paket kurban dengan memanfaatkan teknologi informasi (TI). Qurban App adalah sebuah purwarupa aplikasi pencarian rute distribusi paket kurban. Aplikasi tersebut memudahkan pengguna dalam mengatur pengiriman paket kurban. Pencarian jalur distribusi bagian dari masalah *shortest path*. Untuk menyelesaikan masalah tersebut diperlukan algoritma *Shortest Path* yakni algoritma *Floyd Warshall*. Metode ini digunakan untuk menentukan rute terpendek mana saja yang dapat ditempuh untuk menuju ke tempat pendistribusian paket kurban. Hasilnya adalah aplikasi Qurban App dapat mengelola data pengguna, mengelola data pengiriman paket kurban, baik daging maupun hewan, berupa data muatan dan kapasitasnya, data jenis hewan dan beratnya serta bahan bakar transportasi, menampilkan rute terdekat dan *tracking* terhadap kurir yang mengirimkan paket kurban. Berdasarkan pengujian yang dilakukan, algoritma Floyd Warshall berhasil menentukan rute distribusi dengan total jarak terpendek dan seluruh fitur di Qurban App dapat berjalan sesuai fungsinya.

Kata kunci—*masalah Shortest Path; distribusi paket kurban, algoritma floyd warshall.*

I. PENDAHULUAN

Hari raya Idul Adha merupakan puncak pelaksanaan ibadah Haji bagi umat muslim dan pada hari yang sama, umat muslim melakukan penyembelihan hewan kurban. Guna memudahkan penyaluran hewan kurban, lahir beragam lembaga yang menawarkan jasa penyaluran hewan kurban di Indonesia, salah satunya adalah Lembaga Amil Zakat, Infaq, Shodaqoh Muhammadiyah atau biasa disingkat Lazismu. Lembaga penyalur bekerjasama dengan peternak sapi maupun kambing untuk memenuhi permintaan hewan kurban, kemudian menjadi penghubung antara peternak dengan para donatur kurban. Prosedur yang diterapkan yaitu donatur membeli paket kurban (berupa hewan kurban atau daging kurban) kemudian lembaga penyalur akan mendistribusikan paket kurban yang telah dipilih tersebut kepada pihak yang membutuhkannya.

Ketepatan waktu pengiriman merupakan aspek yang sangat penting diperhatikan dalam proses distribusi paket kurban. Lamanya waktu pengangkutan paket kurban akan menimbulkan stress pada hewan yang diangkut [1]. Jika paket kurban berupa daging sembelihan maka lamanya waktu pengangkutan akan memengaruhi kualitas daging yang diangkut [2]. Lazismu ingin menghindari resiko keterlambatan pengiriman paket kurban ke daerah tujuan pendistribusian. Salah satu faktor yang dapat membuat keterlambatan dalam proses pengiriman tersebut adalah jarak yang tidak diperkirakan antara tempat peternakan hewan kurban menuju lokasi tujuan pendistribusian paket kurban. Masalah pengangkutan juga berdampak pada besar/kecilnya biaya transportasi yang akan ditanggung oleh lembaga penyalur. Berdasarkan hasil wawancara kepada Pimpinan Lazismu unit layanan Umbulharjo Yogyakarta, selama ini, pihak penyalur hanya menggunakan langkah-langkah konvensional untuk mengatasi masalah pengangkutan paket kurban tersebut.

Berdasarkan masalah-masalah di atas, diperlukan suatu upaya untuk menentukan rute-rute distribusi yang singkat dalam pengiriman paket kurban menggunakan pendekatan Teknologi Informasi (IT). Pemanfaatan IT untuk masalah pencarian jalur/rute terpendek (*Shortest Path*) telah diterapkan dalam banyak kasus, diantaranya pencarian rute pengiriman barang dan monitoringnya yang menggunakan algoritma *Dijkstra* [3] dan penentuan rute jaringan pariwisata menggunakan algoritma *Floyd Warshall* [4]. Makalah ini akan menguraikan penentuan rute tersingkat untuk pengiriman paket kurban serta estimasi biaya pengangkutan hewan kurban yang minimum menggunakan algoritma *Shortest Path*. Algoritma *Floyd Warshall* merupakan salah satu dari algoritma *Shortest Path* yang dipilih sebagai solusi karena sesuai dengan kasus distribusi yang dihadapi. Pada kasus distribusi yang dihadapi, memerlukan proses distribusi paket kurban yang optimal. Solusi optimal jelas dibutuhkan sebab menjadi dasar untuk kurir pengantar mengambil keputusan dengan cepat dan tepat sehingga kurir pengantar tidak salah memilih rute yang akhirnya dapat mengakibatkan keterlambatan atau terhalangnya pengiriman paket kurban. Dilihat dari aspek komputasi, *Floyd Warshall* merupakan algoritma yang lebih menjamin keberhasilan penemuan solusi optimum dibandingkan algoritma *Shortest Path* lainnya [5]. Selain itu, algoritma *Floyd Warshall*

memiliki keunggulan lain yakni lebih sederhana dan mudah implementasinya [6].

II. MASALAH SHORTEST PATH

Masalah pencarian jalur distribusi paket kurban dikategorikan sebagai salah satu jenis masalah *Shortest Path*. Topik *Shortest Path* merupakan pengembangan dari teori graf berbobot yaitu graf yang setiap sisinya (*path*) memiliki bobot. Bobot tersebut dapat melambangkan jarak, waktu, biaya dan lain-lain. Setiap *path* dari suatu graf menghubungkan sedikitnya 2 node yang melambangkan node tempat berangkat dan node tujuan.

Ide dari pemecahan masalah *Shortest Path* yakni mencari bobot terkecil dari setiap *path*, hingga ditemukan suatu rute dari node keberangkatan hingga ke node tujuan dengan total bobot terkecil. Oleh karena itu diperlukan suatu algoritma untuk menyelesaikan masalah *shortest path* diantaranya algoritma *Dijkstra*, *Floyd Warshall*, dan lain-lain.

Pada konteks masalah yang diangkat dalam makalah ini, bobot melambangkan jarak dan node dapat menunjukkan 2 hal yakni:

(1) Pada kondisi pertama, node tempat berangkat adalah lokasi awal kurir pengantar paket kurban dan node tujuan adalah lokasi mitra peternak

(2) Kondisi kedua, node tempat berangkat adalah lokasi mitra peternak dan node tujuan adalah lokasi desa/dusun/takmir yang akan menerima paket kurban.

III. ALGORITMA FLOYD WARSHALL

Algoritma *Floyd Warshall* dirumuskan berdasarkan konsep pencarian dari semua node ke semua node lainnya, disinilah letak keunggulannya dibandingkan algoritma *Shortest Path* lainnya sebab tidak perlu melakukan pencarian secara berulang-ulang.

Langkah-langkah penyelesaian menggunakan metode *Floyd Warshall* sebagai berikut [7]:

- 1) $W = W^{(0)}$; $Z = Z^{(0)}$
- 2) W^n
- 3) Untuk $k = 1$ hingga n , lakukan:
 - Untuk $i = 1$ hingga n , lakukan:
 - Untuk $j = 1$ hingga n , lakukan:
 - Jika $W[i,j] > W[i,k] + W[k,j]$ mak:
 - a) tukar $W[i,j]$ dengan $W[i,k] + W[k,j]$
 - b) ganti $Z[i,j]$ dengan $Z[i,k]$
- 4) $W^* = W$

Keterangan:

$W^{(0)}$ = matriks keterhubungan graf berarah berbobot awal

$Z^{(0)}$ = matriks path awal

W^* = matriks keterhubungan minimal

$R[i, j]$ = nilai dari node v_i ke v_j

Algoritma di atas diawali dengan inialisasi matriks hubung graf $W^{(0)}$ sesuai dengan data awal yang diterima. Pada iterasi yang sama pula dilakukan inialisasi awal matriks *path* $Z^{(0)}$ yang dibuat dengan aturan yaitu:

$$z^{(0)} = \begin{cases} j & \text{jika } w_{i,j}^{(0)} \neq \infty \\ 0 & \text{jika } w_{i,j}^{(0)} = \infty \end{cases} \quad (1)$$

Persamaan (1) dimaknai sebagai berikut:

- 1) Jika node v_i pada matriks $W^{(0)}$ dihubungkan langsung dengan node v_j maka elemen matriks *path* $Z^{(0)}$ sama dengan nilai kolom (j);
- 2) Jika matriks hubung graf $W^{(0)}[i,j]$, node v_i tidak memiliki hubungan dengan node v_j maka elemen matriks *path* $Z^{(0)}$ sama dengan nol.

Pada iterasi berikutnya, dilakukan pengujian jarak lewat node v_k , yakni antara node v_i dan node v_j dengan menjumlahkan tiap node pada setiap sel matriks W . Selanjutnya akan diperiksa setiap sel matriks $W[i,j]$. Jika didapatkan nilai hasil penjumlahan antar node v_i dan node v_j lebih besar dari nilai sebenarnya dari sel matriks $W[i,j]$, maka nilai hasil penjumlahan akan disimpan dan nilai sebenarnya matriks $W[i,j]$ akan diganti. Demikian pula nilai sel matriks *path* $Z^{(0)}$ akan ikut mengalami perubahan. Banyaknya iterasi W^n tergantung dari banyaknya node yang akan dihitung. Lakukan terus iterasi hingga node terakhir selesai diperiksa. Iterasi akhir akan menunjukkan hasil yang tetap dengan nilai minimal W^* dari penjumlahan tiap-tiap node.

IV. METODOLOGI

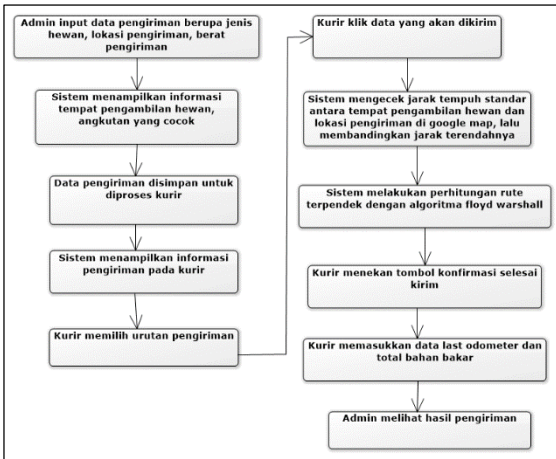
Pembangunan aplikasi pencarian jalur distribusi paket kurban menerapkan model proses *Waterfall* yaitu perangkat lunak dimodelkan dengan proses bertahap yang mengalir ke bawah layaknya air terjun, tahapan selanjutnya dapat dikerjakan apabila tahap berikutnya telah selesai dikerjakan, meliputi: analisis; perancangan; implementasi; dan pengujian [8].

A. Analisis Kebutuhan

Tahap analisis diawali dengan mengumpulkan data-data menyangkut proses bisnis konvensional yang terjadi di Lazismu Umbulharjo DIY ketika menyalurkan paket kurban. Hal ini dilakukan agar memudahkan pembangunan implementasi sehingga dapat diketahui kebutuhan masukan, proses dan keluaran yang diinginkan.

Kebutuhan masukan meliputi input data jenis hewan kurban yang akan disalurkan beserta beratnya, dan lokasi tujuan distribusi. Kebutuhan proses meliputi proses *login*; proses pengambilan posisi dari *Google Maps* dan pengecekan jarak tempuh standar; proses perhitungan, dan pemilihan rute terpendek dan proses perhitungan total biaya transportasi. Kebutuhan keluaran meliputi informasi lokasi pengambilan hewan kurban di mitra peternak, informasi rute pengiriman dan informasi total biaya transportasi.

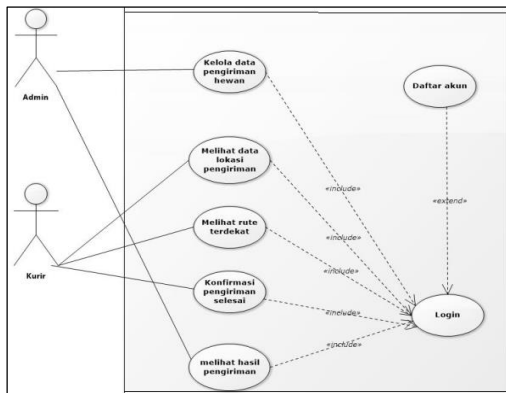
Selanjutnya, sebagai solusi ditawarkan proses bisnis yang baru yakni proses distribusi paket kurban dengan memanfaatkan aplikasi pencarian rute distribusi paket kurban atau disebut Qurban App, ditunjukkan pada Gambar 1.



Gambar 1. Proses bisnis aplikasi pencarian rute distribusi kurban

B. Perancangan Diagram Use Case

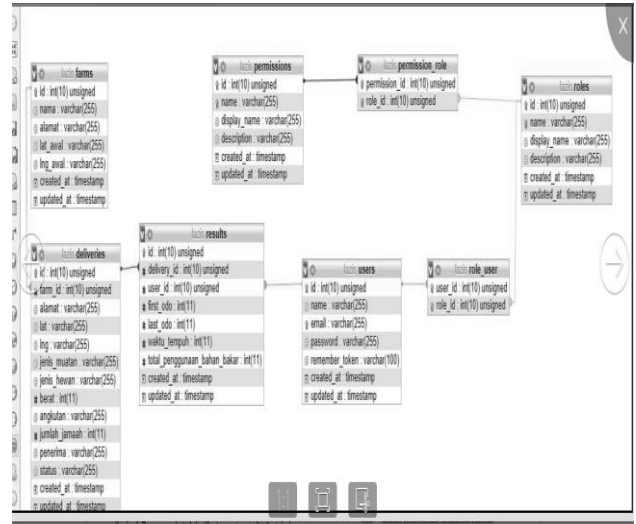
Diagram use case untuk aplikasi pencarian rute distribusi kurban (Qurban App) ditunjukkan oleh Gambar 2.



Gambar 2. Diagram use case aplikasi pencarian rute distribusi kurban

C. Perancangan Relasi Tabel

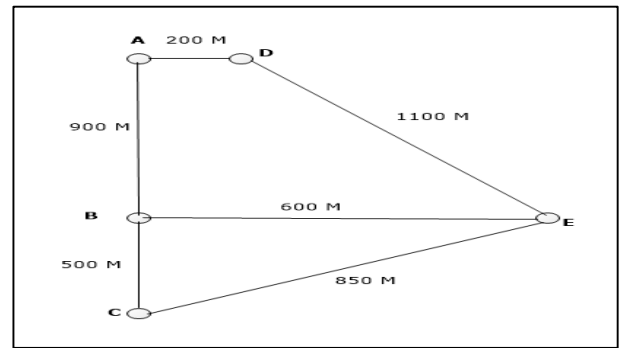
Rancangan untuk relasi tabel aplikasi pencarian rute distribusi kurban (Qurban App) ditunjukkan oleh Gambar 3.



Gambar 3. Rancangan relasi tabel aplikasi pencarian rute distribusi kurban

D. Pemodelan Algoritma Floyd Warshall Pada Kasus

Graf digunakan untuk memodelkan jaringan rute distribusi paket kurban seperti yang ditunjukkan oleh Gambar 4.



Gambar 4. Pemodelan kasus jalur distribusi paket kurban dengan graf

Pada Gambar 4 terdapat 5 node (A, B, C, D, E) yang mewakili lokasi yang akan dikunjungi kurir berikut jarak antar nodenya. Sebenarnya kurir dapat berangkat dari tempat mana saja, namun dalam kasus ini kurir diasumsikan memilih berangkat dari lokasi A dengan tujuan lokasi E. Langkah-langkah penerapan algoritma Floyd Warshall secara matematis sebagai berikut:

- 1) Inisialisasi matriks hubung graf $W^{(0)}$ pada Tabel I dan matriks path $Z^{(0)}$ pada Tabel II yang menerapkan aturan (1)

TABEL I. MATRIKS HUBUNG GRAF $W^{(0)}$

Dari/Ke	A	B	C	D	E
A	∞	900	∞	200	∞
B	900	∞	500	2450	600
C	∞	500	∞	∞	850
D	200	2450	∞	∞	1100
E	∞	600	850	1100	∞

TABEL III. MATRIKS PATH $Z^{(0)}$

Dari\Ke	A	B	C	D	E
A	0	2	0	4	0
B	1	0	3	0	5
C	0	2	0	0	5
D	1	0	0	0	5
E	0	2	3	4	0

- 2) Terdapat 5 lokasi sehingga terdapat 5 node yaitu node A, B,C,D, dan E. Iterasi akan terjadi sebanyak jumlah node.
- 3) Iterasi untuk $k=1$, menguji perjalanan yang melalui node A (node ke 1), diperoleh matriks $W^{(1)}$ pada Tabel III dan matriks $Z^{(1)}$ pada Tabel IV. Pada matriks $W^{(1)}$ diperoleh perubahan nilai sel BD dan DB dari ∞ menjadi 1100, demikian pula di matriks $Z^{(1)}$ mengalami penyesuaian nilai sel BD dan DB, masing-masing diisi sebesar 1, sesuai indeks node yang dilewati.

TABEL IIIII. MATRIKS HUBUNG $W^{(1)}$

Dari\Ke	A	B	C	D	E
A	∞	900	∞	200	∞
B	900	∞	500	1100	600
C	∞	500	∞	∞	850
D	200	1100	∞	∞	1100
E	∞	600	850	1100	∞

TABEL IVV. MATRIKS PATH Z

Dari\Ke	A	B	C	D	E
A	0	2	0	4	0
B	1	0	3	1	5
C	0	2	0	0	5
D	1	1	0	0	5
E	0	2	3	4	0

- 4) Iterasi untuk $k=2$, menguji perjalanan yang melalui node B, diperoleh matriks $W^{(2)}$ pada Tabel V dan matriks $Z^{(2)}$ pada Tabel VI. Pada matriks $W^{(2)}$ diperoleh perubahan nilai sel AC dan CA dari ∞ menjadi 1400, nilai sel AE dan EA menjadi 1500, nilai sel CD dan DC menjadi 1600. Demikian pula di matriks $Z^{(2)}$, nilai sel AC,CA,AE,EA,CD dan DC masing-masing mengalami penyesuaian nilai sesuai indeks node yang dilewati.

TABEL V. MATRIKS HUBUNG $W^{(2)}$

Dari\Ke	A	B	C	D	E
A	∞	900	1400	200	1500
B	900	∞	500	1100	600
C	1400	500	1950	1600	850
D	200	1100	1600	∞	1100
E	1500	600	850	1100	∞

TABEL VI. MATRIKS PATH $Z^{(2)}$

Dari\Ke	A	B	C	D	E
A	0	2	2	4	2
B	1	0	3	1	5
C	2	2	2	2	5
D	1	1	1	0	5
E	2	2	3	4	0

- 5) Iterasi untuk $k=3$, menguji perjalanan yang melalui node C, menguji perjalanan yang melalui node B, diperoleh matriks $W^{(3)}$ pada Tabel VII dan matriks $Z^{(3)}$ pada Tabel VIII. Pada matriks $W^{(3)}$ yang memenuhi syarat untuk mengalami perubahan ada di sel BB dan EE, nilainya menjadi 1950. Pada matriks $Z^{(3)}$, nilai sel BB dan EE mengalami penyesuaian nilai sesuai dengan node yang dikunjungi.

TABEL VII. MATRIKS HUBUNG $W^{(3)}$

Dari\Ke	A	B	C	D	E
A	∞	900	1400	200	1500
B	900	1950	500	1100	600
C	1400	500	1950	1600	850
D	200	1100	1600	∞	1100
E	1500	600	850	1100	1950

TABEL VIII. MATRIKS PATH $Z^{(3)}$

Dari\Ke	A	B	C	D	E
A	0	2	2	4	2
B	1	3	3	1	5
C	2	2	2	2	5
D	1	1	1	0	5
E	2	2	3	4	3

- 6) Iterasi untuk $k=4$, menguji perjalanan yang melalui node D, diperoleh matriks $W^{(4)}$ pada Tabel IX dan matriks $Z^{(4)}$ pada Tabel X. Pada matriks $W^{(4)}$ sel AE dan EA mengalami perubahan nilai dari 1500 menjadi 1300, sel AA dan DD menerima nilai 2800. Pada matriks $Z^{(4)}$, nilai sel AE, EA, AA dan DD mengalami penyesuaian nilai sesuai dengan node yang dikunjungi.

TABEL IX. MATRIKS HUBUNG $W^{(4)}$

Dari\Ke	A	B	C	D	E
A	2800	900	1400	200	1300
B	900	1950	500	1100	600
C	1400	500	1950	1600	850
D	200	1100	1600	∞	1100
E	1300	600	850	1100	1950

TABEL X. MATRIKS PATH $Z^{(4)}$

Dari\Ke	A	B	C	D	E
A	4	2	2	4	4
B	1	3	3	1	5
C	2	2	2	2	5
D	1	1	1	0	5
E	4	2	3	4	3

7) Iterasi untuk $k=5$, menguji perjalanan yang melalui node E, diperoleh matriks $W^{(5)}$ pada Tabel XI dan matriks $Z^{(5)}$ pada Tabel XII. Pada matriks $W^{(5)}$ diperoleh perubahan nilai sel DD dari ∞ menjadi 2800, Demikian pula di matriks $Z^{(2)}$, nilai sel DD mengalami penyesuaian nilai sesuai indeks node yang dilewati.

TABEL XI. MATRIKS HUBUNG $W^{(5)}$

Dari\Ke	A	B	C	D	E
A	2800	900	1400	200	1300
B	900	1950	500	1100	600
C	1400	500	1950	1600	850
D	200	1100	1600	2800	1100
E	1300	600	850	1100	1950

TABEL XII. MATRIKS PATH $Z^{(5)}$

Dari\Ke	A	B	C	D	E
A	4	2	2	4	4
B	1	3	3	1	5
C	2	2	2	2	5
D	1	1	1	5	5
E	4	2	3	4	3

Berdasarkan iterasi terakhir ini, dapat diketahui jalur yang harus dilalui adalah:

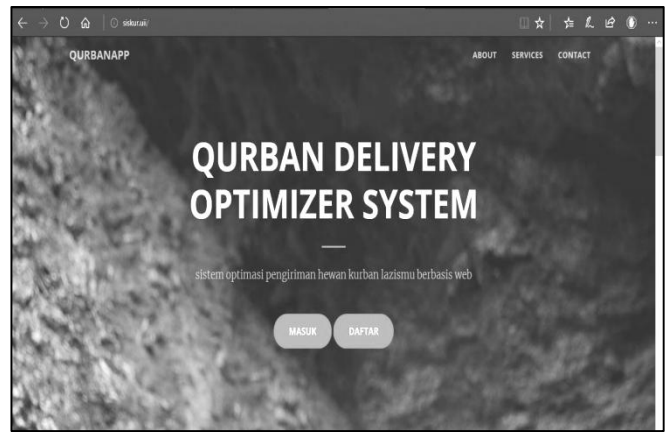
$$Z^{(5)}[A,E] = 4$$

$$Z^{(5)}[D,E] = 5$$

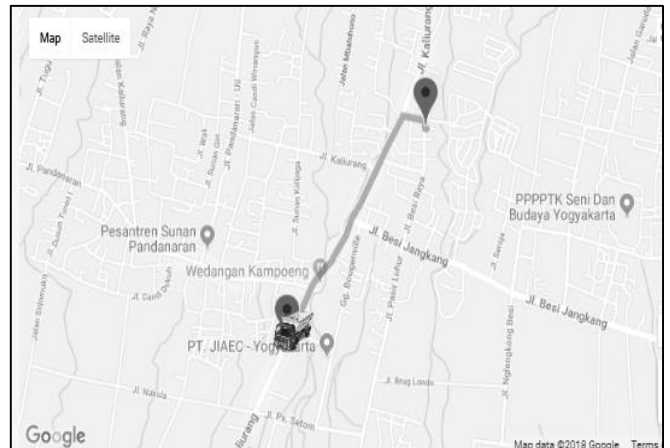
Setelah seluruh node selesai diperiksa, akhirnya diperoleh hasil berupa rute terpendek dari lokasi A menuju lokasi E adalah A-D-E dengan total jarak sebesar 1300 m.

V. HASIL DAN PEMBAHASAN

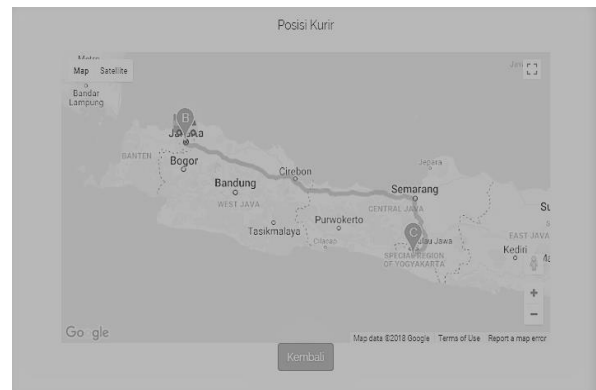
Hasil implementasi dari perancangan aplikasi Qurban App dapat dilihat pada Gambar 5, Gambar 6 dan Gambar 7.



Gambar 5. Halaman muka Qurban App



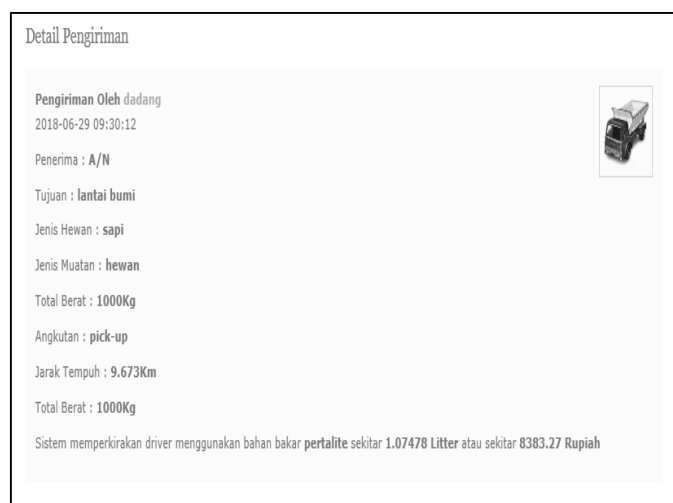
Gambar 6. Halaman hasil rute terpendek distribusi qurban



Gambar 7. Halaman tracking posisi kurir yang mendistribusi qurban

Aplikasi Qurban App memanfaatkan *Google Maps API* untuk melakukan pengolahan data guna merepresentasikan gambar tiap jalur yang memungkinkan dilewati dengan metode mengemudi dan menyimpan nilai jarak dari tiap kelokan yang ada ke dalam indeks *array*. Nilai mentah tiap jalur yang telah diberikan *Google Maps API* diproses oleh sistem. Sistem menyiapkan aturan indeks kemudian menghitung tiap indeks *array* yang berisi data jarak yang sebelumnya diberikan oleh *Google Maps API*, menggunakan algoritma *Floyd Warshall*. Jika ditemukan nilai nol maka sistem akan mengabaikan

titik tersebut karena nilai hitung antara suatu titik dengan dirinya akan tetap bernilai nol. Sistem menyimpan tiap nilai indeks *array*, jika menemukan nilai baru yang lebih kecil, nilai yang lebih besar maka diganti dengan yang lebih kecil, kemudian disimpan ke dalam *array*.



Gambar 8. Halaman estimasi biaya transportasi penyaluran paket kurban

Nilai terkecil (total jarak) yang diperoleh dari perhitungan *Floyd Warshall* disiapkan dan sistem membuat jalur yang kemudian akan dipresentasikan oleh *Google Maps API*. Sistem melakukan penghubungan jalur yang selanjutnya akan diproses oleh *Google Maps API* menjadi direksi jalur. Jalur tersebut kemudian ditampilkan ke peta seperti yang ditunjukkan pada Gambar 6.

Posisi kurir kurban dapat dideteksi oleh lembaga penyalur paket kurban (Gambar 7) dengan memanfaatkan

fungsi geolokasi milik *Google Maps API*. Fungsi ini digunakan untuk mengambil titik koordinat kurir saat menggunakan aplikasi dan menyimpan posisi tersebut ke suatu variabel. Selanjutnya dibuat marker untuk *interface info* posisi diri kurir secara *real time*. Setelah kurir selesai mendistribusikan paket-paket kurban, sebagaimana alur bisnis pada Gambar 1, admin akan menerima informasi total biaya transportasi yang dihabiskan seperti yang ditampilkan pada Gambar 8.

Jenis kendaraan yang dipakai untuk pengiriman hanya jenis *pick-up* dan truk, sistem membuat logika percabangan untuk mengalkulasikan berapa liter bahan bakar yang akan keluar serta berapa rupiah bahan bakar tersebut. Untuk mobil *pick up*, angkutan tersebut menggunakan 1 liter bahan bakar untuk berjalan sejauh 9 kilometer, jadi sistem membagi jarak tempuh yang dilalui kurir dengan 9 kilometer sehingga didapatkan berapa liter bahan bakar keluar sekali jalan. Untuk truk, angkutan tersebut menggunakan 1 liter bahan bakar untuk berjalan sejauh 3,3 kilometer, jadi sistem membagi jarak tempuh yang dilalui kurir dengan 3,3 kilometer lalu didapatkan berapa liter bahan bakar keluar sekali jalan. Bahan bakar untuk mobil *pick up* diatur dalam kondisi *default* menggunakan bahan bakar pertalite, harga pertalite tahun 2018 regional Jogja adalah Rp 7800 rupiah per liter, sedangkan pada truk menggunakan bahan bakar dextrite, harga dextrite tahun 2018 regional Jogja adalah Rp 8100 rupiah per liter.

Pengujian yang dilakukan pada *Qurban App* adalah pengujian fungsional dan pengujian validitas. Pengujian fungsional (Tabel XIII) menggunakan metode *Black Box*, dilakukan dengan menguji semua fitur yang ada di aplikasi *Qurban App* dengan tujuan untuk mengetahui apakah semua fitur tersebut dapat berjalan secara normal, sesuai yang diharapkan dan sesuai dengan kebutuhan *user*. Apabila ada aksi-aksi yang tidak normal, *Qurban App* dapat merespon balik *user* dan menangani kesalahan tersebut.

TABEL XIII. PENGUJIAN FUNGSIONAL

No	Skenario Pengujian	Test Case	Hasil yang diharapkan	Kesimpulan
1	Melakukan login dengan memasukkan email dan password	Email: admin@lazismu.com Password: 123123	Sistem menolak status login dikarenakan password yang dimasukkan pengguna salah atau tidak valid dan memunculkan notifikasi kesalahan	Sesuai
2	Mengakses halaman yang di luar batas hak akses	Kurir mengakses halaman /halaman/rute	Sistem menampilkan notifikasi 404 not found	Sesuai
3	Melakukan reset password dan menekan tombol reset	Email: admiin@lazismu.com	Sistem menolak reset password karena email yang dimasukkan tidak terdaftar dan sistem menampilkan notifikasi alamat email tidak valid	Sesuai
4	Melakukan daftar akun dan memasukkan password	Nama: kurir Email: kurir@lazismu.com Password: 123	Sistem menampilkan notifikasi bahwa password yang dimasukkan kurang dari 6 karakter	Sesuai
5	Melakukan daftar akun dan memasukkan konfirmasi password	Password: kurir123 Konfirmasi password: kurir	Sistem menampilkan notifikasi kesalahan konfirmasi password karena password konfirmasi yang dimasukkan tidak sesuai dengan password sebelumnya	Sesuai
6	Berhasil mendaftarkan akun baru	Redirect halaman beranda	Sistem menampilkan notifikasi bahwa pengguna masih berstatus member dan belum dapat mengakses fitur-fitur	Sesuai

			lainnya sebelum admin mengesahkan jabatan pengguna.	
7	Melakukan penambahan data dengan memasukkan data pengiriman secara lengkap dan menekan tombol tambah data	Alamat: Jenis muatan: hewan Jenis hewan: sapi Berat: 5 Jumlah jamaah: 50	Sistem menampilkan notifikasi gagal tambah data karena alamat kosong dan tidak diisi.	Sesuai
8	Membuka halaman yang terdapat maps	Lokasi perangkat tidak aktif	Sistem menampilkan notifikasi aktifkan lokasi perangkat untuk dapat menggunakan sistem	Sesuai
9	Membuka halaman yang terdapat maps	Lokasi perangkat aktif	Sistem menunjukkan lokasi pengguna saat ini	Sesuai
10	Menghapus akun staff pada halaman kelola staff	Klik aksi hapus pada akun yang ingin dihapus	Sistem menampilkan notifikasi berhasil hapus	Sesuai

Pengujian validitas dilakukan dengan cara membandingkan solusi yang dikeluarkan oleh aplikasi dengan perhitungan manual. Ini dilakukan untuk membuktikan bahwa aplikasi dapat melakukan kalkulasi secara benar sehingga memberikan keluaran yang benar pula. Hasil pengujian menunjukkan bahwa aplikasi dapat berjalan dengan baik sesuai fungsinya dan algoritma *Floyd Warshall* dapat memecahkan pencarian rute terdekat distribusi paket kurban sekaligus dapat menghitung total biaya transportasi pengiriman paket kurban.

VI. KESIMPULAN DAN SARAN

Kesimpulan yang dapat diambil adalah:

- 1) Masalah penentuan rute-rute distribusi yang singkat dalam pengiriman paket kurban telah dapat dipecahkan menggunakan pendekatan Teknologi Informasi (TI), melalui penerapan algoritma Floyd Warshall.
- 2) Hasil perhitungan metode Floyd Warshall tersebut dapat menentukan rute terpendek mana saja yang dapat ditempuh dari lokasi mitra peternak menuju ke tempat pendistribusian paket kurban dengan total jarak tempuh yang tersingkat.
- 3) Implementasi dari perancangan perangkat lunak telah menghasilkan purwarupa aplikasi Qurban App yang dapat mengelola data pengiriman hewan kurban berupa data muatan dan kapasitasnya, data jenis hewan dan beratnya serta bahan bakar transportasi, menampilkan rute terdekat dan melakukan *tracking* terhadap kurir yang mengirimkan paket kurban, semuanya dapat dilakukan secara benar sesuai fungsinya.

Saran bagi pengembangan penelitian yakni hendaknya dalam konteks distribusi paket kurban juga melibatkan penentuan kuantitas paket kurban yang harus disalurkan sesuai kebutuhan lokasi distribusi.

UCAPAN TERIMA KASIH

Terima kasih kepada Jurusan Teknik Informatika UII yang telah memberikan dukungan penuh pendanaan melalui program penelitian kolaborasi dosen dan mahasiswa periode ke 3.

REFERENSI

- [1] Y. Baskara, M. Adam, I. Nasution, T.M. Lubis, T. Armansyah, M. Hasan, "Tinjauan Apek Kesejahteraan Hewan Pada Sapi Potong Di Rumah Pemotongan Hewan Kodya Banda Aceh", *Jurnal Medika Veterinaria*, Vol.9, No.2, pp. 149-153, Agustus 2015.
- [2] K.M. Berutu, "Dampak Lama Transportasi Terhadap Penyusutan Bobot Badan, pH Daging Pasca Potong dan Analisis Biaya Transportasi Sapi Potong Peranakan Ongole (PO) dan Shorthorn", Skripsi, Departemen Peternakan Fakultas Pertanian, USU, pp. 1-3, Desember 2007.
- [3] I.I. Wahyudi, A. Arifin, "Aplikasi Monitoring Pengiriman Barang dengan Algoritma Dijkstra", *Prosiding Konferensi Nasional Teknologi Informasi dan Aplikasinya (KNTIA) Chapter IV*, Fakultas Ilmu Komputer, Universitas Sriwijaya, pp. E1-E5, 08 Oktober 2016.
- [4] F.W. Ningrum, T. Andrasto, "Penerapan Algoritma Floyd Warshall dalam Menentukan Rute Terpendek Pada Pemodelan Jaringan Pariwisata di Kota Semarang", *Jurnal Teknik Elektro* Vol.8, No.1, pp. 21-24, Januari-Juni 2016.
- [5] R.A.D. Novandi, "Perbandingan Algoritma Dijkstra dan Algoritma Floyd Warshall dalam Penentuan Lintasan Terpendek (Single Pair Shortest Path)", *Makalah IF2551 Strategi Algoritmik*, pp.1-5, 2007 (belum dipublikasikan).
- [6] J.J. Siang, "Matematika Diskrit dan Aplikasinya Pada Ilmu Komputer", Yogyakarta, Andi, 2002.
- [7] J.J. Siang, "Riset Operasi Dengan Pendekatan Algoritmik", Yogyakarta, ANDI, 2014.
- [8] R.S. Pressman, "Rekayasa Perangkat Lunak Pendekatan Praktis", Yogyakarta, Andi, 2002.