

IMPLEMENTASI ALGORITME A-STAR UNTUK PEMETAAN KOORDINAT TUMBUHAN LANGKA BERBASIS WEB

Implementation of A-Star Algorithm for Web-Based Rare Plant Coordinating

Ahmad Fauzi, Supriyadi, Yessy Yanitasari

Prodi Informatika, STMIK Kharisma Karawang

Jl. Pangkal Perjuangan Km. 1 Bypass, Tanjungpura, Karawang Barat, Kabupaten Karawang, Jawa Barat 41316, Indonesia

Telp. 08118454800

E-mail : fauzi@pupuk-kujang.co.id, fnfcreator@stmik-kharisma.ac.id, yessy.yanitasari@gmail.com

(Makalah diterima, 15 Desember 2017 – Disetujui, 03 Desember 2018)

ABSTRAK

Pencarian rute yang tepat banyak dibahas dalam dunia informatika. Hal ini terkait dengan semakin meningkatnya kebutuhan transportasi, distribusi, dan industri. Metode untuk menyelesaikan masalah pencarian jalur terpendek dapat dilakukan menggunakan dua metode, yaitu metode konvensional dan heuristik. Metode konvensional menggunakan pendekatan matematis yang mudah dipahami, tetapi membutuhkan waktu relatif lama. Agar waktu pencarian jadi lebih cepat, maka dibutuhkan pendekatan heuristik dengan parameter yang lebih banyak. Algoritme A-star merupakan metode heuristik yang sangat efektif dalam pencarian rute terpendek. Pada penelitian ini dikembangkan aplikasi berbasis web dengan mengintegrasikan algoritme A-star untuk mencari lokasi tanaman langka di kawasan konservasi. Untuk menentukan jarak antarkoordinat tanaman menggunakan formulasai *haversine*. Aplikasi ini sangat penting karena banyaknya tumbuhan dan terbatasnya pemandu peneliti dalam mencari lokasi tanaman langka dalam kawasan konservasi. Panduan disajikan dalam bentuk graph yang mudah dipahami disertai dengan informasi mengenai tumbuhan yang dicari.

Kata kunci: rute terpendek, algoritme A-star, tanaman langka

ABSTRACT

The path finding optimization is the most widely discussed issues in the informatics scope. This was related to the increased needs of the transportation, distribution and industry. The method for solving search problems for the shortest path can be done using two methods, namely the conventional method and the heuristic method. The conventional method uses a mathematical approach that is easy to understand, but the search results require a relatively long time. In order for search time to be faster, a heuristic approach is needed, although it requires more parameters. A-star algorithm is a heuristic route search method that is very effective in finding the shortest route. In this study a web-based application was developed by integrating the A-star algorithm to find the location of rare plants in the conservation area. As for determining the distance between the coordinates of the plant using haversine formula. This application is very important because of the large number of plants and the limited guidance of researchers in finding the location of rare plants in conservation areas. The guide is presented in the form of an easy-to-understand graph accompanied by information about the plants sought.

Key words: the shortest path, A-star algorithm, rare plants

PENDAHULUAN

Semua tempat di bumi pada umumnya dipetakan menggunakan garis lintang dan garis bujur. Seluruh permukaan bumi diproyeksikan dengan empat model yaitu Sinusoidal, Mollweide, Gall-Petters, dan Equal-Earth (Šavrič *et al.*, 2019). Saat ini hampir semua tempat di bumi sudah dipetakan oleh aplikasi komputasi seperti *Google Maps*, *Bing Maps*, *Waze* dan *Map Quest*. Aplikasi tersebut dilengkapi informasi lokasi dan citranya dengan bantuan pencitraan satelit, sehingga memudahkan manusia menuju ke suatu tempat sekalipun belum pernah ke tempat tersebut. Kenyataannya, titik koordinat *longitude* dan *latitude* dalam aplikasi tersebut tidak sama karena perbedaan penggunaan datum (Sajeevan, 2008). Informasi yang dihasilkan dari cara ini cukup lengkap seperti pencarian rute, jarak tempuh, estimasi waktu, lalu lintas, dan banyak lagi informasi lain yang bisa diperoleh. Layanan pencarian rute sangat digemari oleh masyarakat yang bepergian setiap hari. Ketika pergi menuju ke suatu tempat, mereka menginginkan rute terpendek dan ditempuh dengan waktu yang cepat.

Pencarian rute terpendek paling banyak dibahas dan dipelajari sejak akhir tahun 1950 (Purwananto *et al.*, 2005), karena industri tumbuh cepat sehingga kebutuhan akan jalur distribusi meningkat. Pencarian rute terkait erat dengan biaya, terutama untuk transportasi karena mempengaruhi biaya penggunaan bahan bakar dalam pengiriman barang atau jasa dari satu lokasi ke lokasi lain (Juniansyah, 2016). Lokasi tempat atau benda pada peta dinyatakan dalam sistem koordinat. Di Indonesia sendiri digunakan dua sistem koordinat peta, yaitu koordinat geografis dan *Universal Transverse Mercator* atau UTM (Budiyanto, 2016). Sistem koordinat geografis menggunakan satuan derajat menit detik dan terdiri atas garis lintang (*latitude*) dan garis bujur (*longitude*), sedangkan sistem koordinat UTM menggunakan satuan meter untuk menunjukkan jarak sebenarnya di lapangan (Budiyanto, 2016).

Metode pencarian jalur terpendek dapat dibagi menjadi dua, yaitu metode algoritme konvensional dan heuristik (Mutakhiroh *et al.*, 2007). Metode algoritme konvensional menggunakan perhitungan matematis umum, sedangkan metode heuristik menggunakan pendekatan berbasis pengetahuan. Metode algoritme heuristik membutuhkan banyak parameter dan lebih sulit dipahami meskipun menghasilkan jalur yang lebih variatif, sedangkan algoritme konvensional hanya membutuhkan titik awal, titik akhir, dan lebih mudah dipahami meskipun memerlukan waktu lebih lama (Ferdiansyah *et al.*, 2013).

Salah satu metode pencarian rute terpendek heuristik yang paling sering digunakan adalah metode A* (*A-star*). Metode *A-star* ditemukan pertama kali oleh Peter Hart, Nils Nilsson, dan Bertram Raphael pada tahun 1968 (Hart *et al.*, 1968). Metode A* adalah algoritme pencarian rute

terpendek (*shortest path*) yang merupakan perbaikan dari algoritme *Best First Search* (BFS) dengan memodifikasi fungsi heuristik untuk memberikan hasil yang optimal. Implementasi metode tersebut biasanya dilakukan pada peta atau dengan bantuan aplikasi *Geographic Information System* (GIS) untuk wilayah yang sangat luas, misalnya kabupaten.

Pencarian rute menggunakan aplikasi global seperti *Google Maps* saat ini belum bisa menjangkau jalan kecil di perumahan atau di suatu kawasan. Apalagi jalan yang jarang dilalui masyarakat umum seperti jalan di hutan lindung atau kawasan konservasi. Tidak banyak masyarakat umum yang melewati jalan tersebut, biasanya hanya para peneliti dan petugas terkait. Banyaknya varietas tanaman yang ada di kawasan konservasi yang cukup luas menyulitkan peneliti mencari tanaman yang dibutuhkan, walaupun sudah dipandu oleh petugas.

Penerapan algoritme pencarian jarak terdekat bisa diimplementasikan untuk mencari tumbuhan pada suatu kawasan konservasi, dimana tumbuhan tersebut telah ditandai lokasinya berdasarkan koordinat. Penandaan dilakukan oleh peneliti untuk kebutuhan penelitian, identifikasi, dan pendataan keanekaragaman hayati di suatu kawasan agar mudah ditemukan oleh peneliti selanjutnya. Beberapa masalah yang dapat diatasi dengan algoritme pencarian rute pada lokasi konservasi seperti tenaga pemandu yang terbatas, perangkat khusus pencarian rute yang mahal, dan banyaknya tumbuhan yang menyulitkan pencarian dalam kawasan serta belum adanya teknologi khusus untuk pendataan tumbuhan di kawasan konservasi.

Metode A* banyak diimplementasikan untuk menentukan nilai optimum dalam pencarian rute, seperti penelitian sebelumnya yang bisa dilihat pada Tabel 1.

Pada penelitian ini dikembangkan aplikasi komputasi berbasis web untuk mencari jalur terpendek menuju koordinat tanaman langka di kawasan konservasi menggunakan metode *A-star* dan formula haversine. Aplikasi tersebut dikembangkan menggunakan *Software Development Life Cycle* (SDLC) Waterfall (Satzinger *et al.*, 2010). Penelitian mengambil tempat di Taman Kehati PT Pupuk Kujang Cikampek, Jawa Barat. Tujuan penelitian adalah untuk membuat aplikasi yang dapat digunakan dalam membantu pencarian dan pendataan tumbuhan langka pada kawasan konservasi. Penentuan koordinat tanaman diperoleh dengan memanfaatkan layanan *Global Positioning System* (GPS).

BAHAN DAN METODE

Bahan penelitian yang digunakan berupa data tumbuhan yang diambil dari Taman Kehati PT Pupuk Kujang yang terdiri atas 64 familia, 165 spesies, dan 3.410 tumbuhan (PKC, 2015) yang tersebar di 15 blok kawasan konservasi (Tabel 2).

Tabel 1. Penelitian terkait menggunakan algoritme A*

Peneliti	Hasil penelitian
Adipranata <i>et al.</i> (2007)	Pencarian rute pada peta menggunakan metode A <i>star</i> selalu berhasil menemukan solusi rute apabila memang terdapat jalan dari titik awal ke tujuan. Metode A <i>star</i> menghasilkan jarak hasil rute dan total biaya tempuh rute yang lebih baik daripada metode BFS.
Yuliana <i>et al.</i> (2012)	Metode A* lebih baik dalam mendapatkan jalur yang lebih pendek dibandingkan dengan metode <i>Greedy</i> dalam penyelesaian <i>puzzle</i> 8. Tapi metode A* membutuhkan waktu yang lebih lama dalam penyelesaiannya dibanding metode <i>Greedy</i> .
Mutiana <i>et al.</i> (2013)	Metode A- <i>star</i> tidak menjamin rute yang terpilih adalah yang terbaik atau paling optimal, karena agent tidak bersifat <i>full observable</i> setelah proses <i>backtrack</i> dilakukan (menutup node yang macet), sehingga hanya menghitung jarak dari node yang dilaluinya saja.
Irsyad dan Rasila (2015)	Metode A <i>star</i> mampu menentukan rute terdekat untuk mencari ruang dan gedung pada kawasan universitas dengan melakukan 10 kali percobaan secara acak.
Ostrowski <i>et al.</i> (2015)	Perbandingan algoritme pencarian rute terpendek menggunakan A <i>star</i> (A*), Dijkstra, Dijkstra <i>binary</i> , Dijkstra <i>bi-directional</i> , Dijkstra Fibonacci, Bellman-Ford. Hasil pengujian dengan jumlah verteks semakin banyak, A- <i>star</i> memiliki waktu eksekusi paling cepat dan stabil.
Sharma dan Pal (2015)	Algoritme A- <i>star</i> memiliki waktu tercepat untuk menemjkan rute terpendek dibandingkan algoritme Djikstra. Pengujian dilakukan dengan menambahkan rintangan pada rute yang akan diproses.

Tabel 2. Sebaran tumbuhan di kawasan Taman Kehati

Lokasi	Jumlah tumbuhan
Blok hutan 1	325
Blok hutan 2	238
Blok hutan 3	541
Blok hutan 4	328
Blok hutan 5	277
Blok hutan 6	681
Blok hutan 7	1
Blok hutan 8	71
Blok hutan 9	314
Blok hutan 10	16
Blok hutan 12	5
Penyangga blok wisma	3
Golf	38
Rawa	1
Show window	1

Tabel 3. Perangkat lunak

Perangkat	Fungsi
Ubuntu 18 64bit	Sistem Operasi Linux
<i>Geany</i>	Teks <i>editor</i> untuk pengolahan kode program
<i>Hypertext Preprocessor</i>	Bahasa pemrograman <i>script server side</i> untuk pengembangan <i>web</i> .
<i>Firefox, Chrome Browser</i>	<i>Software</i> untuk menguji tampilan <i>web</i>
<i>MySql</i>	<i>Software</i> untuk pengolahan basisdata

Alat yang digunakan meliputi perangkat keras dan perangkat lunak. Untuk perangkat keras menggunakan komputer dengan spesifikasi processor CPU 3337U

@ 1,80 GHz 1,80 GHz RAM 4,00 GB GPU710MB.
Perangkat lunak dapat dilihat pada Tabel 3.

1. Metode SDLC Waterfall

Dalam membangun metode SDLC *Waterfall* digunakan metodologi pengembangan perangkat lunak sistematis dan sekuensial (Gambar 1).

2. Project Planning Phase

Tahap perencanaan adalah proses dasar memahami mengapa sistem pencarian rute menuju lokasi tumbuhan dalam kawasan konservasi harus dibangun dan menentukan bagaimana membangun sistem tersebut. Pada tahapan ini dilakukan penelitian terlebih dahulu untuk mendapatkan data dan informasi terkait. Tujuannya adalah untuk menyusun rencana proyek sehingga membantu dalam bekerja sesuai dengan rencana (Zhou dan Shao, 2012). Teknik pengumpulan data yang dilakukan adalah melalui wawancara dan studi literatur.

3. Analysis Phase

a. Pencarian rute dengan A-star

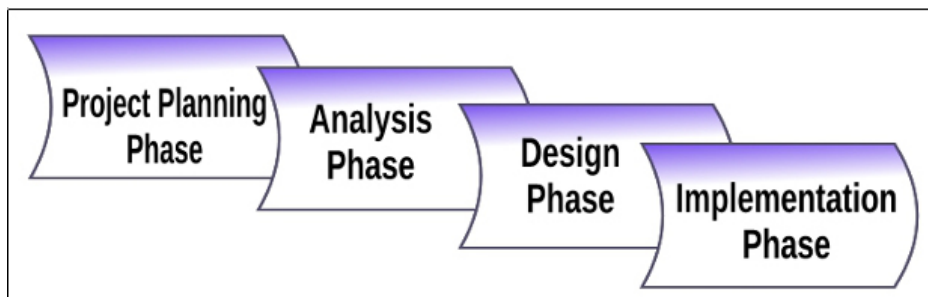
Beberapa titik lokasi (*vertex*) yang terhubung oleh jalan (*edge*) dengan jarak (*d*) dapat digambarkan sebagai sebuah *graph* (Brandes dan Cornelsen, 2003). Secara matematis merupakan himpunan atau sekumpulan *vertex* (*V*) atau *node* (*n*) dan *edge* (*E*) yang diformulasikan sebagai berikut :

$$\begin{aligned} G &= \{V, E\}; \\ V &= \{v_1, v_2, v_3, \dots, v_m\}; \\ E &= \{e_1, e_2, e_3, \dots, e_m\}; \end{aligned}$$

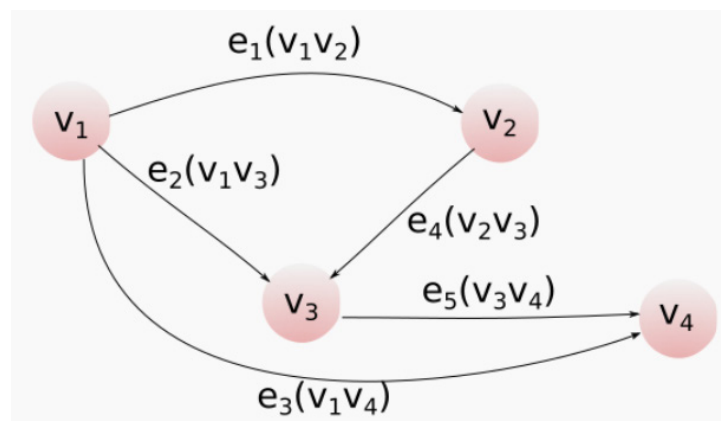
Algoritme A-star merupakan algoritme untuk optimasi dalam pencarian rute. Menurut Hart *et al.* (1968) diumpamakan jarak optimal dari titik awal (*s*) yang melewati *n* menuju node setelah *n* atau *successor(n)* sebagai fungsi *f(n)* dan dikenal juga dengan *evaluation function*. Sedangkan jarak optimal dari titik *s* menuju *n* disebut fungsi *g(n)* dan prediksi jarak optimal dari *n* menuju *successor(n)* disebut fungsi *h(n)*. Secara umum diformulasikan sebagai berikut:

$$f(n) = g(n) + h(n) \dots\dots (1)$$

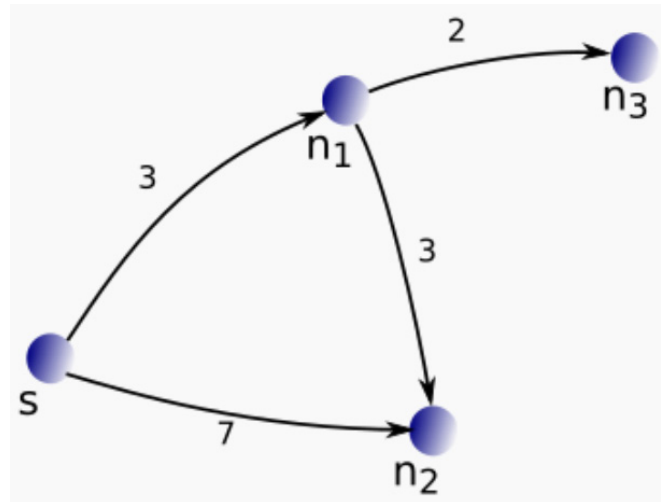
Gambar 3 menunjukkan $g(n_1) = 3$, $g(n_2) = 7$, dan $g(n_3) = 3 + 2 = 5$. Fungsi *h(n)* dapat dihitung dengan fungsi heuristik, yaitu untuk menghitung estimasi jarak dari *n* ke setiap *successor(n)*. Perhitungan nilai *h(n)* bisa menggunakan rumus *euclidean distance* atau menggunakan *Haversine*. Diasumsikan jarak optimal sebenarnya dari *n* menuju *successor(n)* adalah $h^*(n)$. Jika $h(n) < h^*(n)$ maka algoritme A-star layak diaplikasikan (Jiang *et al.*, 2015).



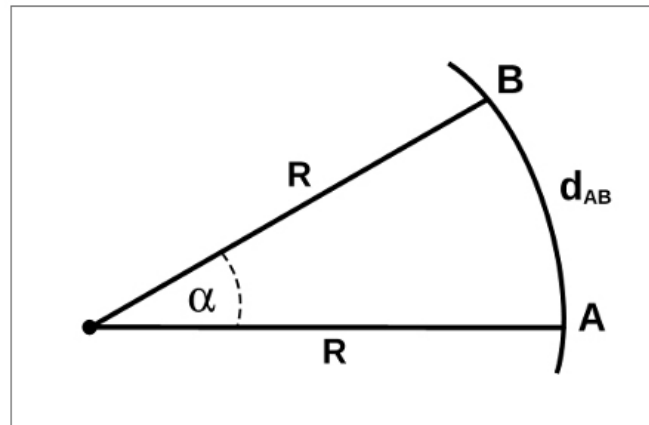
Gambar 1. SDLC Waterfall Approach model



Gambar 2. Struktur graph



Gambar 3. Graph berbobot dengan titik awal s (Hart *et al.*, 1968)



Gambar 4. Keterhubungan jarak (d_{AB}) dengan sudut pusat bumi (α)

b. Pencarian jarak antara dua titik menggunakan *Haversine*

Metode ini ditemukan oleh James Andrew pada tahun 1805 dan digunakan pertama kali oleh Josef de Mendoza y Ríos pada tahun 1801 untuk menghitung jarak di permukaan bumi. Haversine adalah teknik trigonometri untuk menentukan jarak antara dua koordinat dua dimensi dengan mengabaikan tinggi gunung atau kedalaman lembah (Umate dan Bhujade, 2017). Haversine efektif memetakan lokasi *indoor* (Ganesh dan Kumar, 2015) maupun *outdoor* (Chopde dan Nichat, 2013). Rumus dasar fungsi *haversine* (*hav*) adalah sebagai berikut:

$$\begin{aligned} \text{hav}(\theta) &= \sin^2\left(\frac{\theta}{2}\right) \\ &= \frac{1 - \cos(\theta)}{2} \end{aligned}$$

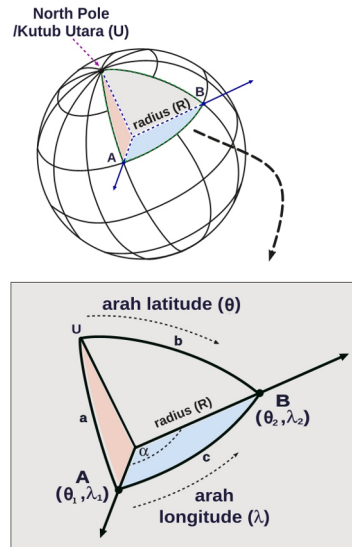
Proses perhitungan jarak di permukaan bumi dapat dihitung dengan fungsi *haversine* dengan melakukan substitusi beberapa variabel (Mwemezi dan Huang, 2011). Variabel-variabel tersebut yaitu jarak dari pusat bumi ke permukaan (R : Radius), sudut pusat bumi antara dua titik di permukaan (α), nilai *latitude* (θ) dan nilai *longitude* (λ).

Jika jarak di permukaan bumi dipetakan ke dalam grafik dua dimensi dengan α sebagai nilai sudut tengah maka hubungan α , R dan d_{AB} diilustrasikan pada Gambar 4.

Permukaan bumi umumnya dipetakan dalam bentuk sebuah bola bumi yang terdiri atas garis *latitude* dan garis *longitude*. Pada Gambar 5 dapat dilihat contoh titik A dan titik B serta jarak antara titik A dan B yang dipetakan dalam bola bumi.

Nilai α diperoleh dengan menghitung invers dari sinus (*arcsin*) seperti pada persamaan berikut:

$$\alpha = 2 \arcsin(\sqrt{\text{hav}(\theta)})$$



Gambar 5. Pemetaan jarak A ke B (d_{AB}) terhadap sudut pusat bumi (α) dan kutub utara (U)

Berdasarkan Gambar 5 dapat dihitung nilai d_{AB} dengan rumus:

$$d_{AB} = R\alpha$$

$$\begin{aligned} h_{AB} &= \text{haversine} \left(\frac{d_{AB}}{R} \right) \\ &= \text{hav} \left(\frac{d_{AB}}{R} \right) \\ &= \text{hav} (\theta_2 - \theta_1) + \cos(\theta_1) * \cos(\theta_2) * \text{hav} (\lambda_2 - \lambda_1) \end{aligned}$$

Mengacu kepada persamaan di atas maka h_{AB} dapat dihitung dengan persamaan:

$$h_{AB} = \sin^2 \left(\frac{\theta_2 - \theta_1}{2} \right) + \cos(\theta_2) * \cos(\theta_1) * \sin^2 \left(\frac{\lambda_2 - \lambda_1}{2} \right)$$

Keterangan:

ϕ_1 = Garis bujur (latitude) X_1

ϕ_2 = Garis bujur (latitude) X_2

λ_1 = Garis lintan (longitude Y_1)

λ_2 = Garis lintan (longitude Y^2)

$$h_{AB} = \text{haversine} \left(\frac{d}{R} \right)$$

c. Analisis pengembangan perangkat lunak

1. *System Activities (Actor Description)*
2. *Class Diagram*
3. *Sequence Diagram*
4. *Object Behavior (Activity Diagram)*

4) Design Phase

Tahapan ini merupakan sekumpulan proses memanipulasi informasi yang harus dilalui sehingga rancangan sistem diperoleh sebaik mungkin (Taylor dan Hoek, 2007). Proses ini berfokus pada struktur data, arsitektur perangkat lunak, representasi *interface*, dan

detail algoritme prosedural. Hasil dari tahapan ini adalah dokumen yang disebut *software requirement* yang akan digunakan - untuk melakukan aktivitas pembuatan sistem. Desain yang dikembangkan oleh peneliti menggunakan pendekatan *Object Oriented Design (OOD)*.

5) Implementation Phase

Tahapan implementasi merupakan tahap pembuatan program, termasuk penulisan kode program, pengetesan program secara terus menerus untuk mencari kesalahan yang terjadi pada program yang dibuat, kemudian meletakkan sistem untuk dioperasikan.

Penulisan kode program dilakukan dengan teknik *Object Oriented Programming (OOP)* secara modular sehingga akan mempermudah memodifikasi kode yang sudah ada. Obyek yang baru dapat dibuat tanpa mengubah kode yang sudah ada. Bagi programmer, cara ini dimungkinkan untuk dapat menyederhanakan kompleksitas dalam mendefinisikan sistem besar dan kompleks menggunakan set yang lebih kecil dari obyek yang saling terkait.

Tujuan penelitian ini adalah untuk mengimplementasikan algoritme A-Star dalam pemetaan koordinat tumbuhan langka berbasis web di Taman Keanekaragaman Hayati PT Pupuk Kujang.

HASIL DAN PEMBAHASAN

Taman Keanekaragaman Hayati (Kehati) PT Pupuk Kujang terletak dalam kawasan industri kujang Cikampek di Kec. Cikampek Kab. Karawang, bersebelahan langsung dengan kasawan perumahan karyawan PT Pupuk Kujang. Berdasarkan observasi terdapat 64 familia, 165 spesies, 3.410 tumbuhan yang tersebar di

15 blok kawasan konservasi. Pada penelitian ini dcoba mencari rute menuju lokasi suatu tumbuhan di belakang kawasan perumahan karyawan, dimana terdapat 105 tikungan dan persimpangan (Mutiana *et al.*, 2013).

Peta Kawasan

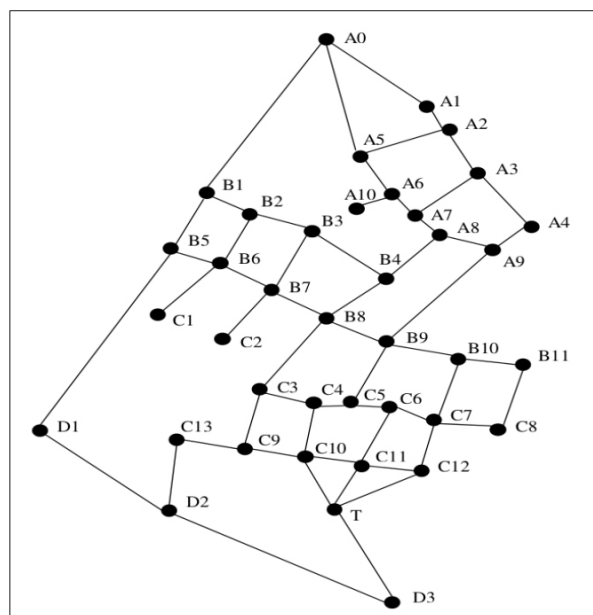
Pada Gambar 6 ditampilkan peta kawasan industri PT Pupuk Kujang Cikampek, dimana kawasan konservasi Taman Keanekaragaman Hayati berada di sebelah selatan

kawasan pabrik dan bersebelahan langsung dengan kawasan perumahan karyawan.

Setelah lokasi persimpangan dan tanaman diberi tanda, dilakukan proses pengambilan titik koordinat *longitude* dan *latitude*. Titik tersebut diambil dengan bantuan aplikasi google map. Peta tersebut kemudian dipetakan menjadi bentuk graph yang terdiri dari verteks dan dan *edges*. Graph juga merupakan gambaran sutu koordinat yang terhubung oleh sebuah fungsi (Kikelomo *et al.*, 2017). Apabila dipetakan ke dalam bentuk graph, lokasi peta kawasan bisa dilihat pada Gambar 7.



Gambar 6. Kawasan Taman Keanekaragaman Hayati PT Pupuk Kujang



Gambar 7. Lokasi percobaan dalam format graph

Perhitungan A-star

Sebagai bahan penelitian dicari rute dari node A_0 menuju tumbuhan di lokasi yang ditandai dengan titik T. Setiap node memiliki koordinat seperti pada Tabel 4:

Langkah 1. Menentukan titik koordinat A_0 dan A_1 , dimana A_0 sebagai titik (X_0, Y_0) dan A_1 sebagai titik (X_1, Y_1) (Gambar 8).

$$X_0 = \theta_0 = -6.412819740^\circ$$

$$Y_0 = \lambda_0 = 107.42163048^\circ$$

$$X_1 = \theta_1 = -6.413395510^\circ$$

$$Y_1 = \lambda_1 = 107.42234230^\circ$$

Tabel 4. Kordinat node dalam kawasan penelitian

No	Lokasi	Latitude (°)	Longitude (°)
1	A0	-6,412819740	107,42163048
2	A1	-6,413395510	107,42234230
3	A2	-6,413688700	107,42239594
4	A3	-6,414243110	107,42275536
5	A4	-6,414824170	107,42323279
6	A5	-6,413992560	107,42181122
7	A6	-6,414440350	107,42205799
8	A7	-6,414653580	107,42222428
9	A8	-6,414957440	107,42240667
10	A9	-6,415138690	107,42287874
11	A10	-6,414627000	107,42177400
12	B1	-6,414477670	107,42051840
13	B2	-6,414696230	107,42080271
14	B3	-6,414962770	107,42131770
15	B4	-6,415383910	107,42185414
16	B5	-6,415112040	107,42015898
17	B6	-6,415293280	107,42046475
18	B7	-6,415597140	107,42097437
19	B8	-6,415916990	107,42143571
20	B9	-6,416199520	107,42203116
21	B10	-6,416402090	107,42259979
22	B11	-6,416487390	107,42321670
23	C1	-6,415980960	107,42016971
24	C2	-6,416210190	107,42069006
25	C3	-6,416823230	107,42101192
26	C4	-6,417031130	107,42146254
27	C5	-6,417009810	107,42173076
28	C6	-6,417068450	107,42192924
29	C7	-6,417180390	107,42237449
30	C8	-6,417297670	107,42305040
31	C9	-6,417489580	107,42092609
32	C10	-6,417601530	107,42141426
33	C11	-6,417681490	107,42187023
34	C12	-6,417734800	107,42228329
35	C13	-6,417367000	107,42024500
36	D1	-6,417121750	107,41905928
37	D2	-6,418225230	107,42025018
38	D3	-6,419270070	107,42212772
39	T	-6,417979970	107,42163048

Langkah 2. Konversi nilai kordinat dari derajat menjadi radian karena operasi perhitungan dilakukan dengan bilangan riil.

Diketahui:

$$\begin{aligned}\pi &= 22/7 = 3,143 \\ R &= 6371000 \text{ m} \\ \theta_0 &= -6,412819740^\circ * (\pi/180) \\ &= -0.1118679078 \\ \theta_1 &= -6,413395510^\circ * (\pi/180) \\ &= -0.1118780256 \\ \Delta\theta &= -0,11197 - (-0,11196) \\ &= -1,005E-5 \\ \lambda_0 &= 107.42163048^\circ * (\pi/180) \\ &= 1.8739106567 \\ \lambda_1 &= 107.42234230^\circ * (\pi/180) \\ &= 1.8757134146 \\ \Delta\lambda &= 1.8757134146 - 1.8739106567 \\ &= 1.24E-5\end{aligned}$$

Langkah 3. Menghitung jarak A_0 ke A_1 atau fungsi $g(A_0A_1)$ dengan metode *Haversine* dalam satuan radian berikut:

$$\begin{aligned}g(A_0A_1) &= \sin^2\left(\frac{-1.005E-5}{2}\right) \cos(-0,112)_s * \\ &\quad \cos(-0,112) * \sin^2\left(\frac{1.24E-5}{2}\right) \\ &= \sin^2(-5.025E-6) \cos(-0,112)_s * \\ &\quad \cos(-0,112) * \sin^2(6.2118E-6) \\ &= (2.52546E-11) + (0.9937) * \\ &\quad (0.9937) * (3.8586E-11) \\ &= (3.8105068790535E-11)\end{aligned}$$

Langkah 4. Menghitung nilai α untuk $g(A_0A_1)$ sebagai berikut:

$$\begin{aligned}Z\alpha &= 2 * \arcsin(\sqrt{g_{A_0A_1}}) \\ &= 2 * \arcsin(\sqrt{3.8105068790535E-11}) \\ &= 1.591867861716E-5\end{aligned}$$

Hitung jarak (d) dalam satuan meter dengan mengalikan α dengan radius bumi (R):

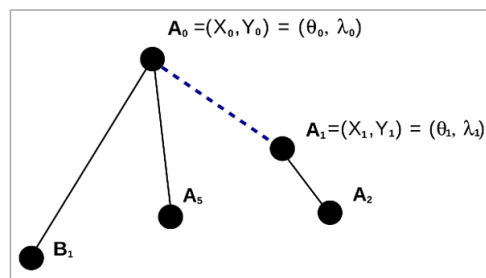
$$\begin{aligned}d(A_0A_1) &= R * \alpha \\ d(A_0A_1) &= (1.591867861716E-5) * 6371000 \\ d(A_0A_1) &= 101.42m\end{aligned}$$

Langkah 5. Menentukan jarak heuristik ($h(A_0A_1)$) antara A_1 dengan lokasi yang dituju (T). Tambahkan jarak dari node saat ini ke tujuan dengan jarak heuristik, kemudian menentukan nilai terkecil. Ilustrasinya dapat dilihat pada Gambar 9.

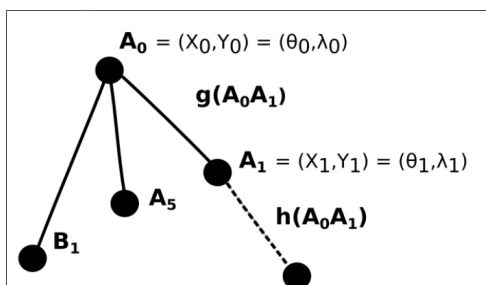
Diketahui data koordinat sebagai berikut:

$$\begin{aligned}X1 &= \theta_1 = -6,413395510 \\ Y1 &= \lambda_1 = 107,42234230 \\ X2 &= \theta_T = -6,413688700 \\ Y2 &= \lambda_T = 107,42239594\end{aligned}$$

$$\begin{aligned}\theta_1 &= -6,413395510 * (\pi/180) \\ &= -0.1119850116 \\ \theta_2 &= -6,417979970 * (\pi/180) \\ &= -0.1120650614 \\ \Delta\theta &= -0.1120650614 - (-0.1119850116) \\ &= -8.005E-05 \\ \lambda_1 &= 107,42234230 * (\pi/180) \\ &= 1.8757134547 \\ \lambda_2 &= 107,42163048 * (\pi/180) \\ &= 1.8757010255 \\ \Delta\lambda &= 1.8757010255 - 1.8757134547 \\ &= -1.24E-05\end{aligned}$$



Gambar 8. Contoh pencarian rute dalam graph



Gambar 9. Lokasi penelitian dalam format graph

$$\begin{aligned}
 h(A_0A_1) &= \sin^2\left(\frac{-8.005E-05}{2}\right) + \cos(-0,112)_s * \\
 &\quad \cos(-0,112) * \sin^2\left(\frac{-1.243E-05}{2}\right) \\
 &= \sin^2(-4.002E-05) + \cos(-0,112)_s * \\
 &\quad \cos(-0,112) * \sin^2(-6.215E-06) \\
 &= (1.602E-09) + (0.9937) * \\
 &\quad (0.9937) * (3.862E-11) \\
 &= (1.6401E-09)
 \end{aligned}$$

Langkah 6. Menghitung nilai α untuk $h(A_0A_1)$ sebagai berikut:

$$\begin{aligned}
 \alpha &= 2 * \arcsin(\sqrt{h_{A_0A_1}}) \\
 &= 2 * \arcsin(\sqrt{1.6401E-09}) \\
 &= 8.09971E-05
 \end{aligned}$$

Hitung jarak perkiraan (d) dalam satuan meter dengan mengalikan α dengan radius bumi (R):

$$\begin{aligned}
 d(A_0A_1) &= R * \alpha \\
 d(A_0A_1) &= (6371000) * 8.09971E-05 \\
 d(A_0A_1) &= 516.03m
 \end{aligned}$$

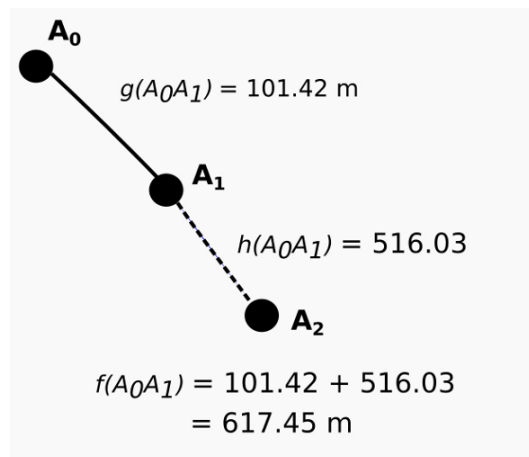
Dengan demikian, hasil $f(n)$ bisa dihitung seperti pada Gambar 10. Perhitungan selanjutnya sama sampai ditemukan rute terpendek (Tabel 5).

Berdasarkan Tabel 5 didapatkan rute untuk menuju lokasi tumbuhan yang dicari adalah melalui $A_0 - A_5 - A_6 - A_7 - A_8 - B_4 - B_8 - C_3 - C_4 - C_{10} - T$

Perangkat Lunak Komputer

Perangkat lunak komputer yang digunakan merupakan aplikasi berbasis web yang dikembangkan dengan pendekatan berorientasi obyek. Kebutuhan analisis perangkat lunak komputer terdiri atas deskripsi aktor yang ditunjukkan pada Tabel 6.

Semua aktor akan berinteraksi dengan sistem atau aplikasi sesuai dengan hak akses nya masing-masing. Hak akses aktor terhadap sistem berkaitan dengan fungsi yang disediakan di dalamnya seperti pilihan menu. Proses komunikasi aktor dengan sistem digambarkan pada gambar 11 yang umumnya dikenal dengan *use case diagram*.



Gambar 10. Penentuan nilai fungsi evaluasi (f)

Tabel 5. Urutan metode A^* untuk menentukan rute

N	N.Next	Nilai	Heuristic	Jarak	Path
A_0	A_1	101.42	516.03	617.45	A_0-A_1
A_0	A_5	131.93	443.83	575.76	A_0-A_5
A_0	B_1	221.55	408.37	629.92	A_0-B_1
A_5	A_2	72.91	484.61	557.52	$A_0-A_5-A_2$
A_5	A_6	56.77	396.41	453.18	$A_0-A_5-A_6$
A_6	A_7	30.00	375.65	405.65	$A_0-A_5-A_6-A_7$
A_6	A_{10}	37.62	373.17	410.79	$A_0-A_5-A_6-A_{10}$
A_7	A_3	74.34	433.71	508.06	$A_0-A_5-A_6-A_7-A_3$
A_7	A_8	39.34	346.86	386.20	$A_0-A_5-A_6-A_7-A_8$
A_8	A_9	55.92	344.73	400.65	$A_0-A_5-A_6-A_7-A_8-A_9$

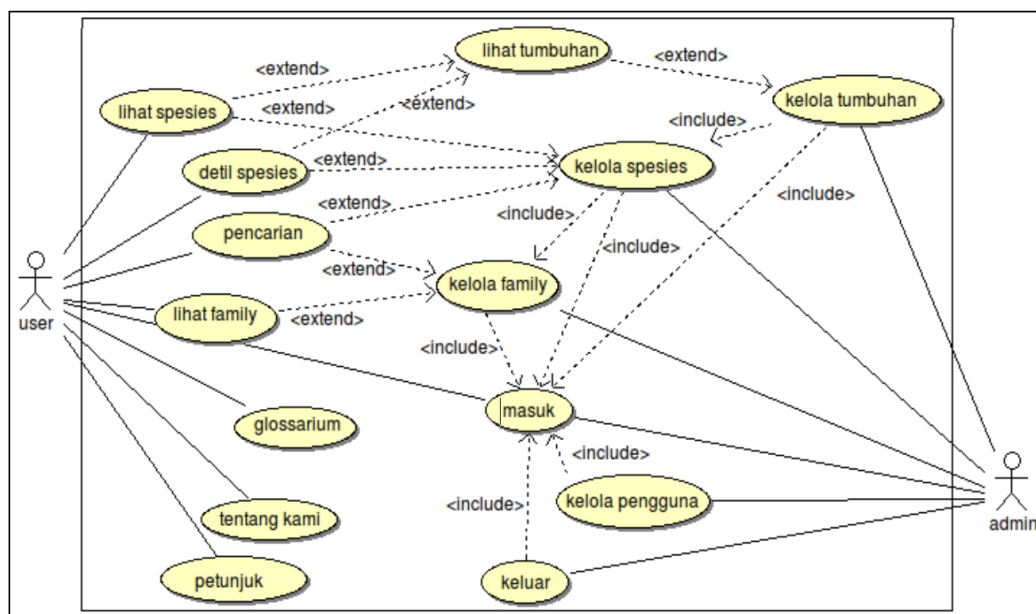
Tabel 5. Urutan metode A* untuk menentukan rute

lanjutan

N	N.Next	Nilai	Heuristic	Jarak	Path
A ₈	B ₄	77.31	289.72	367.03	A ₀ -A ₅ -A ₆ -A ₇ -A ₈ -B ₄
B ₄	B ₄	75.54	337.27	412.81	A ₀ -A ₅ -A ₆ -A ₇ -A ₈ -B ₄ -B ₃
B ₄	B ₈	75.18	230.40	305.58	A ₀ -A ₅ -A ₆ -A ₇ -A ₈ -B ₄ -B ₈
B ₈	B ₇	62.16	274.70	336.86	A ₀ -A ₅ -A ₆ -A ₇ -A ₈ -B ₄ -B ₈ -B ₇
B ₈	B ₉	72.91	202.87	275.78	A ₀ -A ₅ -A ₆ -A ₇ -A ₈ -B ₄ -B ₈ -B ₉
B ₈	C ₃	111.12	145.66	256.77	A ₀ -A ₅ -A ₆ -A ₇ -A ₈ -B ₄ -B ₈ -C ₃
C ₃	C ₄	54.90	107.13	162.02	A ₀ -A ₅ -A ₆ -A ₇ -A ₈ -B ₄ -B ₈ -C ₃ -C ₄
C ₃	C ₉	74.70	95.03	169.73	A ₀ -A ₅ -A ₆ -A ₇ -A ₈ -B ₄ -B ₈ -C ₃ -C ₉
C ₄	C ₃	54.90	145.66	200.55	A ₀ -A ₅ -A ₆ -A ₇ -A ₈ -B ₄ -B ₈ -C ₃ -C ₄ -C ₃
C ₄	C ₅	29.73	108.44	138.18	A ₀ -A ₅ -A ₆ -A ₇ -A ₈ -B ₄ -B ₈ -C ₃ -C ₄ -C ₅
C ₄	C ₁₀	63.65	48.39	112.04	A ₀ -A ₅ -A ₆ -A ₇ -A ₈ -B ₄ -B ₈ -C ₃ -C ₄ -C ₁₀
C ₁₀	C ₉	55.36	95.03	150.39	A ₀ -A ₅ -A ₆ -A ₇ -A ₈ -B ₄ -B ₈ -C ₃ -C ₄ -C ₁₀ -C ₉
C ₁₀	C ₁₁	51.16	42.47	93.63	A ₀ -A ₅ -A ₆ -A ₇ -A ₈ -B ₄ -B ₈ -C ₃ -C ₄ -C ₁₀ -C ₁₁
C ₁₀	T	48.39	0.00	48.39	A ₀ -A ₅ -A ₆ -A ₇ -A ₈ -B ₄ -B ₈ -C ₃ -C ₄ -C ₁₀ -T
SELESAI				A₀-A₅-A₆-A₇-A₈-B₄-B₈-C₃-C₄-C₁₀-T	

Tabel 6. Deskripsi aktor

No	Aktor	Deskripsi
1	Admin	Pengguna sistem yang memiliki akses penuh terhadap sistem
2	User	Pengguna sistem yang hanya dapat melihat informasi yang telah disediakan

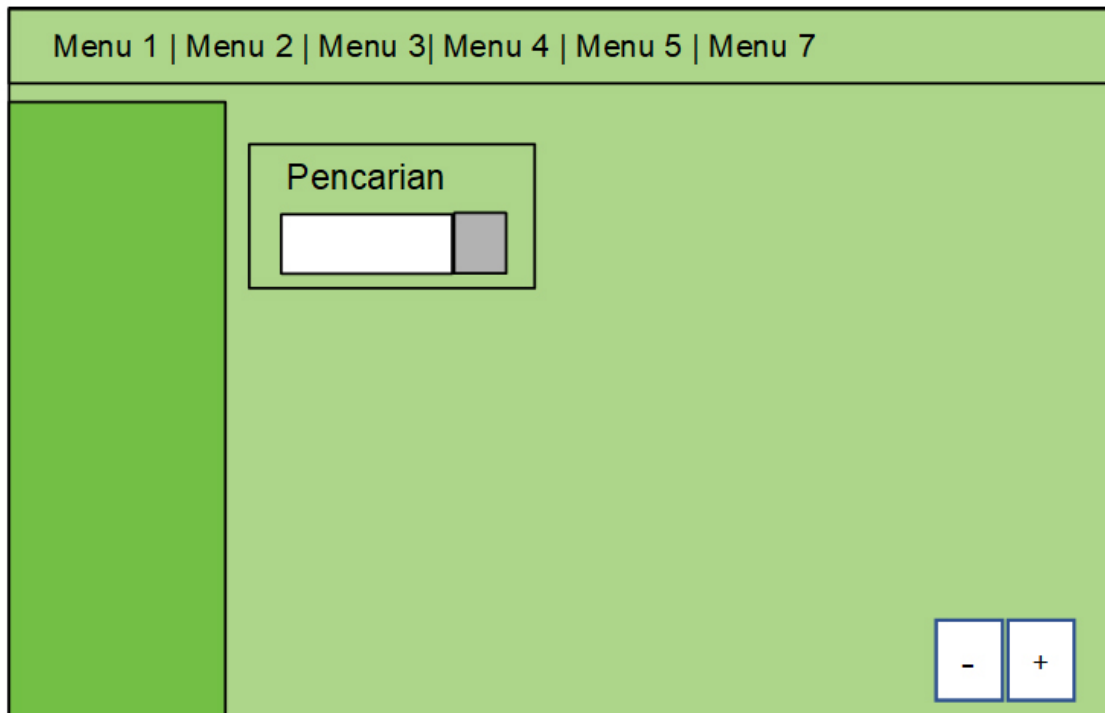


Gambar 11. Use case System

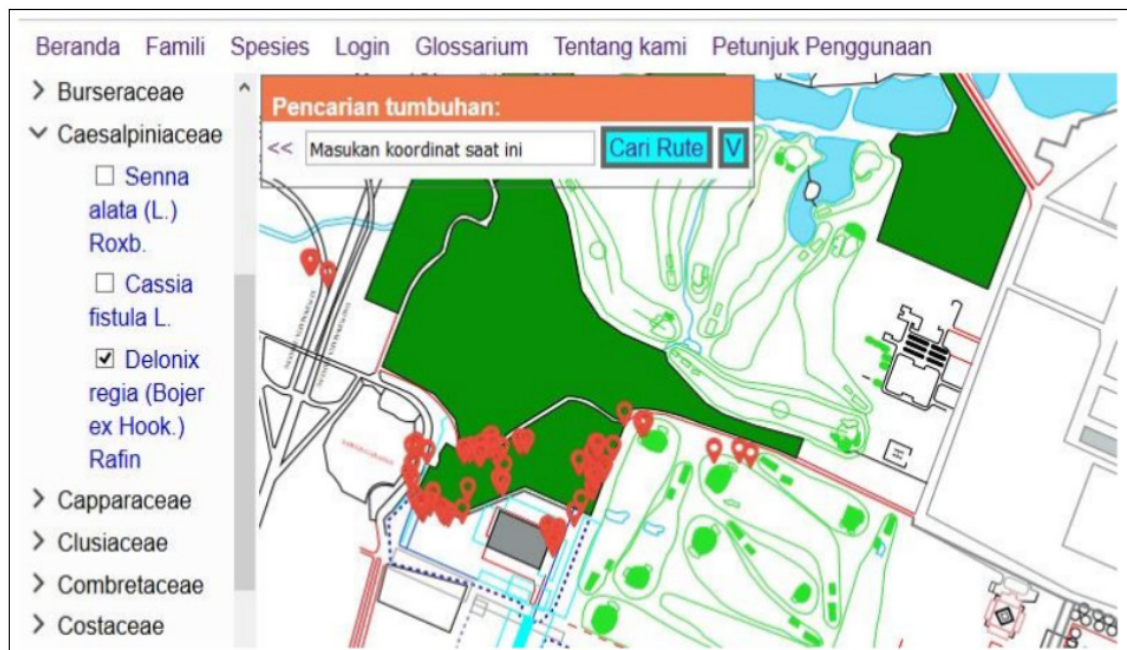
Pada tabel 7 diuraikan mengenai keseluruhan fungsi sistem yaitu terdiri dari 14 proses *use case*. Setiap *use case* ditampilkan di aplikasi dalam menu utama ataupun sub menu.

Perancangan antar muka aplikasi diselaraskan dengan rancangan *use case*, hal tersebut sangat penting

diperhatikan agar implementasi sistem bisa terarah. Adapun tampilan rancangan dan tampilan aplikasi yang dikembangkan ditunjukkan pada gambar 12 dan gambar 13. Aplikasi tersebut dirancang bisa dijalankan pada semua sistem operasi komputer seperti Linux, Windows atau Mac OS.



Gambar 12. Tampilan Rancangan Antar Muka



Gambar 13. Tampilan aplikasi

KESIMPULAN

Algoritme A-star mampu menentukan rute terpendek menuju lokasi tumbuhan yang dituju. Selain fasilitas pencarian rute, aplikasi ini juga mampu mendukung pengelolaan informasi mengenai tumbuhan di dalam kawasan. Berdasarkan penelitian dengan metode A-star maka rute yang dihasilkan berupa *graph* dua dimensi menuju lokasi. *Graph* tersebut akan ditampilkan jika user berada dalam kawasan PT Pupuk Kujang. Berdasarkan hasil penelitian dan temuan di lapangan, aplikasi berbasis web ini perlu dilengkapi dengan sistem *mobile* seperti Android agar dapat langsung memandu *user* ke lokasi tumbuhan dan Teknologi *Augmented Reality* program akan lebih interaktif dan menarik.

UCAPAN TERIMA KASIH

Terima kasih disampaikan kepada narasumber Departemen Keselamatan dan Kesehatan Kerja dan Lingkungan Hidup (K3LH) PT Pupuk Kujang Cikampek Bapak R. Sulisty, S.T. dan Bapak Reyno Pramudyono, S.T., Msi. Tidak lupa terimakasih kepada narasumber Departemen Komunikasi Perusahaan Bapak H. Ade Cahya Kurniawan S.Sos, M.M., Bapak H. Indra Gunawan S.Kom., M.I.Kom dan Bapak Adzinar Nursyahida, S.I.Kom, M.Si yang telah membantu dan mengarahkan penulis selama melakukan penelitian.

DAFTAR PUSTAKA

- Adipranata, R., Handojo, A, dan Setiawan, H. 2007. Aplikasi pencari rute optimum pada peta guna meningkatkan efisiensi waktu tempuh pengguna jalan dengan metode a* dan best first search. *Jurnal Informatika* 8 (2) : 100-108.
- Brandes, U, Cornelsen S. 2003. *Visual ranking of link structures*. JGAA: Journal of Graph Algorithms and Applications 7 (2) : 181-201.
- Budiyanto, E. 2016. Sistem Informasi geografis dengan Quantum GIS. Yogyakarta (ID): Andi Press.
- Chopde, N.R., Nichat, M.K. 2013. *Landmark based shortest path detection by using a* and haversine formula*. International Journal of Innovative Research in Computer and Communication Engineering 1 (2) : 298-302
- Ferdiansyah, .F, Muttaqin A., Aswin, M. 2013. Perbandingan algoritme dijkstra dan algoritme ant colony dalam penentuan jalur terpendek. *Jurnal Mahasiswa Teknik Elektro Universitas Brawijaya* 1(2): 1-6
- Ganesh L, Kumar V. 2015. *Indoor wireless localization using haversine formula*. International Advanced Research Journal in Science, Engineering and Technology 2 (7): 59-63.
- Hart P.E., Nilsson N.J., Raphael B.. 1968. *A formal basis for the heuristic determination of minimum cost paths*. IEEE transactions on Systems Science and Cybernetics 4 (2): 100-107.
- Irsyad, M., Rasila E. 2015. Aplikasi pencarian lokasi gedung dan ruangan universitas islam negeri sultan syarif kasim Riau pada *platform* android menggunakan algoritme A-star (A*). *Jurnal CoreIT (UIN Sultan Syarif Kasim Riau)* 1 (2) : 90-95.
- Jiang, B., Wang Y., Zhao Z. 2015. *Path planning for terrestrial platform based on A-star algorithm*. 3rd International Conference on Mechatronics, Robotics and Automation. Atlantis Press 409-414.
- Juniansyah, A., Mesterjon. 2016. Aplikasi penentuan rute terpendek untuk bagian pemasaran produk roti surya dengan metode *best first search*. *Jurnal Media Infotama* 12 (01): 31-40.
- Kikelomo, A.A., Asafe Y.N., Paul A., Olawale L. 2017. *Design and implementation of mobile map application for finding shortest direction between two pair locations using shortest path algorithm: a case study*. International Journal Advanced Networking and Applications 09 (01) : 3300-3305.
- Mutakhiroh, I., Saptono F., Hasanah N., Wiryadinata R. 2007. Pemanfaatan Metode Heuristik dalam Pencarian Jalur Terpendek dengan algoritme Semut dan algoritme Genetika. Seminar Nasional Aplikasi Teknologi Informasi. 2007 Juni 16, Yogyakarta, Indonesia. B.33 – B.37
- Mutiana, V., Amastini F., Mutiara N. 2013. Optimasi pencarian jalur dengan metode a-star (studi kasus: Area Gading Serpong, Tangerang). *Ultimatics* . 5 (2): 42-47.
- Mwemezi, J., Huang Y. 2011. *Optimal facility location on spherical surfaces: algorithm and application*. New York Science Journal 4 (7) : 21-28.
- [PKC: PT Pupuk Kujang Cikampek.ID]. 2015. Koleksi Penting Tumbuhan di Taman Kehati Pupuk Kujang. Karawang: PT Pupuk Kujang. 5-25.
- Ostrowski, D., Pozniak-Koszalka I., Koszalka L., Kasprzak A. 2015. *Comparative Analysis of the Algorithms for Pathfinding in GPS Systems*. ICN 2015: The Fourteenth International Conference on Networks. Page 102-108.
- Purwananto, Y., Purwitasari, D., Wibowo A.W. 2005. Implementasi dan Analisis algoritme Pencarian Rute Terpendek di Kota Surabaya. *Jurnal Penelitian dan Pengembangan Telekomunikasi* 10 (2) : 94-101.

- Sajeevan. 2008. Latitude and longitude – A Misunderstanding. *Current Science* 94 (5) : 94(5) : 568-569
- Satzinger J.W., Jackson R.B., Burd S.D. 2010. *System Analysis and Design in a Changing World*. 5th Edition. Alex von Rosenberg (Ed.). Boston: Course Technology. 38-45
- Šavrič, B., Patterson T., Jenny B. 2019. *The Equal Earth map projection*. *International Journal Of Geographical Information Science*. 33 (3): 454-465.
- Sharma, S.K., Pal B.L. 2015. *Shortest Path Searching for Road Network using A* Algorithm*. *IJCSMC: International Journal of Computer Science and Mobile Computing* 4 (7) : 513 – 522.
- Taylor, R.N., Hoek AV. 2007. *Software Design and Architecture The once and future focus of software engineering, Proceeding FOSE: Future of Software Engineering*, IEEE Computer Society Washington, DC, USA. pp. 226-243.
- Umate, N. B., Bhujade V.G. 2017. *Design of Targeted Advertising for Location Based Services and User Profile Based Recommendation*. *IJIRCCE: International Journal of Innovative Research in Computer and Communication Engineering* 5 (9): 15102- 5109.
- Yuliana Y., Ananda A., Surya I. 2012. Implementasi algoritme A star pada pemecahan puzzle 8. *Jurnal Teknik Informatika (Politeknik Caltek Riau)* 1(2): 1-8.
- Zhou, G., Shao W. 2012. *The Design and Improvement of a Software Project Management System Based on CMMI*. *Intelligent Information Management*. 04 (06):330-337