

**IMPLEMENTASI *GRAPH DATABASE* UNTUK REKOMENDASI RUTE  
PENDAKIAN GUNUNG YANG TERBAIK**

**SKRIPSI**

Oleh:  
**CONDRO PRIO UTOMO**  
NIM. 200605110167



**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2025**

**IMPLEMENTASI *GRAPH DATABASE* UNTUK REKOMENDASI RUTE  
PENDAKIAN GUNUNG YANG TERBAIK**

**SKRIPSI**

Diajukan kepada:  
Fakultas Sains dan Teknologi  
Universita Islam Negeri Maulana Malik Ibrahim Malang  
Untuk memenuhi Salah Satu Persyaratan dalam  
Memperoleh Gelar Sarjana Komputer (S.Kom)

Oleh:  
**CONDRO PRIO UTOMO**  
NIM. 200605110167

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2025**

**HALAMAN PERSETUJUAN**


**IMPLEMENTASI *GRAPH DATABASE* UNTUK REKOMENDASI RUTE  
PENDAKIAN GUNUNG YANG TERBAIK**

**SKRIPSI**


Oleh:  
**CONDRO PRIO UTOMO**  
NIM. 200605110167

Telah Diperiksa dan Disetujui untuk Diuji:  
Tanggal: 5 Juni 2025

Pembimbing I,


  
Dr. M. Ainul Yaqin, M.Kom  
NIP. 19761013 200604 1 004

Pembimbing II,

  
Dr. Zainal Abidin, M.Kom  
NIP. 19760613 200501 1 004

Mengetahui,  
Ketua Program Studi Teknik Informatika  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri Maulana Malik Ibrahim Malang



  
Dr. Fachrul Kurniawan, M.MT, IPU  
NIP. 19771020 200912 1 001

## HALAMAN PENGESAHAN

### IMPLEMENTASI *GRAPH DATABASE* UNTUK REKOMENDASI RUTE PENDAKIAN GUNUNG YANG TERBAIK

#### SKRIPSI

Oleh:  
**CONDRO PRIO UTOMO**  
NIM. 200605110167

Telah Dipertahankan di Depan Dewan Penguji Skripsi  
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan  
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)  
Tanggal: 16 Juni 2025

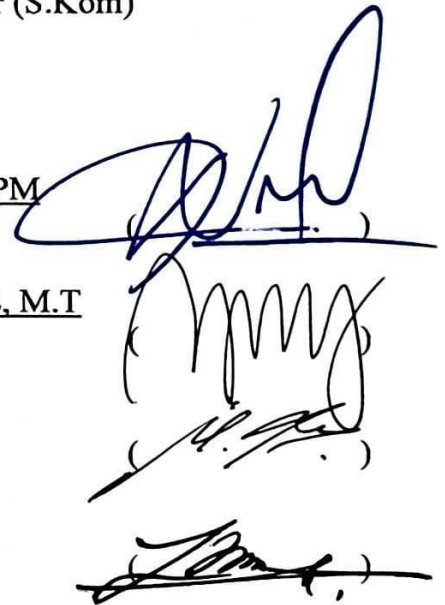
#### Susunan Dewan Penguji

Ketua Penguji : Dr. Ir. Fresy Nugroho, ST., MT, IPM  
NIP. 19710722 201101 1 001

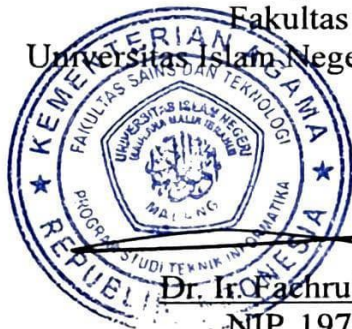
Anggota Penguji I : Dr. Agung Teguh Wibowo Almais, M.T  
NIP. 19860301 202321 1 016

Anggota Penguji II : Dr. M. Ainul Yaqin, M.Kom  
NIP. 19761013 200604 1 004

Anggota Penguji III : Dr. Zainal Abidin, M.Kom  
NIP. 19760613 200501 1 004



Mengetahui,  
Ketua Program Studi Teknik Informatika  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri Maulana Malik Ibrahim Malang



Dr. Ir. Fachrul Kurniawan, M.MT, IPU  
NIP. 19771020 200912 1 001

## PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Condro Prio Utomo  
NIM : 200605110167  
Fakultas / Program Studi : Sains dan Teknologi / Teknik Informatika  
Judul Skripsi : Implementasi *Graph Database* untuk Rekomendasi Rute Pendakian Gunung yang Terbaik

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini merupakan hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 16 Juni 2025  
Yang membuat pernyataan,



Condro Prio Utomo  
NIM. 200605110167

## **MOTTO**

*... Dari Alam Aku Ada, Aku Ada Untuk Alam ...*

## HALAMAN PERSEMBAHAN

*Bismillahirrahmanirrahim.*

*Alhamdulillahillāhi rabbil ‘ālamīn. Alhamdulillahilladzī bini ‘matihī tatimmush shālihātu, wa bifadhlihī tatanazzalul khairātu wal barakātu, wa bitaufiqihī tatahaqqaqul maqāshidu wal ghāyātu.*

Segala puji bagi Allah, Tuhan semesta alam. Segala puji bagi Allah yang dengan nikmat-Nya segala kebaikan menjadi sempurna, dengan anugerah-Nya segala kebaikan dan keberkahan diturunkan, dan dengan pertolongan-Nya segala maksud dan tujuan dapat tercapai. Dengan rahmat dan rida-Nya, penulis dapat menyelesaikan skripsi ini. Shalawat serta salam semoga senantiasa tercurah kepada junjungan kita, Nabi Muhammad saw., beserta keluarga dan para sahabatnya, yang telah menjadi suri teladan bagi seluruh umat manusia. Sebagai wujud syukur dan bakti, karya sederhana ini penulis persembahkan kepada:

Ayahanda dan Ibunda tercinta, yang telah menjadi samudra tak bertepi bagi setiap pengorbanan. Terima kasih atas setiap tetes keringat, dukungan moril dan materiel yang tak ternilai, serta doa-doa yang tak pernah henti terpanjatkan di keheningan malam. Adik-adikku tersayang, yang senantiasa menjadi sumber semangat dan senyum di tengah perjuangan. Segenap keluarga besar, yang selalu memberikan motivasi dan keyakinan untuk tidak pernah menyerah.

## KATA PENGANTAR

*Alhamdulillah* rabbil ‘*alamīn*. Puji syukur penulis panjatkan ke hadirat Allah Swt., yang atas rahmat, karunia, dan taufik-Nya telah melimpahkan kemudahan sehingga penulis dapat menyelesaikan skripsi ini dengan baik. Shalawat serta salam semoga senantiasa tercurah kepada junjungan kita, Nabi Muhammad saw., beserta keluarga dan para sahabatnya, yang telah menjadi suri teladan bagi seluruh umat manusia.

Skripsi yang berjudul “Implementasi *Graph Database* untuk Rekomendasi Rute Pendakian Gunung yang Terbaik” ini disusun sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik pada Jurusan Teknik Informatika, Fakultas Sains dan Teknologi, UIN Maulana Malik Ibrahim Malang. Penulis berharap skripsi ini dapat memberikan kontribusi, baik dalam pengembangan teknologi sistem informasi perencanaan pendakian gunung, maupun dalam menumbuhkan kesadaran kita bersama akan pentingnya menjaga dan melestarikan alam.

Penyelesaian skripsi ini tentu tidak terlepas dari bantuan, bimbingan, dan dukungan tulus dari berbagai pihak. Oleh karena itu, dengan segala kerendahan hati, penulis ingin menghaturkan ucapan terima kasih yang sebesar-besarnya kepada:

1. Prof. Dr. H. M. Zainuddin, M.A., selaku Rektor UIN Maulana Malik Ibrahim Malang, beserta jajaran pimpinan universitas.
2. Prof. Dr. Hj. Sri Harini, M.Si., selaku Dekan Fakultas Sains dan Teknologi UIN Maulana Malik Ibrahim Malang, beserta jajaran pimpinan fakultas.



3. Dr. Fachrul Kurniawan, M.MT., IPM., selaku Ketua Program Studi Teknik Informatika UIN Maulana Malik Ibrahim Malang.
4. Dr. M. Ainul Yaqin, M.Kom., selaku Dosen Pembimbing I dan Dr. Zainal Abidin, M.Kom., selaku Dosen Pembimbing II, yang telah dengan sabar meluangkan waktu, tenaga, dan pikiran untuk memberikan bimbingan, arahan, serta masukan yang sangat berharga dari awal hingga akhir penyusunan skripsi ini.
5. Seluruh Dosen dan Staf Administrasi Jurusan Teknik Informatika UIN Maulana Malik Ibrahim Malang.
6. Rekan-rekan seperjuangan, keluarga besar Teknik Informatika angkatan 2020 (Integer-20), terima kasih atas kebersamaan, semangat, dan kenangan indah selama penulisan skripsi.
7. Semua pihak yang tidak dapat penulis sebutkan satu per satu.

Akhir kata, semoga skripsi ini dapat membawa manfaat bagi pembaca serta pengembangan ilmu pengetahuan.

Malang, 16 Juni 2025

Penulis

## DAFTAR ISI

HALAMAN PERSETUJUAN .....	iii
HALAMAN PENGESAHAN .....	iv
PERNYATAAN KEASLIAN TULISAN .....	v
MOTTO .....	vi
HALAMAN PERSEMBAHAN .....	vii
KATA PENGANTAR .....	viii
DAFTAR ISI .....	x
DAFTAR GAMBAR .....	xii
DAFTAR TABEL .....	xiii
ABSTRAK .....	xv
ABSTRACT .....	xvi
خلاصة .....	xvii
<b>BAB I PENDAHULUAN .....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	4
1.3 Tujuan Penelitian .....	4
1.4 Manfaat Penelitian .....	4
1.5 Batasan Masalah .....	5
<b>BAB II STUDI PUSTAKA .....</b>	<b>6</b>
2.1 <i>Graph</i> .....	6
2.2 <i>Graph Database</i> .....	8
2.3 <i>Weighted Directed Acyclic Graph (WDAG)</i> .....	9
2.4 <i>Analytic Hierarchy Process (AHP)</i> .....	11
2.5 <i>Dijkstra</i> .....	13
2.6 <i>Avenza Maps</i> .....	14
2.7 Trigonometri .....	16
2.8 <i>Naismith Rule</i> .....	16
2.9 Model Ukur Kompleksitas <i>Chyper Query Graph</i> .....	17
2.10 Penelitian Terkait .....	18
<b>BAB III METODOLOGI PENELITIAN .....</b>	<b>22</b>
3.1 Gambaran Umum .....	22
3.2 Sumber Data .....	22
3.3 Desain Penelitian .....	24
3.3.1 Pengolahan Data menggunakan <i>Avenza Maps</i> .....	24
3.3.2 Perancangan Sistem .....	32
3.3.3 Pemodelan Struktur <i>Graph Database</i> .....	34
3.3.4 Normalisasi Data Jalur Pendakian Gunung .....	38
3.3.5 Pembobotan <i>WDAG</i> Pada Jalur Menggunakan <i>AHP</i> .....	39
3.3.6 Pencarian Rekomendasi Rute Menggunakan <i>Dijkstra</i> .....	42
3.4 Desain Eksperimen .....	43
3.4.1 Skenario Uji Coba .....	44
3.4.2 Evaluasi Hasil .....	48
<b>BAB IV HASIL DAN PEMBAHASAN .....</b>	<b>56</b>

4.1	Implementasi Sistem .....	56
4.2	Implementasi Tampilan Sistem .....	57
4.2.1	Tampilan Halaman Utama .....	57
4.2.2	Tampilan Form Lokasi Awal dan Lokasi Tujuan.....	58
4.2.3	Tampilan Hasil Rekomendasi .....	59
4.2.4	Tampilan Detail Hasil Rekomendasi .....	60
4.2.5	Tampilan Profil Pengguna .....	61
4.2.6	Tampilan Pengaturan Preferensi Kriteria.....	62
4.2.7	Tampilan Dashboard Admin .....	63
4.2.8	Tampilan Tambah Data .....	64
4.2.9	Tampilan Edit Data .....	66
4.2.10	Tampilan <i>Login</i> .....	68
4.2.11	Tampilan <i>Register</i> .....	68
4.3	Evaluasi Kompleksitas <i>Chyper Query Graph Database</i> .....	69
4.3.1	<i>Chyper Query Import</i> Data Jalur dan Titik Lokasi Pendakian Gunung	69
4.3.2	<i>Chyper Query</i> Normalisasi dan Pembobotan Data Jalur .....	71
4.3.3	<i>Chyper Query Cyclomatic Complexity</i> .....	72
4.3.4	<i>Chyper Query</i> Algoritma <i>Dijkstra</i> untuk Hasil Pencarian Rute .....	73
4.3.5	<i>Chyper Query</i> Algoritma <i>Dijkstra</i> untuk Detail Hasil Pencarian Rute	75
4.4	Hasil Pengujian Sistem .....	76
4.5	Evaluasi Ketergunaan Sistem dengan <i>System Usability Scale (SUS)</i> .....	84
4.6	Pembahasan.....	85
<b>BAB V KESIMPULAN DAN SARAN .....</b>		<b>95</b>
5.1	Kesimpulan .....	95
5.2	Saran.....	96
5.2.1	Untuk Pengembangan Sistem dan Penelitian Selanjutnya .....	96
5.2.2	Bagi Pengguna (Pendaki) .....	98
5.2.3	Bagi Pengelola Kawasan Pendakian dan Instansi Terkait: .....	98
<b>DAFTAR PUSTAKA</b>		
<b>LAMPIRAN</b>		

## DAFTAR GAMBAR

Gambar 3.1	Peta Jalur Pendakian Gunung Arjuno-Welirang.....	23
Gambar 3.2	Tahapan Penelitian .....	24
Gambar 3.3	<i>Import</i> Peta ke dalam <i>Avenza Maps</i> .....	26
Gambar 3.4	<i>Import</i> Peta ke dalam <i>Avenza Maps</i> .....	26
Gambar 3.5	Penandaan Lokasi dalam Peta Jalur Pendakian .....	28
Gambar 3.6	Penggambaran dan Pengukuran Jalur dalam Peta Jalur Pendakian .....	29
Gambar 3.7	Penggambaran dan Pengukuran Jalur dalam Peta Jalur Pendakian .....	30
Gambar 3.8	Proses Validasi Data menggunakan <i>Google Earth</i> .....	32
Gambar 3.9	<i>Flow Chart</i> Sistem Rekomendasi Rute Pendakian Gunung .....	33
Gambar 3.10	<i>Use Case Diagram</i> Sistem Rekomendasi Rute Pendakian Gunung.....	34
Gambar 3.11	Skema Graf.....	35
Gambar 3.12	Visualisai Graf.....	37
Gambar 4.1	Tampilan Halaman Utama.....	57
Gambar 4.2	Tampilan Form Lokasi Awal dan Lokasi Tujuan .....	58
Gambar 4.3	Tampilan Hasil Rekomendasi .....	59
Gambar 4.4	Tampilan Detail Hasil Rekomendasi.....	60
Gambar 4.5	Tampilan Profil Pengguna.....	61
Gambar 4.6	Tampilan Pengaturan Preferensi Kriteria .....	62
Gambar 4.7	Tampilan Dashboard Admin .....	63
Gambar 4.8	Tampilan Tambah Data Gunung .....	64
Gambar 4.9	Tampilan Tambah Data Lokasi .....	64
Gambar 4.10	Tampilan Tambah Data Jalur.....	65
Gambar 4.11	Tampilan Edit Data Mountain.....	66
Gambar 4.12	Tampilan Edit Data Lokasi.....	66
Gambar 4.13	Tampilan Edit Data Jalur.....	67
Gambar 4.14	Tampilan Login .....	68
Gambar 4.15	Tampilan Register .....	69
Gambar 4.16	Grafik Hubungan Kompleksitas <i>Chyper Query</i> dan Waktu Pemrosesan.....	86
Gambar 4.17	Grafik Hubungan Kompleksitas Graf dan Akurasi .....	88
Gambar 4.18	Grafik Hubungan Kompleksitas Graf dan Waktu Pemrosesan .....	90

## DAFTAR TABEL

Tabel 2.1	Bobot Atribut <i>Chyper Query</i> .....	17
Tabel 2.2	Penelitian Terkait.....	19
Tabel 3.1	Skema Graf.....	35
Tabel 3.2	<i>Cypher Query Node</i> .....	36
Tabel 3.3	<i>Cypher Query Edge</i> .....	36
Tabel 3.4	Matriks Perbandingan Berpasangan.....	39
Tabel 3.5	Matriks Normalisasi.....	40
Tabel 3.6	<i>Cypher Query</i> Normalisasi Data dan Pembobotan Pada Jalur.....	41
Tabel 3.7	<i>Cypher Query</i> algoritma Dijkstra.....	43
Tabel 3.8	Daftar input untuk Uji Coba di Gunung Arjuno.....	45
Tabel 3.9	Daftar input untuk Uji Coba di Gunung Bromo.....	45
Tabel 3.10	Daftar input untuk Uji Coba di Gunung Butak.....	46
Tabel 3.11	Daftar input untuk Uji Coba di Gunung Ciremai.....	46
Tabel 3.12	Daftar input untuk Uji Coba di Gunung Penanggungan.....	46
Tabel 3.13	Daftar input untuk Uji Coba di Gunung Slamet.....	47
Tabel 3.14	Daftar input untuk Uji Coba di Gunung Ungaran.....	47
Tabel 3.15	Daftar input untuk Uji Coba di Gunung Sumbing.....	48
Tabel 3.16	Skala Penerimaan.....	54
Tabel 3.17	Skala Peringkat.....	54
Tabel 3.18	Skala Adjektif.....	54
Tabel 4.1	<i>Cypher Query Import</i> Data Jalur dan Titik Lokasi Pendakian Gunung.....	70
Tabel 4.2	Perhitungan Kompleksitas <i>Chyper Query Import</i> Data.....	70
Tabel 4.3	<i>Cypher Query</i> Normalisasi dan Pembobotan Data Jalur.....	71
Tabel 4.4	Perhitungan Kompleksitas <i>Chyper Query</i> Normalisasi dan Pembobotan.....	72
Tabel 4.5	<i>Cypher Query Cyclomatic Complexity</i> .....	72
Tabel 4.6	Perhitungan Kompleksitas <i>Chyper Query Cyclomatic Complexity</i> .....	73
Tabel 4.7	<i>Cypher Query</i> Algoritma Dijkstra untuk Hasil Pencarian Rute.....	74
Tabel 4.8	Perhitungan Kompleksitas <i>Chyper Query</i> Algoritma Dijkstra.....	74
Tabel 4.9	<i>Cypher Query</i> Algoritma Dijkstra untuk Detail Hasil Pencarian Rute.....	75
Tabel 4.10	Perhitungan Kompleksitas <i>Chyper Query</i> Algoritma Dijkstra.....	75
Tabel 4.11	Hasil Perhitungan Kompleksitas <i>Chyper Query</i> dan Waktu Pemrosesan.....	76
Tabel 4.12	Hasil Perhitungan Kompleksitas <i>Graph</i> pada Setiap Gunung.....	77
Tabel 4.13	Hasil Pengujian di Gunung Arjuno.....	78
Tabel 4.14	Hasil Pengujian di Gunung Bromo.....	79
Tabel 4.15	Hasil Pengujian di Gunung Butak.....	79
Tabel 4.16	Hasil Pengujian di Gunung Ciremai.....	80
Tabel 4.17	Hasil Pengujian di Gunung Penanggungan.....	80
Tabel 4.18	Hasil Pengujian di Gunung Slamet.....	81
Tabel 4.19	Hasil Pengujian di Gunung Sumbing.....	82

Tabel 4.20	Hasil Pengujian di Gunung Ungaran.....	83
Tabel 4.21	Hasil Perhitungan Kompleksitas Graf, Akurasi dan Waktu Pemrosesan.....	83
Tabel 4.22	Ringkasan Hasil Perhitungan <i>System Usability Scale</i> (SUS).....	84

## ABSTRAK

Utomo, Condro Prio. 2025. **Implementasi Graph Database Untuk Rekomendasi Rute Pendakian Gunung yang Terbaik**. Skripsi. Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Dr. M. Ainul Yaqin, M.Kom. (II) Dr. Zainal Abidin, M.Kom.

Kata kunci: pendakian gunung, *Graph Database*, *Dijkstra*

Penelitian ini bertujuan untuk mengimplementasikan *graph database* guna mengembangkan sistem rekomendasi rute pendakian gunung terbaik yang dapat dipersonalisasi sesuai preferensi pengguna, sehingga meningkatkan kenyamanan, pengalaman, dan keselamatan pendaki. Sistem ini menggunakan *graph database Neo4j* untuk memodelkan jaringan jalur pendakian sebagai *Weighted Directed Acyclic Graph (WDAG)*. Metode *Analytic Hierarchy Process (AHP)* diterapkan untuk pembobotan kriteria preferensi pengguna (jarak, selisih elevasi, kemiringan, dan estimasi waktu tempuh), dan algoritma *Dijkstra* digunakan untuk menemukan rute terbaik. Data jalur pendakian dari beberapa gunung di Indonesia bersumber dari situs Gunung Bagging dan diolah menggunakan *Avenza Maps*. Evaluasi sistem melibatkan analisis kompleksitas *query Cypher*, kompleksitas graf (menggunakan *Cyclomatic Complexity*), serta akurasi rekomendasi (*F1-Score*) dan waktu pemrosesan.

Temuan utama menunjukkan bahwa sistem berhasil memberikan rekomendasi rute yang dipersonalisasi sesuai dengan preferensi pengguna. Analisis kinerja mengungkapkan adanya hubungan *non-linear* yang kuat antara kompleksitas *query Cypher* dengan peningkatan waktu pemrosesan. Lebih lanjut, kompleksitas graf juga berpengaruh signifikan: peningkatan kompleksitas graf berdampak pada peningkatan waktu pemrosesan dan sedikit penurunan pada akurasi *F1-Score*. Secara keseluruhan, sistem mencapai rata-rata akurasi *F1-Score* 97.888% dengan waktu pemrosesan rata-rata 65.481 ms pada pengujian di delapan gunung.

Disimpulkan bahwa implementasi *graph database* dengan *Weighted Directed Acyclic Graph (WDAG)*, *Analytical Hierarchy Process (AHP)*, dan *Dijkstra* efektif untuk menghasilkan rekomendasi rute pendakian yang dipersonalisasi. Namun, kompleksitas *query* dan struktur graf memiliki dampak signifikan terhadap kinerja, sehingga optimasi keduanya menjadi krusial untuk menjaga keseimbangan antara akurasi dan efisiensi sistem. Temuan ini relevan untuk pengembangan sistem serupa dan berkontribusi pada peningkatan pengalaman serta keselamatan dalam aktivitas pendakian gunung.

## ABSTRACT

Utomo, Condro Prio. 2025. **Graph Database Implementation for the Best Mountain Climbing Route Recommendation**. Thesis. Department of Informatics Engineering, Faculty of Science and Technology, State Islamic University of Maulana Malik Ibrahim Malang. Advisors: (I) Dr. M. Ainul Yaqin, M.Kom. (II) Dr. Zainal Abidin, M.Kom.

Keywords: mountain climbing, Graph Database, Dijkstra

This study aims to implement a graph database to develop a system for recommending the best mountain climbing routes that can be personalized according to user preferences, thereby increasing climbers' comfort, experience, and safety. This system uses the Neo4j graph database to model the climbing route network as a Weighted Directed Acyclic Graph (WDAG). The Analytic Hierarchy Process (AHP) method is applied to weigh the user's preference criteria (distance, elevation difference, slope, and estimated travel time), and the Dijkstra algorithm is used to find the best route. Data on climbing routes from several mountains in Indonesia are sourced from the Gunung Bagging site and processed using Avenza Maps. The system evaluation involved analysis of Cypher query complexity, graph complexity (using Cyclomatic Complexity), recommendation accuracy (F1-Score), and processing time.

The main findings showed that the system successfully provided personalized route recommendations based on user preferences. Performance analysis revealed a strong non-linear relationship between Cypher query complexity and processing time. Furthermore, graph complexity also had a significant effect: increasing graph complexity resulted in increased processing time and a slight decrease in F1-Score accuracy. Overall, the system achieved an average F1-Score accuracy of 97.888% with an average processing time of 65.481 ms in testing on eight mountains.

It was concluded that implementing a graph database with Weighted Directed Acyclic Graph (WDAG), Analytical Hierarchy Process (AHP), and Dijkstra effectively generated personalized climbing route recommendations. However, query complexity and graph structure had a significant impact on performance, so optimizing both became crucial to maintain a balance between system accuracy and efficiency. These findings are relevant for developing similar systems and contribute to improving the experience and safety in mountaineering activities.



## خلاصة

أوتومو، كوندرو بريو. 2025. تنفيذ قاعدة بيانات الرسوم البيانية لأفضل توصيات مسارات تسلق الجبال. أطروحة. قسم هندسة المعلوماتية، كلية العلوم والتكنولوجيا، جامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرف (I): دكتور محمد عين اليقين ماجستير حاسبات (II) دكتور زينل العابدين ماجستير حاسبات

الكلمات الرئيسية: تسلق الجبال، Graph Database، Dijkstra

تهدف هذه الدراسة إلى تنفيذ graph database لتطوير أفضل نظام توصية لمسار تسلق الجبال والذي يمكن تخصيصه وفقاً لتفضيلات المستخدم، وبالتالي تحسين راحة وتجربة وسلامة المتسلقين. يستخدم هذا النظام graph database Neo4j لنمذجة شبكة مسارات المشي لمسافات طويلة على شكل Weighted Directed Acyclic Graph (WDAG). يتم تطبيق طريقة Analytic Hierarchy Process (AHP) لترجيح معايير تفضيل المستخدم (المسافة، فرق الارتفاع، المنحدر، ووقت السفر المقدر)، ويتم استخدام خوارزمية Dijkstra للعثور على أفضل طريق. تأتي البيانات المتعلقة بمسارات المشي لمسافات طويلة من العديد من الجبال في إندونيسيا من موقع Gunung Bagging ويتم معالجتها باستخدام Avenza Maps. يتضمن تقييم النظام تحليل تعقيد query Cypher، وتعقيد graf (باستخدام Cyclomatic Complexity)، بالإضافة إلى دقة التوصية (F1-Score) ووقت المعالجة.

وتظهر النتائج الرئيسية أن النظام يوفر بنجاح توصيات مسار مخصصة وفقاً لتفضيلات المستخدم. يكشف تحليل الأداء عن وجود علاقة قوية non-linear بين تعقيد query Cypher وزيادة وقت المعالجة. علاوة على ذلك، فإن تعقيد graf له أيضاً تأثير كبير: يؤدي زيادة تعقيد graf إلى زيادة وقت المعالجة وانخفاض طفيف في دقة F1-Score بشكل عام، حقق النظام دقة متوسطة F1-Score بنسبة 97.888% مع متوسط وقت معالجة قدره 65.481 مللي ثانية في الاختبارات عبر ثمانية جبال.

وخلصت الدراسة إلى أن تنفيذ graph database مع Weighted Directed Acyclic Graph (WDAG)، وAnalytical Hierarchy Process (AHP)، وDijkstra فعال في توليد توصيات مخصصة لمسار المشي لمسافات طويلة. ومع ذلك، فإن تعقيد query وبنية graf لهما تأثير كبير على الأداء، لذا فإن تحسين كليهما يصبح أمراً بالغ الأهمية للحفاظ على التوازن بين دقة النظام وكفاءته. وتعتبر هذه النتائج ذات أهمية لتطوير أنظمة مماثلة وتساهم في تحسين تجربة ومستوى السلامة في أنشطة تسلق الجبال.

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Pendakian gunung merupakan aktivitas populer di kalangan pecinta alam dan petualang. Namun, banyak pendaki menghadapi tantangan dalam memilih jalur pendakian yang paling sesuai, karena terdapat berbagai faktor yang perlu dipertimbangkan, seperti jarak, selisih elevasi, kemiringan, dan estimasi waktu tempuh. Oleh karena itu, informasi yang akurat dan terpercaya tentang jalur pendakian sangat penting untuk memastikan keselamatan dan kelancaran perjalanan. Informasi yang tidak lengkap atau tersebar di berbagai sumber dapat meningkatkan risiko kesalahan dalam pengambilan keputusan, bahkan dapat menyebabkan kecelakaan atau pendaki tersesat (Supriady dkk., 2022; Yunanto dkk., 2021).

Meskipun banyak sumber informasi yang tersedia, seperti buku panduan, situs web, dan aplikasi, informasi ini seringkali tidak terintegrasi dengan baik, membuat pendaki kesulitan dalam memilih jalur yang tepat dan meningkatkan risiko kesalahan. Oleh karena itu, diperlukan sebuah sistem yang dapat mengintegrasikan dan menganalisis data jalur pendakian dengan cara yang lebih efisien dan sistematis. (Gumilang, 2022).

Penelitian ini bertujuan untuk mengembangkan sistem rekomendasi jalur pendakian gunung berbasis graph database. Sistem ini memanfaatkan algoritma *Weighted Directed Acyclic Graph (WDAG)*, *Analytic Hierarchy Process (AHP)*, dan *Dijkstra* untuk menghitung jalur pendakian terbaik berdasarkan faktor-faktor

utama, seperti jarak, selisih elevasi, kemiringan, dan estimasi waktu tempuh. *Graph database* akan digunakan untuk menyimpan dan mengelola data jalur pendakian dalam bentuk graf yang memungkinkan pengolahan dan analisis yang lebih fleksibel dan terstruktur (Muharrom, 2020).

Penggunaan teknologi ini diharapkan dapat memberikan rekomendasi rute pendakian yang terbaik dan disesuaikan dengan preferensi serta kemampuan fisik pendaki. Dengan demikian, sistem ini tidak hanya bertujuan untuk meningkatkan pengalaman pendakian, tetapi juga untuk meningkatkan keselamatan pendaki dengan meminimalkan risiko tersesat atau memilih jalur yang tidak sesuai (Tangdisosang & Widiyari, 2022).

Selain itu, penerapan *graph database* dan algoritma terkait dapat memberikan kontribusi dalam pengembangan sistem rekomendasi serupa di berbagai bidang, seperti transportasi dan perencanaan rute, yang dapat meningkatkan kualitas layanan dan pengalaman pengguna secara keseluruhan (Setiawan dkk., 2020).

Sebagaimana telah diuraikan, pendakian gunung sebagai aktivitas populer menghadirkan tantangan signifikan dalam pemilihan rute yang paling sesuai, mengingat beragamnya faktor seperti jarak, selisih elevasi, kemiringan, dan estimasi waktu tempuh yang perlu dipertimbangkan. Ketersediaan informasi yang akurat dan terintegrasi menjadi krusial demi keselamatan dan kelancaran perjalanan, karena informasi yang tidak lengkap atau tersebar dapat meningkatkan risiko kesalahan pengambilan keputusan, bahkan hingga kecelakaan atau tersesatnya pendaki. Dalam konteks inilah, firman Allah SWT dalam QS. An-Nahl:

15 beserta penjelasannya dalam Tafsir Al-Jalalain memberikan perspektif yang mendalam:

{ وألقى في الأرض رواسي { جبالا ثوابت ل { أن { لا { تميد { تتحرك { بكم و { جعل فيها { أنهارا { كالنيل { وسبلا { طرقا { لعلكم تهتدون { إلى مقاصدكم

“(Dan Dia menancapkan gunung-gunung di bumi dengan kokohnya) gunung-gunung yang tegak kokoh supaya (tidak) jangan (bumi itu goncang) bergerak (bersama kalian dan) Dia telah menciptakan padanya (sungai-sungai) seperti sungai Nil (dan jalan-jalan) jalan untuk dilalui (agar kalian mendapat petunjuk) untuk sampai kepada tujuan-tujuan kalian.” (Tafsir Al-Jalalain QS. An-Nahl: 15). (Mahally dkk., 2010)

Ayat ini menegaskan bahwa di samping penciptaan gunung-gunung sebagai penstabil bumi, Allah SWT juga menciptakan jalan-jalan dengan tujuan agar manusia mendapatkan petunjuk untuk mencapai maksud dan tujuan. Penelitian yang berjudul "Implementasi Graph Database untuk Rekomendasi Rute Pendakian Gunung yang Terbaik" ini berupaya selaras dengan hikmah tersebut. Dengan mengembangkan sistem rekomendasi berbasis *graph database* yang memanfaatkan algoritma *Weighted Directed Acyclic Graph (WDAG)*, *Analytic Hierarchy Process (AHP)*, dan Dijkstra, penelitian ini bertujuan untuk menghadirkan sebuah sarana modern dalam mengidentifikasi jalan-jalan atau rute pendakian yang paling baik. Sistem ini dirancang untuk mengolah dan menganalisis data jalur pendakian secara efisien dan sistematis, mempertimbangkan berbagai faktor krusial guna memberikan petunjuk kepada para pendaki dalam memilih jalur yang tidak hanya terbaik dari segi preferensi dan kemampuan fisik, tetapi juga yang paling aman dan efisien. Dengan demikian, penelitian ini merupakan ikhtiar ilmiah untuk memanfaatkan teknologi demi memaksimalkan pemahaman dan penggunaan jalan-jalan alami yang telah Allah sediakan, sehingga perjalanan pendakian gunung dapat

dilakukan dengan lebih terarah, aman, dan minim risiko, selaras dengan tujuan agar manusia mendapat petunjuk untuk sampai kepada tujuan-tujuan kalian.

## **1.2 Rumusan Masalah**

Bagaimana mengembangkan sistem rekomendasi rute pendakian gunung yang dapat memberikan rekomendasi rute yang disesuaikan dengan karakteristik pendaki, sehingga dapat meningkatkan kenyamanan dan pengalaman pendakian?

## **1.3 Tujuan Penelitian**

Penelitian ini bertujuan untuk mengembangkan sistem rekomendasi rute pendakian gunung yang mampu memberikan rekomendasi rute yang disesuaikan dengan karakteristik pendaki, sehingga dapat meningkatkan kenyamanan dan pengalaman pendakian.

## **1.4 Manfaat Penelitian**

Manfaat dari penelitian ini adalah sebagai berikut:

- a. Peningkatan Keselamatan dan Pengalaman Pendakian: Penelitian ini menghasilkan sistem rekomendasi jalur pendakian berbasis graph yang mempertimbangkan faktor jarak, elevasi, kemiringan, dan estimasi waktu tempuh. Sistem ini membantu pendaki mendapatkan rekomendasi jalur yang terbaik, sehingga meningkatkan keselamatan dan kenyamanan dalam pendakian serta mengurangi risiko kecelakaan.
- b. Landasan untuk Sistem Rekomendasi Serupa: Penelitian ini menjadi dasar bagi pengembangan sistem rekomendasi di berbagai sektor, seperti pariwisata, transportasi, logistik, dan manajemen bencana. Metode-metode

yang digunakan dapat diadaptasi untuk menyelesaikan berbagai masalah yang melibatkan pemilihan jalur atau rute.

- c. **Edukasi dan Kesadaran Komunitas Pendakian:** Penelitian ini memberikan wawasan kepada pendaki dan pengelola jalur pendakian mengenai faktor penting dalam memilih jalur yang aman dan efisien. Penelitian ini juga mengedukasi pendaki tentang pentingnya merencanakan pendakian yang sesuai dengan kemampuannya, sehingga meningkatkan kesadaran akan keselamatan dalam kegiatan pendakian.
- d. **Kontribusi Teknologi Informasi:** Penelitian ini berkontribusi dalam pengembangan teknologi berbasis graf untuk memproses dan menganalisis data terhubung.

### **1.5 Batasan Masalah**

Berdasarkan permasalahan yang ada, maka batasan masalah yang ada pada penelitian ini adalah sebagai berikut:

- a. Lokasi penelitian ini difokuskan pada beberapa gunung di Indonesia seperti Gunung Arjuno, Gunung Bromo, Gunung Butak, Gunung Ciremai, Gunung Penanggungan, Gunung Slamet, Gunung Sumbing dan Gunung Ungaran.
- b. Penelitian ini menggunakan *graph database Neo4j*, dengan menerapkan metode *Weighted Directed Acyclic Graph (WDAG)*, *Analytic Hierarchy Process (AHP)* dan algoritma *Dijkstra*.
- c. Data yang digunakan dalam penelitian ini bersumber dari data publik yang tersedia secara online melalui situs web Gunung Bagging.

## BAB II

### STUDI PUSTAKA

#### 2.1 *Graph*

*Graph* merupakan struktur matematis yang digunakan untuk memodelkan hubungan antara objek dalam sebuah jaringan. Dalam konteks ilmu graf, objek-objek tersebut dikenal sebagai simpul (*vertices*), sedangkan hubungan antara simpul-simpul tersebut disebut sisi (*edges*), yang direpresentasikan secara formal sebagai  $G = (V, E)$ . Graf dapat diterapkan di berbagai bidang, mulai dari ilmu komputer hingga kimia, misalnya untuk memodelkan jaringan sosial, struktur molekul, hingga sistem jalan raya. Sebuah graf bisa berupa graf berarah (*digraph*) atau graf tak berarah, tergantung pada sifat relasi antara simpulnya (Sanuri dkk., 2024). Berdasarkan orientasi *edge*-nya, *graph* dapat dibagi menjadi dua jenis:

- a. Graph tak berarah (*Undirected Graph*): Dalam *graph* ini, *edge* tidak memiliki arah, sehingga hubungan antar node bersifat simetris. Jika ada *edge* antara node  $v_1$  dan  $v_2$ , maka  $(v_1, v_2)$  sama dengan  $(v_2, v_1)$ , yang berarti kedua node tersebut saling terhubung tanpa memperhatikan arah. Contoh dari *graph* tak berarah adalah jaringan sosial di mana hubungan "pertemanan" adalah dua arah.
- b. Graph berarah (*Directed Graph* atau *Digraph*): Dalam *graph* ini, setiap *edge* memiliki arah yang menunjukkan urutan atau aliran dari satu node ke node lainnya. Jika ada *edge* dari  $v_1$  dan  $v_2$ , maka  $(v_1, v_2)$  tidak sama dengan  $(v_2, v_1)$ , sehingga arah hubungan menjadi penting. Misalnya, dalam

konteks rute perjalanan, arah dari satu lokasi ke lokasi lain menunjukkan bahwa satu jalur hanya bisa dilalui dalam satu arah tertentu.

*Graph* dapat direpresentasikan melalui beberapa metode, yang masing-masing memiliki cara penggambaran dan keunggulan tersendiri berdasarkan konteks penggunaannya:

- a. Matriks ketetanggaan (*Adjacency Matrix*): Matriks ketetanggaan (*Adjacency Matrix*) merepresentasikan graf dalam bentuk matriks persegi di mana elemen  $a[i][j]$  menunjukkan ada tidaknya sisi antara simpul  $i$  dan  $j$ . Jika sisi tersebut ada, maka  $a[i][j] = 1$ , jika tidak, maka  $a[i][j] = 0$ . Representasi ini mempermudah analisis graf, seperti menentukan derajat simpul atau mencari jalur antara simpul dengan algoritma matriks. Keunggulan utamanya adalah kemudahan implementasi pada algoritma komputasi berbasis matriks dan efisiensi dalam pengujian konektivitas graf (Holly, 2024). Namun, pendekatan ini memerlukan ruang penyimpanan besar untuk graf jarang.
- b. List ketetanggaan (*Adjacency List*): merepresentasikan graf dengan memetakan setiap simpul ke daftar simpul lain yang terhubung dengannya. Setiap simpul memiliki daftar yang berisi semua simpul tetangga, memungkinkan representasi yang efisien untuk graf jarang. Cara ini menggambarkan graf dengan menyimpan hubungan langsung antar simpul tanpa redundansi, menjadikannya hemat ruang dibandingkan matriks ketetanggaan. Keunggulan utama adalah efisiensi penyimpanan dan kecepatan untuk traversal graf, terutama dalam algoritma seperti DFS atau



BFS. *Adjacency List* sangat cocok untuk graf besar dan jarang karena hanya mencatat koneksi yang ada, menghemat memori secara signifikan (Ahmad dkk., 2023).

- c. *Graphic*: merepresentasikan graf melalui visualisasi diagram, di mana simpul digambarkan sebagai titik, dan sisi sebagai garis yang menghubungkan titik-titik tersebut. Penggambaran ini memberikan pandangan intuitif tentang struktur graf, memungkinkan analisis pola, hubungan, atau jaringan secara langsung. Keunggulannya adalah kemudahan interpretasi, karena manusia lebih cepat memahami data visual dibandingkan data numerik. Grafik sangat membantu dalam analisis sosial, kimia, dan jaringan komputer, di mana pola hubungan kompleks sering kali lebih mudah dimengerti secara visual. Penggunaan grafik memfasilitasi identifikasi anomali atau pola tersembunyi dengan cepat (Neiva & Bruno, 2023).

## 2.2 *Graph Database*

*Graph Database* adalah sistem manajemen basis data yang memanfaatkan struktur graf untuk menyimpan, memproses, dan memanipulasi data berbasis hubungan. Dalam basis data ini, entitas-entitas direpresentasikan sebagai simpul (*nodes*), sedangkan hubungan antar-entitas ditampilkan sebagai sisi (*edges*). Berbeda dari basis data relasional, yang mengandalkan tabel dengan relasi yang terbatas, *graph database* dirancang untuk menangani data dengan relasi kompleks dan hierarkis, seperti jaringan sosial, sistem rekomendasi, dan peta navigasi (Fritchey, 2022).

*Graph database* menggabungkan keunggulan model objek dengan fleksibilitas graf, memungkinkan representasi eksplisit dari hubungan, seperti hierarki atau jaringan, yang sulit dimodelkan dalam struktur basis data tradisional. Teknologi ini mendukung bahasa kueri berbasis graf seperti *Cypher* dan *Gremlin*, yang dirancang untuk operasi pencarian pola, traversal graf, dan analisis data berbasis hubungan (Güting, 1994).

Keunggulan utama *graph database* terletak pada efisiensi dalam penanganan data terhubung, memungkinkan pemrosesan kueri lebih cepat dibandingkan basis data relasional untuk data dengan banyak-to-banyak hubungan. Sebagai contoh, pada aplikasi pusat kontak, *graph database* digunakan untuk menyimpan hubungan antara operasi, entitas, dan atribut, memungkinkan integrasi mulus antar modul seperti penugasan kerja dan pelaporan (Güting, 1994).

*Graph database* juga mendukung paralelisasi dan partisi graf, memungkinkan optimalisasi untuk data berskala besar. Melalui teknik seperti *MapReduce*, database ini memungkinkan analisis data skala besar dengan efisiensi tinggi (Dai & Liu, 2023). Aplikasi nyata dari teknologi ini mencakup manajemen logistik, analisis jejaring sosial, dan pemetaan rute perjalanan.

### **2.3 *Weighted Directed Acyclic Graph (WDAG)***

*Weighted Directed Acyclic Graph (WDAG)* adalah representasi graf berbobot yang terarah dan tidak memiliki siklus, di mana bobot (*weight*) diberikan pada setiap *edge* (sisi) untuk merepresentasikan nilai tertentu seperti jarak, biaya, atau probabilitas. Graf ini memiliki karakteristik utama bahwa ia terarah (*directed*), yaitu setiap *edge* memiliki arah tertentu yang menghubungkan node asal ke node

tujuan, dan tidak memiliki siklus (*acyclic*), yang berarti tidak ada jalur di mana sebuah *node* dapat kembali ke dirinya sendiri melalui rangkaian *edge*. Struktur ini sering digunakan dalam berbagai aplikasi seperti perencanaan proyek, analisis jaringan, pembelajaran mesin, dan *blockchain* (Nilsson dkk., 2021).

WDAG secara matematis didefinisikan sebagai  $G = (V, E, W)$  dimana  $V$  adalah himpunan *node* (*vertices*),  $E \subseteq V \times V$  adalah himpunan *edge* terarah dan  $W: E \rightarrow R$  adalah fungsi bobot yang memetakan setiap *edge* ke nilai real.

Keberadaan bobot memberikan konteks tambahan pada hubungan antar *node*, seperti waktu penyelesaian tugas dalam grafik perencanaan proyek atau biaya jalur transportasi. Tidak adanya siklus menjadikan WDAG ideal untuk model yang membutuhkan hierarki atau urutan sebab-akibat, karena tidak ada hubungan yang kontradiktif (Sun dkk., 2023).

Dalam aplikasi nyata, salah satu penggunaan populer WDAG adalah pada algoritma perencanaan jalur, seperti algoritma *Dijkstra* atau *Bellman-Ford*, yang digunakan untuk menemukan jalur terpendek dalam graf berbobot. Dalam ilmu komputer, WDAG juga digunakan dalam perhitungan rangkaian tugas (*task scheduling*) dan jaringan syaraf tiruan untuk menentukan aliran data antara node (Rose dkk., 2024).

Salah satu keuntungan utama WDAG adalah sifatnya yang dapat dimanfaatkan untuk analisis topologis. Urutan topologis memungkinkan kita menyusun node dalam urutan linier berdasarkan ketergantungan (*dependency*) mereka, sehingga mempermudah penyelesaian masalah seperti evaluasi ekspresi matematika atau kompilasi kode sumber (Aisyah dkk., 2020).

Referensi akademik menunjukkan pentingnya WDAG dalam berbagai disiplin ilmu. Sebagai contoh, dalam pembelajaran mesin, WDAG sering digunakan dalam jaringan *Bayesian*, di mana bobot menunjukkan probabilitas bersyarat antar variabel. Dalam sistem terdistribusi seperti *blockchain*, WDAG digunakan untuk mencatat transaksi dalam struktur yang efisien dan tanpa konflik (Yu dkk., 2024).

#### **2.4 Analytic Hierarchy Process (AHP)**

*Analytic Hierarchy Process* (AHP) adalah metode pengambilan keputusan multi-kriteria yang dikembangkan oleh Thomas L. Saaty pada tahun 1970-an untuk membantu individu atau kelompok dalam memilih solusi optimal berdasarkan berbagai alternatif dengan mempertimbangkan beberapa kriteria yang saling terkait. AHP didasarkan pada pendekatan hierarkis, di mana masalah kompleks dipecah menjadi elemen-elemen kecil dalam bentuk struktur hierarki yang terdiri dari tujuan utama, kriteria, subkriteria, dan alternatif. Pendekatan ini memanfaatkan perbandingan berpasangan (*pairwise comparison*) untuk menilai tingkat kepentingan relatif antara elemen-elemen dalam hierarki (Saaty, 2002).

Proses utama dalam AHP melibatkan beberapa langkah, yaitu:

- a. Membangun Hierarki: Menstrukturkan masalah pengambilan keputusan menjadi beberapa tingkatan, mulai dari tujuan di tingkat atas, diikuti oleh kriteria, subkriteria, dan alternatif di bawahnya.
- b. Melakukan Perbandingan Berpasangan: Setiap elemen dibandingkan secara berpasangan dalam hal pentingnya terhadap elemen di atasnya, menggunakan skala yang biasanya berkisar dari 1 (sama penting) hingga 9

(sangat penting). Matrik perbandingan berpasangan ini membantu membangun bobot prioritas.

- c. Menghitung Bobot Prioritas: Melalui normalisasi matriks atau metode *eigenvector*, bobot dari setiap kriteria dan alternatif dihitung.
- d. Memeriksa Konsistensi: Indeks konsistensi (*Consistency Index*) digunakan untuk memastikan bahwa penilaian dalam perbandingan berpasangan cukup konsisten. Jika rasio konsistensi (*Consistency Ratio*) lebih dari 0.1, evaluasi perlu ditinjau ulang.
- e. Membuat Keputusan Akhir: Bobot dari setiap alternatif dijumlahkan berdasarkan bobot kriteria untuk menentukan solusi terbaik.

*AHP* sering digunakan dalam berbagai bidang seperti manajemen, perencanaan strategis, dan rekayasa, di mana pengambilan keputusan melibatkan banyak kriteria yang saling bertentangan. Salah satu kekuatan *AHP* adalah kemampuannya untuk menggabungkan data kuantitatif dan kualitatif, sehingga cocok untuk situasi dengan ketidakpastian atau informasi yang tidak lengkap (Darko dkk., 2019). Misalnya, dalam penentuan lokasi proyek, *AHP* dapat membantu pengambil keputusan menilai alternatif lokasi berdasarkan kriteria seperti biaya, aksesibilitas, dan dampak lingkungan. Metode ini juga dapat diintegrasikan dengan teknik lain, seperti GIS atau *Fuzzy Logic*, untuk meningkatkan akurasi pengambilan keputusan.

Studi menunjukkan bahwa *AHP* adalah metode yang efisien dan fleksibel. Menurut Saaty (2002), *AHP* dapat membantu menangani masalah pengambilan

keputusan yang kompleks dengan mempermudah penguraian masalah dan memberikan solusi yang sistematis (Saaty, 2002).

## 2.5 Dijkstra

Algoritma *Dijkstra* adalah metode yang digunakan untuk menemukan jalur terpendek dari satu titik asal (*source*) ke titik tujuan dalam graf berbobot. Dikembangkan oleh Edsger W. *Dijkstra* pada tahun 1956, algoritma ini bekerja pada graf berarah maupun tak berarah dengan bobot sisi non-negatif. Tujuan utama algoritma ini adalah menentukan jalur dengan bobot minimum antara node awal ke node lainnya dalam graf (Javaid, 2013).

Algoritma *Dijkstra* menggunakan pendekatan *greedy*, yaitu memilih solusi optimal lokal pada setiap langkah dengan asumsi bahwa langkah ini mengarah ke solusi global optimal. Prosesnya melibatkan iterasi melalui node dengan memperbarui jarak terpendek dari node asal ke node lainnya, sambil mempertahankan daftar node yang sudah diproses (Inayah dkk., 2023).

Proses utama dalam AHP melibatkan beberapa langkah, yaitu:

- a. Inisialisasi: Atur jarak awal semua *node* dari *node* asal sebagai tak hingga ( $\infty$ ) kecuali node asal yang diatur ke nol.
- b. Pilih *Node* dengan Jarak Minimum: Pilih *node* yang belum diproses dengan nilai jarak minimum.
- c. Perbarui Jarak: Untuk setiap tetangga dari *node* terpilih, hitung jarak melalui *node* ini. Jika jarak yang dihitung lebih kecil dari jarak sebelumnya, perbarui nilai jarak.

- d. Tandai *Node* Sebagai Diproses: Setelah semua tetangga diperiksa, tandai *node* ini sebagai selesai.
- e. Ulangi: Proses berlanjut hingga semua *node* telah diproses atau jalur terpendek ke *node* tujuan ditemukan.

Rumus utama dalam algoritma *Dijkstra* digunakan untuk memperbarui jarak terpendek ke suatu node  $v$  dinyatakan dalam persamaan (2.1):

$$d[v] = \min(d[v], d[u] + w(u, v)) \quad (2.1)$$

Dimana:

$d[v]$ : jarak terpendek saat ini ke node  $v$ ,

$d[u]$ : jarak terpendek ke node  $u$  (node yang sedang diproses),

$w(u, v)$ : bobot sisi antara node  $u$  dan  $v$ .

Algoritma *Dijkstra* banyak digunakan dalam perencanaan jaringan komputer (misalnya protokol routing), transportasi, dan analisis sistem. Keunggulan utamanya adalah efisiensi dalam graf skala kecil hingga menengah dan ketepatan dalam menemukan solusi terbaik. Namun, kelemahannya adalah algoritma ini tidak dapat menangani bobot sisi negatif, karena asumsi *greedy*-nya tidak lagi valid (Muharrom, 2020).

## 2.6 Avenza Maps

*Avenza Maps* adalah aplikasi pemetaan berbasis *Android* yang memungkinkan pengguna untuk mengakses peta *PDF* secara offline, yang berguna untuk berbagai aktivitas luar ruangan, seperti pendakian gunung atau navigasi di area terpencil. Salah satu fitur utama *Avenza Maps* adalah kemampuannya untuk menampilkan koordinat geografis, elevasi, dan jarak langsung pada peta *PDF*. Ini sangat bermanfaat bagi pendaki gunung yang menggunakan peta dari situs seperti

[www.gunungbagging.com](http://www.gunungbagging.com), yang menyediakan peta topografi dalam format *PDF* untuk berbagai gunung dan jalur pendakian di Indonesia.

Dengan menggunakan *Avenza Maps*, pengguna dapat mengimpor peta *PDF* yang diunduh dari situs seperti [gunungbagging.com](http://gunungbagging.com) ke dalam aplikasi. Setelah peta dimuat, *Avenza Maps* memungkinkan pengguna untuk mendapatkan informasi terkait titik koordinat (*latitude* dan *longitude*) pada setiap lokasi di peta dengan cukup akurat, berkat integrasi *GPS* perangkat *Android*. Informasi ini sangat berguna bagi pendaki yang ingin memverifikasi posisi mereka di lapangan dan memastikan mereka mengikuti jalur yang benar.

Selain itu, *Avenza Maps* juga dapat memberikan informasi elevasi pada titik tertentu, yang penting untuk memahami perubahan ketinggian sepanjang jalur pendakian. Elevasi ini ditampilkan dalam satuan meter atau kaki, tergantung pengaturan pengguna. Kemampuan untuk mengukur jarak antar titik di peta juga menjadi fitur utama, yang memungkinkan pendaki untuk mengetahui sejauh mana mereka telah berjalan dan seberapa jauh lagi mereka harus menempuh perjalanan menuju tujuan.

Fitur-fitur ini, dikombinasikan dengan peta topografi yang rinci dari [gunungbagging.com](http://gunungbagging.com), membuat *Avenza Maps* menjadi alat yang sangat berguna untuk navigasi dan perencanaan jalur pendakian di Indonesia, terutama di daerah yang minim sinyal internet. Dengan kemampuan untuk bekerja secara offline, *Avenza Maps* menjadi alat yang sangat efisien dan efektif dalam meningkatkan keselamatan dan pengalaman pendakian.



## 2.7 Trigonometri

Trigonometri digunakan untuk menghitung kemiringan jalur, terutama dalam konteks perencanaan jalan, konstruksi, dan analisis geospasial. Kemiringan jalur sering kali diukur sebagai sudut antara jalur dan permukaan horizontal (tanah atau dasar). Sudut ini disebut sebagai sudut kemiringan atau gradien dan dihitung menggunakan konsep trigonometri, khususnya fungsi *tangen* (Gur, 2009).

Untuk mendapatkan sudut kemiringan dalam derajat, kita dapat menggunakan *invers* dari *tangen* yang dinyatakan dalam persamaan (2.2):

$$\theta = \text{atan} \frac{\text{Selisih Elevasi}}{\text{Jarak Horizontal}} \quad (2.2)$$

Dimana:

$\theta$  adalah sudut kemiringan,

Selisih elevasi adalah perubahan ketinggian antara dua titik sepanjang jalur,

Jarak horizontal adalah jarak lurus antara dua titik di permukaan tanah.

Kemiringan jalur ini penting dalam berbagai aplikasi, seperti perancangan jalan raya, jalur kereta api, dan pembuatan jalur pendakian, karena mempengaruhi kenyamanan, keamanan, dan efisiensi perjalanan.

## 2.8 Naismith Rule

*Naismith's Rule* adalah suatu rumus yang digunakan untuk menaksir berapa lama waktu yang dibutuhkan untuk menyelesaikan suatu rute dengan berjalan kaki. Teori ini pertama kali diperkenalkan oleh pendaki gunung & pemandu asal Skotlandia, William Naismith, pada tahun 1892. Menurutnya, kecepatan rata-rata orang berjalan adalah 5 km/jam (Scarf, 2007).

Setelah Naismith Rule menjadi acuan dalam memperkirakan estimasi waktu perjalanan yang cukup lama sejak diperkenalkan, seiring berkembangnya waktu,

seorang penulis manual klasik pada bukunya *Mountcraft and Leadership*, Eric Langmuir, memperbarui aturan *Naismith* dengan pembaruan mengurangi kecepatan rata-rata berjalan di medan yang datar menjadi 2 km/jam (Langmuir, 2013).

Dari penjelasan diatas maka dapat dituliskan rumus dasar dalam persamaan (2.3):

$$T = \frac{D}{V} \quad (2.3)$$

Dengan  $D$  = jarak jalur dan  $V$  = kecepatan rata-rata atau 2 km/jam atau 33.34 meter/menit.

## 2.9 Model Ukur Kompleksitas *Chyper Query Graph*

Dalam konteks pengukuran kompleksitas *Chyper Query* pada sistem yang menggunakan *Database* maka diperlukan model yang mempertimbangkan atribut atau parameter. Penelitian yang dilakukan oleh Agus Putra Subali dan Siti Rochimah mengusulkan sebuah model baru untuk mengukur kompleksitas *Chyper Query* dengan mempertimbangkan atribut-atribut *Query* yang digunakan dalam sistem database. Proses penghitungan kompleksitas *Chyper Query* dimulai dengan mengidentifikasi jumlah kelima atribut dalam *Chyper Query* yang digunakan, yaitu: *Variable Output*, *Variable Input*, *Nested Query*, *Relationship*, *Node* (Subali & Rochimah, 2018). Untuk menghitung bobot setiap atribut disini diusulkan nilai bobot pada setiap atribut seperti pada tabel 2.1 berikut.

Tabel 2.1 Bobot Atribut *Chyper Query*

No	Atribut <i>Chyper Query</i>	Bobot
1	<i>Variable Output</i>	0.1
2	<i>Variable Input</i>	0.15
3	<i>Nested Query</i>	0.2
4	<i>Relationship</i>	0.25
5	<i>Node</i>	0.3

Perhitungan kompleksitas *Chyper Query* memperhatikan jumlah setiap atribut *Chyper Query* dan Bobotnya. Rumus untuk menghitung kompleksitas *Chyper Query* dinyatakan dalam persamaan (2.5) berikut:

$$C = \sum_{i=1}^n X_i W_i \quad (2.4)$$

Dimana:

$n$  adalah total atribut chyper query.

$X_i$  adalah jumlah atribut chyper query  $i$ .

$W_i$  adalah nilai bobot atribut chyper query  $i$ .

## 2.10 Penelitian Terkait

Penelitian mengenai *Implementasi Graph Database untuk Menentukan Rute Perjalanan Transportasi Umum* di Kota Malang dan Kota Batu menggunakan pendekatan *Weighted Directed Acyclic Graph* (WDAG) dan *Analytic Hierarchy Process* (AHP). WDAG memetakan hubungan antar titik dan jalur dalam jaringan transportasi, sementara AHP menentukan bobot kriteria seperti jarak, waktu tempuh, dan kemacetan. Implementasi menggunakan *Neo4j* memungkinkan pengolahan data rute yang lebih efisien. Hasilnya menunjukkan akurasi tinggi dalam penentuan rute optimal berdasarkan kriteria yang telah ditetapkan, meningkatkan kualitas rekomendasi rute transportasi umum (Ardani dkk., 2019).

Penelitian *Analisa Perbandingan Algoritma Floyd-Warshall dan Dijkstra* untuk penentuan rute terdekat di Kota Kediri membandingkan dua algoritma populer. *Floyd-Warshall* menemukan semua pasangan jalur terpendek, cocok untuk jaringan besar, sementara *Dijkstra* lebih efisien untuk menentukan jalur terpendek dari satu titik ke titik tujuan tertentu. Hasil menunjukkan *Dijkstra* lebih efisien, terutama pada jaringan rute kecil atau pencarian satu rute (Inayah dkk., 2023).

Penelitian tentang *Implementasi Algoritma Dijkstra dan Metode Haversine pada Penentuan Jalur Terpendek Pendakian Gunung Merapi Jalur Selo* menggunakan *Dijkstra* untuk menentukan jalur terpendek dalam jaringan pendakian dan *Metode Haversine* untuk menghitung jarak berdasarkan koordinat geografis. Aplikasi berbasis Android yang dikembangkan mampu memberikan rute terpendek dengan bobot terkecil sesuai kebutuhan pengguna (Sumaryo dkk., 2020).

Penelitian tentang *Implementasi Graph Database pada Rute Bus Rapid Transit (BRT)* menggunakan *Neo4j* untuk memodelkan jaringan rute BRT, dengan algoritma *Breadth-First Search (BFS)* dan *Depth-First Search (DFS)* untuk traversing. Sistem ini memberikan rekomendasi rute perjalanan dari shelter asal ke tujuan, termasuk perpindahan antar koridor jika diperlukan, sehingga meningkatkan efisiensi perjalanan pengguna (Wirawan & Riyanto, 2017).

Penelitian mengenai *Pencarian Rute Optimal dengan Metode Steepest Ascent Hill Climbing (SAHC)* untuk tempat wisata di Bandung menggunakan SAHC untuk mencari rute optimal berdasarkan nilai heuristik tertinggi. Sistem ini memberikan rute wisata yang lebih efisien dibandingkan metode lain seperti Dijkstra, dan diintegrasikan dalam aplikasi Android untuk memudahkan pengguna mencari rute yang relevan (Nurhasanah dkk., 2021)

Tabel 2.2 Penelitian Terkait

No	Peneliti	Input	Metode	Output	Perbedaan
1	M. Syauqi Hanif Ardani, Muhammad Ainul Yaqin, dan Suhartono	Data transportasi umum angkot di Kota Malang dan Kota Batu	<i>Weighted Directed Acyclic Graph (WDAG)</i> dan <i>Analytic Hierarchy Process (AHP)</i>	Penelitian ini menunjukkan bahwa implementasi <i>Graph Database</i> dan perhitungan AHP menghasilkan nilai <i>Precision</i> yang tinggi.	Kelebihan: Penggunaan graph database <i>Neo4j</i> efisien; Kekurangan: Fokus pada transportasi umum, bukan pendakian gunung

No	Peneliti	Input	Metode	Output	Perbedaan
2	Ahmada Maghfirotul Inayah, Nalsa Cintya Resti, dan Nur Fadilatul Ilmiyah	Rute wilayah Kota Kediri	<i>Algoritma Floyd-Warshall</i> dan <i>Algoritma Dijkstra</i>	<i>Algoritma Dijkstra</i> terbukti efektif dalam menyediakan rute terpendek antara dua titik.	Kelebihan: Perbandingan algoritma; Kekurangan: Tidak mengintegrasikan database graph untuk rekomendasi
3	Roddy Yoto Sumaryo, Paulus Harsadi, dan Didik Nugroho	Jalur pendakian Gunung Merapi melalui jalur Selo	<i>Algoritma Dijkstra</i> dan <i>Metode Haversine</i>	Aplikasi menghasilkan rute terpendek dari titik awal ke tujuan menggunakan <i>Algoritma Dijkstra</i> dengan bobot terkecil.	Kelebihan: Fokus pada pendakian gunung; Kekurangan: Tidak menggunakan graph database sehingga skala data dan kompleksitas terbatas
4	Panji Wisnu Wirawan dan Djalal Er Riyanto	Shelter (halte) dan rute BRT, dengan input shelter asal dan tujuan dari pengguna	<i>Graph Database</i> dan <i>Algoritma Traversing</i>	Informasi rute bus dari shelter asal ke tujuan, termasuk perpindahan koridor, digunakan untuk menyusun program rute BRT yang efisien.	Kelebihan: Pemodelan jaringan transportasi dengan graph database; Kekurangan: Fokus pada transportasi umum, bukan pendakian
5	Youllia Indrawati Nurhasanah, Sofia Umaroh, dan Ninda Halimah Al Ghoniyyah	Daftar tempat wisata di Bandung beserta latitude dan longitude yang diambil dari Google Maps.	<i>Algoritma Steepest Ascent Hill Climbing (SAHC)</i> dan <i>Graph</i>	Hasil pengujian menunjukkan SAHC lebih efektif daripada Dijkstra karena hanya berfokus pada nilai heuristik terbaik tanpa mengunjungi semua node.	Kelebihan: Pendekatan heuristik; Kekurangan: Tidak berbasis graph database dan tidak fokus pada pendakian gunung

Penelitian-penelitian sebelumnya telah menunjukkan berbagai metode dan pendekatan dalam menentukan rute optimal, baik pada transportasi umum maupun jalur pendakian gunung. Penggunaan *graph database* seperti Neo4j terbukti meningkatkan efisiensi pengolahan data jaringan rute (Syauqi dkk., 2019; Wirawan

& Riyanto, 2017), sementara algoritma seperti *Dijkstra* dan *Floyd-Warshall* sering digunakan untuk pencarian jalur terpendek dengan keunggulan dan keterbatasannya masing-masing (Inayah dkk., 2023; Harsadi & Nugroho, 2020).

Namun, sebagian besar penelitian yang mengimplementasikan *graph database* berfokus pada jaringan transportasi umum, dan belum banyak yang mengaplikasikan pada konteks pendakian gunung dengan mempertimbangkan variabel seperti jarak, elevasi, kemiringan dan waktu tempuh secara menyeluruh. Selain itu, beberapa penelitian terkait jalur pendakian hanya menggunakan algoritma jalur terpendek tanpa memanfaatkan *graph database* yang mampu menangani data besar dan kompleks secara efisien.

Penelitian ini bertujuan untuk mengisi gap tersebut dengan mengembangkan sistem rekomendasi rute pendakian yang memanfaatkan *graph database* untuk mengoptimalkan berbagai faktor seperti jarak, elevasi, kemiringan, dan waktu tempuh, sekaligus menyesuaikan rekomendasi dengan karakteristik dan preferensi pendaki guna meningkatkan kenyamanan dan kualitas pengalaman pendakian.

## BAB III

### METODOLOGI PENELITIAN

#### 3.1 Gambaran Umum

Penelitian ini mengembangkan sebuah sistem rekomendasi rute pendakian gunung yang berbasis pada beberapa kriteria, seperti jarak, selisih elevasi, kemiringan, dan estimasi waktu tempuh. Sistem ini menggunakan *Weighted Directed Acyclic Graph* (WDAG) untuk memodelkan jalur pendakian, di mana bobot setiap jalur dihitung menggunakan metode *Analytic Hierarchy Process* (AHP). Bobot ini ditentukan berdasarkan prioritas relatif dari setiap kriteria. Setelah bobot setiap jalur diperoleh, algoritma *Dijkstra* akan mencari rute terbaik dengan total bobot terendah dari titik awal hingga tujuan.

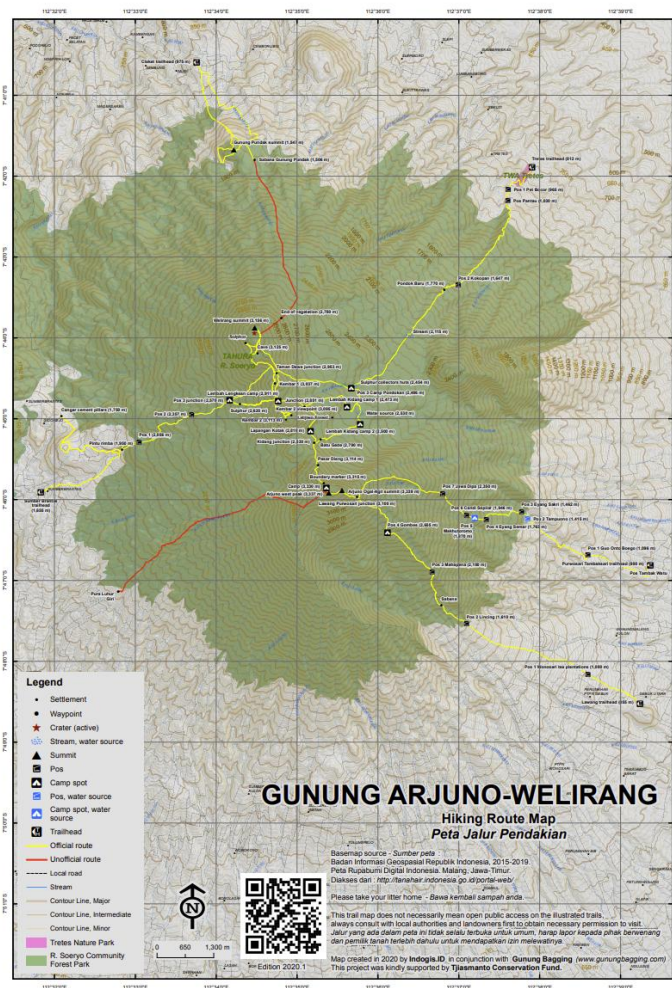
Sistem ini dirancang untuk memberikan fleksibilitas kepada pengguna dalam menentukan preferensi pendakian, seperti memilih gunung, titik awal, dan tujuan pendakian. Berdasarkan input dari pengguna, sistem akan menghasilkan beberapa alternatif rute yang telah diurutkan berdasarkan bobot totalnya. Rute dengan bobot terendah dianggap sebagai rute terbaik.

#### 3.2 Sumber Data

Penelitian ini menggunakan data peta digital dalam format *PDF* yang memuat informasi lengkap mengenai titik lokasi dan jalur pendakian gunung. Data ini diperoleh dari situs [www.gunungbagging.com](http://www.gunungbagging.com), sebuah platform yang mengumpulkan informasi jalur pendakian, elevasi, dan titik-titik penting dari

gunung-gunung di Indonesia. Situs ini merupakan salah satu sumber terpercaya bagi pendaki yang mencari data akurat terkait rute pendakian.

Peta digital pada *Gunung Bagging* bersumber dari Badan Informasi Geospasial (BIG) Republik Indonesia, yang menyediakan data geospasial resmi tentang topografi, elevasi, dan peta rute pendakian. Data ini sangat berguna untuk analisis yang mendalam terkait pemetaan ketinggian, selisih elevasi, dan penentuan rute terbaik berdasarkan kriteria sistem rekomendasi yang digunakan dalam penelitian ini. Gambar 3.1 memperlihatkan contoh peta jalur pendakian Gunung Arjuno Welirang.

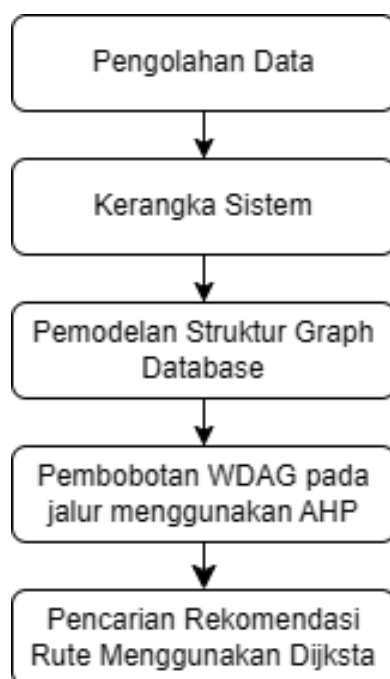


Gambar 3.1 Peta Jalur Pendakian Gunung Arjuno-Welirang



### 3.3 Desain Penelitian

Desain penelitian ini menjelaskan langkah-langkah utama yang ditempuh dalam proses pengembangan sistem rekomendasi rute pendakian gunung, mulai dari pengolahan data hingga pencarian rekomendasi rute. Setiap tahapan dirancang untuk memastikan bahwa data yang digunakan telah diolah dan dianalisis secara optimal agar dapat memberikan rekomendasi jalur pendakian yang terbaik bagi pengguna.



Gambar 3.2 Tahapan Penelitian

Setiap tahapan yang ada pada Gambar 3.2 akan dijelaskan lebih lengkap pada pembahasan berikutnya.

#### 3.3.1 Pengolahan Data menggunakan *Avenza Maps*

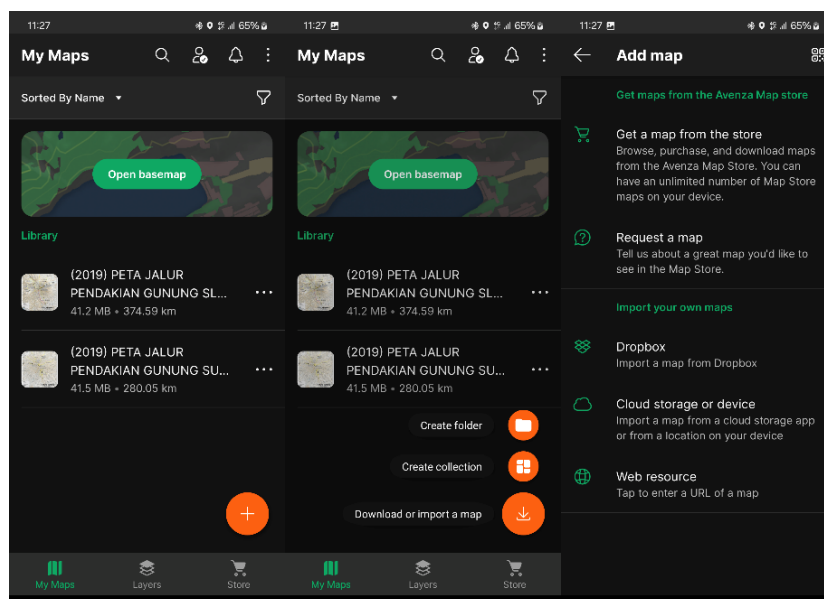
Proses pengolahan data dimulai dengan mengimpor peta dalam format *PDF* ke dalam aplikasi *Avenza Maps*. Setelah peta digital berhasil diimpor, beberapa

tahapan dilakukan, seperti penandaan lokasi, penggambaran dan pengukuran jalur, serta penghitungan kemiringan dan waktu tempuh. Langkah-langkah ini bertujuan untuk menghasilkan data yang terstruktur dalam bentuk tabel berformat *CSV*, agar memudahkan proses analisis dan pemrosesan lebih lanjut. Data *CSV* yang dihasilkan ini akan menjadi dasar bagi pemodelan graf dan perhitungan bobot dalam sistem rekomendasi rute, memastikan bahwa setiap detail topografi jalur telah terekam dengan akurat.

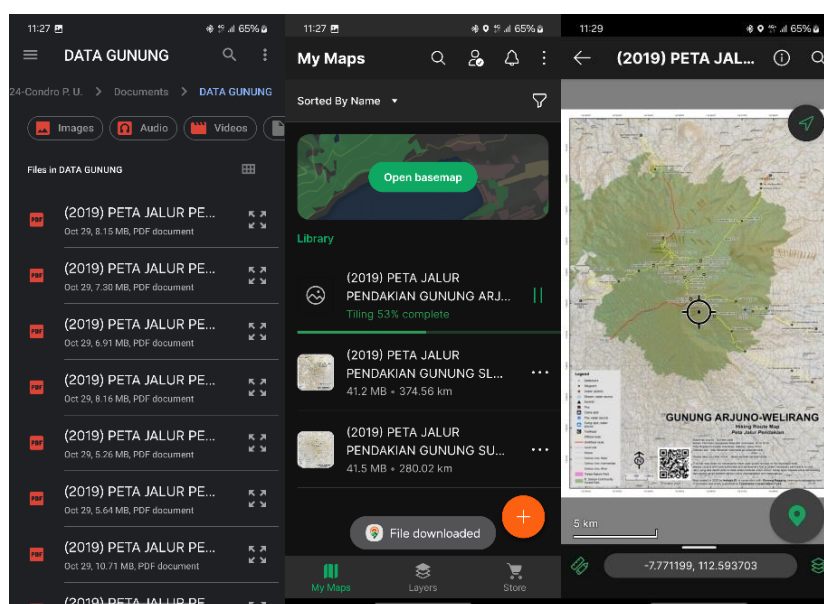
#### **A. Import Peta PDF kedalam *Avenza Maps***

Langkah pertama yaitu import peta *PDF* yang telah diunduh dari situs [www.gunungbagging.com](http://www.gunungbagging.com) kedalam *Avenza Maps*. Berikut adalah langkah-langkah detail yang disertai penjelasan tambahan pada Gambar 3.3 dan 3.4:

- a. Buka aplikasi *Avenza Maps* di perangkat Anda. Di halaman utama "*My Maps*", temukan dan pilih ikon bulat berlambang "+".
- b. Setelah menekan ikon "+", akan muncul beberapa pilihan. Pilih opsi "*Download or import a map*".
- c. Selanjutnya, pilih "*Cloud storage or device*". karena peta sudah kita persiapkan di penyimpanan *device*.
- d. Setelah kita diarahkan ke dalam penyimpanan *device*, cari lokasi folder yang berisi peta dan kemudian pilih peta yang akan diimpor,
- e. Setelah peta dipilih, tunggu proses impor peta selesai. Setelah berhasil diimpor, peta akan siap digunakan.



Gambar 3.3 *Import Peta ke dalam Avenza Maps*



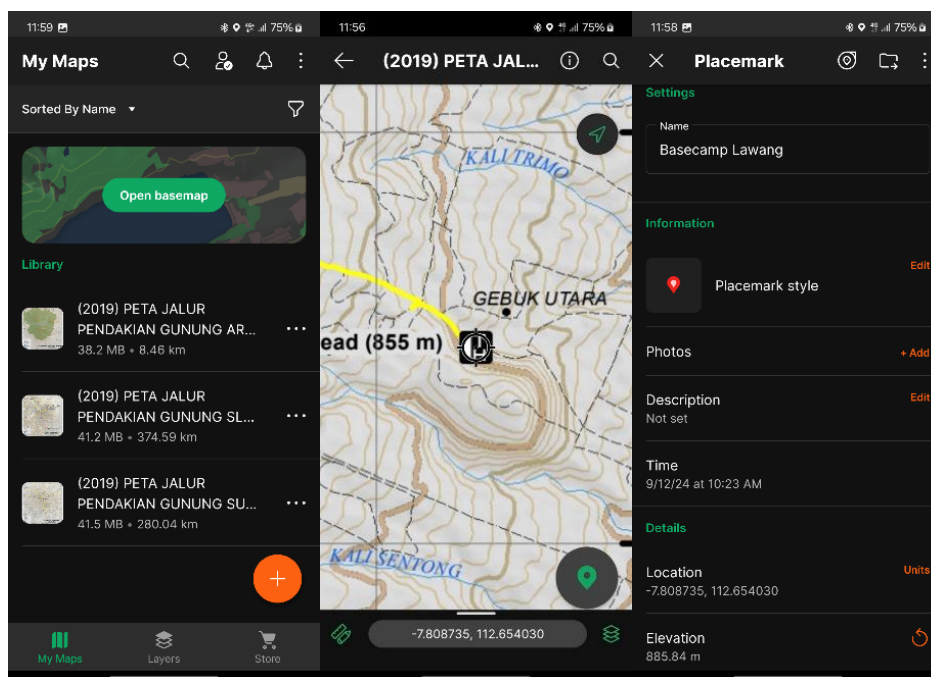
Gambar 3.4 *Import Peta ke dalam Avenza Maps*

## B. Menandai Lokasi dalam Peta Jalur pendakian

Menandai lokasi-lokasi penting dalam peta adalah langkah krusial berikutnya. Proses ini diperlukan untuk mengidentifikasi titik-titik relevan dalam analisis jalur pendakian, seperti *Basecamp*, pos pendakian, persimpangan jalur, dan

puncak. Dalam tahap penandaan lokasi, kita dapat memperoleh data berupa titik koordinat dan elevasi, yang nantinya akan digunakan untuk analisis rute. Berikut adalah penjelasan langkah-langkah dalam proses penandaan lokasi, seperti yang terlihat pada Gambar 3.5:

- a. Pada halaman *My Maps*, pilih peta yang akan dikelola, misalnya Peta Arjuno Welirang. Setelah peta terbuka, posisikan kursor Anda secara akurat pada titik penting yang akan ditandai, semisal "*Basecamp Lawang*". Presisi dalam penempatan kursor sangat penting untuk mendapatkan data lokasi yang akurat.
- b. Selanjutnya, pilih ikon lokasi yang terletak di pojok bawah kanan layar. Setelah itu, halaman *Placemark* akan muncul, di mana Anda dapat mengatur nama lokasi yang ditandai sesuai kebutuhan. Di bagian bawah halaman ini, sistem secara otomatis akan menampilkan koordinat (lintang dan bujur) serta elevasi (ketinggian) dari titik penting yang telah Anda tandai. Informasi ini sangat berharga untuk membangun basis data geografis jalur pendakian.
- c. Setelah mendapatkan semua informasi yang diperlukan, masukkan data Nama Lokasi, Koordinat, dan Elevasi ini ke dalam tabel data lokasi yang telah disiapkan. Proses pencatatan data ini memastikan bahwa setiap titik penting tercatat dengan baik dan dapat diakses untuk analisis lebih lanjut dalam perencanaan rute pendakian.



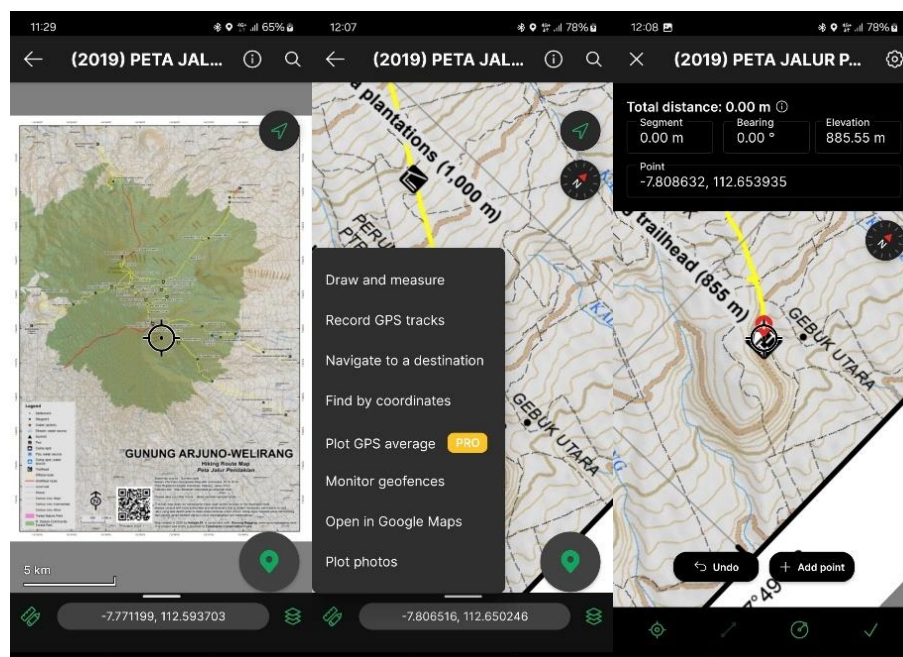
Gambar 3.5 Penandaan Lokasi dalam Peta Jalur Pendakian

### C. Menggambar dan Mengukur Jalur Pendakian

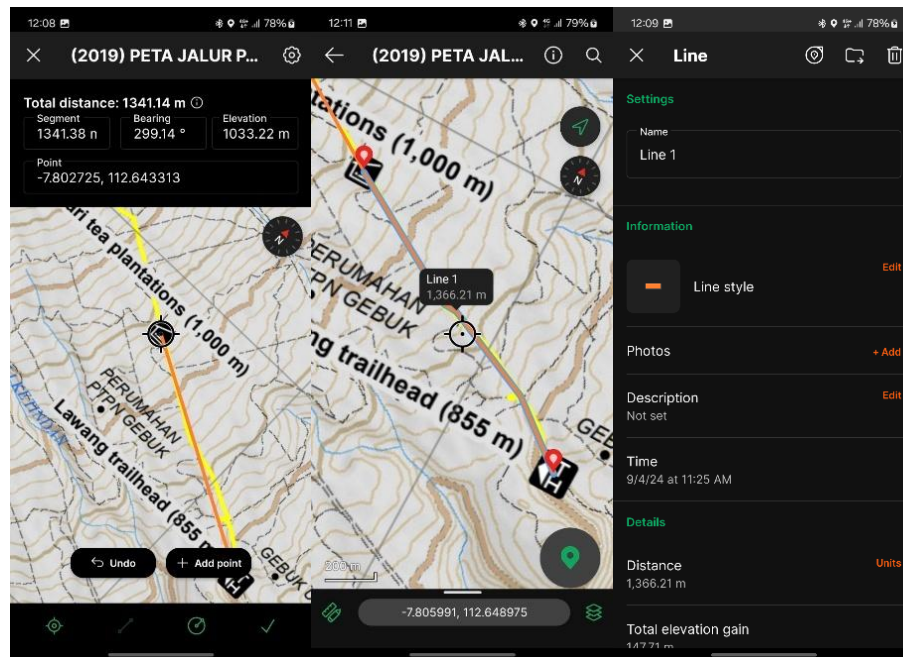
Menggambar dan mengukur jalur pendakian pada peta merupakan proses esensial untuk mendapatkan informasi mengenai **jarak total** dan **jarak horizontal** setiap segmen jalur. Dalam tahapan ini, data penting seperti jarak jalur dan jarak horizontal jalur dapat diperoleh. Informasi ini sangat vital untuk analisis dan pemodelan rute pendakian yang akurat. Berikut adalah penjelasan langkah-langkah dalam proses penggambaran dan pengukuran jalur, seperti yang terlihat pada Gambar 3.6 dan Gambar 3.7:

- a. Pada halaman *My Maps* di aplikasi *Avenza Maps*, pilih peta yang akan Anda kelola, contohnya Peta Arjuno Welirang.
- b. Selanjutnya, pilih ikon "Pensil dan Penggaris" yang biasanya terletak di *toolbar* aplikasi, lalu pilih opsi "*Draw and measure*" untuk memulai proses penggambaran jalur.

- c. Arahkan kursor Anda ke titik awal jalur pada peta, kemudian pilih "Add point" untuk menambahkan titik pertama. Geser kursor Anda secara hati-hati ke titik berikutnya, seperti pada setiap belokan jalur atau perubahan arah yang signifikan, dan teruskan proses ini hingga Anda mencapai titik akhir jalur yang ingin diukur. Penambahan titik-titik ini akan membentuk garis jalur yang presisi sesuai dengan kondisi di lapangan.
- d. Setelah seluruh jalur berhasil digambar dengan menghubungkan semua titik yang diperlukan, pilih ikon "Centang" yang biasanya berada di pojok kanan bawah layar untuk menyelesaikan proses penggambaran. Langkah ini akan menyimpan jalur yang telah Anda buat dan menguncinya dalam peta.
- e. Jalur yang sudah digambar akan tampil, dan ketika kita memilih jalur tersebut, akan muncul halaman "Line" yang berisi informasi, termasuk jarak jalur yang telah diukur secara otomatis.



Gambar 3.6 Penggambaran dan Pengukuran Jalur dalam Peta Jalur Pendakian



Gambar 3.7 Penggambaran dan Pengukuran Jalur dalam Peta Jalur Pendakian

#### D. Menghitung Kemiringan menggunakan Trigonometri

Menghitung kemiringan jalur pendakian penting untuk mengetahui tingkat kemiringan jalur, yang menjadi salah satu aspek utama dalam menentukan rute yang tepat. Kemiringan jalur ini dapat mempengaruhi tingkat kesulitan pendakian dan menjadi pertimbangan dalam memberikan rekomendasi rute. Untuk menghitung kemiringan, digunakan rumus trigonometri untuk menentukan sudut kemiringannya. Berikut adalah contoh perhitungan kemiringan pada jalur antara *Basecamp* Lawang dan Pos 1 Kebun Teh Wonosari, dengan selisih elevasi 145 m dan jarak horizontal 1356 m. Berdasarkan persamaan (2.3) maka hasil perhitungan kemiringan pada jalur tersebut adalah sebagai berikut:

$$\theta = \text{atan} \frac{\text{Selisih Elevasi}}{\text{Jarak Horizontal}} = \text{atan} \frac{145}{1356} = 6.104^\circ$$

### E. Menghitung Waktu Tempuh menggunakan Naismith's Rule

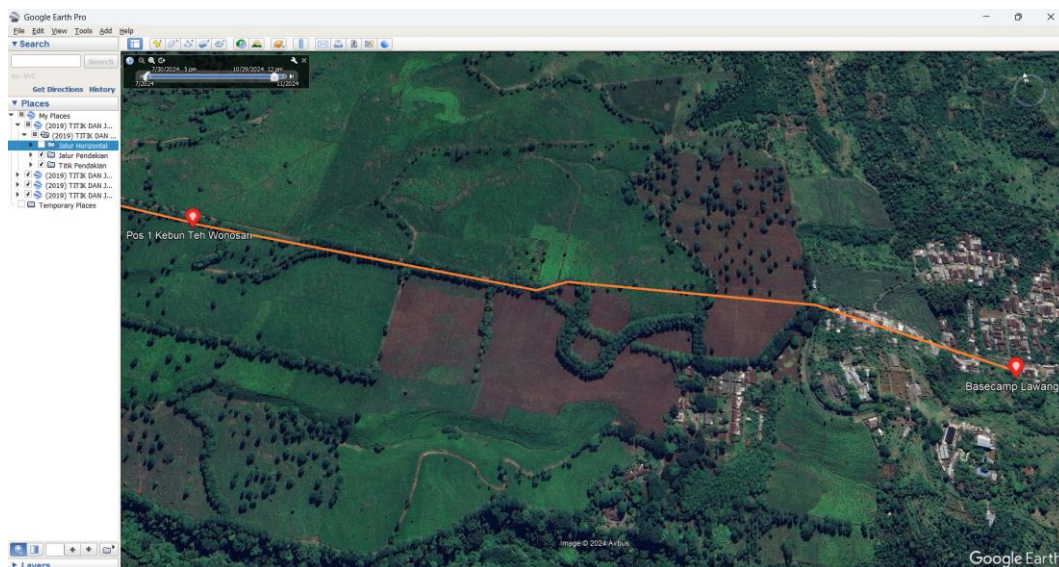
Menghitung waktu tempuh pada setiap jalur pendakian penting untuk mengetahui estimasi durasi perjalanan. Waktu tempuh ini juga menjadi salah satu faktor yang dipertimbangkan oleh sistem dalam memberikan rekomendasi rute. Perhitungan waktu tempuh dilakukan menggunakan *Naismith's Rule*, sebagaimana yang telah dijelaskan dalam pembahasan pada Bab 2. Berikut adalah contoh perhitungan waktu tempuh pada jalur antara *Basecamp* Lawang dan Pos 1 Kebun Teh Wonosari, dengan jarak 1366 m, kenaikan elevasi 145 m, dan kemiringan 6,104 derajat. Berdasarkan persamaan (2.4) maka hasil perhitungan estimasi waktu tempuh pada jalur tersebut adalah sebagai berikut:

$$T_{flat} = \frac{D}{V} = \frac{1366}{66.67} = 40.97 \text{ menit}$$

### F. Validasi Data Jalur menggunakan Google Earth

Untuk memastikan akurasi data, dilakukan validasi data jalur dari Avenza Maps menggunakan Google Earth. Google Earth memungkinkan visualisasi peta *satelit* dan perbandingan jalur yang digambar dengan data geografis yang lebih luas, seperti kontur tanah. Langkah awalnya adalah mengeksport data jalur dari Avenza Maps ke format KML, lalu mengimpornya ke Google Earth. Setelah diimpor, dapat diverifikasi apakah jalur sesuai topografi, memeriksa kelurusan jalur, dan memastikan titik penting (*Basecamp*, pos pendakian, puncak) berada di lokasi yang benar. Proses ini sangat berguna untuk mengidentifikasi potensi kesalahan yang dapat memengaruhi analisis dan rekomendasi rute. Gambar 3.8 menampilkan proses validasi jalur menggunakan Google Earth.





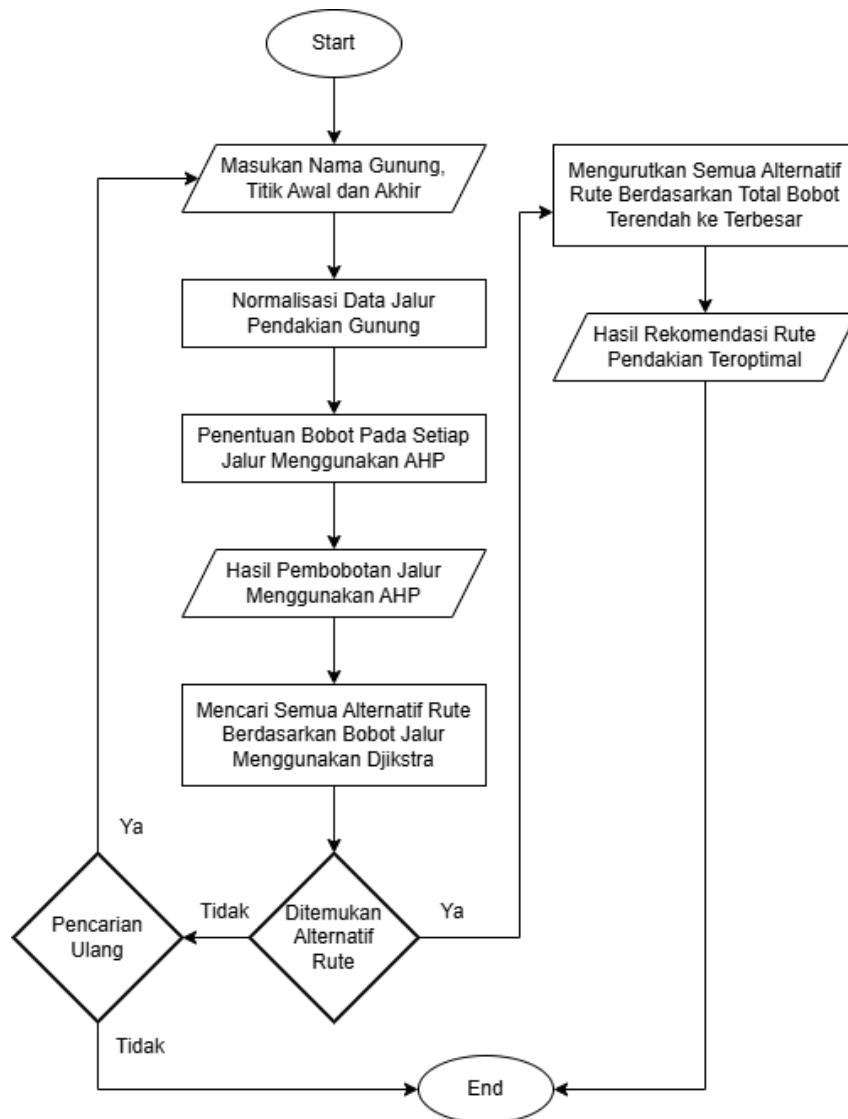
Gambar 3.8 Proses Validasi Data menggunakan *Google Earth*

### 3.3.2 Perancangan Sistem

Dalam membuat sebuah sistem, perancangan yang matang sangat diperlukan untuk memandu setiap langkah pengembangan. Berikut adalah perancangan sistem yang akan dibuat:

#### A. *Flowchart*

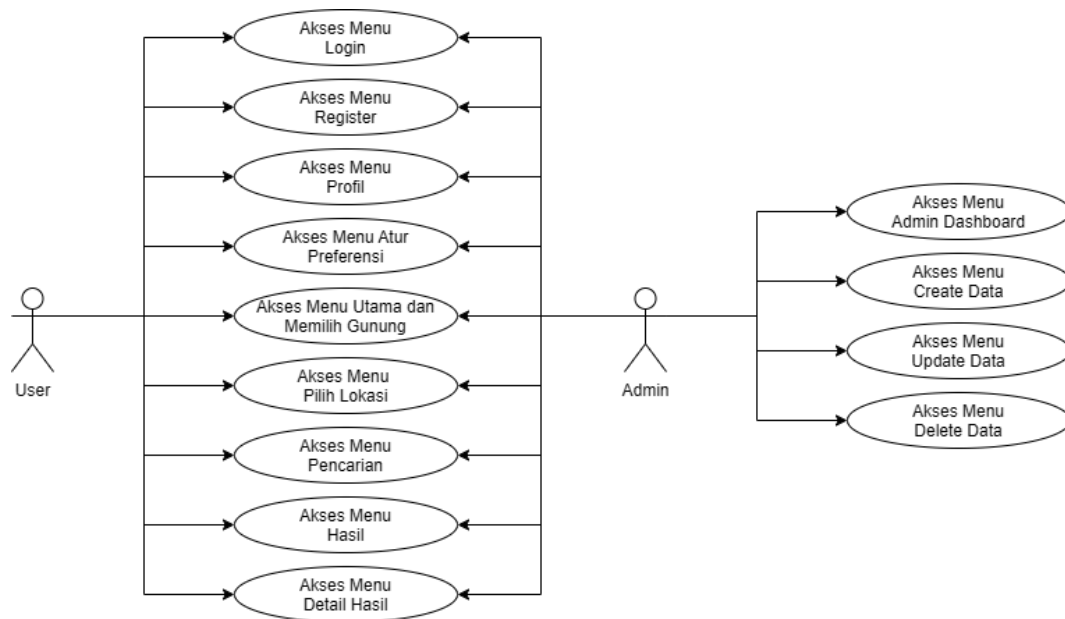
Dalam sistem ini, pengguna memulai dengan memasukkan nama gunung, titik awal, dan titik akhir pendakian yang diinginkan. Setelah input diterima, sistem akan secara otomatis memproses data tersebut. Proses inti melibatkan penggunaan metode *Analytic Hierarchy Process (AHP)* untuk menentukan bobot preferensi kriteria, yang kemudian digunakan oleh algoritma Dijkstra untuk mengidentifikasi alternatif rute terbaik. *Flowchart* dari proses ini dapat dilihat pada Gambar 3.9, yang secara visual menjelaskan alur kerja sistem dari input hingga output rekomendasi rute.



Gambar 3.9 *Flow Chart* Sistem Rekomendasi Rute Pendakian Gunung

## B. Use Case Diagram

*Use Case Diagram* berguna untuk menggambarkan berbagai proses yang berjalan dalam sebuah sistem serta aktor-aktor yang terlibat di dalamnya. Berikut ini adalah gambaran umum *Use Case Diagram* untuk sistem rekomendasi rute pendakian gunung yang akan dibuat, yang dijelaskan dalam Gambar 3.10.



Gambar 3.10 Use Case Diagram Sistem Rekomendasi Rute Pendakian Gunung

### 3.3.3 Pemodelan Struktur Graph Database

Pada tahap ini, data yang telah diekstraksi dan disimpan dalam format *CSV* berupa informasi lokasi dan jalur pendakian gunung akan dimasukkan ke dalam *Graph Database* untuk pemodelan struktur rute pendakian. Pemodelan ini dilakukan dengan beberapa langkah, yaitu:

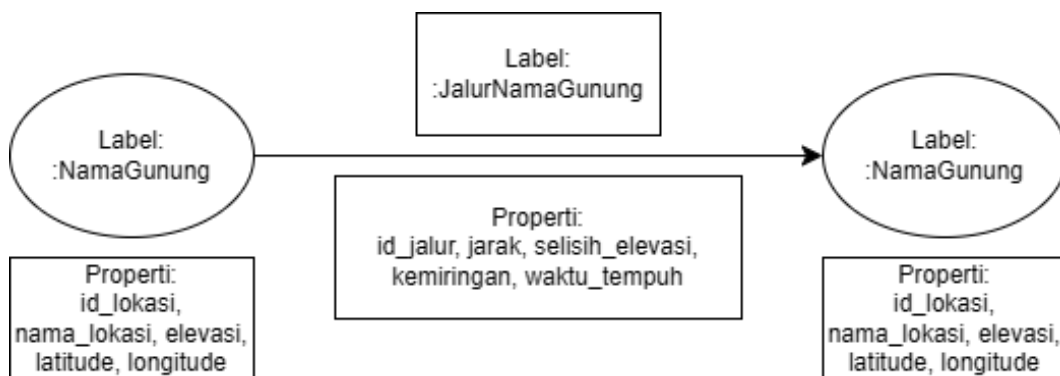
1. Node: Setiap *Basecamp*, pos pendakian, camp ground, dan persimpangan dibuat menjadi sebuah node dalam *Graph Database*. Node ini mewakili lokasi-lokasi penting di sepanjang jalur pendakian.
2. Edge: Setiap jalur pendakian antara dua titik lokasi direpresentasikan sebagai relasi atau yang disebut juga dengan *edge*. *Edge* ini menghubungkan dua *node* dan membawa informasi penting seperti jarak, selisih elevasi, kemiringan, dan estimasi waktu tempuh.

Skema yang digunakan dalam *Graph Database* ini disajikan dalam tabel

3.1 dan direpresentasikan pada gambar 3.10 di bawah ini:

Tabel 3.1 Skema Graf

No	Node/Edge	Label	Properti	Keterangan
1	Node	:NamaGunung	id_lokasi, nama_lokasi, elevasi, latitude, longitude	Untuk membuat titik berupa <i>Basecamp</i> , pos pendakian, camp ground dan persimpangan
2	Edge	:JalurNamaGunung	id_jalur, jarak, selisih_elevasi, kemiringan, waktu_tempuh	Untuk membuat relasi antar node dalam satu gunung



Gambar 3.11 Skema Graf

Untuk merepresentasikan setiap lokasi dan jalur pendakian dalam *Graph Database*, data diimpor menggunakan *Cypher Query* ke dalam database *Neo4j*. Proses ini melibatkan pembuatan *node* untuk setiap lokasi dan *edge* untuk setiap jalur pendakian yang menghubungkan *node-node* tersebut. Berikut ini adalah penjelasan lebih detail mengenai proses input data ke dalam database *Neo4j*.

Berikut adalah *Cypher Query* pada Tabel 3.2 digunakan untuk menginput *node* yang merepresentasikan lokasi-lokasi di Gunung Arjuno-Welirang beserta propertinya, menggunakan data dari file lokasi\_arjunowelirang.csv:

Tabel 3.2 Cypher Query Node

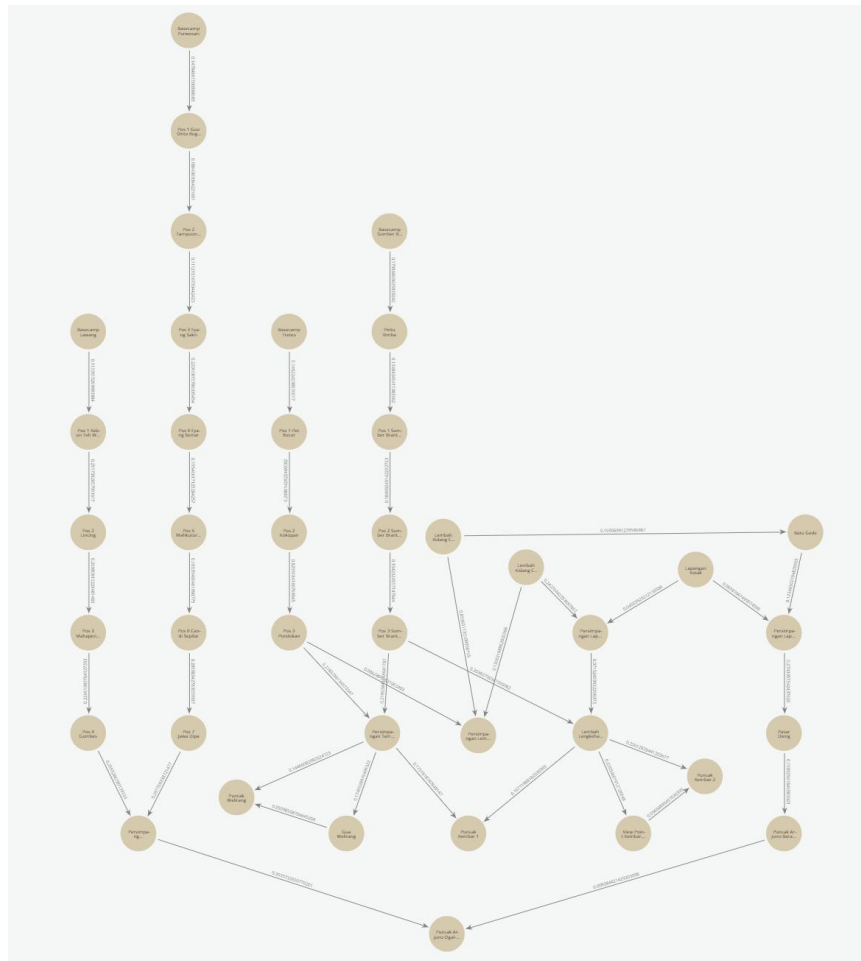
Cypher Query Node
<pre>LOAD CSV WITH HEADERS FROM 'file:///lokasi_arjunowelirang.csv' AS row MERGE (n:ArjunoWelirang {id_lokasi: row.ID_Lokasi}) SET n.nama_lokasi = row&gt;Nama_Lokasi,     n.elevasi = toInteger(row.Elevasi),     n.latitude = toFloat(row.Longitude),     n.longitude = toFloat(row.Longitude)</pre>

Pada query ini, setiap lokasi seperti *Basecamp* dan pos pendakian dari Gunung Arjuno-Welirang dibuat sebagai *node* yang memiliki properti seperti ID\_lokasi, nama, elevasi, koordinat latitude, dan longitude. Pada tabel 3.3 adalah *Cypher Query* untuk mengimpor *edge* yang menggambarkan jalur-jalur yang ada di Gunung Arjuno-Welirang beserta propertinya, berdasarkan data dari file jalur\_arjunowelirang.csv:

Tabel 3.3 Cypher Query Edge

Cypher Query Edge
<pre>LOAD CSV WITH HEADERS FROM 'file:///jalur_arjunowelirang.csv' AS row MATCH (start:ArjunoWelirang {nama_lokasi: row.Titik_Awal}) MATCH (end:ArjunoWelirang {nama_lokasi: row.Titik_Akhir}) MERGE (start)-[e:JalurArjunoWelirang {   id_jalur: toInteger(row.ID_Jalur),   selisih_elevasi: toInteger(row.Selisih_Elevasi),   jarak: toInteger(row.Jarak),   kemiringan: toFloat(row.Kemiringan),   waktu_tempuh: toFloat(row.Waktu_Tempuh) }]-&gt;(end)</pre>

Query ini menciptakan relasi antar node yang menggambarkan jalur pendakian, dengan properti seperti ID\_jalur, selisih elevasi, jarak antar lokasi, kemiringan jalur, dan waktu tempuh. Setiap baris data dari file CSV 'jalur\_arjunowelirang.csv' digunakan untuk mengidentifikasi titik awal dan akhir jalur pendakian. Dengan demikian, query ini secara efektif memodelkan konektivitas dan karakteristik fisik setiap segmen jalur pendakian Gunung Arjuno-Welirang dalam basis data.



Gambar 3.12 Visualisai Graf

Hasil dari proses input *node* dan *edge* ke dalam Graph Database dapat divisualisasikan dalam bentuk graf di *Neo4j*, di mana setiap lokasi pendakian diwakili oleh *node*, dan jalur yang menghubungkan lokasi-lokasi tersebut diwakili oleh *edge*. Visualisasi pada Gambar 3.12 membantu dalam memahami struktur jalur pendakian secara lebih komprehensif, dan memungkinkan sistem untuk melakukan kueri jalur terbaik menggunakan algoritma seperti *Dijkstra*.

### 3.3.4 Normalisasi Data Jalur Pendakian Gunung

Normalisasi data adalah langkah penting dalam analisis yang melibatkan perbandingan antara variabel yang memiliki satuan atau skala yang berbeda. Ketika kita bekerja dengan data yang mencakup berbagai jenis metrik, seperti panjang (dalam meter), waktu (dalam menit), atau kemiringan (dalam derajat), perbedaan skala ini dapat menyebabkan ketidakseimbangan dalam analisis. Proses normalisasi bertujuan untuk mengubah nilai-nilai tersebut menjadi skala yang sebanding atau proporsional, biasanya dalam rentang 0 hingga 1. Dengan cara ini, setiap variabel memiliki kontribusi yang setara dalam analisis, tanpa dipengaruhi oleh perbedaan skala atau satuan. Berdasarkan persamaan (3.1) untuk melakukan perhitungan normalisasi min-max:

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (3.1)$$

Dimana:

$X_{norm}$  adalah nilai yang sudah dinormalisasi.

$X$  adalah nilai asli (misalnya, Jarak, Selisih Elevasi, dll.).

$X_{max}$  adalah nilai maksimum dari data.

$X_{min}$  adalah nilai minimum dari data.

Misalnya kita memiliki tiga data jarak untuk tiga jalur sebagai berikut, Jalur 1: 1366 m, Jalur terpanjang: 3817 m dan Jalur terpendek: 179 m. Untuk normalisasi menggunakan normalisasi min-max, maka  $X_{max} = 3817$  dan  $X_{min} = 179$ . Maka berdasarkan persamaan (3.1) untuk jalur 1 didapatkan hasil:

$$X_{norm} = \frac{1366 - 179}{3817 - 179} = \frac{1187}{3638} = 0.326$$

### 3.3.5 Pembobotan *WDAG* Pada Jalur Menggunakan *AHP*

Pada proses pembobotan *AHP* (*Analytic Hierarchy Process*), dibutuhkan penentuan bobot kriteria yang berfungsi untuk menginisialisasi kriteria mana yang lebih prioritas dibandingkan kriteria lainnya. Untuk mendapatkan bobot kriteria dalam metode *AHP*, perlu dilakukan perbandingan berpasangan pada setiap kriteria. Dalam melakukan perbandingan, dibuat matriks perbandingan berpasangan yang mana nilai dari matriks perbandingan ini didapatkan dari hasil penilaian yang dilakukan kepada pegiat pendakian gunung, yang diminta untuk memberikan nilai perbandingan terhadap atribut-atribut jalur. Hasil penilaian perbandingan atribut ini menjadi dasar kriteria rute pendakian yang diharapkan oleh kalangan pendaki secara umum dan digunakan untuk proses pengujian sistem. Hasil penilaian perbandingan atribut oleh pegiat pendakian disertakan di dalam lampiran. Berikut adalah Tabel 3.4 yang menunjukkan matriks perbandingan berpasangan pada setiap kriteria yang didapat dari hasil survei pegiat pendakian gunung.

Tabel 3.4 Matriks Perbandingan Berpasangan

	Jarak	Selisih Elevasi	Kemiringan	Waktu Tempuh
Jarak	1	2.15938	0.26613	0.31925
Selisih Elevasi	0.46310	1	2.52883	2.30882
Kemiringan	3.75761	0.39544	1	2.26049
Waktu Tempuh	3.13238	0.43312	0.44238	1
Eigen Vector	8.35309	3.98794	4.23734	5.88855

Dari hasil matriks perbandingan berpasangan di atas, selanjutnya dilakukan normalisasi untuk mendapatkan bobot kriteria. Caranya adalah dengan membagi nilai perbandingan setiap elemen dalam matriks dengan nilai eigen vector yang sesuai. Kemudian, pada setiap baris kriteria, diambil rata-ratanya untuk



mendapatkan bobot kriteria. Berikut adalah Tabel 3.5 yang menunjukkan matriks normalisasi.

Tabel 3.5 Matriks Normalisasi

	Jarak	Selisih Elevasi	Kemiringan	Waktu Tempuh	Bobot Kriteria
Jarak	0.11972	0.54148	0.06281	0.05421	0.19455
Selisih Elevasi	0.05544	0.25076	0.59680	0.39209	0.32377
Kemiringan	0.44985	0.09916	0.23600	0.38388	0.29222
Waktu Tempuh	0.37500	0.10861	0.10440	0.16982	0.18946

Setelah mendapatkan nilai bobot kriteria, selanjutnya dilakukan perhitungan bobot untuk setiap jalur pada rute pendakian gunung. Perhitungan ini dinyatakan dalam persamaan (3.2) berikut ini:

$$W = (W_j \times X_j) + (W_{se} \times X_{se}) + (W_k \times X_k) + (W_{wt} \times X_{wt}) \quad (3.2)$$

Dimana:

$W$  adalah bobot jalur.

$W_j$  adalah bobot kriteria jarak.

$X_j$  adalah nilai jarak yang sudah di normalisasi.

$W_{se}$  adalah bobot kriteria selisih elevasi.

$X_{se}$  adalah nilai selisih elevasi yang sudah di normalisasi.

$W_k$  adalah bobot kriteria kemiringan.

$X_k$  adalah nilai kemiringan yang sudah di normalisasi.

$W_{wt}$  adalah bobot kriteria waktu tempuh.

$X_{wt}$  adalah nilai waktu tempuh yang sudah di normalisasi.

Sebagai contoh, pada jalur antara *Basecamp* Lawang dan Pos 1 Kebun Teh Wonosari dengan data Jarak: 1366 m, Selisih Elevasi: 145 m, Kemiringan: 6.104 derajat dan Waktu Tempuh: 34.99 menit. Setelah dinormalisasi, didapatkan nilai sebagai berikut Jarak: 0.326, Selisih Elevasi: 0.586, Kemiringan 0.593 dan Waktu Tempuh: 0.233. Berdasarkan persamaan (3.2) maka perhitungan bobot pada jalur tersebut adalah sebagai berikut:

$$\begin{aligned} W &= (0.195 \times 0.326) + (0.324 \times 0.586) + (0.292 \times 0.593) + (0.189 \times 0.233) \\ &= 0.0636 + 0.1899 + 0.1732 + 0.044 = 0.4707 \end{aligned}$$

Dari perhitungan di atas, maka bobot pada jalur antara *Basecamp* Lawang ke Pos 1 Kebun Teh Wonosari adalah 0.4707. Pembobotan ini dilakukan pada semua jalur yang ada untuk nantinya dijadikan nilai dasar dalam pencarian menggunakan algoritma Dijkstra. Proses normalisasi ini krusial untuk menyamakan skala berbagai properti jalur yang berbeda, memastikan bahwa tidak ada satu properti pun yang mendominasi perhitungan bobot secara tidak proporsional. Dengan bobot yang telah dinormalisasi ini, algoritma *Dijkstra* dapat menghitung jalur terbaik secara akurat, mempertimbangkan berbagai faktor penting dalam pendakian. Berikut pada tabel 3.6 adalah *Cypher Query* untuk melakukan penghitungan normalisasi data dan pembobotan pada jalur.

Tabel 3.6 *Cypher Query* Normalisasi Data dan Pembobotan Pada Jalur

<b><i>Cypher Query</i> Normalisasi Data dan Pembobotan Pada Jalur</b>
<pre>//Menentukan Nilai Minimum dan Maksimum untuk Normalisasi MATCH (a)-[r:JalurArjuno]-&gt;(b) WITH MIN(r.jarak) AS min_jarak, MAX(r.jarak) AS max_jarak, MIN(ABS(r.selisih_elevasi)) AS min_elevasi, MAX(ABS(r.selisih_elevasi)) AS max_elevasi, MIN(ABS(r.kemiringan)) AS min_kemiringan, MAX(ABS(r.kemiringan)) AS max_kemiringan , MIN(r.waktu_tempuh) AS min_waktu_tempuh, MAX(r.waktu_tempuh) AS max_waktu_tempu h //Menormalisasi Data MATCH (a)-[r:JalurArjuno]-&gt;(b) WITH r, min_jarak, max_jarak, min_elevasi, max_elevasi, min_kemiringan, max_kemiringan, min_waktu_tempuh, max_waktu_tempuh WITH r, (r.jarak - min_jarak) / (max_jarak - min_jarak) AS norm_jarak, (ABS(r.selisih_elevasi) - min_elevasi) / (max_elevasi - min_elevasi) AS norm_elevasi, (ABS(r.kemiringan) - min_kemiringan) / (max_kemiringan - min_kemiringan) AS norm_kemiri ngan, (r.waktu_tempuh - min_waktu_tempuh) / (max_waktu_tempuh - min_waktu_tempuh) AS norm _waktu_tempuh //Pembobotan Data WITH r, norm_jarak * 0.194553201 + norm_elevasi * 0.323769675 + norm_kemiringan * 0.2922204 46 + norm_waktu_tempuh * 0.189456678 AS bobot SET r.bobot = bobot RETURN r, r.bobot;</pre>

### 3.3.6 Pencarian Rekomendasi Rute Menggunakan *Dijkstra*

Setelah proses pembobotan setiap jalur dalam Graph Database menggunakan **Analytic Hierarchy Process (AHP)**, algoritma **Dijkstra** diterapkan untuk menemukan beberapa rekomendasi rute terbaik. Dijkstra bekerja dengan mencari jalur terbaik berdasarkan bobot yang telah dihitung, mencerminkan preferensi pengguna. Setiap jalur yang ditemukan akan disimpan bersama total bobotnya, kemudian diurutkan berdasarkan bobot terendah sebagai rute terbaik.

Algoritma *Dijkstra* adalah algoritma yang digunakan untuk menemukan jalur terpendek dari satu *node* (titik) ke node lainnya dalam sebuah *graph* berbobot. Dalam konteks sistem rekomendasi jalur pendakian ini, setiap *node* dalam *graph* mewakili titik penting dalam jalur pendakian, seperti *Basecamp*, pos pendakian, atau puncak, dan setiap *edge* (jalur) antara dua *node* diberi bobot berdasarkan hasil pembobotan AHP yang telah mempertimbangkan kriteria-kriteria seperti jarak, selisih elevasi, kemiringan, dan waktu tempuh (Irawan dkk., 2023).

Cara kerja algoritma Dijkstra adalah sebagai berikut:

- a. *Inisialisasi* Bobot Awal: Setiap node dalam *graph* diberikan bobot awal yang sangat besar (misalnya *infinity*), kecuali node awal yang diberi bobot 0, karena jalur dari node tersebut ke dirinya sendiri tidak memerlukan bobot.
- b. *Evaluasi Node* Tetangga: Algoritma akan memproses setiap node yang sedang dievaluasi dan memeriksa semua node tetangganya. Jika ditemukan jalur yang lebih pendek menuju node tetangga (berdasarkan bobot jalur), maka bobot node tetangga tersebut akan diperbarui dengan bobot yang lebih rendah.

- c. Proses *Iteratif*: Algoritma Dijkstra terus berlanjut secara iteratif, mengevaluasi setiap node yang belum diproses, hingga semua node telah dievaluasi. Dengan demikian, bobot terkecil menuju semua node dari titik awal dapat ditemukan.
- d. Mencapai Titik Tujuan: Setelah algoritma mencapai titik tujuan, bobot akhir yang didapatkan mewakili total bobot jalur terpendek yang melewati node-node tersebut. Jalur ini merupakan jalur dengan bobot paling optimal berdasarkan kriteria yang telah ditetapkan pengguna.

Implementasi algoritma dijkstra ini di dalam graph database menggunakan

*Cypher Query* yang ada pada tabel 3.7.

Tabel 3.7 *Cypher Query* algoritma Dijkstra

<i>Cypher Query</i> algoritma Dijkstra
<pre>//Menentukan Gunung, Titik Awal dan Titik Tujuan MATCH (start:Arjuno {nama_lokasi: 'Basecamp Lawang'}), (end:Arjuno {nama_lokasi: 'Puncak Welirang'}) CALL apoc.algo.dijkstra(start, end, 'JalurArjuno', 'bobot', 0, 100) YIELD path, weight //Menghitung Total Jarak, Selisih Elevasi, Waktu Tempuh, Kemiringan, dan Jumlah Jalur WITH path, weight,   REDUCE(total_jarak = 0, r IN relationships(path)   total_jarak + r.jarak) AS total_jarak,   REDUCE(total_selisih_elevasi = 0, r IN relationships(path)   total_selisih_elevasi + r.selisih_elevasi) AS total_selisih_elevasi,   REDUCE(total_waktu = 0, r IN relationships(path)   total_waktu + r.waktu_tempuh) AS total_waktu,   REDUCE(total_kemiringan = 0, r IN relationships(path)   total_kemiringan + r.kemiringan) AS total_kemiringan,   SIZE(relationships(path)) AS total_jalur //Menghitung Kemiringan Rata-Rata dan Menampilkan Hasil WITH path, weight, total_jarak, total_selisih_elevasi, total_waktu, total_kemiringan, total_jalur, nodes(path) AS nodes_in_path RETURN [n IN nodes_in_path   n.nama_lokasi] AS node_names, AS total_bobot, total_jarak, total_selisih_elevasi, total_waktu, total_kemiringan / total_jalur AS avg_kemiringan;</pre>

### 3.4 Desain Eksperimen

Bagian ini menjelaskan proses pengujian dan evaluasi penerapan metode *Weighted Directed Acyclic Graph (WDAG)* yang dikombinasikan dengan *Analytic Hierarchy Process (AHP)* dan algoritma *Dijkstra* untuk memberikan rekomendasi

rute pendakian gunung. Tujuan utama dari pengujian ini adalah menilai kualitas rekomendasi yang dihasilkan oleh sistem, dengan membandingkan rute-rute yang dihasilkan oleh sistem dengan rute yang dipilih dan dianggap relevan oleh pengguna. Metrik evaluasi yang digunakan dalam analisis ini meliputi *F1-Score*, yang memberikan gambaran mengenai akurasi dari sistem rekomendasi.

Selain itu, pengujian juga meneliti pengaruh kompleksitas graf terhadap kinerja sistem, untuk memahami apakah jumlah node, dan edge yang dianalisis dalam graf memengaruhi akurasi dan efisiensi sistem dalam menghasilkan rekomendasi. Dengan demikian, evaluasi ini tidak hanya menilai seberapa akurat rekomendasi sistem, tetapi juga menyoroti aspek efisiensi dalam menjalankan algoritma.

#### **3.4.1 Skenario Uji Coba**

Data yang digunakan dalam eksperimen ini berasal dari beberapa graf jalur pendakian gunung di Indonesia, yaitu Gunung Arjuno-Welirang, Bromo, Butak, Ciremai, Penanggungan, Slamet, Sumbing dan Ungaran. Pada skenario uji coba ini, sistem akan menghasilkan rekomendasi rute dari setiap *Basecamp* pendakian menuju setiap puncak yang ada pada beberapa graf jalur pendakian seperti yang terdapat pada tabel 3.8 sampai dengan 3.15.

Setiap input akan menghasilkan rekomendasi rute terbaik yang kemudian dicatat berapa waktu pemrosesan untuk menemukan rute terbaik dan pakar pendakian diminta untuk memberikan penilaian terhadap rute yang direkomendasikan untuk kemudian diukur tingkat akurasi setiap rekomendasi rute pendakian. Rata-rata akurasi dan waktu pemrosesan dihitung untuk setiap graf, lalu

dibandingkan dengan kompleksitas graf tersebut untuk mengetahui bagaimana pengaruh kompleksitas terhadap akurasi dan waktu pemrosesan rekomendasi yang dihasilkan.

Tabel 3.8 Daftar input untuk Uji Coba di Gunung Arjuno

No	Lokasi Awal	Lokasi Akhir
1	<i>Basecamp Lawang</i>	Puncak Arjuno Barat
		Puncak Arjuno Ogal-Agil
		Puncak Kembar 1
		Puncak Kembar 2
		Puncak Welirang
2	<i>Basecamp Purwosari</i>	Puncak Arjuno Barat
		Puncak Arjuno Ogal-Agil
		Puncak Kembar 1
		Puncak Kembar 2
		Puncak Welirang
3	<i>Basecamp Sumber Brantas</i>	Puncak Arjuno Barat
		Puncak Arjuno Ogal-Agil
		Puncak Kembar 1
		Puncak Kembar 2
		Puncak Welirang
4	<i>Basecamp Tretes</i>	Puncak Arjuno Barat
		Puncak Arjuno Ogal-Agil
		Puncak Kembar 1
		Puncak Kembar 2
		Puncak Welirang

Tabel 3.9 Daftar input untuk Uji Coba di Gunung Bromo

No	Lokasi Awal	Lokasi Akhir
1	Cemoro Lawang	Puncak Batok
		Puncak Bromo
		Puncak Crater
		Puncak Penanjakan
		Puncak Widodaren
2	Seruni View Point	Puncak Batok
		Puncak Bromo
		Puncak Crater
		Puncak Penanjakan
		Puncak Widodaren

Tabel 3.10 Daftar input untuk Uji Coba di Gunung Butak

No	Lokasi Awal	Lokasi Akhir
1	<i>Basecamp</i> Kraton Gunung Kawi	Puncak Batu Tulis
		Puncak Butak
		Puncak Panderman
2	<i>Basecamp</i> Sirah Kencong	Puncak Batu Tulis
		Puncak Butak
		Puncak Panderman
3	<i>Basecamp</i> Toyomerto	Puncak Batu Tulis
		Puncak Butak
		Puncak Panderman

Tabel 3.11 Daftar input untuk Uji Coba di Gunung Ciremai

No	Lokasi Awal	Lokasi Akhir
1	<i>Basecamp</i> Apuy	Puncak Ciremai
		Puncak Crater
		Puncak Panglongokan
2	<i>Basecamp</i> Linggarjati	Puncak Ciremai
		Puncak Crater
		Puncak Panglongokan
3	<i>Basecamp</i> Palutungan	Puncak Ciremai
		Puncak Crater
		Puncak Panglongokan

Tabel 3.12 Daftar input untuk Uji Coba di Gunung Penanggungan

No	Lokasi Awal	Lokasi Akhir
1	<i>Basecamp</i> Genting	Puncak Bekel
		Puncak Crater
		Puncak Penanggungan
2	<i>Basecamp</i> Jolotundo	Puncak Bekel
		Puncak Crater
		Puncak Penanggungan
3	<i>Basecamp</i> Tamiajeng	Puncak Bekel
		Puncak Crater
		Puncak Penanggungan

Tabel 3.13 Daftar input untuk Uji Coba di Gunung Slamet

No	Lokasi Awal	Lokasi Akhir
1	<i>Basecamp</i> Bambang	Puncak Barat
		Puncak Slamet
		Puncak Surono
2	<i>Basecamp</i> Guci Kompak	Puncak Barat
		Puncak Slamet
		Puncak Surono
3	<i>Basecamp</i> Guci Permadi	Puncak Barat
		Puncak Slamet
		Puncak Surono
4	<i>Basecamp</i> Kaliwadas	Puncak Barat
		Puncak Slamet
		Puncak Surono
5	<i>Basecamp</i> Penakir	Puncak Barat
		Puncak Slamet
		Puncak Surono
6	<i>Basecamp</i> Radenpala-Baturaden	Puncak Barat
		Puncak Slamet
		Puncak Surono
7	<i>Basecamp</i> Sawangan	Puncak Barat
		Puncak Slamet
		Puncak Surono

Tabel 3.14 Daftar input untuk Uji Coba di Gunung Ungaran

No	Lokasi Awal	Lokasi Akhir
1	<i>Basecamp</i> Mawar	Puncak Botak
		Puncak Ungaran
2	<i>Basecamp</i> Medini	Puncak Botak
		Puncak Ungaran



Tabel 3.15 Daftar input untuk Uji Coba di Gunung Sumbing

No	Lokasi Awal	Lokasi Akhir
1	<i>Basecamp Adipuro</i>	Puncak Buntu
		Puncak Kawah
		Puncak Rajawali
		Puncak Sejati
2	<i>Basecamp Banaran</i>	Puncak Buntu
		Puncak Kawah
		Puncak Rajawali
		Puncak Sejati
3	<i>Basecamp Bowongso</i>	Puncak Buntu
		Puncak Kawah
		Puncak Rajawali
		Puncak Sejati
4	<i>Basecamp Butuh</i>	Puncak Buntu
		Puncak Kawah
		Puncak Rajawali
		Puncak Sejati
5	<i>Basecamp Cepit</i>	Puncak Buntu
		Puncak Kawah
		Puncak Rajawali
		Puncak Sejati
6	<i>Basecamp Garung</i>	Puncak Buntu
		Puncak Kawah
		Puncak Rajawali
		Puncak Sejati
7	<i>Basecamp Mangli</i>	Puncak Buntu
		Puncak Kawah
		Puncak Rajawali
		Puncak Sejati
8	<i>Basecamp Sipetung</i>	Puncak Buntu
		Puncak Kawah
		Puncak Rajawali
		Puncak Sejati

### 3.4.2 Evaluasi Hasil

Evaluasi hasil dalam penelitian ini dilakukan menggunakan metrik *F1-Score* sebagai alat utama untuk menilai kualitas prediksi atau hasil rekomendasi yang diberikan oleh sistem. Metrik ini digunakan untuk memahami seberapa baik

sistem dalam merekomendasikan rute yang sesuai dengan preferensi pengguna, serta sejauh mana sistem dapat menangkap berbagai opsi rute yang relevan. Selain metrik akurasi *F1-Score* yang menilai kualitas rekomendasi dari sisi sistem, penelitian ini juga melakukan evaluasi dari sisi pengalaman pengguna (*user experience*) dengan mengukur tingkat *usability* (ketergunaan) dari sistem "Treknesia" yang telah dikembangkan. Untuk tujuan ini, digunakan metode *System Usability Scale (SUS)*.

*F1-Score* adalah ukuran performa yang menggabungkan akurasi rekomendasi dan cakupan dalam satu metrik seimbang. *F1-Score* sangat berguna dalam menyeimbangkan antara *precision* (ketepatan rekomendasi yang relevan) dan *recall* (cakupan rute yang relevan) dengan menggunakan rata-rata harmonis dari keduanya (Yacouby & Axman, 2020). *F1-Score* sangat bermanfaat ketika ada ketidakseimbangan antara rute yang relevan dan tidak relevan dalam data.

Rumus untuk menghitung F1-Score adalah dinyatakan dalam persamaan (3.3) berikut:

$$F1 - Score = 2x \frac{TP}{2 x TP + FP + FN} \quad (3.3)$$

Di mana:

*True Positives (TP)*: Rute yang relevan berdasarkan preferensi pengguna dan juga direkomendasikan oleh sistem. Ini menunjukkan sistem memberikan rekomendasi yang tepat.

*False Positives (FP)*: Rute yang direkomendasikan oleh sistem namun tidak relevan berdasarkan preferensi pengguna. Ini menunjukkan rekomendasi yang tidak akurat.

*False Negatives (FN)*: Rute yang relevan tetapi tidak direkomendasikan oleh sistem. Ini menunjukkan adanya rute relevan yang terlewatkan oleh sistem.

*F1-Score* dihitung dengan memperhitungkan keseimbangan antara *TP*, *FP*, dan *FN*, sehingga memberikan penilaian yang lebih holistik tentang performa sistem rekomendasi.

Setelah menghitung *F1-Score* untuk setiap rute yang direkomendasikan, langkah selanjutnya adalah menghitung rata-rata nilai dari setiap *F1-Score* untuk setiap graf jalur pendakian. Nilai rata-rata ini memberikan gambaran umum tentang performa sistem rekomendasi pada masing-masing jalur pendakian. Berikut persamaan (3.4) untuk mendapatkan rata-rata *F1-Score* pada keseluruhan hasil uji coba.

$$\text{Rata - rata } F1 - \text{Score} = \frac{\sum F1 - \text{Score Setiap Graf}}{\text{Jumlah Graf}} \quad (3.4)$$

Selain menggunakan metrik *F1-Score* sebagai indikator utama, evaluasi ini juga mempertimbangkan pengaruh kompleksitas graf terhadap akurasi dan performa sistem. Kompleksitas graf dihitung berdasarkan jumlah node dan edge yang ada di dalam setiap jalur pendakian, serta interaksi antara node dan edge yang dianalisis menggunakan *Cyclomatic Complexity*. Graf yang lebih kompleks mungkin membutuhkan waktu pemrosesan yang lebih lama, namun juga memberikan lebih banyak opsi rute yang beragam, yang berpotensi meningkatkan akurasi rekomendasi.

Graf yang lebih kompleks, dengan lebih banyak node dan edge, mungkin membutuhkan waktu pemrosesan yang lebih lama oleh algoritma *Dijkstra*, namun memiliki potensi untuk memberikan lebih banyak pilihan rute yang beragam dan, dengan demikian, meningkatkan akurasi rekomendasi. Sebaliknya, graf yang

sederhana, meskipun cepat diproses, mungkin memiliki pilihan rute yang terbatas, yang dapat berdampak pada kualitas rekomendasi.

Kompleksitas graf dalam evaluasi ini dihitung menggunakan *Cyclomatic Complexity*. *Cyclomatic Complexity* adalah metrik yang digunakan untuk mengukur kompleksitas struktural dari suatu program, terutama dalam konteks kontrol alur (flow control). Metrik ini diperkenalkan oleh Thomas McCabe pada tahun 1976 sebagai alat untuk mengukur seberapa sulit sebuah program untuk diuji, serta seberapa banyak cabang (branching) yang ada dalam program tersebut. *Cyclomatic Complexity* mengukur jumlah jalur independen dalam program, yang pada gilirannya dapat digunakan untuk memperkirakan jumlah pengujian yang diperlukan untuk memastikan bahwa seluruh kode telah diuji secara menyeluruh.

*Cyclomatic Complexity* menghitung jumlah jalur independen (independent paths) dalam sebuah grafik kontrol alur (control flow graph) yang mewakili program. Grafik kontrol alur ini menggambarkan bagaimana eksekusi program berjalan melalui struktur kontrol seperti pernyataan bersyarat, loop, dan percabangan.

Untuk menghitung *Cyclomatic Complexity* pada suatu program, kita menggunakan rumus dalam persamaan (3.5) berikut:

$$M = E - N + 2P \quad (3.5)$$

Dimana:

M = Cyclomatic Complexity (kompleksitas siklomatik)

E = Jumlah edge (garis penghubung) dalam graf kontrol alur

N = Jumlah node (titik) dalam graf kontrol alur

P = Jumlah komponen terhubung (biasanya program dianggap satu komponen, sehingga

$P=1$ )

Setelah menghitung kompleksitas setiap graf menggunakan rumus di atas, analisis dilakukan dengan membandingkan hasil rata-rata *F1-Score* pada graf-graf yang memiliki tingkat kompleksitas berbeda. Evaluasi ini dilakukan untuk memahami apakah peningkatan kompleksitas graf berdampak positif atau negatif terhadap kualitas dan efisiensi sistem rekomendasi.

Evaluasi akhir dari hasil percobaan ini bertujuan untuk mengukur apakah ada hubungan signifikan antara kompleksitas graf dan akurasi sistem. Misalnya, jika sistem bekerja pada graf yang lebih kompleks, apakah rekomendasi yang diberikan menjadi lebih akurat namun dengan waktu pemrosesan yang lebih lama? Sebaliknya, pada graf yang lebih sederhana, apakah rekomendasi lebih cepat dihasilkan tetapi dengan cakupan yang lebih terbatas?

Dengan mengukur hasil rata-rata *F1-Score* pada graf dengan tingkat kompleksitas berbeda, sistem dapat dievaluasi untuk memahami bagaimana kompleksitas graf mempengaruhi kualitas rekomendasi yang dihasilkan, serta apakah sistem bekerja secara optimal pada graf yang lebih kompleks.

*System Usability Scale* (SUS) adalah instrumen kuesioner terstandarisasi yang cepat dan andal untuk mengukur persepsi subjektif pengguna terhadap ketergunaan sebuah sistem. Penggunaan SUS dalam penelitian ini bertujuan untuk mendapatkan data kuantitatif mengenai seberapa mudah dan memuaskan sistem rekomendasi ini saat digunakan oleh pengguna akhir.

Langkah-langkah pelaksanaan evaluasi menggunakan SUS dalam penelitian ini adalah sebagai berikut:

- a. Instrumen dan Skala: Kuesioner SUS terdiri dari 10 pernyataan standar yang mencakup berbagai aspek ketergunaan, seperti kompleksitas, kemudahan penggunaan, dan tingkat kepercayaan diri pengguna. Setiap pernyataan dinilai oleh responden menggunakan Skala Likert 5 poin, mulai dari "Sangat Tidak Setuju" (nilai 1) hingga "Sangat Setuju" (nilai 5).
- b. Partisipan dan Skenario Tugas: Evaluasi ini melibatkan sejumlah responden yang memiliki latar belakang sebagai pegiat pendakian gunung, sejalan dengan target pengguna sistem. Sebelum mengisi kuesioner, setiap partisipan diminta untuk melakukan skenario tugas yang telah ditentukan: menggunakan sistem "Treknesia" untuk membuat preferensi bobot kriteria masing-masing, kemudian melakukan pengujian mencari beberapa rute pendakian terbaik dari *Basecamp* ke Puncak tertentu untuk memperhatikan hasil rekomendasi dan detail hasil rekomendasi. Tugas ini memastikan bahwa responden memiliki pengalaman langsung dalam menggunakan fungsionalitas utama sistem sebelum memberikan penilaian.
- c. Prosedur Perhitungan Skor: Skor SUS untuk setiap responden dihitung melalui prosedur yang telah ditetapkan:
  - Untuk pernyataan ganjil (1, 3, 5, 7, 9), skor dihitung dengan rumus: (Nilai Skala - 1).
  - Untuk pernyataan genap (2, 4, 6, 8, 10), skor dihitung dengan rumus: (5 - Nilai Skala).
  - Semua skor dari 10 pernyataan tersebut dijumlahkan untuk mendapatkan total skor (antara 0-40).

- Total skor tersebut kemudian dikalikan dengan 2.5 untuk mendapatkan skor akhir SUS (antara 0-100).
- d. Analisis dan Interpretasi Data: Skor SUS akhir dari seluruh responden akan dianalisis untuk mendapatkan gambaran keseluruhan mengenai ketergunaan sistem. Metrik statistik utama yang akan digunakan meliputi:
- Rata-rata Skor SUS: Nilai rata-rata dari seluruh responden untuk menunjukkan skor ketergunaan sistem secara umum.
  - Standar Deviasi: Untuk menunjukkan seberapa konsisten atau bervariasinya penilaian di antara responden.
  - Interpretasi Skor: Skor rata-rata akan diinterpretasikan menggunakan skala penerimaan, peringkat dan adjektif, yang mana skala-skala SUS ini dijelaskan dalam Tabel 3.16 sampai dengan Tabel 3.18 berikut ini.

Tabel 3.16 Skala Penerimaan

Skor SUS	Skala Penerimaan
> 61	Dapat Diterima
< 61	Tidak Dapat Diterima

Tabel 3.17 Skala Peringkat

Skor SUS	Skala Peringkat
> 90	A
80 – 90	B
70 – 80	C
60 – 70	D
< 60	F

Tabel 3.18 Skala Adjektif

Skor SUS	Skala Adjektif
> 85	Sangat Bagus Sekali
72 – 85	Bagus Sekali
51 - 72	Bagus
39 – 40	Cukup
25 – 39	Buruk
< 25	Sangat Buruk Sekali

Dengan menggabungkan metrik *F1-Score* dan hasil dari SUS, evaluasi sistem tidak hanya mencakup akurasi rekomendasi algoritma, tetapi juga kualitas interaksi dan kepuasan dari sudut pandang pengguna.



## BAB IV

### HASIL DAN PEMBAHASAN

#### 4.1 Implementasi Sistem

Implementasi Sistem membahas penerapan pembahasan yang sudah dijabarkan sebelumnya pada Bab 3 di dalam pembuatan sistem. Dalam hal ini, sistem dibuat dalam bentuk *Web*. Berikut adalah kebutuhan *hardware* dan *software* dalam implementasi program.

##### A. Kebutuhan *Hardware*:

- a. Prosesor: Intel Core i5 (generasi ke-8 atau lebih baru) atau setara.
- b. RAM: 8 GB (minimum 16GB lebih baik).
- c. Penyimpanan: SSD minimum 128 GB.
- d. Kartu Grafis: Tidak diperlukan GPU khusus.
- e. Koneksi Jaringan: Koneksi stabil, minimum 10Mbps.

##### B. Kebutuhan *Software*:

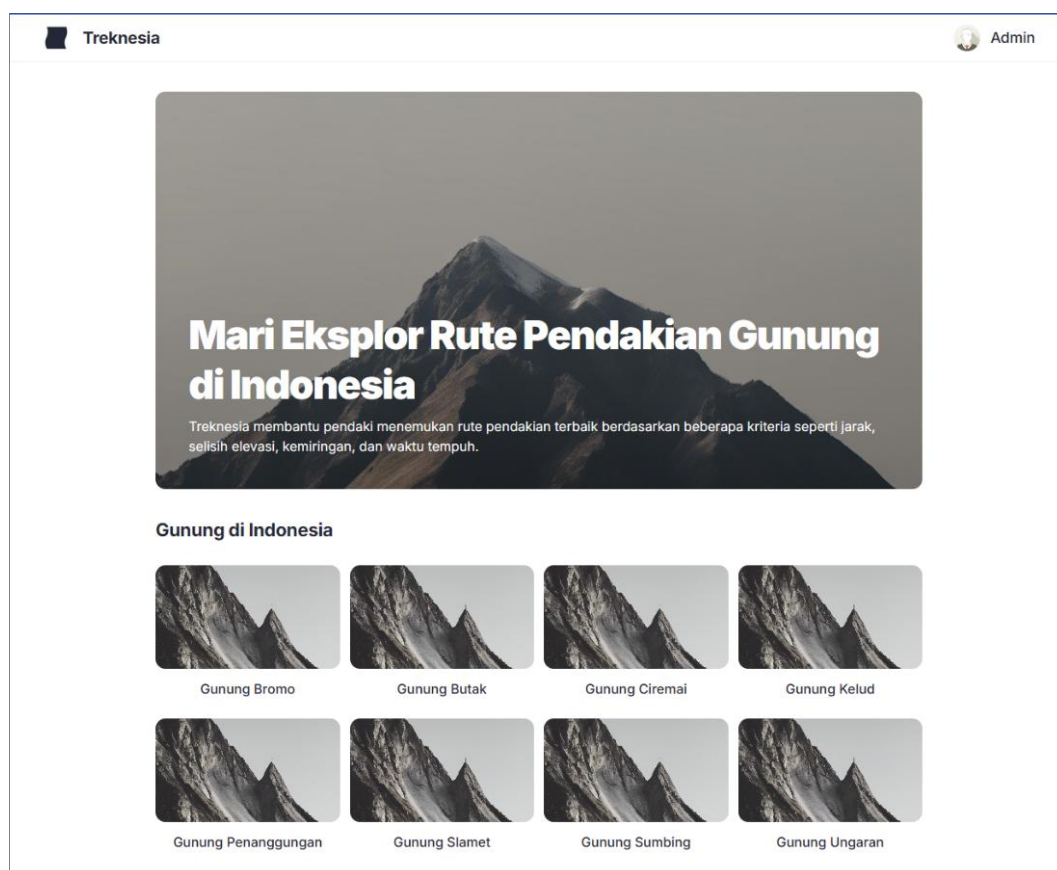
- a. System Operasi: Windows 11.
- b. Database Graf: *Neo4J 5.6.2*.
- c. Bahasa Pemrograman: *Python*.
- d. *Framework Web*: *Flask*.
- e. Antarmuka Pengguna: *HTML, CSS, JavaScript*.
- f. Integrasi *Python* dan *Neo4J*: *Neo4J Python Driver*.
- g. *Web Browser*: *Google Chrome*.
- h. *IDE/Code Editor*: *VS Code*.

i. Manajemen Database: *Neo4J Desktop 1.6.1*.

## 4.2 Implementasi Tampilan Sistem

Implementasi tampilan menjelaskan tahap penerapan antarmuka pengguna yang dirancang untuk mendukung fungsionalitas sistem rekomendasi rute pendakian gunung. Tampilan ini dirancang menggunakan *Tailwind CSS* untuk memberikan pengalaman yang interaktif. Berikut adalah tampilan dari sistem rekomendasi, mulai dari Halaman Utama hingga Detail Hasil Rekomendasi.

### 4.2.1 Tampilan Halaman Utama

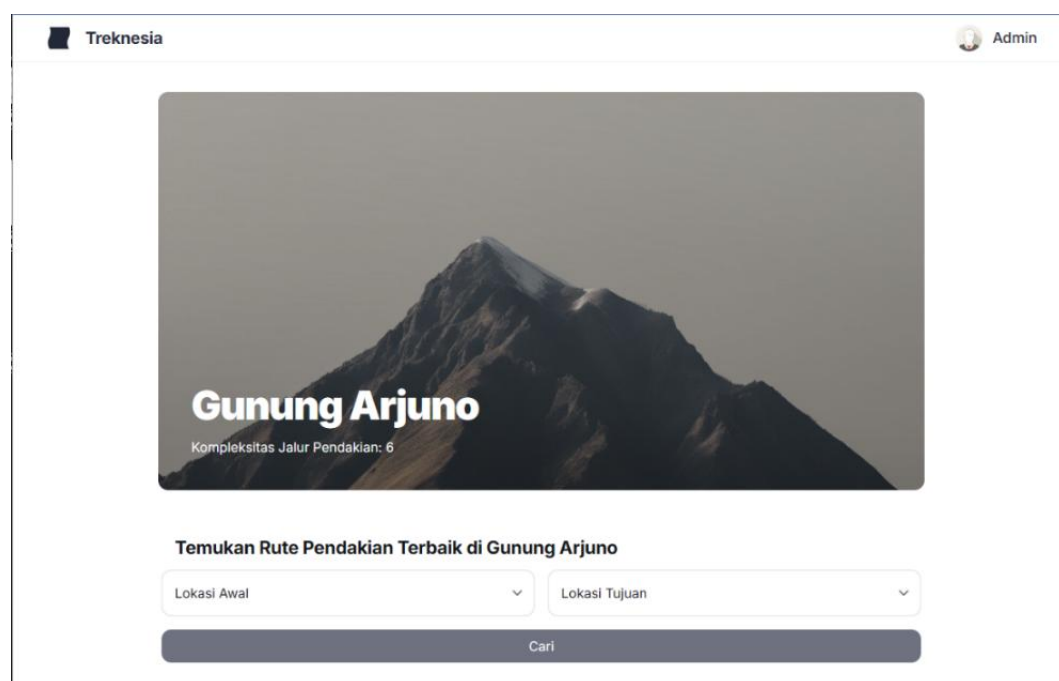


Gambar 4.1 Tampilan Halaman Utama

Pada Gambar 4.1, tampilan halaman utama menunjukkan antarmuka pengguna dari sistem rekomendasi rute pendakian yang diberi nama “Treknesia”

dibagian kiri sekaligus sebagai tombol home dan dibagian kanan menampilkan profil dan username yang sedang login. Di bagian atas halaman, terdapat judul besar yang bertuliskan “Mari Eksplor Rute Pendakian Gunung di Indonesia”, yang menggambarkan tujuan utama situs ini, yaitu membantu pendaki menemukan rute pendakian terbaik di Indonesia. Di bawah judul, ada deskripsi singkat yang menjelaskan fungsi situs ini serta fitur utama dari sistem rekomendasi. Lebih bawah, terdapat beberapa gambar gunung yang mewakili destinasi pendakian di Indonesia, seperti Gunung Arjuno, Gunung Bromo, Gunung Butak, Gunung Ciremai, Gunung Penanggungan, Gunung Slamet, Gunung Sumbing dan Gunung Ungaran. Setiap gambar dilengkapi dengan nama gunung, memberikan pengunjung pilihan untuk memilih gunung yang ingin dijelajahi lebih lanjut dalam sistem rekomendasi rute pendakian.

#### 4.2.2 Tampilan Form Lokasi Awal dan Lokasi Tujuan

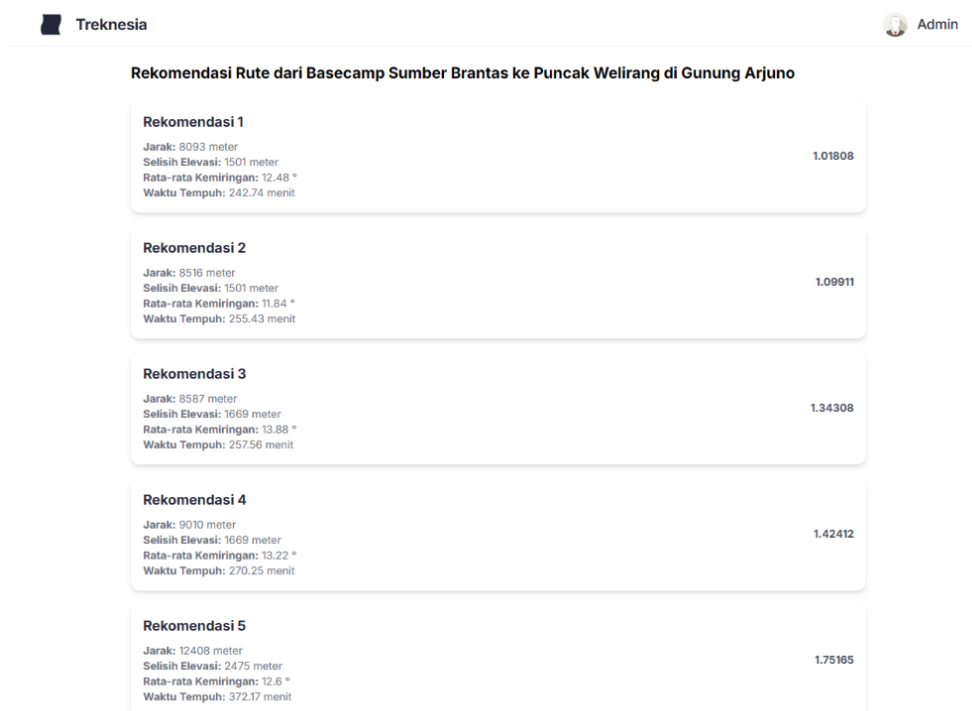


The screenshot displays the user interface of the 'Treknesia' website. At the top left, the logo 'Treknesia' is visible, and at the top right, the user is logged in as 'Admin'. The main content area features a large image of Gunung Arjuno with the text 'Gunung Arjuno' and 'Kompleksitas Jalur Pendakian: 6'. Below the image, there is a section titled 'Temukan Rute Pendakian Terbaik di Gunung Arjuno'. This section contains two dropdown menus labeled 'Lokasi Awal' and 'Lokasi Tujuan', and a 'Cari' button.

Gambar 4.2 Tampilan Form Lokasi Awal dan Lokasi Tujuan

Pada Gambar 4.2, tampilan form input Lokasi Awal dan Lokasi Tujuan menunjukkan antarmuka pengguna dari sistem rekomendasi rute. Di bagian atas halaman, terdapat judul 'Gunung Arjuno' yang menandakan halaman ini khusus untuk rute pendakian di gunung tersebut. Di bawah judul, ada informasi mengenai Kompleksitas Jalur Pendakian dengan nilai 6, yang menunjukkan tingkat kompleksitas rute pendakian di Gunung Arjuno. Di bawahnya, terdapat form pencarian dengan dua input: Lokasi Awal dan Lokasi Tujuan. Pengguna dapat memilih lokasi awal (misalnya *Basecamp*) dan tujuan (misalnya puncak) dari dropdown menu untuk menentukan titik perjalanan. Setelah memilih kedua lokasi, pengguna dapat menekan tombol 'Cari' untuk mendapatkan rekomendasi rute pendakian terbaik.

### 4.2.3 Tampilan Hasil Rekomendasi

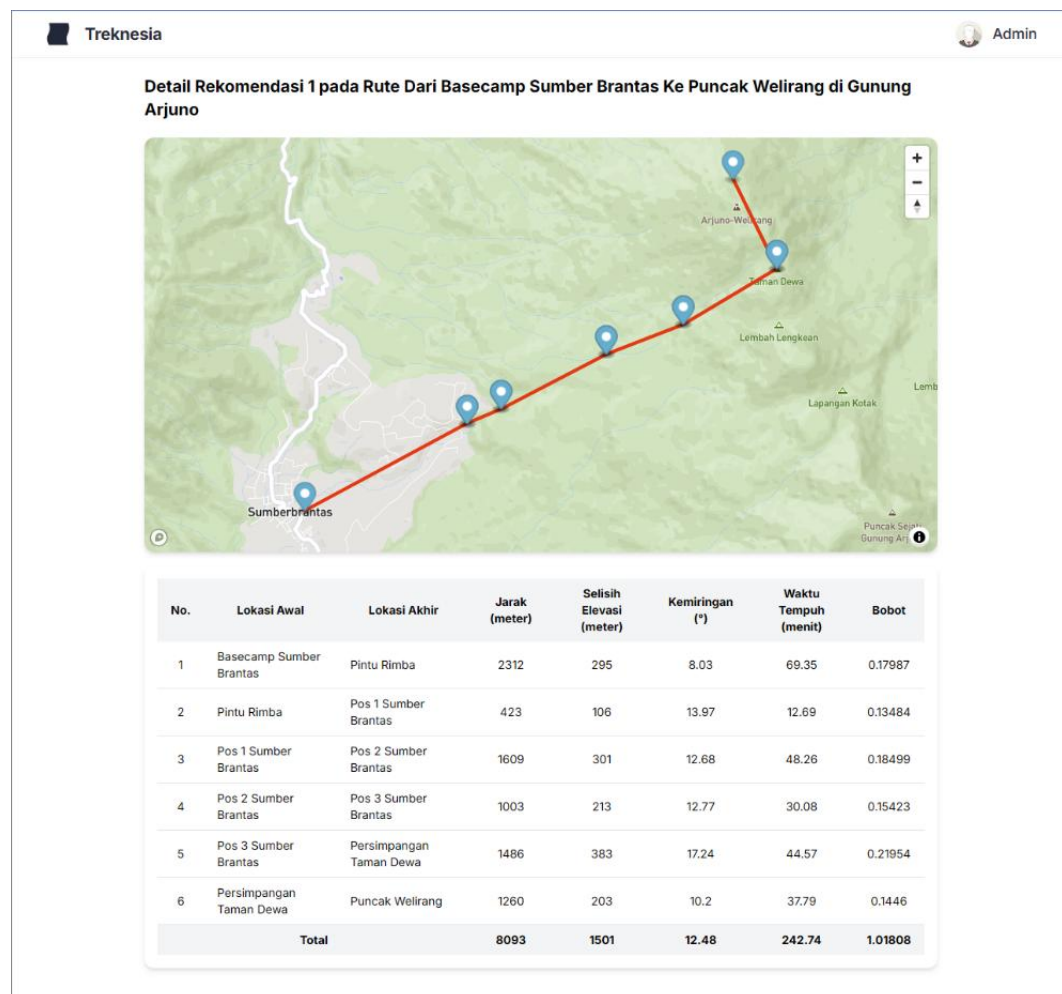


Rekomendasi	Jarak	Selisih Elevasi	Rata-rata Kemiringan	Waktu Tempuh	Nilai
Rekomendasi 1	8093 meter	1501 meter	12.48 °	242.74 menit	1.01808
Rekomendasi 2	8516 meter	1501 meter	11.84 °	255.43 menit	1.09911
Rekomendasi 3	8587 meter	1669 meter	13.88 °	257.56 menit	1.34308
Rekomendasi 4	9010 meter	1669 meter	13.22 °	270.25 menit	1.42412
Rekomendasi 5	12408 meter	2475 meter	12.6 °	372.17 menit	1.75165

Gambar 4.3 Tampilan Hasil Rekomendasi

Pada Gambar 4.3, tampilan halaman hasil rekomendasi menunjukkan rute pendakian dari *Basecamp* Sumber Brantas ke Puncak Welirang di Gunung Arjuno. Di bawah judul, terdapat daftar beberapa rute rekomendasi yang dilengkapi dengan informasi seperti Jarak, Selisih Elevasi, Rata-rata Kemiringan, dan Waktu Tempuh. Di sebelah kanan setiap rekomendasi rute, terdapat angka yang menunjukkan bobot rute berdasarkan perhitungan sistem. Setiap rekomendasi dapat diklik untuk melihat detail rute yang direkomendasikan.

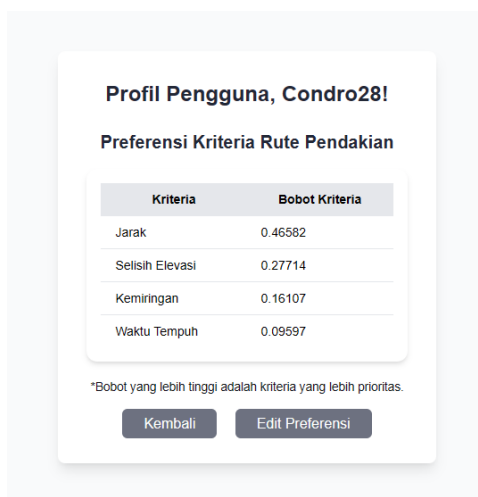
#### 4.2.4 Tampilan Detail Hasil Rekomendasi



Gambar 4.4 Tampilan Detail Hasil Rekomendasi

Pada Gambar 4.4, tampilan halaman detail hasil rekomendasi menunjukkan rincian rute pendakian dari *Basecamp* Sumber Brantas ke Puncak Welirang di Gunung Arjuno. Halaman ini menampilkan peta yang merepresentasikan rute hasil rekomendasi beserta tanda lokasinya. Serta tabel yang berisi rincian rute pendakian, dengan kolom-kolom berikut: Lokasi Awal, Lokasi Akhir, Jarak (m), Selisih Elevasi (m), Kemiringan (derajat), Waktu Tempuh (menit) dan Bobot serta dibaris paling bawah terdapat total dari setiap kolom. Tampilan ini memberikan informasi detail mengenai setiap segmen rute baik melalui peta dan juga tabel, membantu pengguna memahami letak rute, lokasi yang dilalui, jarak, selisih elevasi, kemiringan, dan waktu tempuh pada setiap titik.

#### 4.2.5 Tampilan Profil Pengguna



Gambar 4.5 Tampilan Profil Pengguna

Pada Gambar 4.5, tampilan halaman profil pengguna menampilkan username pengguna kemudian dibawahnya ditampilkan tabel preferensi kriteria rute pendakian yang berisi kriteria dan bobotnya. Tabel ini menunjukkan bobot kriteria yang digunakan oleh sistem untuk memberikan bobot pada jalur sebelum

dilakukan pencarian rekomendasi rute pendakian. Kemudian di bawah tabel terdapat dua tombol berupa tombol Kembali untuk kembali ke halaman utama dan tombol Edit Preferensi untuk melakukan pengeditan preferensi kriteria jalur pendakian.

#### 4.2.6 Tampilan Pengaturan Preferensi Kriteria

**Buat Preferensi Kriteria Rute Versimu!**

\*Buat perbandingan pada beberapa kriteria rute yaitu jarak, selisih elevasi, kemiringan dan waktu tempuh.  
\*Semakin tinggi skala semakin lebih penting kriteria yang kamu pilih.

Perbandingan Jarak dan Selisih Elevasi	Skala Perbandingan
Perbandingan Jarak dan Kemiringan	Skala Perbandingan
Perbandingan Jarak dan Waktu Tempuh	Skala Perbandingan
Perbandingan Selisih Elevasi dan Kemiringan	Skala Perbandingan
Perbandingan Selisih Elevasi dan Waktu Tempuh	Skala Perbandingan
Perbandingan Kemiringan dan Waktu Tempuh	Skala Perbandingan

Gambar 4.6 Tampilan Pengaturan Preferensi Kriteria

Pada Gambar 4.6, tampilan pengaturan preferensi kriteria menampilkan form dengan beberapa input untuk perbandingan kriteria jalur pendakian guna mendapatkan nilai bobot kriteria yang diproses menggunakan matrik perbandingan dan matriks normalisasi. Dibagian bawah form terdapat dua tombol berupa tombol Kembali untuk kembali ke halaman profil pengguna dan tombol Buat untuk membuat matriks perbandingan sehingga menghasilkan matriks normalisasi yang menghasilkan bobot kriteria jalur.

### 4.2.7 Tampilan Dashboard Admin

Pada Gambar 4.7, tampilan dashboard admin menampilkan beberapa tabel, yaitu tabel data gunung, data lokasi di setiap gunung dan data jalur di setiap gunung. Di dalam setiap tabel ada tiga tombol aksi, berupa tombol tambah data untuk melakukan penambahan data, di setiap baris ada kolom aksi berisi tombol edit data untuk menuju ke halaman edit data guna melakukan pengeditan data sesuai baris tombol dan tombol hapus data untuk melakukan penghapusan data sesuai baris tombol.

The screenshot shows the Admin Dashboard for 'Treknesia'. The user is logged in as 'Admin'. The dashboard displays two main data tables:

**Data Gunung di Indonesia**

No.	Nama Gunung	Action
1	Gunung Arjuno	<a href="#">✎</a> <a href="#">🗑</a>
2	Gunung Bromo	<a href="#">✎</a> <a href="#">🗑</a>
3	Gunung Butak	<a href="#">✎</a> <a href="#">🗑</a>
4	Gunung Ciremai	<a href="#">✎</a> <a href="#">🗑</a>
5	Gunung Penanggungan	<a href="#">✎</a> <a href="#">🗑</a>
6	Gunung Slamet	<a href="#">✎</a> <a href="#">🗑</a>
7	Gunung Sumbing	<a href="#">✎</a> <a href="#">🗑</a>
8	Gunung Ungaran	<a href="#">✎</a> <a href="#">🗑</a>

Tambah Data

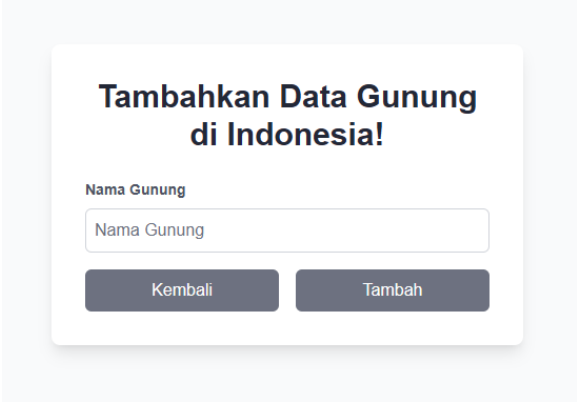
**Data Lokasi di Gunung Arjuno**

No.	Nama Lokasi	Elevasi	Koordinat	Action
1	Basecamp Lawang	855	-7.808734, 112.65403	<a href="#">✎</a> <a href="#">🗑</a>
2	Pos 1 Kebun Teh Wonosari	1000	-7.80269, 112.643327	<a href="#">✎</a> <a href="#">🗑</a>
3	Pos 2 Lincing	1610	-7.792348, 112.618297	<a href="#">✎</a> <a href="#">🗑</a>
4	Pos 3 Mahapena	2180	-7.781589, 112.611201	<a href="#">✎</a> <a href="#">🗑</a>
5	Pos 4 Gombes	2655	-7.773503, 112.602017	<a href="#">✎</a> <a href="#">🗑</a>
6	Basecamp Purwosari	900	-7.780225, 112.656177	<a href="#">✎</a> <a href="#">🗑</a>
7	Pos 1 Guo Onto Bugo	1096	-7.778135, 112.643324	<a href="#">✎</a> <a href="#">🗑</a>
8	Pos 2 Tampuono	1415	-7.770649, 112.630897	<a href="#">✎</a> <a href="#">🗑</a>
9	Pos 3 Eyang Sakri	1462	-7.769225, 112.629627	<a href="#">✎</a> <a href="#">🗑</a>
10	Pos 4 Eyang Semar	1763	-7.770869, 112.622355	<a href="#">✎</a> <a href="#">🗑</a>
11	Pos 5 Mahkutoromo	1870	-7.770598, 112.619966	<a href="#">✎</a> <a href="#">🗑</a>
12	Pos 6 Candi Sepilar	1946	-7.769983, 112.61826	<a href="#">✎</a> <a href="#">🗑</a>
13	Pos 7 Jawa Dipa	2350	-7.765533, 112.613373	<a href="#">✎</a> <a href="#">🗑</a>
14	Basecamp Tretes	812	-7.698118, 112.631784	<a href="#">✎</a> <a href="#">🗑</a>
15	Pos 1 Pet Bocor	965	-7.702753, 112.626839	<a href="#">✎</a> <a href="#">🗑</a>
16	Pos 2 Kokopan	1647	-7.722517, 112.61655	<a href="#">✎</a> <a href="#">🗑</a>
17	Pos 3 Bantakan	2486	-7.742801, 112.604571	<a href="#">✎</a> <a href="#">🗑</a>

Gambar 4.7 Tampilan Dashboard Admin



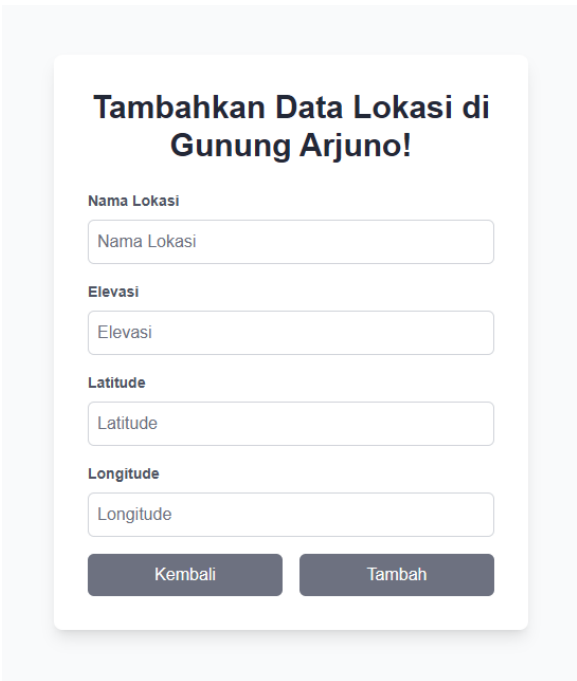
#### 4.2.8 Tampilan Tambah Data



The screenshot shows a web form titled "Tambahkan Data Gunung di Indonesia!". It features a single text input field labeled "Nama Gunung" with the placeholder text "Nama Gunung". Below the input field are two dark grey buttons: "Kembali" (Back) and "Tambah" (Add).

Gambar 4.8 Tampilan Tambah Data Gunung

Pada Gambar 4.8, tampilan halaman tambah data gunung menampilkan form yang terdapat input nama gunung untuk menambahkan data gunung. Dibawah form terdapat dua tombol berupa tombol Kembali untuk kembali ke halaman dashboard admin dan tombol Tambah untuk submit penambahan data gunung.

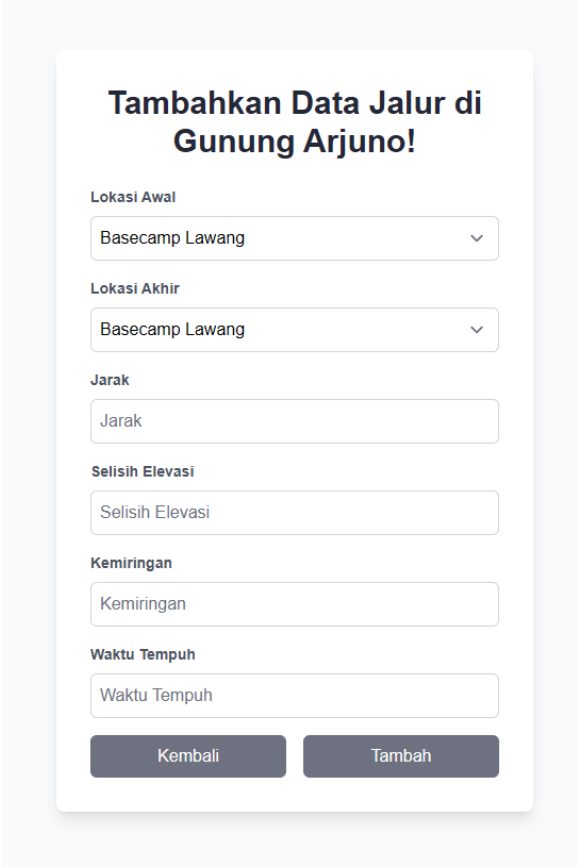


The screenshot shows a web form titled "Tambahkan Data Lokasi di Gunung Arjuno!". It features four text input fields, each with a label above it: "Nama Lokasi" (placeholder: "Nama Lokasi"), "Elevasi" (placeholder: "Elevasi"), "Latitude" (placeholder: "Latitude"), and "Longitude" (placeholder: "Longitude"). Below the input fields are two dark grey buttons: "Kembali" (Back) and "Tambah" (Add).

Gambar 4.9 Tampilan Tambah Data Lokasi

Pada Gambar 4.9, tampilan halaman tambah data lokasi menampilkan form yang terdapat input nama lokasi, elevasi, latitude dan longitude untuk menambahkan data lokasi. Dibawah form terdapat dua tombol berupa tombol Kembali untuk kembali ke halaman dashboard admin dan tombol Tambah untuk submit penambahan data lokasi.

Pada Gambar 4.10, tampilan halaman tambah data jalur menampilkan form yang terdapat input lokasi awal, lokasi akhir, jarak, selisih elevasi, kemiringan dan waktu tempuh untuk menambahkan data jalur. Dibawah form terdapat dua tombol berupa tombol Kembali untuk kembali ke halaman dashboard admin dan tombol Tambah untuk submit penambahan data jalur.



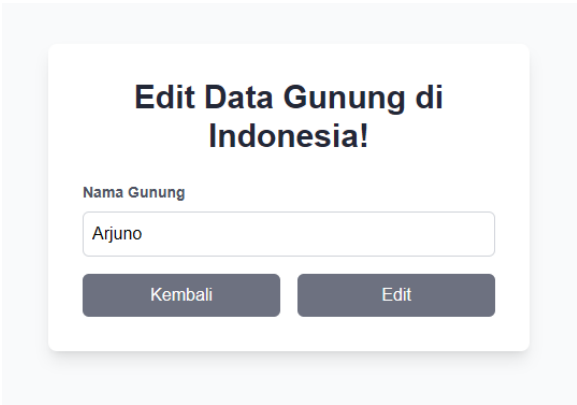
The image shows a web form titled "Tambahkan Data Jalur di Gunung Arjuno!". The form contains the following fields and buttons:

- Lokasi Awal:** A dropdown menu with "Basecamp Lawang" selected.
- Lokasi Akhir:** A dropdown menu with "Basecamp Lawang" selected.
- Jarak:** A text input field with the placeholder "Jarak".
- Selisih Elevasi:** A text input field with the placeholder "Selisih Elevasi".
- Kemiringan:** A text input field with the placeholder "Kemiringan".
- Waktu Tempuh:** A text input field with the placeholder "Waktu Tempuh".
- Buttons:** Two buttons at the bottom: "Kembali" (Return) and "Tambah" (Add).

Gambar 4.10 Tampilan Tambah Data Jalur

#### 4.2.9 Tampilan Edit Data

Pada Gambar 4.11, tampilan halaman edit data gunung menampilkan form yang terdapat input nama gunung untuk mengedit data gunung. Dibawah form terdapat dua tombol berupa tombol Kembali untuk kembali ke halaman dashboard admin dan tombol Tambah untuk submit pengeditan data gunung.

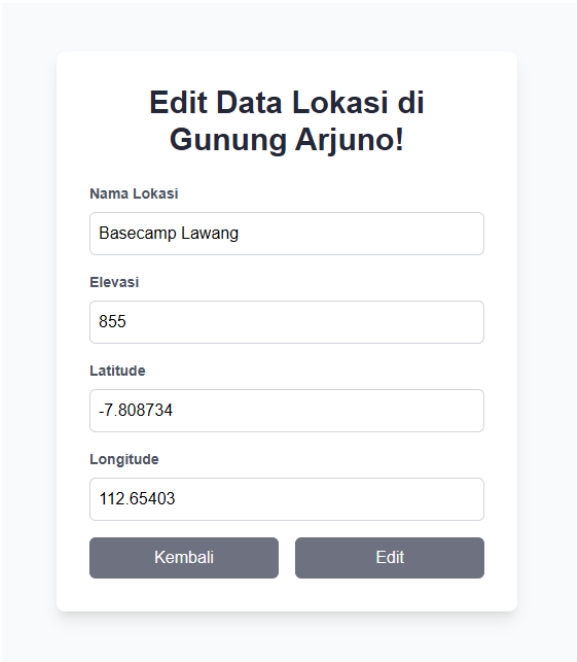


**Edit Data Gunung di Indonesia!**

Nama Gunung

Kembali Edit

Gambar 4.11 Tampilan Edit Data Mountain



**Edit Data Lokasi di Gunung Arjuno!**

Nama Lokasi

Elevasi

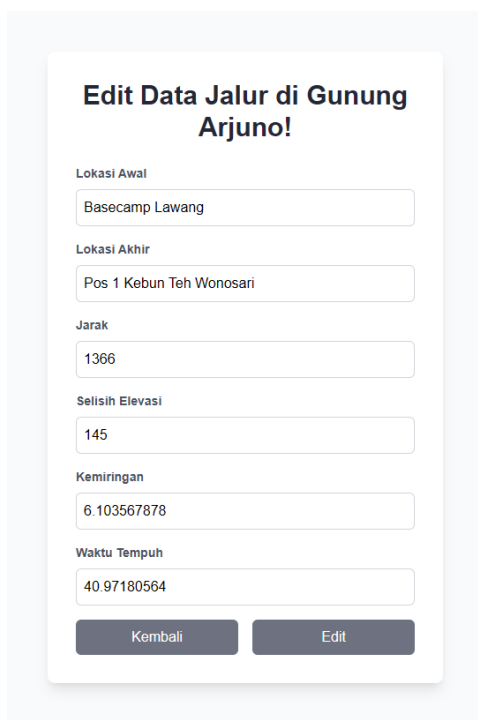
Latitude

Longitude

Kembali Edit

Gambar 4.12 Tampilan Edit Data Lokasi

Pada Gambar 4.12, tampilan halaman edit data lokasi menampilkan form yang terdapat input nama lokasi, elevasi, latitude dan longitude untuk mengedit data lokasi. Dibawah form terdapat dua tombol berupa tombol Kembali untuk kembali ke halaman dashboard admin dan tombol Tambah untuk submit pengeditan data lokasi.



**Edit Data Jalur di Gunung Arjuno!**

Lokasi Awal  
Basecamp Lawang

Lokasi Akhir  
Pos 1 Kebun Teh Wonosari

Jarak  
1366

Selisih Elevasi  
145

Kemiringan  
6.103567878

Waktu Tempuh  
40.97180564

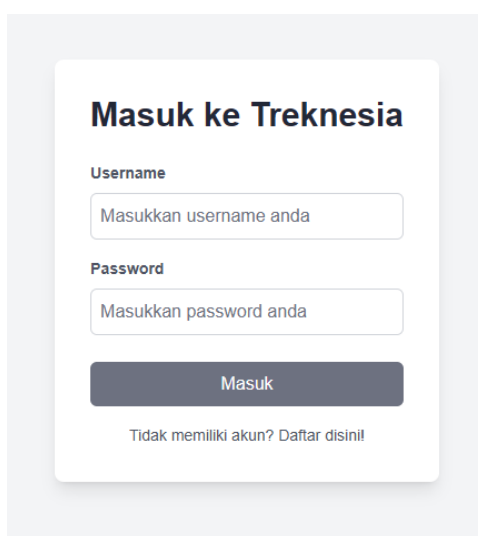
Kembali Edit

Gambar 4.13 Tampilan Edit Data Jalur

Pada Gambar 4.13, tampilan halaman edit data jalur menampilkan form yang terdapat input lokasi awal, lokasi akhir, jarak, selisih elevasi, kemiringan dan waktu tempuh untuk mengedit data jalur. Dibawah form terdapat dua tombol berupa tombol Kembali untuk kembali ke halaman dashboard admin dan tombol Tambah untuk submit pengeditan data jalur.

#### 4.2.10 Tampilan *Login*

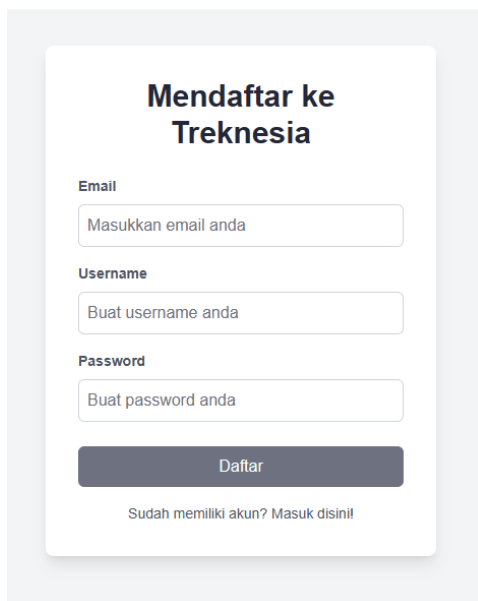
Pada Gambar 4.14, tampilan halaman *login* menampilkan form untuk input username dan password untuk verifikasi masuk ke dalam halaman utama *web*. Dibagian bawah ada teks “Tidak memiliki akun? Mendaftar disini!” yang bisa diklik untuk menuju ke halaman pendaftaran akun bagi user yang belum memiliki akun. Tampilan *login* inilah yang secara default ditampilkan oleh *web* ketika *web* dibuka sebelum pengguna melakukan *login*.

The image shows a login form titled "Masuk ke Treknesia". It contains two input fields: "Username" with the placeholder text "Masukkan username anda" and "Password" with the placeholder text "Masukkan password anda". Below the fields is a dark grey button labeled "Masuk". At the bottom of the form, there is a link that says "Tidak memiliki akun? Daftar disini!".

Gambar 4.14 Tampilan Login

#### 4.2.11 Tampilan *Register*

Pada Gambar 4.15, tampilan halaman *register* menampilkan form untuk input email, username dan password untuk melakukan pendaftaran akun untuk masuk ke halaman utama *web*. Dibagian bawah form terdapat teks “Sudah memiliki akun? Masuk disini!” yang bisa diklik untuk menuju ke halaman *login* bagi pengguna yang sudah memiliki akun.



**Mendaftar ke  
Treknesia**

Email  
Masukkan email anda

Username  
Buat username anda

Password  
Buat password anda

Daftar

Sudah memiliki akun? Masuk disini!

Gambar 4.15 Tampilan Register

### 4.3 Evaluasi Kompleksitas *Chyper Query Graph Database*

Pada evaluasi kompleksitas *chyper query* disini menjelaskan implementasi dan evaluasi beberapa *chyper query* yang digunakan dalam pengolahan data jalur pendakian gunung menggunakan graph database *Neo4j*. Evaluasi dilakukan dengan menguji kompleksitas *query* dan waktu pemrosesan dalam mengimpor data, melakukan normalisasi dan pembobotan jalur, serta menerapkan algoritma pencarian jalur terbaik menggunakan algoritma *Dijkstra*.

#### 4.3.1 *Chyper Query Import Data Jalur dan Titik Lokasi Pendakian Gunung*

Pada tahap awal, data titik lokasi dan jalur pendakian diimpor ke dalam *graph database Neo4j*. Proses ini merupakan fondasi untuk membangun struktur data graf yang merepresentasikan ekosistem pendakian, di mana setiap titik lokasi menjadi *node* dan setiap jalur menjadi relasi antar *node*. Berikut pada Tabel 4.1 adalah *Cypher query* yang digunakan dalam proses import data.

Tabel 4.1 *Cypher Query Import Data Jalur dan Titik Lokasi Pendakian Gunung*

<i>Cypher Query Import Data Jalur dan Titik Lokasi Pendakian Gunung</i>
<pre> LOAD CSV WITH HEADERS FROM 'file:///lokasi_arjunowelirang.csv' AS row MERGE (n:ArjunoWelirang {id_lokasi: row.ID_Lokasi}) SET n.nama_lokasi = row&gt;Nama_Lokasi,     n.elevasi = toInteger(row.Elevasi),     n.latitude = toFloat(row.Latitude),     n.longitude = toFloat(row.Longitude)  LOAD CSV WITH HEADERS FROM 'file:///jalur_arjunowelirang.csv' AS row MATCH (start:ArjunoWelirang {nama_lokasi: row.Titik_Awal}) MATCH (end:ArjunoWelirang {nama_lokasi: row.Titik_Akhir}) MERGE (start)-[e:JalurArjunoWelirang {   id_jalur: toInteger(row.ID_Jalur),   selisih_elevasi: toInteger(row.Selisih_Elevasi),   jarak: toInteger(row.Jarak),   kemiringan: toFloat(row.Kemiringan),   waktu_tempuh: toFloat(row.Waktu_Tempuh) }]-&gt;(end) </pre>

Setelah dilakukan analisis pada *cypher query* dalam Tabel 4.1 menggunakan model ukur kompleksitas *cypher query* berikut pada Tabel 4.2 adalah hasil perhitungan kompleksitas *cypher query*nya:

Tabel 4.2 Perhitungan Kompleksitas *Cypher Query Import Data*

No.	Atribut Query	Bobot Atribut	Jumlah Atribut	Bobot
1	<i>Variable Output</i>	0.1	0	0
2	<i>Variable Input</i>	0.15	11	1.65
3	<i>Nested Query</i>	0.2	0	0
4	<i>Relathionship</i>	0.25	1	0.25
5	<i>Node</i>	0.3	3	0.9
Kompleksitas <i>Cypher Query</i>				2.8
Waktu Pemrosesan (ms)				328

Pada Tabel 4.2 menunjukkan hasil perhitungan kompleksitas *cypher query* dari *query* pada Tabel 4.1. Hasil analisis dari *cypher query* mendapatkan jumlah atribut yang kemudian dihitung menggunakan persamaan (2.5) menghasilkan kompleksitas *cypher query* selanjutnya dijalankan di *Neo4J* untuk mencatat waktu pemrosesan *cypher query*.

### 4.3.2 *Cypher Query* Normalisasi dan Pembobotan Data Jalur

Setelah data jalur berhasil diimpor, langkah selanjutnya adalah melakukan normalisasi dan pembobotan pada setiap jalur untuk memastikan setiap kriteria memiliki skala yang sebanding. Proses normalisasi ini krusial untuk menghilangkan bias yang mungkin timbul akibat perbedaan satuan atau rentang nilai antar kriteria, seperti jarak, elevasi, atau kemiringan. Dengan demikian, semua faktor dapat berkontribusi secara adil dalam perhitungan bobot akhir. Berikut pada Tabel 4.3 adalah *Cypher query* yang digunakan dalam proses normalisasi dan pembobotan data jalur.

Tabel 4.3 *Cypher Query* Normalisasi dan Pembobotan Data Jalur

<i>Cypher Query</i> Normalisasi dan Pembobotan Data Jalur
<pre> MATCH (a)-[r:JalurArjuno]-&gt;(b) WITH   MIN(r.jarak) AS min_jarak, MAX(r.jarak) AS max_jarak,   MIN(ABS(r.selisih_elevasi)) AS min_elevasi, MAX(ABS(r.selisih_elevasi)) AS max_elevasi,   MIN(ABS(r.kemiringan)) AS min_kemiringan, MAX(ABS(r.kemiringan)) AS max_kemiringan,   MIN(r.waktu_tempuh) AS min_waktu_tempuh, MAX(r.waktu_tempuh) AS max_waktu_tempuh  MATCH (a)-[r:JalurArjuno]-&gt;(b) WITH r,   min_jarak, max_jarak,   min_elevasi, max_elevasi,   min_kemiringan, max_kemiringan,   min_waktu_tempuh, max_waktu_tempuh  WITH r,   (r.jarak - min_jarak) / (max_jarak - min_jarak) AS norm_jarak,   (ABS(r.selisih_elevasi) - min_elevasi) / (max_elevasi - min_elevasi) AS norm_elevasi,   (ABS(r.kemiringan) - min_kemiringan) / (max_kemiringan - min_kemiringan) AS norm_kemiringan,   (r.waktu_tempuh - min_waktu_tempuh) / (max_waktu_tempuh - min_waktu_tempuh) AS norm_waktu_tempuh  WITH r,   norm_jarak * 0.194553201 +   norm_elevasi * 0.323769675 +   norm_kemiringan * 0.292220446 +   norm_waktu_tempuh * 0.189456678 AS bobot  SET r.bobot = bobot RETURN r, r.bobot; </pre>



Setelah dilakukan analisis pada *chyper query* dalam Tabel 4.3 menggunakan model ukur kompleksitas *chyper query* berikut pada Tabel 4.4 adalah hasil perhitungan kompleksitas *chyper query*nya:

Tabel 4.4 Perhitungan Kompleksitas *Chyper Query* Normalisasi dan Pembobotan

No.	Atribut Query	Bobot w	Xi	Wi.Xi
1	Variable Output	0.1	2	0.2
2	Variable Input	0.15	1	0.15
3	Nested Query	0.2	0	0
4	Relathionship	0.25	1	0.25
5	Node	0.3	2	0.6
Kompleksitas <i>Chyper Query</i>				1.2
Waktu Pemrosesan (ms)				47

Pada Tabel 4.4 menunjukkan hasil perhitungan kompleksitas *chyper query* dari *query* pada Tabel 4.3. Hasil analisis dari *chyper query* mendapatkan jumlah atribut yang kemudian dihitung menggunakan persamaan (2.5) menghasilkan kompleksitas *chyper query* selanjutnya dijalankan di *Neo4J* untuk mencatat waktu pemrosesan *chyper query*.

#### 4.3.3 *Chyper Query Cyclomatic Complexity*

Pada bagian ini dilakukan penghitungan kompleksitas siklomatik (*cyclomatic complexity*) dari graph jalur pendakian menggunakan *cypher query* di *Neo4j*. Berikut pada Tabel 4.5 adalah *chyper query* yang digunakan dalam perhitungan *cyclomatic complexity*.

Tabel 4.5 *Cypher Query Cyclomatic Complexity*

<i>Cypher Query Cyclomatic Complexity</i>
<pre> MATCH (n:Arjuno)   WITH COUNT(n) AS total_nodes MATCH ()-[r:JalurArjuno]-&gt;()   WITH total_nodes, COUNT(r) AS total_edges RETURN (total_edges - total_nodes + 2) AS cyclomatic_complexity;</pre>

Setelah dilakukan analisis pada *chyper query* dalam Tabel 4.5 menggunakan model ukur kompleksitas *chyper query* berikut pada Tabel 4.6 adalah hasil perhitungan kompleksitas *chyper query*nya:

Tabel 4.6 Perhitungan Kompleksitas *Chyper Query Cyclomatic Complexity*

No.	Atribut Query	Bobot w	Xi	Wi.Xi
1	Variable Output	0.1	1	0.1
2	Variable Input	0.15	2	0.3
3	Nested Query	0.2	0	0
4	Relathionship	0.25	1	0.25
5	Node	0.3	1	0.3
Kompleksitas <i>Chyper Query</i>				0.95
Waktu Pemrosesan (ms)				18

Pada Tabel 4.6, ditunjukkan hasil perhitungan **kompleksitas Cypher query** dari *query* pada Tabel 4.5. Hasil analisis dari *Cypher query* ini mendapatkan jumlah atribut yang kemudian dihitung menggunakan persamaan (2.5), menghasilkan kompleksitas *Cypher query*. Selanjutnya, *query* tersebut dijalankan di Neo4j untuk mencatat waktu pemrosesannya.

#### 4.3.4 *Chyper Query* Algoritma *Dijkstra* untuk Hasil Pencarian Rute

Implementasi algoritma *Dijkstra* pada *graph database* memungkinkan pencarian rute pendakian dengan bobot total terkecil dari titik awal ke titik tujuan. Algoritma ini secara efisien menemukan jalur optimal dengan mempertimbangkan berbagai faktor yang telah dibobot sebelumnya, seperti jarak, elevasi, dan waktu tempuh. Dengan demikian, sistem dapat merekomendasikan rute yang paling efisien atau paling sesuai dengan preferensi pengguna. Berikut pada Tabel 4.7 adalah *Cypher query* yang digunakan dalam proses pencarian rute pendakian terbaik.

Tabel 4.7 *Cypher Query* Algoritma *Dijkstra* untuk Hasil Pencarian Rute

<b><i>Cypher Query Algoritma Dijkstra untuk Hasil Pencarian Rute</i></b>
<pre> MATCH (start:Arjuno {nama_lokasi: 'Basecamp Lawang'}), (end:Arjuno {nama_lokasi: 'Puncak Welirang'}) CALL apoc.algo.dijkstra(start, end, 'JalurArjuno', 'bobot', 0, 100) YIELD path, weight   WITH path, weight,     REDUCE(total_jarak = 0, r IN relationships(path)   total_jarak + r.jarak) AS total_jarak,     REDUCE(total_selisih_elevasi = 0, r IN relationships(path)   total_selisih_elevasi + r.selisih_elevasi) AS total_selisih_elevasi,     REDUCE(total_waktu = 0, r IN relationships(path)   total_waktu + r.waktu_tempuh) AS total_waktu,     REDUCE(total_kemiringan = 0, r IN relationships(path)   total_kemiringan + r.kemiringan) AS total_kemiringan,     SIZE(relationships(path)) AS total_jalur   WITH path, weight, total_jarak, total_selisih_elevasi, total_waktu, total_kemiringan, total_jalur, nodes(path) AS nodes_in_path   RETURN [n IN nodes_in_path   n.nama_lokasi] AS node_names,     weight AS total_bobot, total_jarak, total_selisih_elevasi, total_waktu,     total_kemiringan / total_jalur AS avg_kemiringan; </pre>

Setelah dilakukan analisis pada *cypher query* dalam Tabel 4.7 menggunakan model ukur kompleksitas *cypher query* berikut pada Tabel 4.8 adalah hasil perhitungan kompleksitas *cypher query*nya:

Tabel 4.8 Perhitungan Kompleksitas *Cypher Query* Algoritma *Dijkstra*

No.	Atribut Query	Bobot w	Xi	Wi.Xi
1	Variable Output	0.1	6	0.6
2	Variable Input	0.15	2	0.3
3	Nested Query	0.2	1	0.2
4	Relationship	0.25	1	0.25
5	Node	0.3	2	0.6
Kompleksitas <i>Cypher Query</i>				1.95
Waktu Pemrosesan (ms)				197

Pada Tabel 4.8 menunjukkan hasil perhitungan kompleksitas *cypher query* dari *query* pada Tabel 4.7. Hasil analisis dari *cypher query* mendapatkan jumlah atribut yang kemudian dihitung menggunakan persamaan (2.5) menghasilkan kompleksitas *cypher query* selanjutnya dijalankan di *Neo4J* untuk mencatat waktu pemrosesan *cypher query*.

#### 4.3.5 *Cypher Query* Algoritma *Dijkstra* untuk Detail Hasil Pencarian Rute

Query ini memperdalam hasil pencarian dengan menampilkan detail masing-masing segmen jalur yang direkomendasikan oleh algoritma *Dijkstra*. Tidak hanya rute keseluruhan yang ditampilkan, tetapi juga informasi spesifik untuk setiap bagian dari jalur, seperti jarak, elevasi, dan waktu tempuh per segmen. Hal ini memberikan pemahaman yang lebih komprehensif kepada pengguna tentang karakteristik setiap tahapan pendakian. Berikut pada Tabel 4.9 adalah *Cypher query* yang digunakan dalam menampilkan detail hasil pencarian rute.

Tabel 4.9 *Cypher Query* Algoritma *Dijkstra* untuk Detail Hasil Pencarian Rute

<b><i>Cypher Query</i> Algoritma <i>Dijkstra</i> untuk Detail Hasil Pencarian Rute</b>
<pre> MATCH (start:Arjuno {nama_lokasi: 'Basecamp Lawang'}), (end:Arjuno {nama_lokasi: 'Puncak Welirang'}) CALL apoc.algo.dijkstra(start, end, 'JalurArjuno', 'bobot', 0, 100) YIELD path, weight   WITH path, weight, relationships(path) AS edges, nodes(path) AS nodes   ORDER BY weight ASC   SKIP 2   LIMIT 1   UNWIND range(0, SIZE(nodes) - 2) AS idx   WITH nodes[idx] AS lokasi_awal, nodes[idx + 1] AS lokasi_akhir, edges[idx] AS edge   RETURN     lokasi_awal.nama_lokasi AS Lokasi_Awal, lokasi_akhir.nama_lokasi AS Lokasi_Akhir,     edge.jarak AS Jarak, edge.selisih_elevasi AS Selisih_Elevasi, edge.kemiringan AS Kemiringan,     edge.waktu_tempuh AS Waktu_Tempuh; </pre>

Setelah dilakukan analisis pada *cypher query* dalam Tabel 4.9 menggunakan model ukur kompleksitas *cypher query* berikut pada Tabel 4.10 adalah hasil perhitungan kompleksitas *cypher query*nya:

Tabel 4.10 Perhitungan Kompleksitas *Cypher Query* Algoritma *Dijkstra*

No.	Atribut Query	Bobot w	$X_i$	$W_i \cdot X_i$
1	Variable Output	0.1	6	0.6
2	Variable Input	0.15	2	0.3
3	Nested Query	0.2	1	0.2
4	Relationship	0.25	1	0.25
5	Node	0.3	2	0.6
Kompleksitas <i>Cypher Query</i>				1.95
Waktu Pemrosesan (ms)				147

Pada Tabel 4.10 menunjukkan hasil perhitungan kompleksitas *chyper query* dari *query* pada Tabel 4.9. Hasil analisis dari *chyper query* mendapatkan jumlah atribut yang kemudian dihitung menggunakan persamaan (2.5) menghasilkan kompleksitas *chyper query* selanjutnya dijalankan di Neo4J untuk mencatat waktu pemrosesan *chyper query*.

Setelah didapatkan masing-masing hasil perhitungan kompleksitas pada beberapa *chyper query* di atas dan hasil pengujian waktu pemrosesannya. Berikut pada Tabel 4.11 adalah tabel ringkasan hasil perhitungan kompleksitas dan waktu pemrosesan pada masing-masing *chyper query*.

Tabel 4.11 Hasil Perhitungan Kompleksitas *Chyper Query* dan Waktu Pemrosesan

No.	Atribut Query	Kompleksitas	Waktu Pemrosesan (ms)
1	<i>Variable Output</i>	2.8	328
2	<i>Variable Input</i>	1.2	47
3	<i>Nested Query</i>	0.95	18
4	<i>Relathionship</i>	1.95	197
5	<i>Node</i>	1.95	147

Pada Tabel 4.11 diatas menunjukkan ringkasan hasil perhitungan kompleksitas dan waktu pemrosesan dari masing-masing *chyper query* yang digunakan di dalam sistem rekomendasi pendakian gunung ini.

#### 4.4 Hasil Pengujian Sistem

Penelitian ini melakukan pengujian sistem untuk memperoleh nilai kompleksitas *graph*, akurasi, dan waktu pemrosesan pada setiap gunung. Perhitungan kompleksitas *graph* menggunakan *Cyclomatic Complexity*, kemudian perhitungan akurasi menggunakan *FI-Score* yang nilainya diperoleh dari penilaian pendaki yang berpengalaman. Sementara itu, waktu pemrosesan didapat dari seberapa lama sistem dapat memberikan *output* rekomendasi rute. Berikut adalah

beberapa tabel hasil pengujian secara ringkas; hasil pengujian lebih lengkapnya terdapat pada tabel di halaman lampiran.

Tabel 4.12 Hasil Perhitungan Kompleksitas *Graph* pada Setiap Gunung

No	Gunung	Node	Edge	Kompleksitas
1	Arjuno-Welirang	40	44	6
2	Bromo	19	25	8
3	Butak	36	38	4
4	Ciremai	31	31	2
5	Penanggungan	31	35	6
6	Slamet	48	47	1
7	Sumbing	42	43	3
8	Ungaran	16	18	4

Pada Tabel 4.12, setiap *graph* gunung yang ada dilakukan perhitungan kompleksitas *graph* menggunakan persamaan (3.5) berdasarkan jumlah *node* dan *edge*, sehingga menghasilkan kompleksitas *graph* dari masing-masing gunung. Perhitungan ini memberikan gambaran kuantitatif mengenai seberapa rumit struktur jaringan jalur pendakian pada setiap gunung, mencerminkan kepadatan konektivitas antar lokasi (*node*) dan banyaknya jalur (*edge*) yang tersedia. Semakin tinggi nilai kompleksitas, semakin banyak pilihan rute dan interkoneksi yang ada dalam *graph* gunung tersebut. Informasi ini penting untuk memahami karakteristik topologi setiap gunung dalam konteks pendakian dan dapat menjadi salah satu faktor dalam perencanaan rute yang efisien. Analisis kompleksitas ini juga membantu dalam memprediksi performa sistem saat beroperasi pada *graph* dengan skala yang berbeda, karena *graph* yang lebih kompleks mungkin memerlukan sumber daya komputasi yang lebih besar. Oleh karena itu, pemahaman tentang kompleksitas *graph* ini esensial untuk optimasi sistem di masa mendatang dan pengembangan fitur yang lebih canggih.

Tabel 4.13 Hasil Pengujian di Gunung Arjuno

No	Lokasi Awal	Lokasi Akhir	Bobot	Akurasi	Waktu
1	Basecamp Lawang	Puncak Arjuno Barat	1.428	100	82
2		Puncak Arjuno Ogal-Agil	1.42	100	65
3		Puncak Kembar 1	2.239	97.18	68.0
4		Puncak Kembar 2	2.298	97.87	77
5		Puncak Welirang	2.557	97.82	65
6	Basecamp Purwosari	Puncak Arjuno Barat	1.939	100	64
7		Puncak Arjuno Ogal-Agil	1.931	100	61
8		Puncak Kembar 1	2.75	98.83	72
9		Puncak Kembar 2	2.808	98.21	62
10		Puncak Welirang	3.068	98.69	63
11	Basecamp Tretes	Puncak Arjuno Barat	2.184	96.00	65
12		Puncak Arjuno Ogal-Agil	2.193	97.25	65
13		Puncak Kembar 1	1.654	99.17	66
14		Puncak Kembar 2	2.02	99.44	73
15		Puncak Welirang	1.626	100	65
16	Basecamp Sumber Brantas	Puncak Arjuno Barat	1.501	98.18	67
17		Puncak Arjuno Ogal-Agil	1.509	97.91	62
18		Puncak Kembar 1	1.025	100	62
19		Puncak Kembar 2	1.084	100	63
20		Puncak Welirang	1.018	99.29	67
Rata-rata				98.793	66.7

Pada Tabel 4.13, disajikan hasil perhitungan akurasi menggunakan persamaan (3.3) dan waktu pemrosesan untuk setiap rekomendasi rute pendakian Gunung Arjuno, dari setiap *basecamp* ke puncaknya. Hasil rata-rata akurasi dalam pengujian pada *graf* Gunung Arjuno sangat tinggi, mencapai 98.793%, menunjukkan bahwa sistem ini mampu memberikan rekomendasi rute yang sangat sesuai dengan kriteria yang diinginkan dengan tingkat kesalahan yang minimal. Ini membuktikan efektivitas sistem dalam mengidentifikasi jalur optimal. Sementara itu, rata-rata waktu pemrosesannya hanya 66.7 ms, menunjukkan efisiensi dan responsivitas sistem yang luar biasa dalam memberikan rekomendasi secara cepat.

Tabel 4.14 Hasil Pengujian di Gunung Bromo

No	Lokasi Awal	Lokasi Akhir	Bobot	Akurasi	Waktu
1	Cemoro Lawang	Puncak Batok	0.54414	90.76	67
2		Puncak Bromo	0.21364	92.68	68
3		Puncak Crater	0.17302	91.14	66
4		Puncak Penanjakan	0.58407	94.00	67
5		Puncak Widodaren	0.33532	93.33	72
6	Seruni View Point	Puncak Batok	0.69963	92.56	66
7		Puncak Bromo	0.39414	91.80	65
8		Puncak Crater	0.35351	93.44	67
9		Puncak Penanjakan	0.40357	86.67	65
10		Puncak Widodaren	0.51581	96.89	67
Rata-rata				92.328	67

Pada Tabel 4.14 adalah hasil perhitungan akurasi menggunakan persamaan (3.3) dan waktu pemrosesan pada setiap hasil rekomendasi rute pendakian gunung dari setiap *Basecamp* ke setiap puncak di gunung Bromo. Yang mana hasil rata-rata akurasi dalam pengujian di graf gunung Bromo hasilnya 92.328% dan rata-rata waktu pemrosesannya selama 67 ms.

Tabel 4.15 Hasil Pengujian di Gunung Butak

No	Lokasi Awal	Lokasi Akhir	Bobot	Akurasi	Waktu
1	<i>Basecamp</i> Kraton Gunung Kawi	Puncak Batu Tulis	1.57639	100	66
2		Puncak Butak	2.17183	96.36	66
3		Puncak Panderman	4.45792	100	67
4	<i>Basecamp</i> Sirah Kencong	Puncak Batu Tulis	3.13206	98.43	66
5		Puncak Butak	2.55098	97.69	69
6		Puncak Panderman	5.10643	100	72
7	<i>Basecamp</i> Toyomerto	Puncak Batu Tulis	2.3043	99.33	66
8		Puncak Butak	1.99258	97.86	61
9		Puncak Panderman	0.93231	100	61
Rata-rata				98.853	66

Pada Tabel 4.15 adalah hasil perhitungan akurasi menggunakan persamaan (3.3) dan waktu pemrosesan pada setiap hasil rekomendasi rute pendakian gunung dari setiap *Basecamp* ke setiap puncak di gunung Butak. Yang mana hasil rata-rata



akurasi dalam pengujian di graf gunung Butak hasilnya 98.853% dan rata-rata waktu pemrosesannya selama 66 ms.

Tabel 4.16 Hasil Pengujian di Gunung Ciremai

No	Lokasi Awal	Lokasi Akhir	Bobot	Akurasi	Waktu
1	Basecamp Apuy	Puncak Ciremai	1.663	100	67
2		Puncak Crater	1.634	100	64
3		Puncak Panglongokan	1.652	98.18	64
4	Basecamp Linggarjati	Puncak Ciremai	2.447	100	67
5		Puncak Crater	2.418	100	72
6		Puncak Panglongokan	2.401	100	67
7	Basecamp Palutungan	Puncak Ciremai	2.424	98.76	61
8		Puncak Crater	2.395	100	59
9		Puncak Panglongokan	2.413	98.85	62
Rata-rata				99.532	64.778

Pada Tabel 4.16, hasil perhitungan akurasi dan waktu pemrosesan untuk rekomendasi rute pendakian Gunung Ciremai. Sistem menunjukkan rata-rata akurasi sangat tinggi sebesar 99.532%, menandakan rekomendasi rute yang sangat sesuai, dan rata-rata waktu pemrosesan yang cepat, 64.778 ms, menjamin efisiensi.

Tabel 4.17 Hasil Pengujian di Gunung Penanggungan

No	Lokasi Awal	Lokasi Akhir	Bobot	Akurasi	Waktu
1	Basecamp Genting	Puncak Bekel	1.86312	94.00	65
2		Puncak Crater	2.2774	95.56	71
3		Puncak Penanggungan	2.40442	94.95	62
4	Basecamp Jolotundo	Puncak Bekel	1.04	93.53	66
5		Puncak Crater	1.56061	97.14	69
6		Puncak Penanggungan	1.68763	95.54	67
7	Basecamp Tamiajeng	Puncak Bekel	2.48715	94.56	67
8		Puncak Crater	1.56377	96.70	68
9		Puncak Penanggungan	1.43675	100	66
Rata-rata				95.775	66.778

Pada Tabel 4.17 adalah hasil perhitungan akurasi menggunakan persamaan (3.3) dan waktu pemrosesan pada setiap hasil rekomendasi rute pendakian gunung

dari setiap *Basecamp* ke setiap puncak di gunung Penanggungan. Yang mana hasil rata-rata akurasi dalam pengujian di graf gunung Penanggungan hasilnya 95.775% dan rata-rata waktu pemrosesannya selama 66.778 ms.

Tabel 4.18 Hasil Pengujian di Gunung Slamet

No	Lokasi Awal	Lokasi Akhir	Bobot	Akurasi	Waktu
1	<i>Basecamp</i> Bambang	Puncak Barat	2.152	100	62
2		Puncak Slamet	2.03	100	63
3		Puncak Surono	2.075	100	65
4	<i>Basecamp</i> Guci Kompak	Puncak Barat	1.419	100	60
5		Puncak Slamet	1.541	100	60
6		Puncak Surono	1.496	100	61
7	<i>Basecamp</i> Guci Permadi	Puncak Barat	1.371	100	60
8		Puncak Slamet	1.493	100	71
9		Puncak Surono	1.448	100	58
10	<i>Basecamp</i> Kaliwadas	Puncak Barat	1.616	100	61
11		Puncak Slamet	1.738	100	60
12		Puncak Surono	1.694	100	59
13	<i>Basecamp</i> Penakir	Puncak Barat	2.354	99.55	67
14		Puncak Slamet	2.232	98.9	62
15		Puncak Surono	2.277	100	59
16	<i>Basecamp</i> Radenpala-Baturaden	Puncak Barat	1.791	100	59
17		Puncak Slamet	1.758	99.37	65
18		Puncak Surono	1.714	99.29	62
19	<i>Basecamp</i> Sawangan	Puncak Barat	1.145	100	63
20		Puncak Slamet	1.267	99.45	64
21		Puncak Surono	1.222	100	67
Rata-rata				99.836	62.286

Pada Tabel 4.18 adalah hasil perhitungan akurasi menggunakan persamaan (3.3) dan waktu pemrosesan pada setiap hasil rekomendasi rute pendakian gunung dari setiap basecamp ke setiap puncak di gunung Slamet. Yang mana hasil rata-rata akurasi dalam pengujian di graf gunung Slamet hasilnya 99.836% dan rata-rata waktu pemrosesannya selama 62.286 ms.

Tabel 4.19 Hasil Pengujian di Gunung Sumbing

No	Lokasi Awal	Lokasi Akhir	Bobot	Akurasi	Waktu
1	<i>Basecamp Adipuro</i>	Puncak Buntu	1.761	99.58	66
2		Puncak Kawah	1.451	98.02	62
3		Puncak Rajawali	1.377	97.78	68
4		Puncak Sejati	1.276	100	64
5	<i>Basecamp Banaran</i>	Puncak Buntu	1.982	99.09	64
6		Puncak Kawah	1.672	98.89	64
7		Puncak Rajawali	1.597	98.11	62
8		Puncak Sejati	1.497	100	62
9	<i>Basecamp Bowongso</i>	Puncak Buntu	1.03	100	61
10		Puncak Kawah	1.255	100	67
11		Puncak Rajawali	1.329	97.87	65
12		Puncak Sejati	1.733	98.49	64
13	<i>Basecamp Butuh</i>	Puncak Buntu	1.61	99.16	65
14		Puncak Kawah	1.3	98.99	65
15		Puncak Rajawali	1.225	97.78	63
16		Puncak Sejati	1.125	100	71
17	<i>Basecamp Cepit</i>	Puncak Buntu	1.706	98.51	66
18		Puncak Kawah	1.396	97.50	65
19		Puncak Rajawali	1.344	98.14	65
20		Puncak Sejati	1.296	97.78	63
21	<i>Basecamp Garung</i>	Puncak Buntu	1.256	100	64
22		Puncak Kawah	1.566	100	64
23		Puncak Rajawali	1.641	98.77	69
24		Puncak Sejati	2.044	98.64	65
25	<i>Basecamp Mangli</i>	Puncak Buntu	1.701	99.54	66
26		Puncak Kawah	1.392	99.44	65
27		Puncak Rajawali	1.317	98.11	67
28		Puncak Sejati	1.217	100	64
29	<i>Basecamp Sipetung</i>	Puncak Buntu	0.971	100	69
30		Puncak Kawah	1.281	100	64
31		Puncak Rajawali	1.356	99.37	63
32		Puncak Sejati	1.76	98.64	62
Rata-rata				99.006	64.813

Pada Tabel 4.19 adalah hasil perhitungan akurasi menggunakan persamaan (3.3) dan waktu pemrosesan pada setiap hasil rekomendasi rute pendakian gunung dari setiap *Basecamp* ke setiap puncak di gunung Sumbing. Yang mana hasil rata-

rata akurasi dalam pengujian di graf gunung Sumbing hasilnya 99.006% dan rata-rata waktu pemrosesannya selama 64.813 ms.

Tabel 4.20 Hasil Pengujian di Gunung Ungaran

No	Lokasi Awal	Lokasi Akhir	Bobot	Akurasi	Waktu
1	Basecamp Mawar	Puncak Botak	1.27576	98.75	66
2		Puncak Ungaran	1.21367	99.28	64
3	Basecamp Medini	Puncak Botak	2.1855	99	67
4		Puncak Ungaran	2.12341	98.89	65
Rata-rata				98.98	65.5

Pada Tabel 4.20 adalah hasil perhitungan akurasi menggunakan persamaan (3.3) dan waktu pemrosesan pada setiap hasil rekomendasi rute pendakian gunung dari setiap *Basecamp* ke setiap puncak di gunung Ungaran. Yang mana hasil rata-rata akurasi dalam pengujian di graf gunung Ungaran hasilnya 98.98% dan rata-rata waktu pemrosesannya selama 65.5 ms.

Dari hasil perhitungan rata-rata kompleksitas graf, akurasi dan waktu pemrosesan pada masing-masing gunung yang telah dilakukan sebelumnya didapatkan ringkasan hasil yang ada pada Tabel 4.21.

Tabel 4.21 Hasil Perhitungan Kompleksitas Graf, Akurasi dan Waktu Pemrosesan

No	Gunung	Kompleksitas Graph	Akurasi	Waktu Pemrosesan
1	Slamet	1	99.836	62.286
2	Ciremai	2	99.532	64.778
3	Sumbing	3	99.006	64.813
4	Ungaran	4	98.980	65.5
5	Butak	4	98.853	66
6	Arjuno-Welirang	6	98.793	66.7
7	Penanggungan	6	95.775	66.778
8	Bromo	8	92.328	67
Rata-rata		4.25	97.888	65.481875

Pada Tabel 4.21 ini adalah ringkasan dari hasil hasil perhitungan rata-rata kompleksitas graf, akurasi dan waktu pemrosesan pada masing-masing gunung yang dilakukan dalam proses pengujian.

#### 4.5 Evaluasi Ketergunaan Sistem dengan *System Usability Scale* (SUS)

Selain melakukan evaluasi terhadap kinerja dan akurasi sistem secara teknis, penelitian ini juga mengukur aspek pengalaman pengguna (*user experience*) melalui evaluasi ketergunaan (*usability*). Pengujian ini bertujuan untuk memperoleh data kuantitatif mengenai persepsi pengguna terhadap kemudahan penggunaan, efisiensi, dan kepuasan saat berinteraksi dengan sistem rekomendasi "Treknesia". Sebagaimana telah diuraikan pada BAB III, metode yang digunakan adalah *System Usability Scale* (SUS), yang diujikan kepada 30 responden yang merupakan target pengguna sistem, yaitu para pegiat pendakian gunung.

Setelah responden melakukan skenario tugas menggunakan sistem, data dari kuesioner SUS yang disebar diolah untuk menghitung skor ketergunaan. Setiap skor dari 30 responden dihitung menggunakan prosedur standar SUS, yang kemudian dianalisis untuk mendapatkan nilai rata-rata, standar deviasi, skala penerimaan, skala peringkat dan skala adjektif. Hasil perhitungan statistik dari data yang terkumpul disajikan pada Tabel di dalam lampiran dan berikut ringkasan hasil perhitungannya pada Tabel 4.22.

Tabel 4.22 Ringkasan Hasil Perhitungan *System Usability Scale* (SUS)

METRIK	NILAI
Jumlah Responden	30
Rata-rata Skor SUS	89.916
Standar Deviasi	7.67
Skala Penerimaan	Dapat Diterima
Skala Peringkat	B
Skala Adjektif	Sangat Bagus Sekali

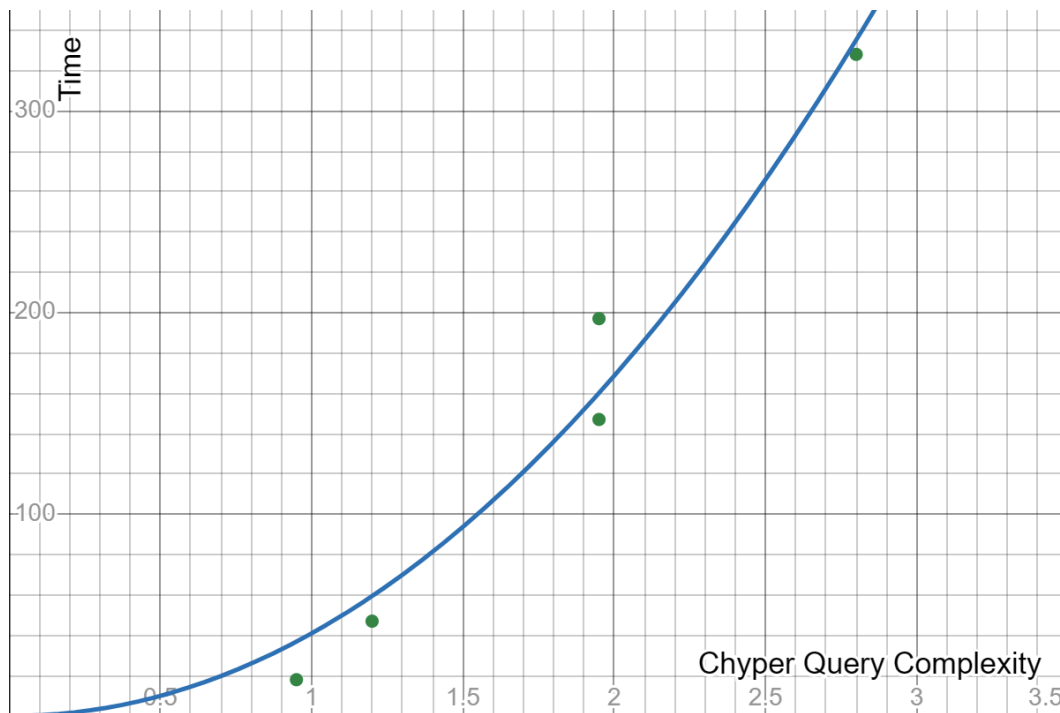
Berdasarkan hasil analisis pada Tabel 4.22, sistem rekomendasi "Treknesia" memperoleh skor SUS rata-rata sebesar 89.916. Skor ini merupakan indikator utama dari tingkat ketergunaan yang dipersepsikan oleh pengguna. Berdasarkan pembahasan interpretasi skor yang telah dijelaskan di dalam pembahasan *system usability scale* (SUS) pada Tabel 3.16 sampai dengan 3.18. Dengan skor 89.916, sistem ini masuk ke dalam kategori "Dapat Diterima" dengan peringkat "B" dan dikategorikan "Sangat Bagus Sekali". Hal ini menunjukkan bahwa pengguna secara umum merasa sistem sangat mudah digunakan, efisien, dan memuaskan untuk digunakan dalam mencari rekomendasi rute pendakian.

Nilai standar deviasi sebesar 7.67 mengindikasikan tingkat sebaran skor di antara 30 responden. Nilai ini tergolong rendah, yang menyiratkan bahwa persepsi pengguna terhadap sistem cenderung konsisten. Artinya, sebagian besar responden memiliki pengalaman positif yang serupa dan tidak terdapat perbedaan pendapat yang drastis mengenai kemudahan penggunaan sistem.

#### **4.6 Pembahasan**

Penelitian ini berhasil mengimplementasikan sistem rekomendasi rute pendakian gunung "Treknesia" menggunakan *graph database* Neo4j, yang memodelkan jalur sebagai *Weighted Directed Acyclic Graph* (WDAG). Dengan metode *Analytic Hierarchy Process* (AHP) untuk pembobotan preferensi pengguna dan algoritma Dijkstra untuk pencarian rute, sistem ini terbukti mampu menyediakan rekomendasi yang dipersonalisasi, sesuai dengan tujuan utama penelitian.

Salah satu aspek krusial dalam implementasi sistem berbasis *graph database* adalah kinerja *query* yang digunakan untuk mengakses dan memanipulasi data. Dalam penelitian ini, analisis dilakukan terhadap hubungan antara kompleksitas *cypher query* dengan waktu pemrosesan yang dibutuhkan.



Gambar 4.16 Grafik Hubungan Kompleksitas *Chyper Query* dan Waktu Pemrosesan

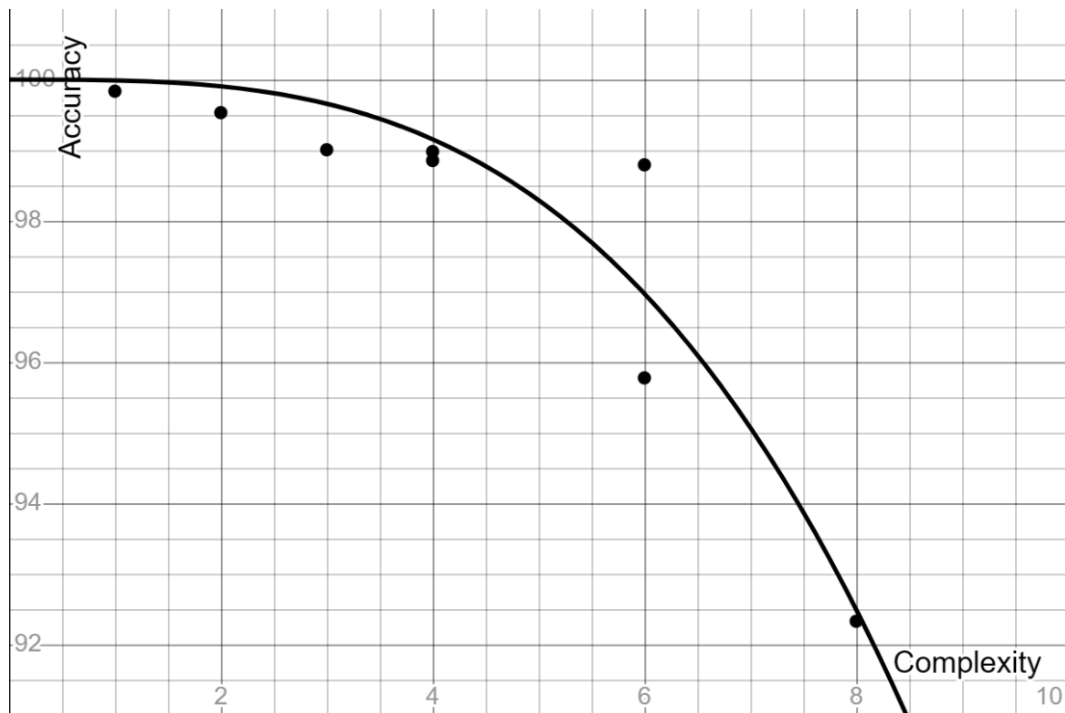
Sebagaimana ditunjukkan pada Gambar 4.16 dan didukung oleh data pada Tabel 4.11, terdapat hubungan yang kuat antara kompleksitas *query Cypher* dan waktu pemrosesannya. Grafik tersebut menghasilkan persamaan  $y = 40.95491x^{2.04145}$  dengan koefisien determinasi ( $R^2$ ) sebesar 0.9662. Nilai  $R^2$  yang mendekati 1 ini mengindikasikan bahwa model persamaan tersebut sangat baik dalam menjelaskan variasi waktu pemrosesan berdasarkan kompleksitas *query*. Secara spesifik, grafik menunjukkan bahwa peningkatan kompleksitas *query* cenderung diikuti oleh peningkatan waktu pemrosesan secara non-linear.

Temuan ini sejalan dengan teori umum mengenai kinerja *database query*. *Query* yang lebih kompleks, misalnya yang melibatkan lebih banyak klausa *MATCH*, *WHERE*, penggunaan fungsi agregasi yang rumit, atau traversal graf yang lebih dalam dan luas, secara inheren membutuhkan lebih banyak sumber daya komputasi. *Neo4j*, meskipun dioptimalkan untuk operasi graf, tetap tunduk pada prinsip ini. Setiap elemen dalam *chyper query* (seperti *node pattern*, *relationship pattern*, filter, dan operasi lainnya) berkontribusi pada total beban kerja yang harus dieksekusi oleh *engine database*. Peningkatan non-linear ini dapat diatribusikan pada sifat kombinatorial dari beberapa operasi graf, di mana jumlah jalur atau pola yang mungkin untuk dievaluasi dapat meningkat secara eksponensial dengan bertambahnya kedalaman atau keluasan pencarian.

Implikasi dari temuan ini sangat signifikan bagi pengembangan sistem berbasis *graph database* seperti *Neo4j*. Efisiensi dalam perancangan *query Cypher* menjadi faktor kunci untuk menjaga performa sistem, terutama ketika berhadapan dengan *dataset* yang besar atau skenario penggunaan dengan tuntutan *real-time*. Pengembang perlu cermat dalam menyusun *query*, memanfaatkan *indexing* dengan tepat, dan memahami bagaimana *Neo4j* mengeksekusi *query* (misalnya melalui *EXPLAIN* atau *PROFILE*) untuk mengidentifikasi dan mengatasi potensi *bottleneck*.

Selain kompleksitas *query*, kompleksitas struktur graf itu sendiri (jumlah *node* dan *edge*) juga dievaluasi pengaruhnya terhadap akurasi rekomendasi dan waktu pemrosesan. Data untuk analisis ini diringkas dalam Tabel 4.21.





Gambar 4.17 Grafik Hubungan Kompleksitas Graf dan Akurasi

Gambar 4.17 menyajikan hubungan antara kompleksitas graf (yang diukur dari jumlah *node* dan *edge* pada data jalur pendakian setiap gunung) dengan rata-rata akurasi rekomendasi, yang diukur menggunakan F1-Score. Grafik ini menggunakan persamaan  $y = 100 - 0.0107062x^{3.15111}$  dan menghasilkan koefisien determinasi ( $R^2$ ) sebesar 0.8808. Nilai  $R^2$  ini menunjukkan bahwa model tersebut cukup baik dalam menggambarkan hubungan antara kedua variabel.

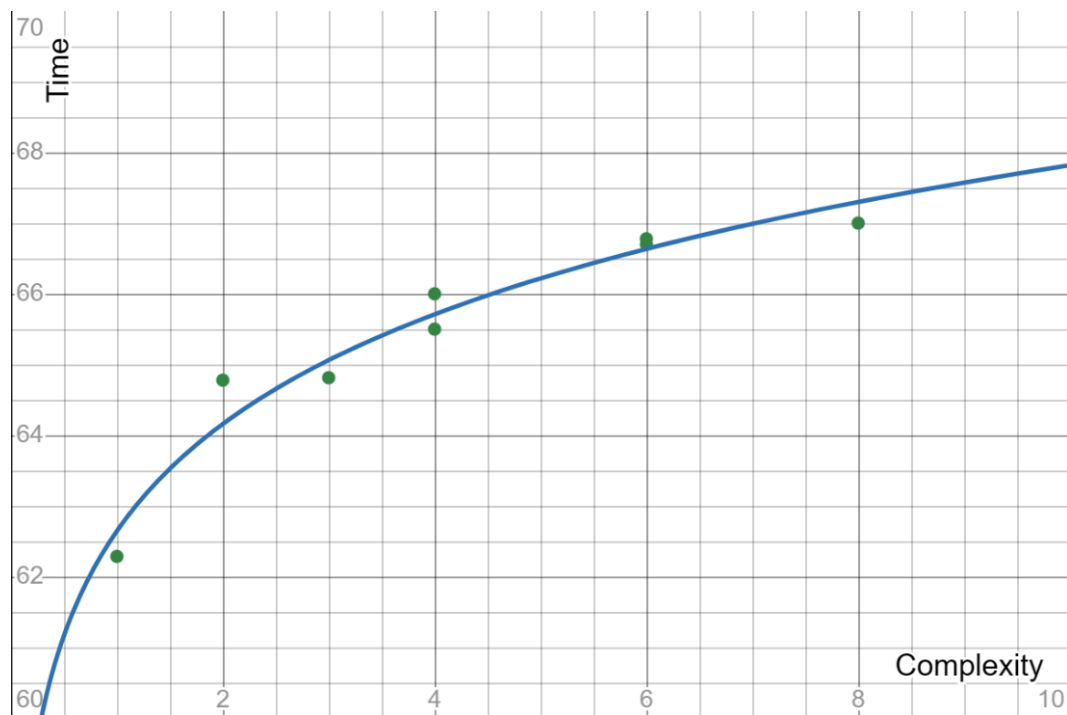
Secara menarik, hasil analisis pada Gambar 4.18 menunjukkan bahwa dengan meningkatnya kompleksitas graf, rata-rata akurasi (F1-Score) cenderung mengalami sedikit penurunan. Pada pandangan pertama, hal ini mungkin tampak berlawanan dengan intuisi, karena graf yang lebih kompleks berarti representasi jalur yang lebih detail dan lebih banyak pilihan rute. Namun, penurunan akurasi ini dapat diinterpretasikan dari beberapa sudut pandang:

Dengan lebih banyaknya alternatif jalur pada graf yang kompleks, definisi "rute terbaik" menurut preferensi pengguna yang diukur melalui AHP mungkin menjadi lebih sulit untuk dicocokkan secara presisi oleh sistem. Algoritma Dijkstra akan selalu menemukan jalur dengan bobot terendah berdasarkan kriteria AHP, namun persepsi subjektif pengguna mengenai "rute ideal" bisa jadi lebih nuansif dan tidak sepenuhnya tertangkap oleh bobot AHP, terutama ketika banyak jalur memiliki skor bobot yang berdekatan.

F1-Score adalah rata-rata harmonik dari precision dan recall (seperti dijelaskan pada sub-bab 3.4.2 dan Persamaan 3.3 & 3.4). Pada graf yang lebih kompleks, sistem mungkin merekomendasikan rute yang secara teknis terbaik menurut bobot, tetapi mungkin ada rute lain yang juga dianggap relevan oleh pengguna (mempengaruhi recall) atau sistem merekomendasikan rute yang tidak sepenuhnya sesuai dengan harapan pengguna (mempengaruhi precision). Peningkatan jumlah kemungkinan rute dapat meningkatkan potensi False Positives (FP) atau False Negatives (FN) jika pembobotan AHP tidak secara sempurna mencerminkan semua nuansa preferensi pengguna. Meskipun pengujian dilakukan pada beberapa gunung dengan kompleksitas graf yang berbeda, karakteristik spesifik setiap gunung dan preferensi pengguna yang digunakan dalam pengujian dapat memengaruhi hasil F1-Score.

Temuan ini menggarisbawahi pentingnya metode pembobotan preferensi (AHP) yang akurat dan mungkin perlunya mekanisme untuk menangani ambiguitas atau memberikan beberapa alternatif rekomendasi teratas pada graf yang sangat kompleks. Hal ini menunjukkan bahwa akurasi rekomendasi rute tidak hanya

bergantung pada algoritma pencarian jalur, tetapi juga pada seberapa baik sistem memahami dan menginterpretasikan preferensi pengguna yang beragam. Oleh karena itu, penyempurnaan pada model preferensi dan kemampuan sistem untuk menawarkan variasi rute dapat meningkatkan kepuasan pengguna secara keseluruhan, terutama pada kondisi graf yang menantang.



Gambar 4.18 Grafik Hubungan Kompleksitas Graf dan Waktu Pemrosesan

Gambar 4.18 mengilustrasikan hubungan antara kompleksitas graf dan rata-rata waktu pemrosesan untuk menghasilkan rekomendasi. Grafik ini menggunakan persamaan  $y = 62.64487x^{0.0344613}$  dan menunjukkan koefisien determinasi ( $R^2$ ) yang sangat tinggi, yaitu 0.9517. Ini menandakan bahwa model persamaan tersebut sangat akurat dalam menggambarkan bahwa waktu pemrosesan sangat dipengaruhi oleh kompleksitas graf.

Dengan meningkatnya jumlah *node* dan *edge* pada graf, jumlah operasi yang harus dilakukan oleh algoritma untuk menjelajahi graf dan menemukan jalur terpendek juga meningkat. Peningkatan ini cenderung bersifat non-linear, seperti yang juga tercermin dalam grafik. Implikasi dari temuan ini adalah pada aspek skalabilitas sistem. Untuk gunung-gunung dengan jaringan jalur yang sangat rapat dan kompleks, waktu yang dibutuhkan untuk menghasilkan rekomendasi dapat meningkat secara signifikan. Hal ini perlu menjadi pertimbangan dalam desain sistem, terutama jika sistem diharapkan dapat memberikan respons cepat kepada pengguna.

Dari analisis kinerja ini, dapat disimpulkan bahwa optimasi kompleksitas, baik pada struktur graf maupun pada *query*, menjadi sangat krusial untuk menjaga keseimbangan antara akurasi rekomendasi dan efisiensi waktu pemrosesan sistem.

Di samping evaluasi teknis, keberhasilan sistem juga diukur dari persepsi pengguna akhir. Melalui metode *System Usability Scale* (SUS) yang melibatkan 30 responden pegiat pendakian gunung, sistem "Treknesia" memperoleh skor rata-rata 89.916 dengan standar deviasi 7.67. Berdasarkan skala interpretasi, skor ini termasuk dalam kategori "Dapat Diterima", dengan peringkat "B", dan skala adjektif "Sangat Bagus Sekali".

Hasil ini mengindikasikan bahwa pengguna mempersepsikan sistem sangat mudah digunakan, tidak rumit, dan mereka merasa percaya diri saat mengoperasikannya. Standar deviasi yang rendah menunjukkan konsistensi pengalaman positif di antara responden. Dengan demikian, dapat disimpulkan bahwa sistem ini tidak hanya berhasil secara fungsional dalam merekomendasikan

rute yang optimal, tetapi juga berhasil memberikan pengalaman pengguna yang sangat memuaskan, yang secara langsung menjawab tujuan penelitian untuk "meningkatkan kenyamanan dan pengalaman pendakian".

Implementasi teknologi dalam penelitian ini terbukti efektif. Penggunaan *graph database* Neo4j sangat sesuai untuk memodelkan jaringan jalur pendakian yang kompleks secara alami. Metode AHP berhasil menerjemahkan preferensi subjektif pengguna menjadi bobot kuantitatif yang andal, yang menjadi dasar personalisasi rekomendasi. Algoritma Dijkstra juga secara efisien menemukan rute dengan bobot terkecil berdasarkan kriteria tersebut.

Secara keseluruhan, pembahasan ini menjawab rumusan masalah yang diajukan di awal penelitian. Sistem yang dikembangkan berhasil menyediakan rekomendasi rute pendakian yang disesuaikan dengan karakteristik pendaki, dengan mengintegrasikan analisis mendalam terhadap faktor-faktor krusial seperti jarak, elevasi, kemiringan, dan waktu tempuh untuk meningkatkan kenyamanan dan kualitas pengalaman pendakian.

Keberhasilan sistem "Treknesia" dalam menyediakan rekomendasi rute yang optimal, serta memberikan pengalaman pengguna yang sangat memuaskan, tidak hanya memiliki implikasi teknis, tetapi juga dapat dimaknai dari perspektif yang lebih dalam. Jika dihubungkan dengan nilai-nilai keislaman, upaya menghadirkan kemudahan, keamanan, dan petunjuk dalam sebuah perjalanan memiliki landasan yang kuat, sebagaimana diisyaratkan dalam firman Allah SWT dalam QS. Saba': 18:

{ وجعلنا بينهم } بين سبأ وهم باليمن { وبين القرى التي باركنا فيها } بالماء والشجر وهي قرى الشام التي يسرون إليها للتجارة { قرى ظاهرة } متواصلة من اليمن إلى الشام { وقدرنا فيها السير } بحيث يقبلون في واحدة ويبيتون في أخرى إلى انتهاء سفرهم ولا يحتاجون فيه إلى حمل زاد وماء أي وقلنا { سيروا فيها ليالي وأياما آمنين } لا تخافون في ليل ولا في نهار

*“(Dan Kami jadikan antara mereka) yakni penduduk negeri Saba yang berada di Yaman (dan antara negeri-negeri yang Kami limpahkan berkat kepadanya) dengan melimpahnya air dan banyaknya pohon-pohonan, yang dimaksud adalah kampung-kampung negeri Syam tempat lalu mereka untuk tujuan berdagang (beberapa negeri yang berdekatan) mulai dari Yaman sampai ke Syam (dan Kami tetapkan antara negeri-negeri itu jarak-jarak perjalanan) hingga mereka dapat beristirahat pada suatu tempat, kemudian menginap pada tempat lainnya sampai pada akhir perjalanan mereka. Di dalam perjalanan, mereka tak perlu lagi membawa bekal dan air. Kami katakan, ("Berjalanlah kalian di kota-kota itu pada malam dan siang hari dengan aman") tanpa merasa takut lagi, baik kalian melakukan perjalanan pada malam hari maupun pada siang hari”.* (Tafsir Al-Jalalain QS. Saba': 18) (Mahally dkk., 2010)

Ayat ini secara indah melukiskan sebuah koridor perjalanan yang dirancang dengan sempurna, sebuah perwujudan nyata dari rahmat Allah SWT. Rute perjalanan tersebut terbentang dari negeri Saba', yang lokasinya berada di sekitar Ma'rib di Yaman saat ini, menuju negeri Syam yang kini mencakup wilayah Suriah, Lebanon, Palestina, dan Yordania. Perjalanan yang kini difasilitasi oleh infrastruktur canggih, dimulai dengan menyusuri jaringan jalan raya yang membentang ke utara. Rute ini secara bertahap menyingkapkan bentang alam yang menakjubkan, seperti keindahan puncak Gunung Sauda' di Arab Saudi yang sejuk, sebuah kontras dramatis dari gurun di bawahnya. Infrastruktur modern ini, dengan kota-kota dan stasiun layanan yang tersebar secara berurutan di sepanjang jalan tol yang melintasi gurun, dapat diinterpretasikan sebagai manifestasi kontemporer dari konsep negeri-negeri yang berdekatan, yang mentransformasi perjalanan berat menjadi sebuah pengalaman yang indah. Perjalanan yang melalui kota Madinah, di

mana terdapat tempat singgah yang memberikan ketenangan yaitu Masjid Nabawi dan pemandangan ke arah Gunung Uhud yang historis menjadi titik hening di tengah perjalanan.

Melanjutkan perjalanan ke utara, akan melintasi serangkaian situs yang menjadi saksi bisu peradaban besar. Situs kuno Hegra di Mada'in Salih dengan makam-makam monumental kaum Nabath yang terukir di batu pasir, serta Benteng Tabuk yang strategis, kini menjadi destinasi bersejarah yang dapat diakses dengan mudah, tidak lagi sebagai pos pertahanan vital melainkan sebagai jendela menuju masa lalu. Memasuki Yordania, lanskap dramatis Wadi Rum menawarkan keagungan alam yang magis, sementara puncak perjalanan penuh sejarah ini terwujud di kota kuno Petra, sebuah mahakarya arsitektur Nabath yang tersembunyi di balik tebing batu. Perjalanan yang juga melewati Benteng Kerak yang megah, akhirnya mencapai titik akhir di Amman Citadel, di mana lapisan-lapisan sejarah dari berbagai era bertemu.

Dengan demikian, pengalaman perjalanan di rute Saba'-Syam ini menjadi sebuah pembuktian. Janji perjalanan yang aman pada malam dan siang hari dalam QS. Saba': 18 menemukan keselarasan dalam kenyamanan dan keamanan yang ada saat ini. Kemudahan yang dirasakan tidak hanya mengurangi lelah, tetapi juga memungkinkan setiap musafir untuk meresapi jejak sejarah dan keagungan alam dengan rasa takjub yang mendalam. Rahmat Tuhan yang dahulu berwujud sumber air dan jarak antar desa atau kota yang terukur, kini terwujud dalam bentuk jalan raya yang terawat dan kota-kota yang terhubung, menjadikan perjalanan ini sebuah petualangan yang aman melintasi koridor waktu dan peradaban.

## BAB V

### KESIMPULAN DAN SARAN

#### 5.1 Kesimpulan

Penelitian ini berhasil mengimplementasikan *graph database* dengan Neo4j untuk mengembangkan sistem rekomendasi rute pendakian gunung yang efektif dan dapat dipersonalisasi. Sistem ini mampu memberikan rekomendasi rute terbaik yang disesuaikan dengan preferensi pengguna mencakup jarak, selisih elevasi, kemiringan, dan estimasi waktu tempuh dengan memanfaatkan pemodelan *Weighted Directed Acyclic Graph* (WDAG), pembobotan kriteria menggunakan *Analytic Hierarchy Process* (AHP), dan pencarian rute menggunakan algoritma Dijkstra. Keberhasilan ini secara langsung menjawab tujuan penelitian untuk meningkatkan kenyamanan, pengalaman, dan keselamatan pendaki melalui rekomendasi yang terpersonalisasi.

Evaluasi kinerja sistem menunjukkan hasil yang sangat positif. Secara keseluruhan, sistem mencapai tingkat akurasi yang tinggi dengan rata-rata *F1-Score* sebesar 97.888% dan waktu pemrosesan yang efisien, dengan rata-rata 65.481 ms pada pengujian di delapan gunung. Selain itu, evaluasi ketergunaan sistem "Treknesia" menggunakan *System Usability Scale* (SUS) menghasilkan skor rata-rata 89.916, yang termasuk dalam kategori "Sangat Bagus Sekali," mengonfirmasi bahwa sistem tidak hanya akurat secara teknis tetapi juga sangat mudah digunakan dan memuaskan bagi pengguna.



Analisis lebih lanjut mengungkap hubungan signifikan antara kompleksitas dengan kinerja sistem. Ditemukan adanya hubungan non-linear yang kuat antara kompleksitas *query Cypher* dan waktu pemrosesan, di mana *query* yang lebih kompleks memerlukan waktu eksekusi yang lebih lama. Begitu pula dengan kompleksitas struktur graf (jumlah *node* dan *edge*), yang terbukti berpengaruh signifikan terhadap peningkatan waktu pemrosesan dan sedikit penurunan pada akurasi *F1-Score*. Temuan ini menegaskan bahwa implementasi *graph database* dengan metode yang diusulkan sangat efektif, namun optimasi pada kompleksitas *query* dan struktur graf menjadi faktor krusial untuk menjaga keseimbangan antara akurasi dan efisiensi sistem.

## 5.2 Saran

Berdasarkan temuan dan kesimpulan yang telah diuraikan dari penelitian ini, terdapat beberapa saran yang dapat menjadi pertimbangan untuk pengembangan sistem lebih lanjut, penelitian di masa mendatang, serta pemanfaatan praktis oleh para pemangku kepentingan. Saran-saran ini dirumuskan untuk mengatasi keterbatasan yang ada dan memaksimalkan potensi dari sistem yang telah dibangun.

### 5.2.1 Untuk Pengembangan Sistem dan Penelitian Selanjutnya

- a. Optimasi Performa dan Skalabilitas: Mengingat temuan bahwa kompleksitas *query Cypher* dan kompleksitas graf berpengaruh signifikan terhadap waktu pemrosesan, penelitian selanjutnya disarankan untuk fokus pada optimasi performa.

- b. Pengayaan Atribut dan Kriteria Dinamis: Untuk meningkatkan akurasi dan personalisasi rekomendasi, sistem dapat dikembangkan dengan menambahkan atribut jalur yang lebih komprehensif, seperti jenis medan (misalnya, tanah, bebatuan, lumpur), keberadaan sumber air, titik rawan bahaya, *shelter*, dan *point of interest* (POI) lainnya. Selain itu, dapat dipertimbangkan integrasi faktor dinamis seperti tingkat kebugaran dan pengalaman pendaki, komposisi grup, serta kondisi cuaca secara *real-time* untuk menghasilkan rekomendasi yang lebih adaptif.
- c. Ekspansi Dataset dan Eksplorasi Algoritma Alternatif: Cakupan basis data perlu diperluas dengan menambahkan data jalur pendakian dari lebih banyak gunung di Indonesia dan bahkan mancanegara untuk meningkatkan manfaat sistem. Terkait algoritma, penelitian mendatang dapat mengeksplorasi penggunaan algoritma pencarian lain seperti A\* (A-Star) yang menggunakan heuristik untuk pencarian lebih cepat, atau algoritma *Yen's K-shortest paths* untuk menyajikan beberapa alternatif rute terbaik kepada pengguna, tidak hanya satu.
- d. Pengembangan Platform *Mobile* dan Peningkatan Antarmuka: Mengingat target pengguna adalah para pendaki, pengembangan sistem dalam bentuk aplikasi *mobile* (*Android* atau *iOS*) sangat disarankan untuk kemudahan akses dan penggunaan di lapangan, terutama dengan fungsionalitas *offline maps*. Peningkatan pada antarmuka pengguna (UI/UX) juga perlu dilakukan agar lebih intuitif dan interaktif, misalnya dengan visualisasi peta 3D atau profil elevasi rute yang lebih dinamis.

### **5.2.2 Bagi Pengguna (Pendaki)**

Sistem ini dapat dimanfaatkan sebagai alat bantu yang sangat informatif untuk merencanakan pendakian, khususnya dalam membandingkan dan memilih rute yang paling sesuai dengan preferensi, kemampuan fisik, dan alokasi waktu. Namun, pengguna dianjurkan untuk tidak menjadikan sistem ini sebagai satu-satunya sumber informasi. Sangat penting untuk tetap melengkapi perencanaan dengan informasi dari sumber terpercaya lainnya, seperti dari pengelola taman nasional, komunitas pendaki, serta melakukan persiapan fisik, mental, dan logistik yang matang untuk menjamin keselamatan dan kenyamanan selama pendakian.

### **5.2.3 Bagi Pengelola Kawasan Pendakian dan Instansi Terkait:**

Data dan model yang dihasilkan dari penelitian ini berpotensi menjadi masukan berharga bagi pengelola kawasan pendakian (misalnya, Taman Nasional, Perhutani, atau pemerintah daerah) dan instansi terkait seperti Badan Informasi Geospasial (BIG). Informasi ini dapat dimanfaatkan untuk manajemen jalur pendakian yang lebih baik, penyediaan informasi yang akurat dan terstandarisasi kepada calon pendaki, serta mendukung upaya peningkatan keselamatan dalam aktivitas wisata minat khusus. Di samping itu, disarankan adanya kolaborasi antara pengembang sistem dengan pihak pengelola untuk memastikan validitas dan kemitakhiran data, sehingga dapat tercipta sebuah ekosistem informasi pendakian yang komprehensif dan bermanfaat bagi masyarakat luas.

## DAFTAR PUSTAKA

- Ahmad, O., Jalil, O. A., Nazir, U., & Taj, M. (2023). Learning Adjacency Matrix For Dynamic Graph Neural Network. *Computer Vision and Graphics Lab, LUMS, 1*.
- Aisyah, K. N., Sungkono, K. R., & Sarno, R. (2020). A new similarity method based on weighted-linear temporal logic tree and weighted directed acyclic graph for graph-based business process models. *International Journal of Intelligent Engineering and Systems, 13*(5), 356–367.
- Ardani, M. S. H., Yaqin, M. A., & Suhartono, M. H. (2019). Implementasi Graph Database untuk Menentukan Rute Perjalanan Transportasi Umum Clustering View project Fraud Detection View project. *Seminar Nasional Inovasi dan Aplikasi Teknologi di Industri, February*, 36–42.
- Dai, R., & Liu, G. (2023). Graph Database. *Graph Database and Graph Computing for Power System Analysis, 17–39*.  
<https://doi.org/10.1002/9781119903895.CH2>
- Darko, A., Chan, A. P. C., Ameyaw, E. E., Owusu, E. K., Pärn, E., & Edwards, D. J. (2019). Review of application of analytic hierarchy process (AHP) in construction. *International Journal of Construction Management, 19*(5), 436–452.
- Fritchey, G. (2022). Graph Databases. *SQL Server 2022 Query Performance Tuning, 597–617*. [https://doi.org/10.1007/978-1-4842-8891-7\\_20](https://doi.org/10.1007/978-1-4842-8891-7_20)
- Gumilang, M. R. (2022). Sistem Informasi Jalur Pendakian Gunung di Indonesia Berbasis Web. *Prosiding Seminar Nasional Hasil Penelitian Dan Pengabdian Kepada Masyarakat (Snhp), 3*(352031004), 115–118.
- Gur, H. (2009). Trigonometry Learning. *New Horizons in Education, 57*(1), 67–80.
- Güting, R. H. (1994). GraphDB - Modeling and querying graphs in databases. *Proceedings of the International Conference on Very Large Data Bases, 297–297*.
- Holly, M. (2024). *Adjacency Matrix for the Graph Exponential*.  
<https://doi.org/10.20944/PREPRINTS202405.1667.V1>
- Inayah, A. M., Cintya Resti, N., & Kediri, I. (2023). Analisa Perbandingan Algoritma Floyd-Warshall Dan Algoritma Dijkstra untuk Penentuan Rute Terdekat. *Jurnal Ilmiah Matematika Realistik (JI-MR), 4*(2), 146–155.
- Irawan, J., Nuryani, S., & Pandin, R. M. (2023). Penentuan Jalur Evakuasi Terbaik Bencana Gunung Tangkuban Perahu Menggunakan Algoritma Dijkstra. *CICES (Cyberpreneurship Innovative and Creative Exact and Social Science), 9*(2), 174–184.

- Javaid, M. A. (2013). Understanding Dijkstra's Algorithm. *SSRN Electronic Journal*. <https://doi.org/10.2139/SSRN.2340905>
- Langmuir, Eric. (2013). *Mountaincraft and leadership: a handbook for mountaineers and hillwalking leaders in the British Isles* / Eric Langmuir. Mountain Training England and Mountain Training Scotland. [https://books.google.com/books/about/Mountaincraft\\_and\\_Leadership.html?id=k4kllQEACAAJ](https://books.google.com/books/about/Mountaincraft_and_Leadership.html?id=k4kllQEACAAJ)
- Mahally, S. J. Al, Suyuthi, S. J. A., Fara, A. El, & Hakim, L. (2010). *Terjemah Tafsir Jalalain | Lengkap Dengan Asbabun Nuzul & Nasikh Mansukh*. Bariklana.
- Muharrom, M. (2020). Implementasi Algoritma Dijkstra Dalam Penentuan Jalur Terpendek Studi Kasus Jarak Tempat Kuliah Terdekat. *Indonesian Journal of Business Intelligence (IJUBI)*, 3(1), 25–30.
- Neiva, M. B., & Bruno, O. M. (2023). Exploring ordered patterns in the adjacency matrix for improving machine learning on complex networks. *Physica D: Nonlinear Phenomena*, 626, 129086–129086. <https://doi.org/10.1016/J.PHYSA.2023.129086>
- Nilsson, A., Bonander, C., Strömberg, U., & Björk, J. (2021). A directed acyclic graph for interactions. *International Journal of Epidemiology*, 50(2), 613–619.
- Nurhasanah, Y. I., Umaroh, S., & Al Ghoniyyah, N. H. (2021). Pencarian Rute Optimal Dengan Metode Steepest Ascent Hill Climbing Untuk Tempat Wisata Di Bandung Menggunakan Android. *Infotek: Jurnal Informatika dan Teknologi*, 4(2), 113–124.
- Rose, R. A., Cosgrove, J. A., & Lee, B. R. (2024). Directed Acyclic Graphs in Social Work Research and Evaluation: A Primer. *Journal of the Society for Social Work and Research*, 15(2), 391–415.
- Saaty, T. L. (2002). Decision making with the Analytic Hierarchy Process. *Scientia Iranica*, 9(3), 215–229. <https://doi.org/10.1504/ijssci.2008.017590>
- Sanuri, R., Muzakkar, M., Ismail, A., Rahma Yogyakarta, E., Studi Informatika, P., & El Rahma Yogyakarta, S. (2024). Transformasi Database Graph Dokumen Standar Mutu SPMI STMIK El Rahma Yogyakarta. *FAHMA: Jurnal Informatika Komputer, Bisnis dan Manajemen*, 22(2), 19–27.
- Scarf, P. (2007). Route choice in mountain navigation, Naismith's rule, and the equivalence of distance and climb. *Journal of Sports Sciences*, 25(6), 719–726. <https://doi.org/10.1080/02640410600874906>
- Setiawan, S. P. G., Sujaini, H., & Irwansyah, M. A. (2020). Sistem Informasi Objek Wisata dengan Algoritma Dijkstra untuk Rute Terdekat dan Metode Analytical Hierarchy Process (AHP) untuk Rekomendasi. (Studi Kasus Kabupaten Bengkayang). *JUSTIN (Jurnal Sistem dan Teknologi Informasi)*, 8(2), 191–198.

- Subali, M. A. P., & Rochimah, S. (2018). A new model for measuring the complexity of SQL commands. *Proceedings of 2018 10th International Conference on Information Technology and Electrical Engineering: Smart Technology for Better Society, ICITEE 2018*, 1–5. <https://doi.org/10.1109/ICITEED.2018.8534782>
- Sumaryo, R. Y., Harsadi, P., & Nugroho, D. (2020). Implementasi Algoritma Dijkstra Dan Metode Haversine Pada Penentuan Jalur Terpendek Pendakian Gunung Merapi Jalur Selo Berbasis Android. *Jurnal Teknologi Informasi dan Komunikasi (TIKOMSiN)*, 8(1), 61–67.
- Sun, J., Duncan, S., Pal, S., & Kong, M. (2023). Directed Acyclic Graph Assisted Method For Estimating Average Treatment Effect. *Journal of Biopharmaceutical Statistics*, 1–20. <https://doi.org/10.1080/10543406.2023.2296047>
- Supriady, A., Trisari Schiff, N., Ramadhani, M., Pasundan, S., & Ramadhani, M. (2022). Aktivitas Pendakian Taman Nasional Gunung Gede Pangrango Climbing Activities of Gunung Gede Pangrango Nasional Park. *Journal of Physical and Outdoor Education*, 4(2), 164–173.
- Tangdisosang, S. P., & Widiyari, I. R. (2022). Sistem Informasi Pendakian Gunung Jawa Timur Berbasis Web Menggunakan Framework Codeigniter. *JIKO (Jurnal Informatika dan Komputer)*, 5(1), 7–11.
- Wirawan, P. W., & Riyanto, D. E. (2017). Kajian Implementasi Graph Database pada Rute Bus Rapid Transit. *Jurnal Nasional Teknologi dan Sistem Informasi*, 3(3), 313–319.
- Yacoub, R., & Axman, D. (2020). Probabilistic Extension of Precision, Recall, and F1 Score for More Thorough Evaluation of Classification Models. *Association for Computational Linguistics*, 79–91.
- Yu, C., Gao, D., Guo, X., Ma, H., Zhao, Y., & Wang, G. (2024). dwMLCS: An Efficient MLCS Algorithm based on Dynamic and Weighted Directed Acyclic Graph. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 1–14. <https://doi.org/10.1109/TCBB.2024.3431558>
- Yunanto, P. W., Nugraheni, M., & Nugraha, N. (2021). Sistem Informasi Penjejak Pendakian Gunung Berbasis Web. *Jurnal Sistem dan Teknologi Informasi Indonesia (SINTESIA)*, 1(1), 13–30.

# LAMPIRAN

## LAMPIRAN

Tabel Perbandingan dan Skala Atribut Jalur Pendakian Oleh Pegiat Pendakian

Jarak dan Selisih Elevasi	Skala	Jarak dan Kemiringan	Skala	Jarak dan Waktu Tempuh	Skala	Selisih Elevasi dan Kemiringan	Skala	Selisih Elevasi dan Waktu Tempuh	Skala	Kemiringan dan Waktu Tempuh	Skala
Jarak	2	Jarak	3	Jarak	4	Selisih Elevasi (Ketinggian)	2	Selisih Elevasi (Ketinggian)	3	Kemiringan	2
Jarak	2	Kemiringan	2	Waktu Tempuh	2	Kemiringan	3	Selisih Elevasi (Ketinggian)	2	Kemiringan	2
Selisih Elevasi (Ketinggian)	6	Kemiringan	7	Waktu Tempuh	9	Kemiringan	6	Waktu Tempuh	8	Waktu Tempuh	8
Selisih Elevasi (Ketinggian)	1	Kemiringan	1	Waktu Tempuh	1	Kemiringan	1	Selisih Elevasi (Ketinggian)	1	Kemiringan	1
Selisih Elevasi (Ketinggian)	5	Kemiringan	7	Waktu Tempuh	6	Kemiringan	5	Waktu Tempuh	5	Kemiringan	5
Selisih Elevasi (Ketinggian)	2	Kemiringan	3	Waktu Tempuh	3	Selisih Elevasi (Ketinggian)	1	Selisih Elevasi (Ketinggian)	2	Kemiringan	2
Selisih Elevasi (Ketinggian)	8	Kemiringan	1	Waktu Tempuh	1	Selisih Elevasi (Ketinggian)	6	Selisih Elevasi (Ketinggian)	1	Kemiringan	1
Jarak	5	Kemiringan	7	Waktu Tempuh	5	Kemiringan	5	Waktu Tempuh	5	Kemiringan	7
Jarak	6	Jarak	6	Jarak	1	Selisih Elevasi (Ketinggian)	1	Selisih Elevasi (Ketinggian)	1	Kemiringan	1
Jarak	9	Kemiringan	9	Waktu Tempuh	9	Kemiringan	1	Selisih Elevasi (Ketinggian)	1	Kemiringan	1
Jarak	2	Kemiringan	3	Waktu Tempuh	3	Kemiringan	3	Waktu Tempuh	3	Kemiringan	3
Selisih Elevasi (Ketinggian)	5	Kemiringan	8	Waktu Tempuh	7	Kemiringan	5	Waktu Tempuh	7	Waktu Tempuh	8



Lanjutan

Jarak dan Selisih Elevasi	Skala	Jarak dan Kemiringan	Skala	Jarak dan Waktu Tempuh	Skala	Selisih Elevasi dan Kemiringan	Skala	Selisih Elevasi dan Waktu Tempuh	Skala	Kemiringan dan Waktu Tempuh	Skala
Selisih Elevasi (Ketinggian)	4	Kemiringan	1	Waktu Tempuh	1	Kemiringan	1	Selisih Elevasi (Ketinggian)	1	Waktu Tempuh	1
Jarak	1	Jarak	3	Waktu Tempuh	8	Kemiringan	1	Waktu Tempuh	8	Waktu Tempuh	8
Jarak	3	Jarak	1	Waktu Tempuh	4	Selisih Elevasi (Ketinggian)	4	Waktu Tempuh	1	Waktu Tempuh	1
Selisih Elevasi (Ketinggian)	7	Kemiringan	6	Jarak	6	Kemiringan	6	Selisih Elevasi (Ketinggian)	6	Kemiringan	6
Jarak	1	Jarak	5	Jarak	1	Selisih Elevasi (Ketinggian)	5	Waktu Tempuh	1	Waktu Tempuh	1
Jarak	6	Kemiringan	6	Waktu Tempuh	6	Selisih Elevasi (Ketinggian)	6	Waktu Tempuh	6	Waktu Tempuh	6
Selisih Elevasi (Ketinggian)	5	Kemiringan	5	Waktu Tempuh	5	Selisih Elevasi (Ketinggian)	5	Selisih Elevasi (Ketinggian)	4	Kemiringan	5
Selisih Elevasi (Ketinggian)	7	Kemiringan	7	Waktu Tempuh	7	Kemiringan	7	Waktu Tempuh	7	Waktu Tempuh	7
Jarak	2	Jarak	2	Jarak	2	Selisih Elevasi (Ketinggian)	2	Selisih Elevasi (Ketinggian)	2	Kemiringan	2
Selisih Elevasi (Ketinggian)	9	Kemiringan	6	Jarak	7	Selisih Elevasi (Ketinggian)	7	Selisih Elevasi (Ketinggian)	8	Kemiringan	7
Jarak	1	Kemiringan	2	Waktu Tempuh	2	Selisih Elevasi (Ketinggian)	2	Waktu Tempuh	4	Kemiringan	3
Jarak	7	Jarak	7	Waktu Tempuh	1	Kemiringan	3	Waktu Tempuh	7	Waktu Tempuh	7
Jarak	1	Jarak	1	Jarak	1	Selisih Elevasi (Ketinggian)	1	Selisih Elevasi (Ketinggian)	1	Waktu Tempuh	1
Jarak	1	Kemiringan	1	Jarak	1	Selisih Elevasi (Ketinggian)	1	Selisih Elevasi (Ketinggian)	1	Kemiringan	2

Lanjutan

Jarak dan Selisih Elevasi	Skala	Jarak dan Kemiringan	Skala	Jarak dan Waktu Tempuh	Skala	Selisih Elevasi dan Kemiringan	Skala	Selisih Elevasi dan Waktu Tempuh	Skala	Kemiringan dan Waktu Tempuh	Skala
Selisih Elevasi (Ketinggian)	5	Kemiringan	5	Waktu Tempuh	6	Selisih Elevasi (Ketinggian)	5	Waktu Tempuh	5	Waktu Tempuh	5
Selisih Elevasi (Ketinggian)	2	Kemiringan	1	Waktu Tempuh	5	Selisih Elevasi (Ketinggian)	1	Waktu Tempuh	1	Waktu Tempuh	4
Jarak	1	Jarak	7	Jarak	5	Selisih Elevasi (Ketinggian)	5	Selisih Elevasi (Ketinggian)	5	Waktu Tempuh	5
Jarak	1	Kemiringan	2	Waktu Tempuh	2	Selisih Elevasi (Ketinggian)	2	Selisih Elevasi (Ketinggian)	3	Waktu Tempuh	2
Jarak	7	Jarak	7	Jarak	7	Selisih Elevasi (Ketinggian)	3	Waktu Tempuh	5	Waktu Tempuh	5
Selisih Elevasi (Ketinggian)	5	Kemiringan	6	Waktu Tempuh	6	Selisih Elevasi (Ketinggian)	4	Selisih Elevasi (Ketinggian)	6	Waktu Tempuh	7
Selisih Elevasi (Ketinggian)	5	Kemiringan	7	Jarak	1	Kemiringan	3	Selisih Elevasi (Ketinggian)	5	Kemiringan	8
Selisih Elevasi (Ketinggian)	7	Kemiringan	7	Jarak	5	Kemiringan	6	Selisih Elevasi (Ketinggian)	7	Kemiringan	7
Selisih Elevasi (Ketinggian)	5	Kemiringan	8	Waktu Tempuh	5	Selisih Elevasi (Ketinggian)	5	Selisih Elevasi (Ketinggian)	6	Waktu Tempuh	8
Jarak	3	Jarak	5	Waktu Tempuh	1	Selisih Elevasi (Ketinggian)	3	Waktu Tempuh	3	Waktu Tempuh	3
Jarak	6	Kemiringan	4	Jarak	5	Kemiringan	6	Waktu Tempuh	6	Kemiringan	5
Jarak	3	Jarak	3	Waktu Tempuh	3	Selisih Elevasi (Ketinggian)	6	Selisih Elevasi (Ketinggian)	3	Waktu Tempuh	4
Selisih Elevasi (Ketinggian)	5	Kemiringan	8	Waktu Tempuh	5	Kemiringan	6	Waktu Tempuh	3	Kemiringan	2
Jarak	1	Kemiringan	9	Waktu Tempuh	8	Selisih Elevasi (Ketinggian)	9	Waktu Tempuh	9	Waktu Tempuh	9

Lanjutan

Jarak dan Selisih Elevasi	Skala	Jarak dan Kemiringan	Skala	Jarak dan Waktu Tempuh	Skala	Selisih Elevasi dan Kemiringan	Skala	Selisih Elevasi dan Waktu Tempuh	Skala	Kemiringan dan Waktu Tempuh	Skala
Jarak	9	Kemiringan	9	Waktu Tempuh	9	Kemiringan	9	Waktu Tempuh	9	Kemiringan	9
Selisih Elevasi (Ketinggian)	8	Kemiringan	8	Waktu Tempuh	8	Kemiringan	8	Waktu Tempuh	8	Waktu Tempuh	8
Jarak	5	Jarak	5	Waktu Tempuh	5	Selisih Elevasi (Ketinggian)	7	Waktu Tempuh	8	Waktu Tempuh	8
Selisih Elevasi (Ketinggian)	7	Kemiringan	9	Jarak	7	Kemiringan	7	Selisih Elevasi (Ketinggian)	6	Kemiringan	7
Selisih Elevasi (Ketinggian)	6	Kemiringan	8	Waktu Tempuh	7	Kemiringan	8	Selisih Elevasi (Ketinggian)	6	Kemiringan	8
Jarak	5	Kemiringan	8	Waktu Tempuh	1	Kemiringan	7	Waktu Tempuh	8	Waktu Tempuh	8
Selisih Elevasi (Ketinggian)	2	Jarak	4	Jarak	4	Selisih Elevasi (Ketinggian)	7	Selisih Elevasi (Ketinggian)	7	Waktu Tempuh	5
Jarak	2	Jarak	3	Jarak	4	Selisih Elevasi (Ketinggian)	2	Selisih Elevasi (Ketinggian)	3	Kemiringan	2
Jarak	2	Jarak	3	Jarak	4	Selisih Elevasi (Ketinggian)	2	Selisih Elevasi (Ketinggian)	3	Kemiringan	2
Selisih Elevasi (Ketinggian)	5	Kemiringan	4	Waktu Tempuh	4	Selisih Elevasi (Ketinggian)	3	Selisih Elevasi (Ketinggian)	4	Kemiringan	4
Jarak	2	Jarak	3	Jarak	4	Selisih Elevasi (Ketinggian)	2	Selisih Elevasi (Ketinggian)	3	Kemiringan	2
Jarak	9	Jarak	9	Jarak	9	Selisih Elevasi (Ketinggian)	9	Selisih Elevasi (Ketinggian)	9	Kemiringan	9
Selisih Elevasi (Ketinggian)	9	Kemiringan	9	Jarak	9	Kemiringan	9	Selisih Elevasi (Ketinggian)	9	Waktu Tempuh	7
Selisih Elevasi (Ketinggian)	3	Kemiringan	5	Waktu Tempuh	5	Selisih Elevasi (Ketinggian)	5	Waktu Tempuh	5	Waktu Tempuh	5

Lanjutan

Jarak dan Selisih Elevasi	Skala	Jarak dan Kemiringan	Skala	Jarak dan Waktu Tempuh	Skala	Selisih Elevasi dan Kemiringan	Skala	Selisih Elevasi dan Waktu Tempuh	Skala	Kemiringan dan Waktu Tempuh	Skala
Jarak	6	Kemiringan	6	Waktu Tempuh	6	Kemiringan	5	Waktu Tempuh	4	Waktu Tempuh	5
Jarak	2	Jarak	2	Jarak	2	Selisih Elevasi (Ketinggian)	2	Selisih Elevasi (Ketinggian)	2	Kemiringan	2
Jarak	7	Jarak	7	Jarak	7	Selisih Elevasi (Ketinggian)	7	Selisih Elevasi (Ketinggian)	7	Kemiringan	7
Selisih Elevasi (Ketinggian)	1	Kemiringan	1	Jarak	1	Kemiringan	3	Selisih Elevasi (Ketinggian)	1	Kemiringan	1
Selisih Elevasi (Ketinggian)	7	Kemiringan	9	Waktu Tempuh	9	Selisih Elevasi (Ketinggian)	9	Waktu Tempuh	9	Waktu Tempuh	9

Tabel Jumlah Pemilihan Atribut dalam Perbandingan dan Skala Atribut Jalur Pendakian Oleh Pegiat Pendakian

Perbandingan	Atribut	Jumlah
Jarak dan Selisih Elevasi	Jarak	32
	Selisih Elevasi	27
Jarak dan Kemiringan	Jarak	20
	Kemiringan	39
Jarak dan Waktu Tempuh	Jarak	23
	Waktu Tempuh	36
Selisih Elevasi dan Kemiringan	Selisih Elevasi	34
	Kemiringan	25
Selisih Elevasi dan Waktu Tempuh	Selisih Elevasi	33
	Waktu Tempuh	26
Kemiringan dan Waktu Tempuh	Kemiringan	31
	Waktu Tempuh	28

Tabel Perbandingan dan Skala Atribut Jalur Pendakian Oleh Pegiat Pendakian

Jarak dan Selisih Elevasi	Skala	Jarak dan Kemiringan	Skala	Jarak dan Waktu Tempuh	Skala	Selisih Elevasi dan Kemiringan	Skala	Selisih Elevasi dan Waktu Tempuh	Skala	Kemiringan dan Waktu Tempuh	Skala
Jarak	2	Jarak	0.333333333	Jarak	0.25	Selisih Elevasi (Ketinggian)	2	Selisih Elevasi (Ketinggian)	3	Kemiringan	2
Jarak	2	Kemiringan	2	Waktu Tempuh	2	Kemiringan	0.333333333	Selisih Elevasi (Ketinggian)	2	Kemiringan	2
Selisih Elevasi (Ketinggian)	0.166666667	Kemiringan	7	Waktu Tempuh	9	Kemiringan	0.166666667	Waktu Tempuh	0.125	Waktu Tempuh	0.125
Selisih Elevasi (Ketinggian)	1	Kemiringan	1	Waktu Tempuh	1	Kemiringan	1	Selisih Elevasi (Ketinggian)	1	Kemiringan	1
Selisih Elevasi (Ketinggian)	0.2	Kemiringan	7	Waktu Tempuh	6	Kemiringan	0.2	Waktu Tempuh	0.2	Kemiringan	5
Selisih Elevasi (Ketinggian)	0.5	Kemiringan	3	Waktu Tempuh	3	Selisih Elevasi (Ketinggian)	1	Selisih Elevasi (Ketinggian)	2	Kemiringan	2
Selisih Elevasi (Ketinggian)	0.125	Kemiringan	1	Waktu Tempuh	1	Selisih Elevasi (Ketinggian)	6	Selisih Elevasi (Ketinggian)	1	Kemiringan	1
Jarak	5	Kemiringan	7	Waktu Tempuh	5	Kemiringan	0.2	Waktu Tempuh	0.2	Kemiringan	7
Jarak	6	Jarak	0.166666667	Jarak	1	Selisih Elevasi (Ketinggian)	1	Selisih Elevasi (Ketinggian)	1	Kemiringan	1
Jarak	9	Kemiringan	9	Waktu Tempuh	9	Kemiringan	1	Selisih Elevasi (Ketinggian)	1	Kemiringan	1
Jarak	2	Kemiringan	3	Waktu Tempuh	3	Kemiringan	0.333333333	Waktu Tempuh	0.333333333	Kemiringan	3
Selisih Elevasi (Ketinggian)	0.2	Kemiringan	8	Waktu Tempuh	7	Kemiringan	0.2	Waktu Tempuh	0.142857143	Waktu Tempuh	0.125

Lanjutan

Jarak dan Selisih Elevasi	Skala	Jarak dan Kemiringan	Skala	Jarak dan Waktu Tempuh	Skala	Selisih Elevasi dan Kemiringan	Skala	Selisih Elevasi dan Waktu Tempuh	Skala	Kemiringan dan Waktu Tempuh	Skala
Selisih Elevasi (Ketinggian)	0.25	Kemiringan	1	Waktu Tempuh	1	Kemiringan	1	Selisih Elevasi (Ketinggian)	1	Waktu Tempuh	1
Jarak	1	Jarak	0.33333333	Waktu Tempuh	8	Kemiringan	1	Waktu Tempuh	0.125	Waktu Tempuh	0.125
Jarak	3	Jarak	1	Waktu Tempuh	4	Selisih Elevasi (Ketinggian)	4	Waktu Tempuh	1	Waktu Tempuh	1
Selisih Elevasi (Ketinggian)	0.142857143	Kemiringan	6	Jarak	0.16666667	Kemiringan	0.16666667	Selisih Elevasi (Ketinggian)	6	Kemiringan	6
Jarak	1	Jarak	0.2	Jarak	1	Selisih Elevasi (Ketinggian)	5	Waktu Tempuh	1	Waktu Tempuh	1
Jarak	6	Kemiringan	6	Waktu Tempuh	6	Selisih Elevasi (Ketinggian)	6	Waktu Tempuh	0.16666667	Waktu Tempuh	0.16666667
Selisih Elevasi (Ketinggian)	0.2	Kemiringan	5	Waktu Tempuh	5	Selisih Elevasi (Ketinggian)	5	Selisih Elevasi (Ketinggian)	4	Kemiringan	5
Selisih Elevasi (Ketinggian)	0.142857143	Kemiringan	7	Waktu Tempuh	7	Kemiringan	0.142857143	Waktu Tempuh	0.142857143	Waktu Tempuh	0.142857143
Jarak	2	Jarak	0.5	Jarak	0.5	Selisih Elevasi (Ketinggian)	2	Selisih Elevasi (Ketinggian)	2	Kemiringan	2
Selisih Elevasi (Ketinggian)	0.111111111	Kemiringan	6	Jarak	0.142857143	Selisih Elevasi (Ketinggian)	7	Selisih Elevasi (Ketinggian)	8	Kemiringan	7
Jarak	1	Kemiringan	2	Waktu Tempuh	2	Selisih Elevasi (Ketinggian)	2	Waktu Tempuh	0.25	Kemiringan	3

Lanjutan

Jarak dan Selisih Elevasi	Skala	Jarak dan Kemiringan	Skala	Jarak dan Waktu Tempuh	Skala	Selisih Elevasi dan Kemiringan	Skala	Selisih Elevasi dan Waktu Tempuh	Skala	Kemiringan dan Waktu Tempuh	Skala
Jarak	7	Jarak	0.142 8571 43	Waktu Tempuh	1	Kemiringan	0.333 3333 33	Waktu Tempuh	0.142 85714 3	Waktu Tempuh	0.142 8571 43
Jarak	1	Jarak	1	Jarak	1	Selisih Elevasi (Ketinggian)	1	Selisih Elevasi (Ketinggian)	1	Waktu Tempuh	1
Jarak	1	Kemiringan	1	Jarak	1	Selisih Elevasi (Ketinggian)	1	Selisih Elevasi (Ketinggian)	1	Kemiringan	2
Selisih Elevasi (Ketinggian)	0.2	Kemiringan	5	Waktu Tempuh	6	Selisih Elevasi (Ketinggian)	5	Waktu Tempuh	0.2	Waktu Tempuh	0.2
Selisih Elevasi (Ketinggian)	0.5	Kemiringan	1	Waktu Tempuh	5	Selisih Elevasi (Ketinggian)	1	Waktu Tempuh	1	Waktu Tempuh	0.25
Jarak	1	Jarak	0.142 8571 43	Jarak	0.2	Selisih Elevasi (Ketinggian)	5	Selisih Elevasi (Ketinggian)	5	Waktu Tempuh	0.2
Jarak	1	Kemiringan	2	Waktu Tempuh	2	Selisih Elevasi (Ketinggian)	2	Selisih Elevasi (Ketinggian)	3	Waktu Tempuh	0.5
Jarak	7	Jarak	0.142 8571 43	Jarak	0.142 8571 43	Selisih Elevasi (Ketinggian)	3	Waktu Tempuh	0.2	Waktu Tempuh	0.2
Selisih Elevasi (Ketinggian)	0.2	Kemiringan	6	Waktu Tempuh	6	Selisih Elevasi (Ketinggian)	4	Selisih Elevasi (Ketinggian)	6	Waktu Tempuh	0.142 8571 43
Selisih Elevasi (Ketinggian)	0.2	Kemiringan	7	Jarak	1	Kemiringan	0.333 3333 33	Selisih Elevasi (Ketinggian)	5	Kemiringan	8
Selisih Elevasi (Ketinggian)	0.142 8571 43	Kemiringan	7	Jarak	0.2	Kemiringan	0.166 6666 67	Selisih Elevasi (Ketinggian)	7	Kemiringan	7

Lanjutan

Jarak dan Selisih Elevasi	Skala	Jarak dan Kemiringan	Skala	Jarak dan Waktu Tempuh	Skala	Selisih Elevasi dan Kemiringan	Skala	Selisih Elevasi dan Waktu Tempuh	Skala	Kemiringan dan Waktu Tempuh	Skala
Selisih Elevasi (Ketinggian)	0.2	Kemiringan	8	Waktu Tempuh	5	Selisih Elevasi (Ketinggian)	5	Selisih Elevasi (Ketinggian)	6	Waktu Tempuh	0.125
Jarak	3	Jarak	0.2	Waktu Tempuh	1	Selisih Elevasi (Ketinggian)	3	Waktu Tempuh	0.3333333	Waktu Tempuh	0.3333333
Jarak	6	Kemiringan	4	Jarak	0.2	Kemiringan	0.1666667	Waktu Tempuh	0.1666667	Kemiringan	5
Jarak	3	Jarak	0.3333333	Waktu Tempuh	3	Selisih Elevasi (Ketinggian)	6	Selisih Elevasi (Ketinggian)	3	Waktu Tempuh	0.25
Selisih Elevasi (Ketinggian)	0.2	Kemiringan	8	Waktu Tempuh	5	Kemiringan	0.1666667	Waktu Tempuh	0.3333333	Kemiringan	2
Jarak	1	Kemiringan	9	Waktu Tempuh	8	Selisih Elevasi (Ketinggian)	9	Waktu Tempuh	0.1111111	Waktu Tempuh	0.1111111
Jarak	9	Kemiringan	9	Waktu Tempuh	9	Kemiringan	0.1111111	Waktu Tempuh	0.1111111	Kemiringan	9
Selisih Elevasi (Ketinggian)	0.125	Kemiringan	8	Waktu Tempuh	8	Kemiringan	0.125	Waktu Tempuh	0.125	Waktu Tempuh	0.125
Jarak	5	Jarak	0.2	Waktu Tempuh	5	Selisih Elevasi (Ketinggian)	7	Waktu Tempuh	0.125	Waktu Tempuh	0.125
Selisih Elevasi (Ketinggian)	0.142857143	Kemiringan	9	Jarak	0.142857143	Kemiringan	0.142857143	Selisih Elevasi (Ketinggian)	6	Kemiringan	7



Lanjutan

Jarak dan Selisih Elevasi	Skala	Jarak dan Kemiringan	Skala	Jarak dan Waktu Tempuh	Skala	Selisih Elevasi dan Kemiringan	Skala	Selisih Elevasi dan Waktu Tempuh	Skala	Kemiringan dan Waktu Tempuh	Skala
Selisih Elevasi (Ketinggian)	0.16666667	Kemiringan	8	Waktu Tempuh	7	Kemiringan	0.125	Selisih Elevasi (Ketinggian)	6	Kemiringan	8
Jarak	5	Kemiringan	8	Waktu Tempuh	1	Kemiringan	0.142857143	Waktu Tempuh	0.125	Waktu Tempuh	0.125
Selisih Elevasi (Ketinggian)	0.5	Jarak	0.25	Jarak	0.25	Selisih Elevasi (Ketinggian)	7	Selisih Elevasi (Ketinggian)	7	Waktu Tempuh	0.2
Jarak	2	Jarak	0.33333333	Jarak	0.25	Selisih Elevasi (Ketinggian)	2	Selisih Elevasi (Ketinggian)	3	Kemiringan	2
Jarak	2	Jarak	0.33333333	Jarak	0.25	Selisih Elevasi (Ketinggian)	2	Selisih Elevasi (Ketinggian)	3	Kemiringan	2
Selisih Elevasi (Ketinggian)	0.2	Kemiringan	4	Waktu Tempuh	4	Selisih Elevasi (Ketinggian)	3	Selisih Elevasi (Ketinggian)	4	Kemiringan	4
Jarak	2	Jarak	0.33333333	Jarak	0.25	Selisih Elevasi (Ketinggian)	2	Selisih Elevasi (Ketinggian)	3	Kemiringan	2
Jarak	9	Jarak	0.11111111	Jarak	0.11111111	Selisih Elevasi (Ketinggian)	9	Selisih Elevasi (Ketinggian)	9	Kemiringan	9
Selisih Elevasi (Ketinggian)	0.11111111	Kemiringan	9	Jarak	0.11111111	Kemiringan	0.11111111	Selisih Elevasi (Ketinggian)	9	Waktu Tempuh	0.142857143
Selisih Elevasi (Ketinggian)	0.33333333	Kemiringan	5	Waktu Tempuh	5	Selisih Elevasi (Ketinggian)	5	Waktu Tempuh	0.2	Waktu Tempuh	0.2

Lanjutan

Jarak dan Selisih Elevasi	Skala	Jarak dan Kemiringan	Skala	Jarak dan Waktu Tempuh	Skala	Selisih Elevasi dan Kemiringan	Skala	Selisih Elevasi dan Waktu Tempuh	Skala	Kemiringan dan Waktu Tempuh	Skala
Jarak	6	Kemiringan	6	Waktu Tempuh	6	Kemiringan	0.2	Waktu Tempuh	0.25	Waktu Tempuh	0.2
Jarak	2	Jarak	0.5	Jarak	0.5	Selisih Elevasi (Ketinggian)	2	Selisih Elevasi (Ketinggian)	2	Kemiringan	2
Jarak	7	Jarak	0.142857143	Jarak	0.142857143	Selisih Elevasi (Ketinggian)	7	Selisih Elevasi (Ketinggian)	7	Kemiringan	7
Selisih Elevasi (Ketinggian)	1	Kemiringan	1	Jarak	1	Kemiringan	0.333333333	Selisih Elevasi (Ketinggian)	1	Kemiringan	1
Selisih Elevasi (Ketinggian)	0.142857143	Kemiringan	9	Waktu Tempuh	9	Selisih Elevasi (Ketinggian)	9	Waktu Tempuh	0.111111111	Waktu Tempuh	0.111111111
Rata-rata	2.159375841	Rata-rata	3.757613667	Rata-rata	3.132378262	Rata-rata	2.528827011	Rata-rata	2.308817595	Rata-rata	2.260485607

Tabel Penilaian Hasil Rekomendasi Rute di Gunung Arjuno

Nama Lengkap	Instansi/ Komunitas	Jumlah Pendakian	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN
Khurotul Aini	SLH Saunggalih	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	1
Zahrotul Qotrun Nada	SLH Saunggalih	2	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	1	0	0	0
Indah Rifatud Diniyah	Mapala Tursina	3	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0	1
Wahyu Nur Efendi	Inayah	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Suhendik Prastiya, S.A.P	ORPADA	4	0	0	0	0	0	2	0	1	0	0	0	0	0	0	0	1	1	0	0	0
Ranisya	SLH Saunggalih	3	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Fairus In'am Pratama	Mapala Tursina	4	0	0	0	0	0	1	0	1	1	2	0	0	0	0	0	0	0	0	0	0
Achmad Qosi'an	Ispala	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Mochammad Khoirul Basyar	Inayah	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Tasrif Al Fajar	Inayah	6	0	0	0	0	1	1	1	1	1	1	0	0	0	0	1	1	1	1	1	1
Rata-rata		4	0	0	0	0	0.2	0.6	0.2	0.4	0.3	0.4	0	0	0	0	0.1	0.3	0.5	0.1	0.2	0.3

Lanjutan

Nama Lengkap	Instansi/ Komunitas	Jumlah Pendakian	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN
Khurotul Aini	SLH Saunggalih	4	0	0	1	0	0	1	0	0	0	0	0	1	1	1	0	0	0	0	0	1
Zahrotul Qotrun Nada	SLH Saunggalih	2	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Indah Rifatud Diniyah	Mapala Tursina	3	1	2	0	0	0	0	1	0	0	0	1	0	1	0	0	0	0	0	0	0
Wahyu Nur Efendi	Inayah	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Suhendik Prastiya, S.A.P	ORPADA	4	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Ranisya	SLH Saunggalih	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Fairus In'am Pratama	Mapala Tursina	4	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Achmad Qosi'an	Ispala	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Mochammad Khoirul Basyar	Inayah	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Tasrif Al Fajar	Inayah	6	1	1	1	1	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0
Rata-rata		4	0.4	0.4	0.4	0.2	0	0.1	0.1	0	0	0	0.2	0.2	0.3	0.2	0	0	0	0	0	0.1

Tabel Penilaian Hasil Rekomendasi Rute di Gunung Bromo

Nama Lengkap	Instansi/ Komunitas	Jumlah Pendakian	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN
Wahyu Nur Efendi	Inayah	3	1	1	0	0	1	0	1	0	0	0	1	0	1	1	0	0	1	2	0	1
Mochammad Khoirul Basyar	Inayah	4	0	0	1	1	0	0	0	0	1	1	1	2	0	0	1	1	0	0	0	0
Tasrif Al Fajar	Inayah	3	1	1	0	0	1	1	0	1	0	0	0	0	1	1	0	0	0	0	0	0
Achmad Qosi'an	Ispala	2	0	0	0	1	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0
Fairus In'am Pratama	Mapala Tursina	5	1	0	0	0	1	0	0	0	0	0	0	0	1	2	0	0	1	1	1	1
Indah Rifatud Diniyah	Mapala Tursina	4	0	0	0	0	0	0	2	0	0	0	0	0	0	0	1	1	1	1	0	1
Suhendik Prastiya, S.A.P	ORPADA	4	1	0	0	0	0	0	0	0	1	1	0	0	0	0	1	2	0	0	0	0
Khurotul Aini	SLH Saunggalih	5	1	2	0	0	1	2	0	1	0	0	1	2	0	1	0	0	1	0	1	0
Ranisya	SLH Saunggalih	3	0	0	1	2	0	0	0	1	1	1	0	0	0	0	0	1	0	0	0	0
Zahrotul Qotrun Nada	SLH Saunggalih	5	1	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
Rata-rata		3.8	0.6	0.5	0.2	0.4	0.4	0.3	0.3	0.3	0.4	0.4	0.4	0.5	0.4	0.6	0.3	0.5	0.4	0.4	0.2	0.3

Tabel Penilaian Hasil Rekomendasi Rute di Gunung Butak

Nama Lengkap	Instansi/ Komunitas	Jumlah Pendakian	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN
Wahyu Nur Efendi	Inayah	3	0	0	1	1	0	0	1	1	1	1	0	0	0	0	1	1	0	0
Mochammad Khoirul Basyar	Inayah	4	0	0	1	1	0	0	1	1	0	0	0	0	1	0	1	1	0	0
Tasrif Al Fajar	Inayah	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Achmad Qosi'an	Ispala	4	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Fairus In'am Pratama	Mapala Tursina	3	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
Indah Rifatud Diniyah	Mapala Tursina	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
Suhendik Prastiya, S.A.P	ORPADA	3	0	0	1	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Khurotul Aini	SLH Saunggalih	3	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0
Ranisya	SLH Saunggalih	4	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
Zahrotul Qotrun Nada	SLH Saunggalih	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Rata-rata		3.2	0	0	0.4	0.4	0	0	0.3	0.2	0.3	0.3	0	0	0.2	0	0.3	0.3	0	0

Tabel Penilaian Hasil Rekomendasi Rute di Gunung Ciremai

Nama Lengkap	Instansi/ Komunitas	Jumlah Pendakian	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN
Khurotul Aini	SLH Saunggalih	2	0	0	0	0	1	1	0	0	0	0	0	0	0	1	0	0	0	0
Fairus In'am Pratama	Mapala Tursina	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Suhendik Prastiya, S.A.P	ORPADA	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Zahrotul Qotrun Nada	SLH Saunggalih	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Indah Rifatud Diniyah	Mapala Tursina	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Wahyu Nur Efendi	Inayah	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Ranisya	SLH Saunggalih	2	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	1	1
Achmad Qosi'an	Ispala	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Mochammad Khoirul Basyar	Inayah	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Tasrif Al Fajar	Inayah	4	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
Rata-rata		3	0	0	0	0	0.2	0.2	0	0	0	0	0	0	0.1	0.2	0	0	0.1	0.2

Tabel Penilaian Hasil Rekomendasi Rute di Gunung Penanggungan

Nama Lengkap	Instansi/ Komunitas	Jumlah Pendakian	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN
Wahyu Nur Efendi	Inayah	3	1	1	1	1	1	1	0	0	0	0	0	0	1	1	1	2	0	0
Mochammad Khoirul Basyar	Inayah	3	1	1	1	1	1	1	1	0	0	0	1	0	1	1	0	0	0	0
Tasrif Al Fajar	Inayah	3	1	1	0	0	1	1	0	0	1	2	0	0	1	1	0	0	0	0
Achmad Qosi'an	Ispala	4	1	1	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0
Fairus In'am Pratama	Mapala Tursina	3	0	0	0	0	0	0	1	2	0	0	0	0	0	0	0	0	0	0
Indah Rifatud Diniyah	Mapala Tursina	2	1	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0
Suhendik Prastiya, S.A.P	ORPADA	3	0	0	1	2	0	0	1	1	0	0	1	1	1	0	0	0	0	0
Khurotul Aini	SLH Saunggalih	4	0	0	0	0	1	0	1	1	1	0	1	1	0	0	1	2	0	0
Ranisya	SLH Saunggalih	2	1	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Zahrotul Qotrun Nada	SLH Saunggalih	4	0	0	0	0	0	0	1	0	0	0	1	0	1	2	0	0	0	0
Rata-rata		3.1	0.6	0.6	0.4	0.4	0.6	0.4	0.5	0.4	0.2	0.2	0.5	0.2	0.7	0.6	0.2	0.4	0	0





Lanjutan

Nama Lengkap	Instansi/ Komunitas	Jumlah Pendakian	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN
Khurotul Aini	SLH Saunggalih	5	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Ranisya	SLH Saunggalih	3	0	0	0	0	0	1	0	0	0	0	1	0	0	1	0	0	0	1	0	0
Achmad Qosi'an	Ispala	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Indah Rifatud Diniyah	Mapala Tursina	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Suhendik Prastiya, S.A.P	ORPADA	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Fairus In'am Pratama	Mapala Tursina	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Wahyu Nur Efendi	Inayah	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Zahrotul Qotrun Nada	SLH Saunggalih	3	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Mochammad Khoirul Basyar	Inayah	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Tasrif Al Fajar	Inayah	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Rata-rata		3.7	0	0	0	0.1	0	0.2	0	0	0	0	0.1	0	0	0.1	0	0	0	0.1	0	0

Tabel Penilaian Hasil Rekomendasi Rute di Gunung Sumbing

Nama Lengkap	Instansi/ Komunitas	Jumlah Pendakian	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN
Achmad Qosi'an	Ispala	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Fairus In'am Pratama	Mapala Tursina	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Indah Rifatud Diniyah	Mapala Tursina	4	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0
Khurotul Aini	SLH Saunggalih	4	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Mochammad Khoirul Basyar	Inayah	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Ranisya	SLH Saunggalih	3	0	0	0	0	1	1	0	0	0	0	0	1	0	0	0	0
Suhendik Prastiya, S.A.P	ORPADA	3	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Tasrif Al Fajar	Inayah	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Wahyu Nur Efendi	Inayah	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Zahrotul Qotrun Nada	SLH Saunggalih	5	1	0	0	1	1	1	0	0	1	1	1	0	1	1	0	0
Rata-rata		4	0.1	0	0.1	0.3	0.2	0.2	0	0	0.1	0.1	0.1	0.1	0.2	0.1	0	0

Lanjutan

Nama Lengkap	Instansi/ Komunitas	Jumlah Pendakian	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN
Achmad Qosi'an	Ispala	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Fairus In'am Pratama	Mapala Tursina	5	0	0	0	0	0	0	1	0	1	0	0	0	1	1	0	0
Indah Rifatud Diniyah	Mapala Tursina	4	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
Khurotul Aini	SLH Saunggalih	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Mochammad Khoirul Basyar	Inayah	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Ranisya	SLH Saunggalih	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Suhendik Prastiya, S.A.P	ORPADA	3	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0
Tasrif Al Fajar	Inayah	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Wahyu Nur Efendi	Inayah	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Zahrotul Qotrun Nada	SLH Saunggalih	5	0	0	0	0	0	1	1	1	1	0	1	1	1	1	0	0
Rata-rata		4	0	0	0	0	0.1	0.2	0.2	0.1	0.2	0	0.2	0.1	0.2	0.2	0	0

Lanjutan

Nama Lengkap	Instansi/ Komunitas	Jumlah Pendakian	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	
Achmad Qosi'an	Ispala	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
Fairus In'am Pratama	Mapala Tursina	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Indah Rifatud Diniyah	Mapala Tursina	4	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1
Khurotul Aini	SLH Saunggalih	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Mochammad Khoirul Basyar	Inayah	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Ranisya	SLH Saunggalih	3	0	0	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0
Suhendik Prastiya, S.A.P	ORPADA	3	0	1	1	1	0	1	0	0	0	0	0	0	0	0	1	0	1
Tasrif Al Fajar	Inayah	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Wahyu Nur Efendi	Inayah	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Zahrotul Qotrun Nada	SLH Saunggalih	5	1	0	1	1	1	0	2	2	0	0	0	0	0	0	0	0	0
Rata-rata		4	0.1	0.2	0.4	0.4	0.1	0.2	0.2	0.2	0	0	0	0	0	0	0.22	0.1	0.2

Lanjutan

Nama Lengkap	Instansi/ Komunitas	Jumlah Pendakian	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP	FN
Achmad Qosi'an	Ispala	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Fairus In'am Pratama	Mapala Tursina	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Indah Rifatud Diniyah	Mapala Tursina	4	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
Khurotul Aini	SLH Saunggalih	4	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Mochammad Khoirul Basyar	Inayah	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Ranisya	SLH Saunggalih	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Suhendik Prastiya, S.A.P	ORPADA	3	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
Tasrif Al Fajar	Inayah	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Wahyu Nur Efendi	Inayah	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Zahrotul Qotrun Nada	SLH Saunggalih	5	1	0	1	0	1	0	0	0	0	0	0	0	0	0	1	1
Rata-rata		4	0.1	0	0.1	0	0.2	0.1	0	0	0	0	0	0	0.1	0	0.1	0.2

Tabel Penilaian Hasil Rekomendasi Rute di Gunung Ungaran

Nama Lengkap	Instansi/ Komunitas	Jumlah Pendakian	FP	FN	FP	FN	FP	FN	FP	FN
Wahyu Nur Efendi	Inayah	3	0	0	0	0	0	0	0	0
Mochammad Khoirul Basyar	Inayah	2	0	0	0	0	0	0	0	0
Tasrif Al Fajar	Inayah	3	0	1	0	0	0	0	1	1
Achmad Qosi'an	Ispala	2	0	0	1	0	0	0	0	0
Fairus In'am Pratama	Mapala Tursina	2	0	0	0	0	1	0	0	0
Indah Rifatud Diniyah	Mapala Tursina	2	0	0	0	0	0	1	0	0
Suhendik Prastiya, S.A.P	ORPADA	2	0	0	0	0	0	0	0	0
Khurotul Aini	SLH Saunggalih	3	0	0	0	0	0	0	0	0
Ranisya	SLH Saunggalih	4	1	0	0	0	0	0	0	0
Zahrotul Qotrun Nada	SLH Saunggalih	2	0	0	0	0	0	0	0	0
Rata-rata		2.5	0.1	0.1	0.1	0	0.1	0.1	0.1	0.1

Tabel Perhitungan Akurasi dan Waktu Pemrosesan di Gunung Arjuno

No	START	FINISH	BOBOT	JARAK	R	TP	FP	FN	F1-Score (%)	Waktu (ms)
1	<i>Basecamp Lawang</i>	Puncak Arjuno Barat	1.428	9521	8	8	0	0	100	82
2		Puncak Arjuno Ogal-Agil	1.42	9181	7	7	0	0	100	65
3		Puncak Kembar 1	2.239	13025	14	13.8	0.2	0.6	97.18	68.0
4		Puncak Kembar 2	2.298	13181	14	13.8	0.2	0.4	97.87	77
5		Puncak Welirang	2.557	14628	16	15.7	0.3	0.4	97.82	65
6	<i>Basecamp Purwosari</i>	Puncak Arjuno Barat	1.939	9211	11	11	0	0	100	64
7		Puncak Arjuno Ogal-Agil	1.931	8871	10	10	0	0	100	61
8		Puncak Kembar 1	2.75	12715	17	16.9	0.1	0.3	98.83	72
9		Puncak Kembar 2	2.808	12871	17	16.5	0.5	0.1	98.21	62
10		Puncak Welirang	3.068	14318	19	18.8	0.2	0.3	98.69	63
11	<i>Basecamp Tretes</i>	Puncak Arjuno Barat	2.184	11615	10	9.6	0.4	0.4	96.00	65
12		Puncak Arjuno Ogal-Agil	2.193	11955	11	10.6	0.4	0.2	97.25	65
13		Puncak Kembar 1	1.654	10139	6	6	0	0.1	99.17	66
14		Puncak Kembar 2	2.02	11296	9	8.9	0.1	0	99.44	73
15		Puncak Welirang	1.626	11056	6	6	0	0	100	65
16	<i>Basecamp Sumber Brantas</i>	Puncak Arjuno Barat	1.501	9632	11	10.8	0.2	0.2	98.18	67
17		Puncak Arjuno Ogal-Agil	1.509	9972	12	11.7	0.3	0.2	97.91	62
18		Puncak Kembar 1	1.025	6984	7	7	0	0	100	62
19		Puncak Kembar 2	1.084	7140	7	7	0	0	100	63
20		Puncak Welirang	1.018	8093	7	7	0	0.1	99.29	67
Rata-rata									98.793	66.700



Tabel Perhitungan Akurasi dan Waktu Pemrosesan di Gunung Bromo

No	START	FINISH	BOBOT	JARAK	R	TP	FP	FN	F1-Score (%)	Waktu (ms)
1	Cemoro Lawang	Puncak Batok	0.54414	2814	6	5.4	0.6	0.5	90.76	67
2		Puncak Bromo	0.21364	4010	4	3.8	0.2	0.4	92.68	68
3		Puncak Crater	0.17302	3624	4	3.6	0.4	0.3	91.14	66
4		Puncak Penanjakan	0.58407	4766	5	4.7	0.3	0.3	94.00	67
5		Puncak Widodaren	0.33532	6292	6	5.6	0.4	0.4	93.33	72
6	Seruni View Point	Puncak Batok	0.69963	5250	6	5.6	0.4	0.5	92.56	66
7		Puncak Bromo	0.39414	6876	6	5.6	0.4	0.6	91.80	65
8		Puncak Crater	0.35351	6490	6	5.7	0.3	0.5	93.44	67
9		Puncak Penanjakan	0.40357	1900	3	2.6	0.4	0.4	86.67	65
10		Puncak Widodaren	0.51581	9158	8	7.8	0.2	0.3	96.89	67
Rata-rata									92.328	67.00

Tabel Perhitungan Akurasi dan Waktu Pemrosesan di Gunung Butak

No	START	FINISH	BOBOT	JARAK	R	TP	FP	FN	F1-Score (%)	Waktu (ms)
1	Basecamp Kraton Gunung Kawi	Puncak Batu Tulis	1.57639	3786	6	6	0	0	100	66
2		Puncak Butak	2.17183	6743	11	10.6	0.4	0.4	96.36	66
3		Puncak Panderman	4.45792	18375	21	21	0	0	100	67
4	Basecamp Sirah Kencong	Puncak Batu Tulis	3.13206	9138	16	15.7	0.3	0.2	98.43	66
5		Puncak Butak	2.55098	6325	13	12.7	0.3	0.3	97.69	69
6		Puncak Panderman	5.10643	19147	25	25	0	0	100	72
7	Basecamp Toyomerto	Puncak Batu Tulis	2.3043	12950	15	14.8	0.2	0	99.33	66
8		Puncak Butak	1.99258	11327	14	13.7	0.3	0.3	97.86	61
9		Puncak Panderman	0.93231	2837	6	6	0	0	100	61
Rata-rata									98.853	66.00

Tabel Perhitungan Akurasi dan Waktu Pemrosesan di Gunung Ciremai

No	START	FINISH	BOBOT	JARAK	R	TP	FP	FN	F1-Score (%)	Waktu (ms)
1	<i>Basecamp Apuy</i>	Puncak Ciremai	1.663	6180	10	10	0	0	100	67
2		Puncak Crater	1.634	6218	10	10	0	0	100	64
3		Puncak Panglongokan	1.652	6784	11	10.8	0.2	0.2	98.18	64
4	<i>Basecamp Linggarjati</i>	Puncak Ciremai	2.447	9018	13	13	0	0	100	67
5		Puncak Crater	2.418	8943	13	13	0	0	100	72
6		Puncak Panglongokan	2.401	8377	12	12	0	0	100	67
7	<i>Basecamp Palutungan</i>	Puncak Ciremai	2.424	9575	12	11.9	0.1	0.2	98.76	61
8		Puncak Crater	2.395	9613	12	12	0	0	100	59
9		Puncak Panglongokan	2.413	10179	13	12.9	0.1	0.2	98.85	62
Rata-rata									99.532	64.778

Tabel Perhitungan Akurasi dan Waktu Pemrosesan di Gunung Penanggungan

No	START	FINISH	BOBOT	JARAK	R	TP	FP	FN	F1-Score (%)	Waktu (ms)
1	<i>Basecamp Genting</i>	Puncak Bekel	1.86312	4148	10	9.4	0.6	0.6	94.00	65
2		Puncak Crater	2.2774	4039	9	8.6	0.4	0.4	95.56	71
3		Puncak Penanggungan	2.40442	4253	10	9.4	0.6	0.4	94.95	62
4	<i>Basecamp Jolotundo</i>	Puncak Bekel	1.04	3353	7	6.5	0.5	0.4	93.53	66
5		Puncak Crater	1.56061	3477	7	6.8	0.2	0.2	97.14	69
6		Puncak Penanggungan	1.68763	3691	8	7.5	0.5	0.2	95.54	67
7	<i>Basecamp Tamiajeng</i>	Puncak Bekel	2.48715	6249	12	11.3	0.7	0.6	94.56	67
8		Puncak Crater	1.56377	4430	9	8.8	0.2	0.4	96.70	68
9		Puncak Penanggungan	1.43675	4216	8	8	0	0	100	66
Rata-rata									95.775	66.778

Tabel Perhitungan Akurasi dan Waktu Pemrosesan di Gunung Slamet

No	START	FINISH	BOBOT	JARAK	R	TP	FP	FN	F1-Score (%)	Waktu (ms)
1	<i>Basecamp</i> Bambangan	Puncak Barat	2.152	7103	11	11	0	0	100	62
2		Puncak Slamet	2.03	5936	9	9	0	0	100	63
3		Puncak Surono	2.075	6642	10	10	0	0	100	65
4	<i>Basecamp</i> Guci Kompak	Puncak Barat	1.419	8214	7	7	0	0	100	60
5		Puncak Slamet	1.541	9381	9	9	0	0	100	60
6		Puncak Surono	1.496	8675	8	8	0	0	100	61
7	<i>Basecamp</i> Guci Permadi	Puncak Barat	1.371	9465	7	7	0	0	100	60
8		Puncak Slamet	1.493	10632	9	9	0	0	100	71
9		Puncak Surono	1.448	9926	8	8	0	0	100	58
10	<i>Basecamp</i> Kaliwadas	Puncak Barat	1.616	9989	7	7	0	0	100	61
11		Puncak Slamet	1.738	11156	9	9	0	0	100	60
12		Puncak Surono	1.694	10450	8	8	0	0	100	59
13	<i>Basecamp</i> Penakir	Puncak Barat	2.354	7212	11	11	0	0.1	99.55	67
14		Puncak Slamet	2.232	6045	9	9	0	0.2	98.9	62
15		Puncak Surono	2.277	6751	10	10	0	0	100	59
16	<i>Basecamp</i> Radenpala-Baturaden	Puncak Barat	1.791	8544	8	8	0	0	100	59
17		Puncak Slamet	1.758	8789	8	7.9	0.1	0	99.37	65
18		Puncak Surono	1.714	8083	7	7	0	0.1	99.29	62
19	<i>Basecamp</i> Sawangan	Puncak Barat	1.145	9403	7	7	0	0	100	63
20		Puncak Slamet	1.267	10570	9	9	0	0.1	99.45	64
21		Puncak Surono	1.222	9864	8	8	0	0	100	67
Rata-rata									99.836	62.286

Tabel Perhitungan Akurasi dan Waktu Pemrosesan di Gunung Sumbing

No	START	FINISH	BOBOT	JARAK	R	TP	FP	FN	F1-Score (%)	Waktu (ms)
1	<i>Basecamp Adipuro</i>	Puncak Buntu	1.761	8858	12	11.9	0.1	0	99.58	66
2		Puncak Kawah	1.451	8472	10	9.9	0.1	0.3	98.02	62
3		Puncak Rajawali	1.377	8310	9	8.8	0.2	0.2	97.78	68
4		Puncak Sejati	1.276	7506	8	8	0	0	100	64
5	<i>Basecamp Banaran</i>	Puncak Buntu	1.982	8341	11	10.9	0.1	0.1	99.09	64
6		Puncak Kawah	1.672	7955	9	8.9	0.1	0.1	98.89	64
7		Puncak Rajawali	1.597	7793	8	7.8	0.2	0.1	98.11	62
8		Puncak Sejati	1.497	6989	7	7	0	0	100	62
9	<i>Basecamp Bowongso</i>	Puncak Buntu	1.03	6453	6	6	0	0	100	61
10		Puncak Kawah	1.255	6253	6	6	0	0	100	67
11		Puncak Rajawali	1.329	6415	7	6.9	0.1	0.2	97.87	65
12		Puncak Sejati	1.733	7677	10	9.8	0.2	0.1	98.49	64
13	<i>Basecamp Butuh</i>	Puncak Buntu	1.61	6679	12	11.8	0.2	0	99.16	65
14		Puncak Kawah	1.3	6293	10	9.8	0.2	0	98.99	65
15		Puncak Rajawali	1.225	6131	9	8.8	0.2	0.2	97.78	63
16		Puncak Sejati	1.125	5327	8	8	0	0	100	71
17	<i>Basecamp Cepit</i>	Puncak Buntu	1.706	6845	10	9.9	0.1	0.2	98.51	66
18		Puncak Kawah	1.396	6459	8	7.8	0.2	0.2	97.50	65
19		Puncak Rajawali	1.344	6871	8	7.9	0.1	0.2	98.14	65
20		Puncak Sejati	1.296	7199	9	8.8	0.2	0.2	97.78	63

Lanjutan

No	START	FINISH	BOBOT	JARAK	R	TP	FP	FN	F1-Score (%)	Waktu (ms)
21	<i>Basecamp Garung</i>	Puncak Buntu	1.256	6313	5	5	0	0	100	64
22		Puncak Kawah	1.566	6699	7	7	0	0	100	64
23		Puncak Rajawali	1.641	6861	8	8	0	0.2	98.77	69
24		Puncak Sejati	2.044	8123	11	10.9	0.1	0.2	98.64	65
25	<i>Basecamp Mangli</i>	Puncak Buntu	1.701	6775	11	10.9	0.1	0	99.54	66
26		Puncak Kawah	1.392	6389	9	8.9	0.1	0	99.44	65
27		Puncak Rajawali	1.317	6227	8	7.8	0.2	0.1	98.11	67
28		Puncak Sejati	1.217	5423	7	7	0	0	100	64
29	<i>Basecamp Siptung</i>	Puncak Buntu	0.971	6767	5	5	0	0	100	69
30		Puncak Kawah	1.281	7153	7	7	0	0	100	64
31		Puncak Rajawali	1.356	7315	8	7.9	0.1	0	99.37	63
32		Puncak Sejati	1.76	8577	11	10.9	0.1	0.2	98.64	62
Rata-rata									99.006	64.813

Tabel Perhitungan Akurasi dan Waktu Pemrosesan di Gunung Ungaran

No	START	FINISH	BOBOT	JARAK	R	TP	FP	FN	F1-Score (%)	Waktu (ms)
1	<i>Basecamp Mawar</i>	Puncak Botak	1.27576	4405	8	7.9	0.1	0.1	98.75	66
2		Puncak Ungaran	1.21367	3706	7	6.9	0.1	0	99.28	64
3	<i>Basecamp Medini</i>	Puncak Botak	2.1855	6290	10	9.9	0.1	0.1	99	67
4		Puncak Ungaran	2.12341	5591	9	8.9	0.1	0.1	98.89	65
Rata-rata									98.980	65.500

Tabel Penilaian *System Usability Scale (SUS)*

Nama Lengkap/ Panggilan	Instansi/ Komunitas	Pendakian 1th Terakhir	Familiar dengan Aplikasi Serupa	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
Mochammad Khoirul Basyar	Inayah	1-2 kali	Kurang Familiar	5	1	4	2	5	2	4	2	4	1
Tasrif Al Fajar	Inayah	3-5 kali	Kurang Familiar	5	1	4	2	5	2	4	2	5	2
Wahyu Nur Efendi	Inayah	3-5 kali	Kurang Familiar	5	2	5	2	4	1	4	1	4	1
Achmad Qosi'an	Ispala	Lebih dari 5 kali	Kurang Familiar	5	1	5	1	5	1	5	1	5	1
Indah Rifatud Diniyah	Mapala Tursina	Lebih dari 5 kali	Cukup Familiar	5	2	4	1	5	1	4	1	5	1
Moh. Fatkhur Rohman	Mapala Tursina	Lebih dari 5 kali	Kurang Familiar	5	1	5	1	5	1	5	1	5	1
Jihan Hilmi Amila	Mapala Tursina	1-2 kali	Cukup Familiar	5	2	4	1	4	3	4	1	4	2
Tegar Maulana Bil Haqqi	Mapala Tursina	3-5 kali	Kurang Familiar	5	1	5	1	5	1	5	1	5	1
Rizwan Fadilah	Mapala Tursina	3-5 kali	Cukup Familiar	5	1	4	1	4	2	5	2	4	3
Neny Khoirun Nisa'	Mapala Tursina	3-5 kali	Kurang Familiar	5	1	5	1	5	2	5	2	5	2
Gumelar Satra Hidayat	Mapala Tursina	3-5 kali	Kurang Familiar	5	1	5	1	5	1	5	1	5	1
Mohammad Wasik Maulana	Mapala Tursina	Lebih dari 5 kali	Cukup Familiar	5	1	5	1	5	1	5	1	5	2
Alfy Nur Hidayah Ginanjar	Mapala Tursina	Lebih dari 5 kali	Kurang Familiar	5	2	5	2	4	1	3	2	4	1
Laily Faiqotul Maghfiroh	Mapala Tursina	3-5 kali	Tidak Familiar Sama Sekali	4	2	5	1	4	1	4	2	5	2
Moh. Irfan Lutfirrahman	Mapala Tursina	Lebih dari 5 kali	Kurang Familiar	5	1	5	1	5	1	5	2	5	1
M. Minsyahril	Mapala Tursina	1-2 kali	Tidak Familiar Sama Sekali	4	1	4	1	4	1	5	2	4	2
Akhmad Sigit Pamungkas	Mapala Tursina	3-5 kali	Kurang Familiar	5	1	5	1	4	1	5	2	5	1
Erischa Yana Maprilia	Mapala Tursina	3-5 kali	Cukup Familiar	4	1	5	1	5	1	5	1	5	1
Briyana Qusnul Kholifah	Mapala Tursina	3-5 kali	Kurang Familiar	5	1	4	2	5	1	5	2	4	2

Lanjutan

<b>Nama Lengkap/ Panggilan</b>	<b>Instansi/ Komunitas</b>	<b>Pendakian 1th Terakhir</b>	<b>Familiar dengan Aplikasi Serupa</b>	<b>Q1</b>	<b>Q2</b>	<b>Q3</b>	<b>Q4</b>	<b>Q5</b>	<b>Q6</b>	<b>Q7</b>	<b>Q8</b>	<b>Q9</b>	<b>Q10</b>
Rizka Dwi Andika	Mapala Tursina	3-5 kali	Cukup Familiar	5	1	5	1	5	1	3	2	5	1
Andi Darusalam	Mapala Tursina	Lebih dari 5 kali	Kurang Familiar	5	1	5	1	5	1	5	1	5	1
Ahmad Kemal Fahreza	Mapala Tursina	3-5 kali	Tidak Familiar Sama Sekali	3	3	4	2	5	1	4	2	3	3
Najidah Rahmadina	Mapala Tursina	1-2 kali	Kurang Familiar	5	1	5	1	4	2	4	1	5	2
Megantara Afwan Zusroni	Mapala Tursina	Lebih dari 5 kali	Kurang Familiar	5	1	5	2	5	2	4	2	4	1
Aisyah Rochmah Nuraini	Mapala Tursina	3-5 kali	Cukup Familiar	5	1	5	1	5	1	5	1	5	1
Fairuz In'am Pratama	Mapala Tursina	Lebih dari 5 kali	Cukup Familiar	4	2	5	2	4	1	5	2	5	2
Suhendik Prasetya, S.A.P	ORPADA (Organisasi Pemuda Peduli Lingkungan dan Alam)	1-2 kali	Kurang Familiar	5	1	3	2	4	1	5	1	5	1
Zahrotul Qotrun Nada	SLH Saunggalih	1-2 kali	Cukup Familiar	4	2	5	2	5	1	4	2	4	1
Ranisya	SLH Saunggalih	3-5 kali	Cukup Familiar	4	3	4	2	5	1	4	1	5	2
Khurotul Aini	SLH Saunggalih	1-2 kali	Kurang Familiar	4	2	5	2	4	1	3	2	5	1

Tabel Perhitungan *System Usability Scale (SUS)*

Nama Lengkap/ Panggilan	Instansi/ Komunitas	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Total Skor	Skor SUS
Mochammad Khoirul Basyar	Inayah	4	4	3	3	4	3	3	3	3	4	34	85
Zahrotul Qotrun Nada	SLH Saunggalih	3	3	4	3	4	4	3	3	3	4	34	85
Tasrif Al Fajar	Inayah	4	4	3	3	4	3	3	3	4	3	34	85
Ranisya	SLH Saunggalih	3	2	3	3	4	4	3	4	4	3	33	82.5
Wahyu Nur Efendi	Inayah	4	3	4	3	3	4	3	4	3	4	35	87.5
Khurotul Aini	SLH Saunggalih	3	3	4	3	3	4	2	3	4	4	33	82.5
Achmad Qosi'an	Ispala	4	4	4	4	4	4	4	4	4	4	40	100
Suhendik Prasetya, S.A.P	ORPADA	4	4	2	3	3	4	4	4	4	4	36	90
Indah Rifatud Diniyah	Mapala Tursina	4	3	3	4	4	4	3	4	4	4	37	92.5
Moh. Fatkhur Rohman	Mapala Tursina	4	4	4	4	4	4	4	4	4	4	40	100
Jihan Hilmi Amila	Mapala Tursina	4	3	3	4	3	2	3	4	3	3	32	80
Tegar Maulana Bil Haqqi	Mapala Tursina	4	4	4	4	4	4	4	4	4	4	40	100
Rizwan Fadilah	Mapala Tursina	4	4	3	4	3	3	4	3	3	2	33	82.5
Neny Khoirun Nisa'	Mapala Tursina	4	4	4	4	4	3	4	3	4	3	37	92.5
Gumelar Satra Hidayat	Mapala Tursina	4	4	4	4	4	4	4	4	4	4	40	100
Mohammad Wasik Maulana	Mapala Tursina	4	4	4	4	4	4	4	4	4	3	39	97.5
Alfy Nur Hidayah Ginanjar	Mapala Tursina	4	3	4	3	3	4	2	3	3	4	33	82.5
Laily Faiqotul Maghfiroh	Mapala Tursina	3	3	4	4	3	4	3	3	4	3	34	85
Moh. Irfan Lutfirrahman	Mapala Tursina	4	4	4	4	4	4	4	3	4	4	39	97.5
M. Minsyahril	Mapala Tursina	3	4	3	4	3	4	4	3	3	3	34	85
Akhmad Sigit Pamungkas	Mapala Tursina	4	4	4	4	3	4	4	3	4	4	38	95
Erischa Yana Maprilia	Mapala Tursina	3	4	4	4	4	4	4	4	4	4	39	97.5
Briyana Qusnul Kholifah	Mapala Tursina	4	4	3	3	4	4	4	3	3	3	35	87.5
Rizka Dwi Andika	Mapala Tursina	4	4	4	4	4	4	2	3	4	4	37	92.5
Andi Darusalam	Mapala Tursina	4	4	4	4	4	4	4	4	4	4	40	100
Ahmad Kemal Fahreza	Mapala Tursina	2	2	3	3	4	4	3	3	2	2	28	70



Lanjutan

<b>Nama Lengkap/ Panggilan</b>	<b>Instansi/ Komunitas</b>	<b>Q1</b>	<b>Q2</b>	<b>Q3</b>	<b>Q4</b>	<b>Q5</b>	<b>Q6</b>	<b>Q7</b>	<b>Q8</b>	<b>Q9</b>	<b>Q10</b>	<b>Total Skor</b>	<b>Skor SUS</b>
Najidah Rahmadina	Mapala Tursina	4	4	4	4	3	3	3	4	4	3	36	90
Megantara Afwan Zusroni	Mapala Tursina	4	4	4	3	4	3	3	3	3	4	35	87.5
Aisyah Rochmah Nuraini	Mapala Tursina	4	4	4	4	4	4	4	4	4	4	40	100
Fairuz In'am Pratama	Mapala Tursina	3	3	4	3	3	4	4	3	4	3	34	85
Rata-rata												35.966	89.916
Standar Deviasi													7.67