



**PENERAPAN METODE *SIMPLE ADDITIVE WEIGHTING* (SAW) DAN
BORDA DALAM SISTEM REKOMENDASI PEMILIHAN
PARIWISATA DI KABUPATEN JEMBER**

SKRIPSI

Oleh :

Hafiz Ardi Hidayat

162410101050

PROGRAM STUDI SISTEM INFORMASI

FAKULTAS ILMU KOMPUTER

UNIVERSITAS JEMBER

2020



**PENERAPAN METODE *SIMPLE ADDITIVE WEIGHTING* (SAW) DAN
BORDA DALAM SISTEM REKOMENDASI PEMILIHAN
PARIWISATA DI KABUPATEN JEMBER**

SKRIPSI

diajukan guna melengkapi tugas akhir dan memenuhi salah satu syarat untuk menyelesaikan Pendidikan Sarjana (S1) Program Studi Sistem Informasi Fakultas Ilmu Komputer Universitas Jember dan mencapai gelar Sarjana Komputer

Oleh :

Hafiz Ardi Hidayat

162410101050

**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS ILMU KOMPUTER
UNIVERSITAS JEMBER**

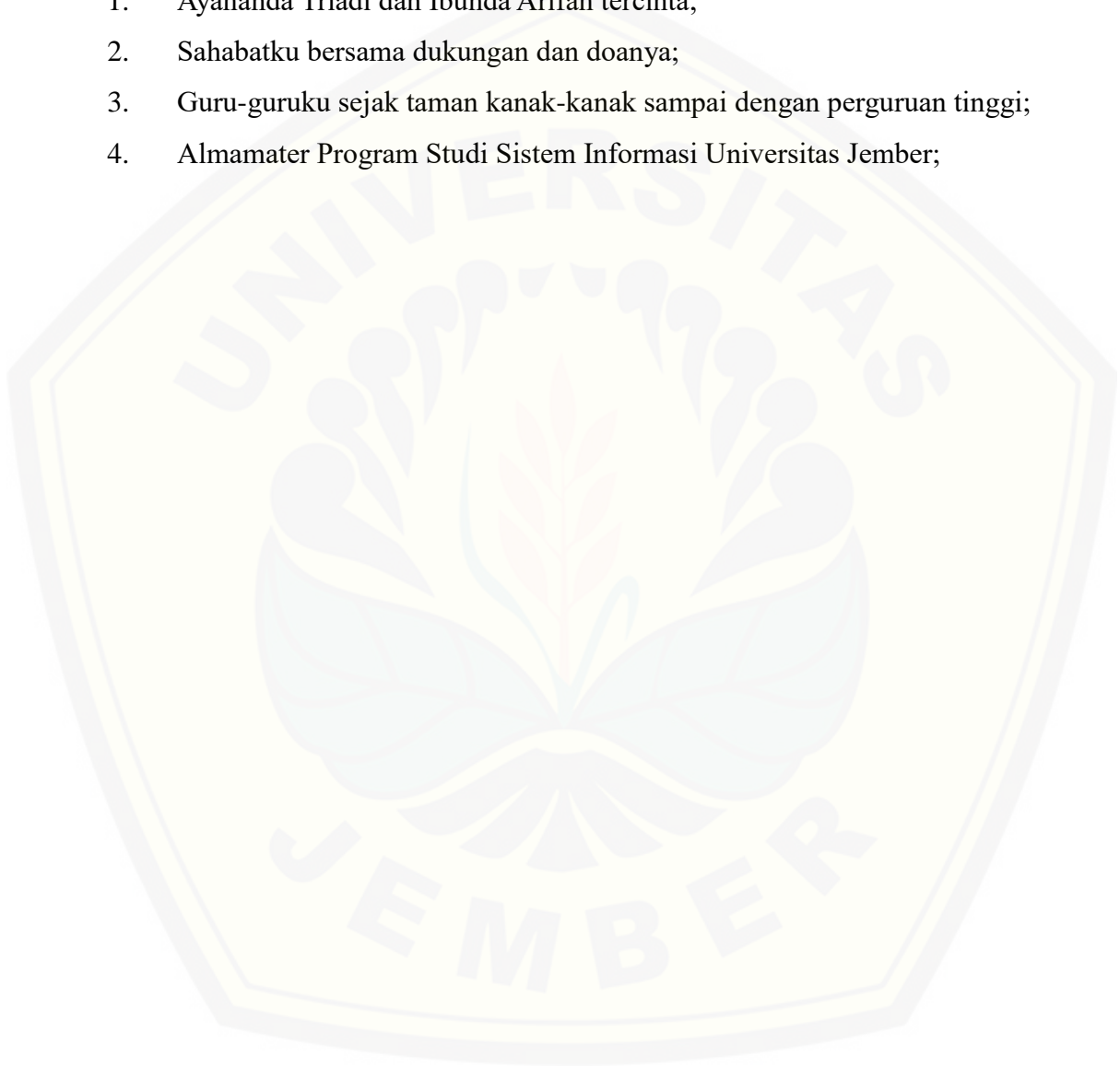
2020

ii

PERSEMBAHAN

Skripsi ini saya persembahkan untuk :

1. Ayahanda Triadi dan Ibunda Arifah tercinta;
2. Sahabatku bersama dukungan dan doanya;
3. Guru-guruku sejak taman kanak-kanak sampai dengan perguruan tinggi;
4. Almamater Program Studi Sistem Informasi Universitas Jember;



MOTTO

“Selalu ada harapan bagi mereka yang sering berdoa dan pantang menyerah”



PERNYATAAN

Saya yang bertanda tangan di bawah ini :

Nama : Hafiz Ardi Hidayat

NIM : 162410101050

Menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul “Penerapan Metode *Simple Additive Weighting* (SAW) dan Borda Dalam Sistem Rekomendasi Pemilihan Pariwisata di Kabupaten Jember”, adalah benar-benar hasil karya sendiri, kecuali jika dalam pengutipan substansi disebutkan sumbernya, belum pernah diajukan pada institusi manapun, dan bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa adanya tekanan dan paksaan dari pihak manapun serta bersedia mendapat sanksi akademik jika di kemudian hari pernyataan ini tidak benar.

Jember, 10 Juli 2020

Yang menyatakan,



Hafiz Ardi Hidayat
NIM 162410101050

SKRIPSI

**PENERAPAN METODE SIMPLE ADDITIVE WEIGHTING (SAW) DAN
BORDA DALAM SISTEM REKOMENDASI PEMILIHAN
PARIWISATA DI KABUPATEN JEMBER**

Oleh

Hafiz Ardi Hidayat

162410101050

Pembimbing :

Dosen Pembimbing Utama : Diah Ayu Retnani Wulandari S.T.,M.Eng

Dosen Pembimbing Pendamping : Muhammad Arief Hidayat S.Kom.,M.Kom

PENGESAHAN PEMBIMBING

Skripsi berjudul “Penerapan Metode Simple Additive Weighting (SAW) dan Borda dalam Sistem Rekomendasi Pemilihan Pariwisata di Kabupaten Jember”, telah diuji dan disahkan pada :

hari, tanggal : Jum’at, 10 Juli 2020

tempat : Program Studi Sistem Informasi Universitas Jember

Disetujui Oleh:

Pembimbing I,

Pembimbing II,



Diah Ayu Retnani Wulandari S.T.,M.Eng

Muhammad Arief Hidayat S.Kom.,M.Kom

NIP. 198603052014042001

NIP. 198101232010121003

PENGESAHAN PENGUJI

Skripsi berjudul “Penerapan Metode Simple Additive Weighting (SAW) dan Borda dalam Sistem Rekomendasi Pemilihan Pariwisata di Kabupaten Jember”, telah diuji dan disahkan pada:

hari, tanggal : Jum’at, 10 Juli 2020

tempat : Program Studi Sistem Informasi Universitas Jember

Disetujui Oleh :

Penguji I,



Drs. Antonius Cahya Prihandoko, M.App.Sc., Ph.D

NIP.196909281993021001

Penguji II,



Nova El Maidah S.Si., M.Cs.

NIP. 198411012015042001

Mengesahkan

Dekan Fakultas Ilmu Komputer,



Prof. Dr. Saiful Bukhori, ST., M.Kom

NIP. 196811131994121001

RINGKASAN

Penerapan Metode *Simple Additive Weighting* (SAW) dan Borda dalam Sistem Rekomendasi Pemilihan Pariwisata di Kabupaten Jember; Hafiz Ardi Hidayat, 162410101050; 2020, 251 halaman; Program Studi Sistem Informasi Universitas Jember.

Pariwisata merupakan segala bentuk perusahaan obyek, usaha pariwisata, dan daya tarik yang berhubungan dengan wisata. Sektor pariwisata di Indonesia saat ini memiliki perkembangan yang sangat pesat. Perkembangan sektor pariwisata menyebabkan peningkatan pengunjung pariwisata yang akan berdampak juga pada peningkatan devisa negara, namun, terdapat beberapa kendala dalam sector pariwisata salah satunya yaitu ketersediaan informasi.

Informasi terkait objek wisata bagi wisatawan sangatlah penting agar wisatawan tidak salah memilih tujuan tempat wisata terutama ketika wisatawan belum pernah mengunjungi daerah tersebut. Informasi terkait objek wisata yang akan dikunjungi serta jarak objek wisata juga ikut menjadi pertimbangan para wisatawan sebelum mengunjungi suatu daerah, namun pada saat ini, ketersediaan informasi akan wisata masih kurang dan masih harus mencari satu persatu untuk mendapatkan suatu informasi tersebut. Oleh karena itu dibutuhkan sebuah system rekomendasi wisata yang bisa menjadi sarana pencarian informasi dan bisa mendapatkan hasil rekomendasi wisata didalam sistem berdasarkan dengan kebutuhan yang diinginkan oleh wisatawan itu sendiri.

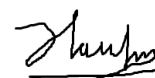
PRAKATA

Puji syukur kehadiran Allah swt yang telah memberikan rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan skripsi yang berjudul “Penerapan Metode Simple Additive Weighting (SAW) dan Borda Dalam Sistem Rekomendasi Pemilihan Pariwisata di Kabupaten Jember”. Skripsi ini disusun untuk memenuhi salah satu syarat menyelesaikan pendidikan Strata Satu (S1) pada Program Studi Sistem Informasi Universitas Jember.

Penyusunan skripsi ini tidak lepas dari bantuan berbagai pihak. Oleh karena itu, penulis menyampaikan terimakasih pada :

1. Diah Ayu Retnani Wulandari S.T.,M.Eng selaku Dosen Pembimbing Utama dan M.Arief Hidayat, S.Kom, M.Kom, selaku Dosen Pembimbing Anggota dan Dosen Pembimbing Akademik;
2. Ayahanda Triadi dan Ibunda Arifah tercinta;
3. Sahabat-sahabatku yang telah memberikan semangat;
4. Teman-teman seperjuangan INFINITY angkatan 2016;
5. Keluarga besar Badan Perwakilan Mahasiswa (BPM) periode 2017-2018;
6. Semua pihak yang tidak dapat disebutkan satu persatu;

Jember, 2020



Penulis

DAFTAR ISI

PERSEMBAHAN	iii
MOTTO.....	iv
PERNYATAAN	v
SKRIPSI	vi
PENGESAHAN PEMBIMBING.....	vii
PENGESAHAN PENGUJI.....	viii
RINGKASAN	ix
PRAKATA	x
DAFTAR ISI.....	xi
DAFTAR GAMBAR	xiv
DAFTAR TABEL	xvi
BAB 1 PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	4
1.3. Tujuan dan Manfaat	4
1.4. Batasan Masalah	6
BAB 2 TINJAUAN PUSTAKA.....	7
2.1. Penelitian Terdahulu.....	7
2.2. Metode Simple Additive Weighting (SAW).....	10
2.3. Metode Borda.....	11
2.4. Destinasi Wisata	13
2.5. Sistem Rekomendasi	13
2.6. Penginapan Atau Hotel.....	13
2.7. Kabupaten Jember.....	14

BAB 3	METODOLOGI PENELITIAN.....	15
3.1.	Jenis Penelitian.....	15
3.2.	Deskripsi Umum Sistem	15
3.3.	Tahapan Penelitian	16
3.3.1.	Studi Permasalahan.....	17
3.3.2.	Studi Literatur	17
3.3.3.	Pengumpulan Data Dan Akuisisi Pengetahuan.....	17
3.3.4.	Pengolahan Data	18
3.3.5.	Penerapan Metode SAW Dan Borda.....	18
3.3.6.	Pengembangan Sistem	19
3.3.7.	Pengujian Sistem.....	21
3.3.8.	Hasil Dan Analisa	22
3.4.	Kesimpulan	22
BAB 4	PENGEMBANGAN SISTEM.....	23
4.1.	Analisis Kebutuhan Sistem	23
4.1.1.	Statement Of Purpose (SOP)	24
4.1.2.	Kebutuhan Fungsional	27
4.1.3.	Kebutuhan Non-Fungsional.....	28
4.1.4.	Fungsi Sistem.....	28
4.2.	Desain Sistem.....	29
4.2.1.	Bagan Input, Proses, Output.....	29
4.2.2.	Usecase Diagram	30
4.2.3.	Skenario Sistem	35
4.2.4.	Sequence Diagram	40
4.2.5.	Activity Diagram	45
4.2.6.	Class Diagram.....	49
4.2.7.	Entity Relationship Diagram.....	49
4.3.	Pengujian Sistem.....	51

4.3.1. Pengujian <i>White Box</i>	51
4.3.2. Pengujian <i>Black Box</i>	79
BAB 5 HASIL DAN PEMBAHASAN.....	81
5.1. Hasil Penelitian	81
5.1.1. Fitur Trip	81
5.2. Pembahasan.....	85
BAB 6 PENUTUP.....	124
DAFTAR PUSTAKA	128
LAMPIRAN	130
Lampiran A	130
Lampiran B	189
Lampiran C	224
Lampiran D Pengujian <i>Black Box</i>	261

DAFTAR GAMBAR

Gambar 2.1 Perhitungan metode Borda	12
Gambar 3.1 Tahapan Penelitian	16
Gambar 3.2 Tahapan <i>Waterfall</i> (Mitch Kramer, 2018)	20
Gambar 4.1 <i>Input, Proses, Output</i>	30
Gambar 4.2 <i>Usecase Diagram</i>	31
Gambar 4.3 <i>Sequence Diagram Login Sistem</i>	41
Gambar 4.4 <i>Activity Diagram Login Sistem</i>	48
Gambar 4.5 <i>Entity Diagram</i>	49
Gambar 4.6 <i>Class Diagram</i>	50
Gambar 4.7 <i>Listing Program function __construct</i>	52
Gambar 4.8 <i>Listing Program function index</i>	52
Gambar 4.9 <i>Listing Program function detailWisata</i>	52
Gambar 4.10 <i>Listing Program function detailHotel</i>	53
Gambar 4.11 <i>Listing Program function detailKuliner</i>	53
Gambar 4.12 <i>Listing Program fuction dostartroute</i>	53
Gambar 4.13 Pengecekan Kriteria	54
Gambar 4.14 Melakukan Penentuan Penggunaan Metode.....	55
Gambar 4.15 Menghitung menggunakan metode Borda	56
Gambar 4.16 Diagram Alir <i>function __construct</i>	57
Gambar 4.17 Diagram Alir <i>function index</i>	57
Gambar 4.18 Diagram Alir <i>function detailWisata</i>	58
Gambar 4.19 Diagram Alir <i>function detailHotel</i>	58
Gambar 4.20 Diagram Alir <i>function detailKuliner</i>	59
Gambar 4.21 Diagram Alir <i>function formTour</i>	59
Gambar 4.22 Diagram Alir <i>function dostartroute</i>	60

Gambar 4.23 Diagram Alir Pengecekan Kriteria	61
Gambar 4.24 Diagram Alir Penentuan Penggunaan Metode	67
Gambar 4.25 Diagram Alir Penggunaan metode Borda.....	72
Gambar 5.1 <i>startTouring</i>	82
Gambar 5.2 Wisata Model.....	83
Gambar 5.3 Hotel Model.....	83
Gambar 5.4 Kuliner Model	84
Gambar 5.5 Range Model	84
Gambar 5.6 Data Prediksi Model.....	84
Gambar 5.7 Tampilan Fitur Trip	85
Gambar 5.8 Menentukan Kriteria.....	94
Gambar 5.9 Menentukan Kriteria yang Berbeda	103
Gambar 5.10 Form Kelompok Wisatawan.....	106
Gambar 5.11 Hasil Rekomendasi Wisata	117
Gambar 5.12 Pemilihan Kriteria yang Berbeda	118

DAFTAR TABEL

Tabel 4.1 Data Tempat Wisata.....	25
Tabel 4.2 Data Tempat Hotel dan Kuliner.....	26
Tabel 4.3 Data Lokasi Awal.....	26
Tabel 4.4 Definisi Aktor	32
Tabel 4.5 Definisi <i>Usecase</i>	32
Tabel 4.6 Mengelola Data <i>User Profile</i>	38
Tabel 4.7 <i>Test Case construct</i>	75
Tabel 4.8 <i>Test Case index</i>	76
Tabel 4.9 <i>Test Case detailWisata</i>	76
Tabel 4.10 <i>Test Case detailHotel</i>	77
Tabel 4.11 <i>Test Case detaiKuliner</i>	77
Tabel 4.12 <i>Test Case formTour</i>	77
Tabel 4.13 <i>Test Case dostartroute</i>	78
Tabel 4.14 <i>Test Case startTouring</i>	78
Tabel 5.1 Kriteria, <i>Range</i> , Bobot, <i>Fuzzy</i>	86
Tabel 5.2 Data Objek Wisata.....	88
Tabel 5.3 Bobot Kriteria.....	91
Tabel 5.4 Normalisasi Kriteria	95
Tabel 5.5 Menentukan Nilai V	98
Tabel 5.6 Menentukan Ranking V.....	100
Tabel 5.7 Menentukan Nilai V	103
Tabel 5.8 Nilai V2	106
Tabel 5.9 Menentukan Ranking V2.....	109
Tabel 5.10 Tabel Nilai Akhir Borda	112
Tabel 5.11 Tabel Bobot dan Peringkat Borda.....	114

Tabel 5.12 Nilai V1	118
Tabel 5.13 Nilai V2	120
Tabel 5.14 Nilai Borda	122



BAB 1 PENDAHULUAN

Bab ini merupakan bab yang menjelaskan tentang latar belakang, rumusan masalah, tujuan dan manfaat, dan batasan masalah.

1.1. Latar Belakang

Pariwisata merupakan serangkaian kegiatan perjalanan untuk mengunjungi tempat wisata yang dapat dilakukan oleh perorangan maupun secara berkelompok (Soedarso dkk, 2014). Ruang lingkup pariwisata telah dijelaskan pada Undang-Undang Republik Indonesia nomor 9 Tahun 1990 Bab 1 pasal 1, Pariwisata merupakan segala bentuk pengusahaan obyek, usaha pariwisata, dan daya tarik yang berhubungan dengan wisata. Sektor pariwisata di Indonesia saat ini memiliki perkembangan yang sangat pesat. Perkembangan sektor pariwisata menyebabkan peningkatan pengunjung pariwisata yang akan berdampak juga pada peningkatan devisa negara. Sektor pariwisata menjadi salah satu sumber devisa negara, hal ini terbukti pada tahun 2010 sektor pariwisata menghasilkan sekitar \$5,7 triliun dengan mempekerjakan 235 juta orang sebagai bentuk usaha pariwisata yang telah dilakukan (Maslim, 2013). Kesimpulan dari pernyataan tersebut bahwa perkembangan sektor pariwisata sangat pesat dan mampu berpotensi lebih berkembang daripada sebelumnya. Salah satu kendala yang sering terjadi di sektor pariwisata yang dapat menghambat perkembangan sektor wisata adalah ketersediaan informasi.

Informasi terkait objek wisata bagi wisatawan sangatlah penting agar wisatawan tidak salah memilih tujuan tempat wisata terutama ketika wisatawan belum pernah mengunjungi daerah tersebut. Informasi terkait objek wisata yang akan dikunjungi serta jarak objek wisata juga ikut menjadi pertimbangan para

wisatawan sebelum mengunjungi suatu daerah. Menurut Sunaki dkk (2015), kebutuhan akan informasi memicu seorang wisatawan untuk melakukan pencarian informasi. Informasi terkini mengenai tempat wisata menjadi kebutuhan bagi para wisatawan, karena wisatawan sendiri tidak ingin mengunjungi tempat wisata yang tidak sesuai dengan keinginan dan kebutuhan dirinya.

Pencarian informasi tentang objek wisata oleh wisatawan pada umumnya melalui media *Google* dengan mencari satu persatu objek wisata yang diinginkannya, dan dapat melalui aplikasi-aplikasi yang telah menyediakan informasi yang berkaitan dengan pariwisata dari yang hanya menyediakan berupa informasi objek wisata hingga aplikasi berbasis sistem penunjang keputusan untuk menjadi sarana menyediakan informasi yang lebih kompleks bagi wisatawan. Mayoritas aplikasi-aplikasi tersebut hanya menyediakan satu jenis informasi saja, seperti halnya menyediakan informasi tentang penginapan, kuliner, tetapi tidak menjadi satu dengan informasi tempat objek wisata. Hal tersebut memerlukan waktu yang cukup lama terlebih lagi informasi yang di dapat kurang lengkap serta untuk memilih destinasi objek wisata yang menurut wisatawan menarik juga membutuhkan waktu yang tidak sedikit. Penentuan objek wisata yang dilakukan secara berkelompok oleh kelompok wisatawan juga dirasa masih menjadi kendala untuk saat ini akibat dari perbedaan pendapat dari masing-masing individu tentang objek wisata yang ingin mereka kunjungi, sehingga sangat dibutuhkan suatu sistem yang dapat menyediakan satu paket informasi dan dapat memberikan rekomendasi objek wisata baik untuk kebutuhan wisatawan secara individu maupun secara berkelompok.

Perkembangan sektor pariwisata di Kabupaten Jember, berdasarkan data Badan Pusat Statistika Tahun 2017, menunjukkan bahwa Jumlah objek wisata yang terdapat di Jember sebanyak 60 objek wisata menurut jarak dan jenisnya.

Wisata yang ada di Jember antara lain pantai, air terjun, agrowisata, pemandian, wisata religi dan lainnya. Kondisi ini, membuktikan bahwa Kabupaten Jember memiliki potensi untuk menarik wisatawan datang, akan tetapi Kabupaten Jember juga memiliki permasalahan yang sama dalam hal penyediaan informasi bagi para wisatawan untuk mengenal lebih jauh wisata-wisata yang ada di Jember sendiri.

Berdasarkan permasalahan tersebut maka perlu dibangun sebuah sistem yang bertujuan untuk mengatasi permasalahan-permasalahan tersebut. Sistem ini dibuat dengan menggabungkan kedua metode yaitu metode Simple Additive Weighting (SAW) dan metode Borda. Penggabungan kedua metode ini dilakukan karena terkait dari karakteristik atau fungsi dari kedua metode tersebut jika dijadikan satu maka akan dapat memenuhi kebutuhan wisatawan baik secara individu maupun berkelompok. Metode SAW merupakan metode yang memberikan kriteria-kriteria tertentu yang memiliki bobot nilai masing-masing sehingga hasil penjumlahan bobot tersebut akan diperoleh hasil yang menjadi keputusan akhirnya (Frieyadi, 2016), sedangkan metode Borda merupakan metode yang dipakai untuk menetapkan peringkat pada pemungutan suara berguna untuk wisatawan yang berkelompok dengan masing-masing anggota kelompok memiliki keragaman keinginan dan tujuan wisata masing-masing sehingga dibutuhkan pemungutan suara untuk mencapai keputusan wisata apa yang dituju. Metode Borda tidak dapat berdiri sendiri melainkan membutuhkan metode-metode sebelumnya dalam perhitungannya (Saputra dan Wardoyo, 2017).

Kriteria dan cara kerja kedua algoritma yang ada pada penelitian terdahulu tersebut sangat cocok untuk digunakan dalam pengembangan sistem ini karena, di dalam sistem menggunakan kriteria-kriteria dalam penilaiannya, kriteria tersebut adalah jarak, harga dan rating. Kriteria-kriteria tersebut sangat cocok diimplementasikan dengan menggunakan metode SAW dikarenakan sifat-sifat

kriteria tersebut termasuk ke dalam atribut keuntungan (benefit) maupun atribut biaya (cost) yang hanya terdapat dalam metode SAW sendiri, serta dalam sistem juga menyediakan pengambilan keputusan secara kelompok dengan menggunakan metode Borda yang juga sangat cocok untuk diimplementasikan pada sistem ini. Karena baik kelompok wisatawan maupun wisatawan yang datang secara individu dapat memilih jenis objek wisata berdasarkan kebutuhan masing-masing. Dengan adanya sistem ini, diharapkan dapat membantu mempermudah wisatawan dalam menentukan pemilihan wisata yang telah ada khususnya di Kabupaten Jember.

1.2. Rumusan Masalah

Berdasarkan uraian yang telah disampaikan pada latar belakang, permasalahan yang harus diselesaikan pada penulisan ini, yaitu:

1. Bagaimana cara menentukan rekomendasi pemilihan pariwisata yang berada di Kabupaten Jember dengan banyak kriteria yang dapat dipilih di dalam sebuah sistem rekomendasi pariwisata?
2. Bagaimana merancang dan membangun suatu sistem untuk pemilihan pariwisata berdasarkan kriteria-kriteria tertentu dengan menggunakan metode SAW dan Borda?

1.3. Tujuan dan Manfaat

Tujuan dan manfaat yang dipaparkan merupakan hasil yang ingin dicapai dan dapat digunakan dari berbagai aspek penelitian ini.

D1. Tujuan

Adapun tujuan yang ingin dicapai dalam penelitian ini meliputi:

1. Membangun sebuah sistem rekomendasi pemilihan pariwisata yang disesuaikan dengan kriteria jarak, harga, dan rating yang diinginkan oleh wisatawan sehingga dapat membantu wisatawan dalam pemilihan pariwisata yang berada di Kabupaten Jember.
2. Menghasilkan prediksi dan rekomendasi pemilihan pariwisata di Kabupaten Jember menggunakan metode Simple Additive weighting (SAW) dan metode Borda.
3. Menjelaskan langkah-langkah perhitungan dari metode Simple Additive weighting (SAW) dan metode Borda.

D2. Manfaat

Manfaat yang dihasilkan dari penelitian ini yaitu :

1. Bagi Peneliti

Melatih kemampuan dan menerapkan ilmu pengetahuan yang telah diperoleh selama masa studi untuk membangun sebuah sistem terintegrasi sehingga dapat memberikan informasi-informasi terkait rekomendasi pemilihan pariwisata.

2. Bagi Dinas Pariwisata Kabupaten Jember

Dapat membantu Dinas Pariwisata Kabupaten Jember untuk mengarahkan wisatawan kepada destinasi wisata yang telah diberikan oleh sistem.

3. Bagi Pengguna

Membantu para wisatawan untuk mendapatkan rekomendasi informasi-informasi terkait wisata yang ada di kabupaten Jember.

1.4. Batasan Masalah

Batasan masalah yang ditentukan dalam penelitian ini adalah sebagai berikut:

1. Sistem ini menghasilkan rekomendasi pemilihan pariwisata berdasarkan kriteria yang dipilih yaitu jarak, harga, serta rating.
2. Sistem ini difokuskan bagi wisatawan dari luar Kabupaten Jember yang ingin mengetahui wisata yang berada di Kabupaten Jember, sehingga untuk jarak titik awalnya dari Bandara, terminal serta perbatasan kota Jember dengan kota yang lain.

BAB 2 TINJAUAN PUSTAKA

Tinjauan pustaka merupakan bagian yang menjelaskan teori-teori yang dapat digunakan untuk penelitian. Teori-teori ini diambil dari berbagai sumber seperti halnya jurnal, literatur maupun internet. Berikut ini teori-teori yang dibahas meliputi tentang.

2.1. Penelitian Terdahulu

Penelitian yang dilakukan oleh Ismanto dan Effendi (2017) dengan judul “Sistem Pendukung Keputusan Penerimaan Karyawan Dengan Metode Simple Additive Weighting (SAW)”. Penelitian ini bertujuan untuk mendapatkan orang yang tepat bagi suatu jabatan tertentu, dapat mampu bekerja secara optimal dan mampu bertahan lama. Penelitian ini memiliki 8 kriteria yaitu pendidikan, pengalaman kerja, penampilan, test, wawancara, usia, status dan alamat, dimana 8 kriteria tersebut didapat dari pengumpulan data penelitian dilapangan. Hasil penelitian ini didapatkan bahwa metode SAW dapat diimplementasikan dalam sebuah sistem rekomendasi pemilihan calon karyawan baru namun kriteria telah ditetapkan secara statis sehingga tidak dapat di rubah kembali, maka dalam penelitian kali ini peneliti mencoba untuk memberikan kriteria yang dapat diubah secara dinamis sehingga pengguna mendapatkan kebebasan untuk memilih sendiri.

Penelitian selanjutnya yang dilakukan oleh Sukerti (2018) yang berjudul “Sistem Pendukung Keputusan Menggunakan Simple Additive Weighting Methode (SAW) dalam Merekomendasikan Objek Wisata di Pulau Nusa Penida”. Penelitian ini bertujuan untuk meningkatkan kunjungan wisatawan dan memudahkan wisatawan untuk memilih objek wisata di pulau Nusa Penida. Terdapat 5 kriteria dalam penelitian ini yaitu biaya wisata, jarak tempuh, fasilitas

wisata, waktu kunjungan dan rating wisata dimana masing-masing kriteria memiliki nilai bobot-bobot tersendiri. Penelitian ini hanya meneliti tentang bagaimana sebuah sistem dapat membantu membuat keputusan pemilihan objek wisata dengan menggunakan metode SAW. Hasil penelitian ini yaitu keberhasilan penerapan SAW sehingga dapat diimplementasikan untuk dijadikan sistem pendukung keputusan pemilihan objek wisata. Namun, dalam penelitian ini kriteria yang tersedia bersifat statis karena peneliti sendiri yang menentukan tingkat penilaian pembobotan dalam suatu kriteria, hal ini menyebabkan kurangnya kebebasan user dalam mengambil suatu kriteria. Pada penelitian ini peneliti akan menggunakan atau mengambil hanya 3 kriteria utama yang menjadi dasar untuk melakukan rekomendasi wisata yaitu kriteria jarak, harga dan rating.

Pada penelitian yang dilakukan oleh Cahyana dan Aribowo (2014) dengan judul “Group Decision Support System (GDSS) Untuk Menentukan Prioritas Proyek”. Penelitian ini menggunakan gabungan metode *Weighted Product* (WP) dengan metode Borda. Penelitian ini bertujuan untuk menentukan prioritas proyek sehingga proyek dapat dikerjakan secara efisien dan efektif. Hasil penelitian ini didapatkan bahwa metode borda dapat dikombinasikan dengan baik pada metode *Weighted Product* (WP). Penelitian selanjutnya yang dilakukan oleh Oei (2013) dengan judul “Group Decision Support System Untuk Pembelian Rumah dengan Menggunakan Analytical Hierarchy Process (AHP) dan Borda”. Tujuan penelitian ini yaitu untuk mempercepat dan mempermudah proses pemilihan rumah mana yang paling optimal dimana kriteria-kriteria telah ditentukan sebelumnya. Terdapat 4 kriteria yang digunakan dalam penelitian ini yaitu lokasi, harga, fasilitas, dan developer. Hasil dari penelitian ini yaitu keberhasilan dalam metode AHP dan borda untuk menentukan hasil perhitungan persepsi yang berbeda dari setiap anggota keluarga calon pembeli rumah, dan Borda dapat dikombinasikan

dengan baik dengan metode AHP. Dalam kedua penelitian terdahulu ini dapat disimpulkan bahwa metode borda dapat di kombinasikan dengan berbagai metode lainnya, oleh karena itu pada penelitian selanjutnya maka akan dilakukan kombinasi antara metode borda dengan metode yang belum dilakukan sebelumnya yaitu akan dikombinasikan dengan metode SAW.

Berdasarkan beberapa penelitian terdahulu seperti yang telah dijabarkan diatas dapat disimpulkan bahwa mayoritas penggunaan metode SAW yang telah dilakukan oleh peneliti sebelumnya seperti dalam penelitian ismanto dan sukerti, penelitian dilakukan secara statis dalam penentuan peringkat kepentingan kriteria nya sehingga *user* tidak diberikan kebebasan untuk merangking kriterianya sendiri, oleh karena itu dalam penelitian ini peneliti mencoba untuk memberikan jenis kriteria yang bersifat dinamis yang sewaktu-waktu dapat dirubah nilainya sehingga nilai prediksinya juga dapat berubah-ubah, selain itu, dalam penelitian ini mengambil beberapa kriteria yang diambil berdasarkan penelitian oleh Sukerti diantaranya kriteria jarak, harga dan rating yang harus dapat diseimbangkan dengan adanya jumlah pengguna yang seimbang dengan jumlah objek pariwisata yang ditawarkan. Kemudian peneliti menggunakan metode borda dikarenakan dalam penelitian ini akan disediakan pemilihan rekomendasi wisata baik untuk individu maupun kelompok wisatawan dan dalam penelitian terdahulu seperti halnya penelitian yang dilakukan oleh vania, dan oei berdarakan penelitiannya, metode borda memberikan hasil yang optimal dalam pengambilan keputusan berkelompok serta masih belum adanya penggabungan antara metode SAW dengan metode Borda, mayoritas peneliti menggabungkan metode Borda dengan AHP maupun WP.

2.2. Metode Simple Additive Weighting (SAW)

Menurut Fishburn dan MacCrimmon (2013) mengemukakan bahwa Metode Simple Additive Weight (SAW), sering juga dikenal dengan istilah metode penjumlahan terbobot. Metode SAW membutuhkan proses normalisasi matriks keputusan (X) ke suatu skala yang dapat diperbandingkan dengan semua rating alternatif yang ada (Eniyati, 2011). Adapun langkah-langkah metode *Simple Additive Weighting* (SAW) menurut Fishburn dan MacCrimmon dalam (Munthe, 2013) adalah sebagai berikut:

1. Menentukan kriteria yang akan menjadi acuan dalam perhitungan.
2. Menentukan rating kecocokan setiap alternatif dengan cara memasukkan nilai dari masing-masing kriteria kedalam tabel rating kecocokan.
3. Membuat matriks keputusan, nilai dari hasil rating kecocokan kemudian dijadikan matriks keputusan.
4. Melakukan normalisasi matriks, dengan mengelompokkan kriteria kriteria tersebut kedalam dua jenis atribut yaitu atribut keuntungan (*benefit*) yaitu jika nilai terbesar adalah nilai terbaik dan atribut biaya (*cost*) jika nilai terkecil adalah nilai terbaik seperti yang terlihat dari persamaan ke-1 sebagai berikut

$$R_{ij} \begin{cases} \frac{X_{ij}}{\text{Max}X_{ij}} & \text{jika } j \text{ adalah atribut keuntungan (benefit)} \\ \frac{\text{Min}X_{ij}}{X_{ij}} & \text{jika } j \text{ adalah atribut biaya (cost) } \dots \dots \dots \end{cases} \quad (2.1)$$

Dimana:

- R_{ij} = Rating kinerja ternormalisasi
- $\text{Max}X_{ij}$ = Nilai maksimum dari setiap baris dan kolom
- $\text{Min}X_{ij}$ = Nilai minimum dari setiap baris dan kolom
- X_{ij} = Baris dan kolom dari matriks

5. Setelah melakukan normalisasi matriks seperti yang terdapat pada persamaan ke-1, kemudian mencari hasil akhir yang diperoleh dari penjumlahan dari perkalian matriks ternormalisasi R dengan bobot (W) seperti yang terdapat dalam persamaan ke-2 sebagai berikut.

$$V_i = \sum_{j=1}^n W_j R_{ij} \dots \dots \dots (2.2)$$

Dimana:

- V_i = Nilai akhir dari alternatif
- W_j = Bobot yang telah ditentukan
- R_{ij} = Normalisasi matriks

2.3. Metode Borda

Borda adalah metode yang digunakan untuk keputusan kelompok yang dilakukan dengan perkalian dari nilai referensinya dengan bobot dari rangking. (Vania, *et al*, 2017). Metode Borda umumnya digunakan untuk pengambilan keputusan secara *voting*, dan dalam penerapannya metode Borda selalu dikombinasikan dengan metode lainnya. Contoh perhitungan metode Borda untuk

kasus perangkaan proyek menurut Cahyana dan Aribowo (2014) adalah sebagai berikut:

- a) Dari hasil kuisisioner penilaian proyek dihitung dengan jumlah responden yang menyatakan ranking untuk tiap proyek. Misalnya terdapat 4 responden yang menyatakan proyek A berada di peringkat 2 dan 3 responden menyatakan proyek A berada di peringkat 3, maka tuliskan angka 4 pada kolom proyek A peringkat 2 dan 3 pada kolom proyek A peringkat 3. Hal yang sama dilakukan untuk jenis yang lain.
- b) Berikan bobot borda pada masing-masing peringkat dengan ketentuan memberikan nilai dari yang besar ke yang terkecil pada peringkat pertama dan seterusnya. Kalikan angka pada kolom peringkat dengan bobot yang telah diberikan, kemudian tambahkan dengan hasil perkalian pada proyek yang sama, kemudian isikan hasilnya pada kolom ranking. Misalnya untuk proyek A $(0 \times 2) + (4 \times 1) + (3 \times 0) = 4$.
- c) Jumlahkan hasil ranking, yang dalam contoh ini berarti $4 + 11 + 5 = 20$
- d) Untuk mencari bobot tiap proyek, bagi ranking dengan jumlah total ranking, misalkan Proyek A = $4/20 = 0,2$ dan seterusnya.
- e) Proyek dengan bobot tertinggi merupakan yang terpilih untuk mendapatkan prioritas utama.

T

Proyek	Peringkat			Score Akhir	Bobot Setelah Normalisasi	Rangking
	1	2	3			
A	0	4	3	4	0.2	3
B	5	1	1	11	0.55	1
C	1	3	3	5	0.25	2
Bobot Peringkat	2	1	0			

2.1 Perhitungan metode Borda

2.4. Destinasi Wisata

Berdasarkan Undang-Undang Republik Indonesia No. 10 tahun 2009 Tentang kepariwisataan, Destinasi Pariwisata adalah kawasan geografis yang didalamnya terdapat daya tarik wisata, fasilitas umum, fasilitas pariwisata, aksesibilitas, serta masyarakat yang saling terkait dan melengkapi terwujudnya kepariwisataan.

2.5. Sistem Rekomendasi

Sistem rekomendasi adalah suatu sistem yang mencoba untuk memprediksi atau merekomendasikan suatu hal berdasarkan informasi yang diperoleh dari pengguna (Priyono, 2016). Sistem rekomendasi dapat membantu para pengguna untuk memberikan saran suatu informasi yang dibutuhkan dengan cara menyimpulkan apa yang pengguna butuhkan berdasarkan kegiatan sebelumnya.

2.6. Penginapan Atau Hotel

Menurut Dirjen Pariwisata (1998) pengertian hotel adalah suatu jenis akomodasi yang menyediakan jasa penginapan dan dikelola secara komersial. Menurut Surat Keputusan Menteri Perhubungan R.I No. PM 10/PW – 301/Phb.

77, tanggal 12 Desember 1977 hotel adalah suatu bentuk akomodasi atau fasilitas yang dikelola secara komersial, disediakan untuk setiap orang yang membutuhkan penginapan, makanan maupun minuman.

2.7. Kabupaten Jember

Kabupaten Jember adalah kabupaten di Provinsi Jawa Timur yang berbatasan dengan Kabupaten Probolinggo dan Kabupaten Bondowoso di sebelah utara, Kabupaten Banyuwangi di Timur, Samudera Hindia di Selatan dan Kabupaten Lumajang di Barat. Berdasarkan Staatsblad No. 46/1941 tanggal 1 Maret 1941 wilayah distrik di kabupaten Jember dipecah menjadi 25 onderdistrik yaitu:

- Distrik Jember, meliputi onderdistrik Jember, Wirolegi, dan Arjasa
- Distrik Kalisat, meliputi onderdistrik Kalisat, Ledokombo, Sumberjambe, dan Sukowono.
- Distrik Rambipuji, meliputi onderdistrik Rambipuji, Panti, Mangli, dan Jenggawah.
- Distrik Mayang, meliputi onderdistrik Mayang, Silo, Mumbulsari, dan Tempurejo.
- Distrik Tanggul, meliputi onderdistrik Tanggul, Sumberbaru, dan Bangsalsari.
- Distrik Puger, meliputi onderdistrik Puger, Kencong Gumukmas, dan Umbulsari.
- Distrik Wuluhan, meliputi onderdistrik Wuluhan, Ambulu, dan Balung.



BAB 3 METODOLOGI PENELITIAN

Bagian ini menjelaskan proses atau tahapan yang sistematis untuk mendapatkan suatu data yang digunakan untuk keperluan penelitian sehingga dapat mencapai tujuan sebenarnya dari penelitian. Adapun tahap-tahap metodologi penelitian yaitu:

3.1. Jenis Penelitian

Tahap ini merupakan tahap dimana peneliti menggunakan metode penelitian kuantitatif. Metode penelitian kuantitatif adalah metode yang menggunakan data-data angka yang digunakan untuk menganalisis suatu penelitian. Data-data angka yang dimaksud yaitu seperti halnya data jarak, data rating dan data harga wisata, hotel maupun kuliner serta data-data perhitungan prediksi.

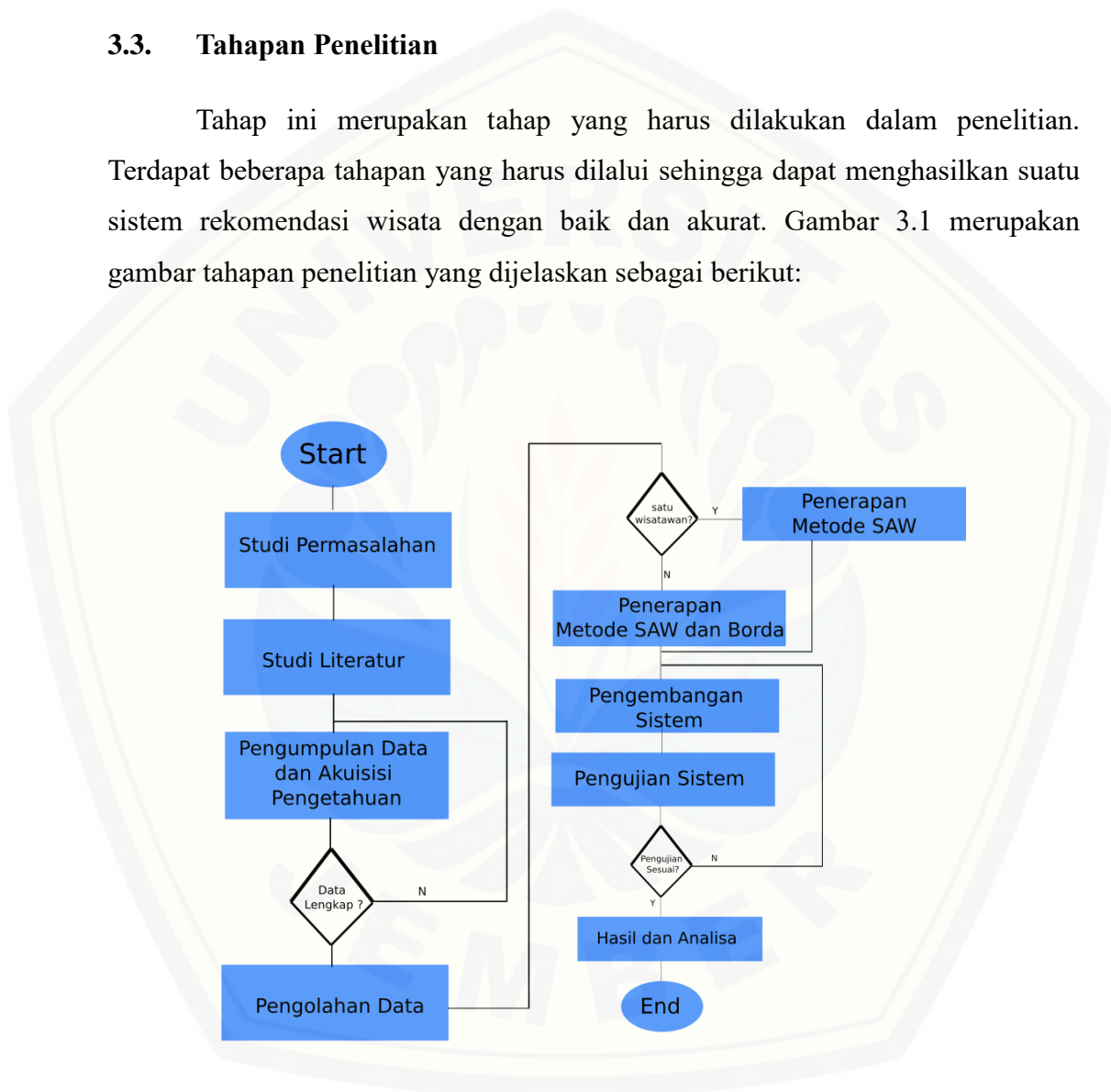
3.2. Deskripsi Umum Sistem

Sistem ini menggunakan bahasa pemrograman PHP dengan *framework* laravel sebagai pendukungnya. Sistem ini mempunyai fitur utama yaitu rekomendasi pemilihan-pemilihan wisata berdasarkan kebutuhan-kebutuhan yang telah ditetapkan sebelumnya oleh *user*. Kebutuhan-kebutuhan tersebut antara lain, kebutuhan akan wisata, kuliner, penginapan, jarak, harga serta rating. Sistem ini menerapkan metode Simple Additive Weighting (SAW) dan metode Borda untuk menghasilkan rekomendasi wisata baik untuk kebutuhan kelompok wisatawan maupun individu. Metode SAW dalam sistem ini digunakan untuk menghitung hasil dari kriteria-kriteria yang telah diinputkan oleh pengguna, sedangkan metode borda digunakan sebagai lanjutan perhitungan setelah metode SAW telah dilakukan. Perhitungan dengan metode borda dilakukan ketika terdapat lebih dari

satu pengguna yang menginputkan kriteria yang berbeda-beda sehingga diharapkan mendapatkan hasil rekomendasi terbaik dari suatu sistem.

3.3. Tahapan Penelitian

Tahap ini merupakan tahap yang harus dilakukan dalam penelitian. Terdapat beberapa tahapan yang harus dilalui sehingga dapat menghasilkan suatu sistem rekomendasi wisata dengan baik dan akurat. Gambar 3.1 merupakan gambar tahapan penelitian yang dijelaskan sebagai berikut:



Gambar 3.1 Tahapan Penelitian

3.3.1. Studi Permasalahan

Studi permasalahan merupakan tahap awal dalam penelitian. Studi masalah didapatkan dari melihat, mencari permasalahan yang ada di sekitar kita mengenai objek yang akan diteliti, solusi dari permasalahan tersebut serta bagaimana penyelesaiannya. Studi permasalahan dalam penelitian ini pada awalnya melihat dari Kabupaten Jember yang memiliki potensi wisata yang cukup untuk menarik wisatawan di karenakan Jember dinobatkan sebagai kota festival karena JFC (Jember Fashion Carnaval), namun sampai sekarang Jember masih belum ada sebuah peningkatan dalam penanganan wisata, sehingga masih banyak wisatawan yang belum mengetahui potensi yang dimiliki oleh Jember. Berdasarkan permasalahan tersebut, maka peneliti menjadikan hal tersebut untuk dijadikan objek penelitian.

3.3.2. Studi Literatur

Studi Literatur merupakan tahapan dimana dilakukannya pencarian referensi teori yang relevan dengan permasalahan yang dihadapi. Referensi dapat berupa buku, jurnal, artikel maupun informasi yang tersedia pada internet.

3.3.3. Pengumpulan Data Dan Akuisisi Pengetahuan

Tahap ini merupakan tahap pengumpulan data dengan cara wawancara dan survey yang ditujukan kepada responden yaitu Dinas Pariwisata Jember. Wawancara adalah teknik pengambilan data atau informasi dengan cara melakukan pengajuan pertanyaan kepada pihak yang bersangkutan, sedangkan survey dilakukan dengan cara mengunjungi langsung objek wisata yang ada di Jember. Pengumpulan data juga diambil pada *google maps* untuk mengetahui rating dan jarak, data di ambil pada bulan Januari dan terdapat kemungkinan

adanya perubahan data tersebut mengingat data tersebut diambil pada *google maps*. Pengumpulan data dikatakan lengkap ketika data objek wisata, penginapan, kuliner, jarak, harga, dan rating telah terpenuhi. Setelah pengumpulan data dilakukan, maka dilakukan akuisisi pengetahuan dengan cara :

a. Identifikasi

Identifikasi yaitu menganalisis atau menentukan komponen-komponen utama dalam penelitian yang akan dibangun.

b. Konseptualisasi

Konseptualisasi merupakan pembangunan atau perancangan konsep penelitian dari data-data yang telah didapatkan sebelumnya.

3.3.4. Pengolahan Data

Tahap pengolahan data ini menjelaskan tentang bagaimana proses pengolahan data berupa data pariwisata, penginapan, dan tempat kuliner. Data-data tersebut pada mulanya di kelompokkan terlebih dahulu dengan cara menghitung jarak pada masing-masing data dengan lokasi awal dari wisatawan. Selanjutnya, melakukan perhitungan manual dengan rumus metode SAW dan Borda yang akan digunakan untuk mencocokkan hasil akhir rekomendasi wisata dengan hasil akhir rekomendasi wisata yang ada di sistem.

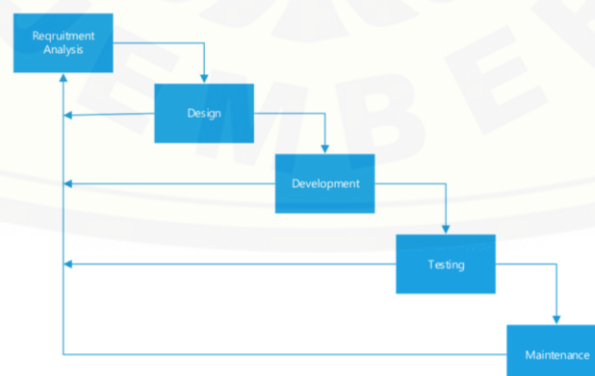
3.3.5. Penerapan Metode SAW Dan Borda

Tahap ini menjelaskan langkah-langkah bagaimana menerapkan metode SAW dan borda dengan data yang telah di dapat dan telah diolah sebelumnya. Setelah data diolah pada tahap pengolahan data, maka data-data tersebut digunakan untuk melakukan perhitungan metode SAW terlebih dahulu. Dalam

metode SAW data dikelompokkan dalam beberapa kriteria terlebih dahulu, kemudian memberikan bobot pada tiap-tiap kriteria, Setelah itu, dilakukan normalisasi matriks data dengan bobot kriteria tersebut untuk menentukan nilai akhir dari alternatif (nilai V), kemudian nilai akhir tersebut dilakukan perangkingan pada setiap data wisata tersebut. Kemudian ketika *output* dari metode SAW lebih dari satu nilai akhir (V), maka akan dilanjutkan dengan perhitungan menggunakan metode Borda. Pada metode Borda ini, setiap nilai V pada tiap list wisata akan di data jumlah dan peringkat kepentingannya, setelah itu memberikan bobot tiap peringkat, setelah itu melakukan total jumlah peringkat dengan bobotnya kemudian hasilnya akan dilakukan perangkingan ulang untuk dapat menghasilkan rekomendasi wisata berdasarkan hasil *voting* dari beberapa keinginan wisatawan.

3.3.6. Pengembangan Sistem

Metode yang digunakan dalam penelitian ini yaitu dengan menggunakan metode *waterfall*. Metode *waterfall* merupakan metode yang dalam melakukan tahapan-tahapannya harus dilakukan secara sistematis dan berurutan. Adapun tahapan dari metode *waterfall* seperti yang terdapat dalam gambar 3.2 sebagai berikut:



Gambar 3.2 Tahapan *Waterfall* (Mitch Kramer, 2018)

Penjelasan dari setiap tahapan pada gambar 3.2 adalah sebagai berikut:

1. Requirements dan Analysis

Tahap ini merupakan tahap pertama dalam metode *waterfall*, pada tahap ini merupakan tahap analisis untuk mendapatkan data yang dibutuhkan dalam membangun sistem. Analisis data dapat dilakukan dengan melakukan wawancara, survei maupun observasi. Proses wawancara dilakukan kepada pihak Dinas Pariwisata Jember, dan kepada calon wisatawan yang akan melakukan perjalanan wisata di kabupaten Jember.

2. System Design

System design merupakan tahap untuk menyiapkan, merancang dan membuat desain sistem yang akan di bangun. Desain sistem dapat membantu dalam mendefinisikan arsitektur sistem secara keseluruhan.

3. Development

Tahap ini merupakan tahap untuk melakukan pembuatan sistem sesuai dengan hasil yang telah dilakukan pada proses sebelumnya, yaitu hasil dari perancangan yang telah dibuat diimplementasikan kedalam baris-baris program, serta menerapkan metode SAW dan Borda. Adapun komponen pendukung yang digunakan dalam tahap pembuatan untuk sistem rekomendasi pemilihan wisata di kabupaten Jember adalah menggunakan bahasa pemrograman PHP dan database MYSQL.

4. Testing

Sistem yang telah dibuat pada tahap development akan dilakukan tahap pengujian sistem kepada pihak dinas pariwisata kabupaten Jember serta beberapa

pengguna untuk menghindari terjadinya kegagalan dan kesalahan dalam penentuan hasil akhirnya.

5. Maintenance

Tahap ini merupakan tahap dilakukannya pemeliharaan sistem secara berkala. Proses ini bertujuan untuk menghindari adanya kesalahan-kesalahan yang muncul seperti halnya kesalahan perhitungan pada metode SAW dan Borda maupun kesalahan dalam menginputkan nilai kriteria-kriteria yang dibutuhkan. Tahap maintenance berakhir ketika kesalahan tersebut telah teratasi.

3.3.7. Pengujian Sistem

Tahap ini merupakan proses pengujian sistem dengan menggunakan parameter-parameter yang telah tersedia. Tujuan dari pengujian sistem ini yaitu untuk menghindari atau memperbaiki kesalahan yang terjadi. Sutu sistem dapat dikatakan lolos pengujian ketika program yang dibuat, metode yang diimplementasikan dan hasil rekomendasi yang dibuat sistem sama dengan hasil dari perhitungan secara manual dari kedua metode. Pengujian sistem akan menggunakan dua metode, yakni metode *white box* dan metode *black box*.

1. *White Box Testing*

White Box Testing merupakan pengujian sistem yang dilakukan dengan cara melakukan pengecekan detail desain program, mengevaluasi apakah sistem dapat dijalankan sesuai dengan yang diharapkan atau tidak.

2. *Black Box Testing*

Black box testing merupakan pengujian sistem yang dilakukan dengan cara menguji fungsionalitas suatu program apakah bertentangan dengan struktur internal atau kerja dari suatu sistem.

3.3.8. Hasil Dan Analisa

Tahap ini merupakan tahap untuk mendapatkan hasil dari sistem yang telah dibuat dan akan melakukan analisis sistem yang telah dibuat apakah sistem yang dibangun dapat menampilkan rekomendasi pemilihan wisata yang sesuai dengan keinginan pengguna atau tidak, menganalisis apakah metode yang digunakan yaitu metode SAW dan borda dapat diimplementasikan kedalam sebuah sistem rekomendasi pemilihan wisata atau tidak.

3.4. Kesimpulan

Pada bagian ini menjelaskan tentang menyimpulkan penelitian yang telah dilakukan, apakah metode SAW dan metode Borda berhasil diterapkan dalam sistem rekomendasi pariwisata ataupun sebaliknya. Pada bagian ini juga menjelaskan faktor-faktor apa saja yang mendukung keberhasilan atau kegagalan dalam penelitian ini.

BAB 4 PENGEMBANGAN SISTEM

Bab ini akan menjelaskan tahapan-tahapan pengembangan sistem seperti halnya tahap analisis kebutuhan, desain, implementasi, dan pengujian sistem. Tahapan-tahapan tersebut merupakan tahapan-tahapan yang sesuai dengan model pengembangan SDLC Waterfall.

4.1. Analisis Kebutuhan Sistem

Tahap ini merupakan tahap awal dari tahap pengembangan sistem. Tahap analisis ini merupakan tahap menganalisis maupun pengumpulan data untuk mendapatkan kebutuhan fungsional maupun kebutuhan non fungsional dari sebuah sistem yang akan dibangun.

Pada penelitian ini, tahap awal untuk mendapatkan kebutuhan sistem dilakukan dengan melakukan wawancara kepada Dinas Kebudayaan dan Pariwisata Kabupaten Jember, selanjutnya mengumpulkan data-data kuantitatif meliputi data nama objek wisata dari Dinas Kebudayaan dan Pariwisata, data rating dan jarak dari *Google Maps* maupun dari survey harga yang dilakukan mulai tanggal 1 Januari 2020 hingga tanggal 15 Januari 2020. Survey harga dilakukan dengan cara mencari informasi dari wisatawan yang pernah berkunjung pada suatu objek wisata, setelah mendapatkan data-data tersebut, maka akan dijadikan acuan yang akan digunakan sebagai kriteria didalam sistem yang kemudian akan menjadi kebutuhan sistem baik fungsional maupun non fungsional.

4.1.1. Statement Of Purpose (SOP)

Sistem Rekomendasi pemilihan wisata di kabupaten jember ini akan diimplementasikan di kota Jember dengan memiliki dua hak akses yaitu admin dan user. Admin disini merupakan Dinas Kebudayaan dan Pariwisata Kabupaten Jember serta user disini merupakan masyarakat umum khususnya bagi wisatawan yang ingin datang ke Kabupaten Jember.

Sistem ini dapat membantu merekomendasikan wisata bagi para wisatawan yang ingin mengunjungi tempat wisata di Kabupaten Jember. Sistem ini menyediakan rekomendasi wisata sesuai dengan keinginan pengguna, pengguna dapat dengan bebas memilih kriteria-kriteria seperti halnya kriteria jarak, harga dan rating, sebagai contoh pengguna dapat memilih ingin mengetahui rekomendasi wisata berdasarkan jarak yang terdekat dari lokasi dirinya berada saat ini, dapat mengetahui harga wisatawan termurah dan rating wisata yang paling baik yang ada di Kabupaten Jember. Selain itu, didalam sistem juga terdapat fitur untuk mengelola data wisata untuk menambahkan data-data wisata seperti halnya data wisata pantai, data hotel dan data kuliner.

4.1.1.1. Data Tempat Wisata

Data tempat wisata ini merupakan data utama atau pokok dalam penelitian ini yang didapatkan dengan melakukan wawancara maupun pengambilan data secara kuantitatif. Data ini diambil dari Dinas Pariwisata dan Kebudayaan Kabupaten Jember. Berikut merupakan beberapa contoh data tempat wisata yang didapatkan dari Dinas Pariwisata dan Kebudayaan Kabupaten Jember yang akan ditunjukkan pada tabel 4.1

Tabel 4.1 Data Tempat Wisata

Nama Wisata	Alamat	Harga	Rating
Papuma	Wuluhan, 68172, Area Kebun, Lojejer, Wuluhan, Kabupaten Jember	15000	5
Watu Ulo	Watu Ulo, Sumberrejo, Ambulu	7500	4
Nanggalan	Jalan Cangak Indah, Curahnongko, Tempurejo,	5000	5
Puger	Jalan Pelabuhan, Puger Wetan, Puger, Jember	5000	4

Sumber:(Dinas pariwisata, dan *Google Maps*, Januari 2020)

4.1.1.2. Data Tempat Hotel Dan Kuliner

Data ini merupakan kumpulan data yang berisi data hotel dan kuliner yang didapatkan dari beberapa sumber seperti halnya dari *Google Maps* dan responden yang pernah mengunjungi hotel dan kuliner yang ada di Kabupaten Jember. Data hotel dan kuliner disajikan seperti halnya yang akan ditunjukkan pada tabel 4.2 berikut ini.

Tabel 4.2 Data Tempat Hotel dan Kuliner

Nama	Alamat	Harga	Rating
Hotel 88	Jalan Diponegoro No. 43. Jember. Kaliwates. Jember.	320.226	4
Hotel Sulawesi	Jalan KH A Dahlan 33. Kaliwates. Kaliwates. Jember	244.480	4
Legian Restaurant	Jl. Gajah Mada No.234, Kaliwates, Kec. Kaliwates, Kabupaten Jember	34.900	5
Resto Papuma	Jl. Sentot Prawirodirdjo No.44A, Telengsah, Jember Kidul, Kec. Kaliwates, Kabupaten Jember	50.000	5

Sumber:(Dinas pariwisata, wisatawan dan *Google Maps*, Januari 2020)

4.1.1.3. Data Lokasi Awal Wisatawan

Data ini merupakan kumpulan data lokasi awal yang akan disediakan dalam sistem. Data ini meliputi lokasi tempat-tempat yang mempunyai potensi untuk dilalui pertama kali oleh wisatawan yang akan masuk atau mengunjungi wilayah Kabupaten Jember. Berikut merupakan contoh data lokasi awal yang akan ditunjukkan pada tabel 4.3.

Tabel 4.3 Data Lokasi Awal

Nama Lokasi	Alamat
Stasiun Arjasa	Jl. Supriadi, Jawaan, Patemon, Kec. Pakusari, Kabupaten Jember, Jawa Timur
Stasiun Bangsalsari	Kedung Suko, Bangsalsari, Jember Regency.
Terminal Tawang Alun	Jl. Dharmawangsa, Krajan, Kaliwining, Kec. Rambipuji, Kabupaten Jember, Jawa Timur.
Terminal Pakusari	Jl. Brigjend Katamso No. 29, Kertosari, Pakusari, Kabupaten Jember, Jawa Timur 68124

Sumber:(*Google Maps*, Januari 2020)

4.1.2. Kebutuhan Fungsional

Kebutuhan fungsional sistem merupakan kebutuhan yang berisi tentang fitur-fitur utama yang ada pada sistem yang dapat di fungsikan sesuai dengan tujuan dan kebutuhan yang akan dibahas dalam penelitian. Berikut merupakan kebutuhan fungsional dari sistem yaitu:

1. Sistem dapat melakukan login untuk hak akses Admin sebelum Admin masuk ke dalam sistem.
2. Sistem dapat mengubah data *user profile*.
3. Sistem dapat mengelola data wisata meliputi menambah, merubah, dan menghapus data wisata.
4. Sistem dapat mengelola data Hotel meliputi menambah, merubah, dan menghapus data hotel maupun data kamar hotel.

5. Sistem dapat mengelola data Kuliner meliputi menambah, merubah dan menghapus data kuliner.
6. Sistem dapat mengelola data perhitungan yang dibutuhkan untuk melakukan perhitungan metode SAW dan Borda.
7. Sistem dapat menampilkan hasil dari rekomendasi wisata sesuai dengan nilai kriteria yang telah diinputkan sebelumnya oleh pengguna.
8. Sistem mampu melakukan *logout* untuk Admin yang akan keluar dari sistem.

4.1.3. Kebutuhan Non-Fungsional

Kebutuhan non-fungsional merupakan kebutuhan yang dapat mendukung kebutuhan fungsional dari suatu sistem untuk dapat memenuhi kebutuhan dari pengguna. Berikut merupakan kebutuhan non-fungsional meliputi:

1. Sistem menggunakan *password* yang di enkripsi sebelum admin masuk kedalam sistem.
2. Sistem memiliki tampilan yang mudah dimengerti oleh user (*user friendly*) dan Sistem berbasis *website*.

4.1.4. Fungsi Sistem

Fungsi sistem merupakan tujuan dari dibangunnya sistem tersebut, fungsi sistem dalam penelitian ini yaitu untuk membantu mempermudah wisatawan untuk merekomendasikan destinasi wisata yang ada di Kabupaten Jember. Adapun fungsi sistem secara khusus dibagi sesuai dengan hak aksesnya meliputi:

1. Admin

Admin merupakan pengelola dalam sistem ini yang sebelumnya telah diberikan *username* dan *password* untuk mengaksesnya. Admin berfungsi untuk mengelola data-data tempat wisata, hotel, kuliner, maupun data perhitungan yang digunakan untuk merekomendasi tempat wisata.

2. User

User merupakan pengguna umum dalam sistem ini, user tidak perlu melakukan *login* untuk dapat mengakses sistem. User dapat langsung memilih beberapa kriteria yang disediakan sistem dan dapat langsung mendapatkan rekomendasi destinasi wisata.

4.2. Desain Sistem

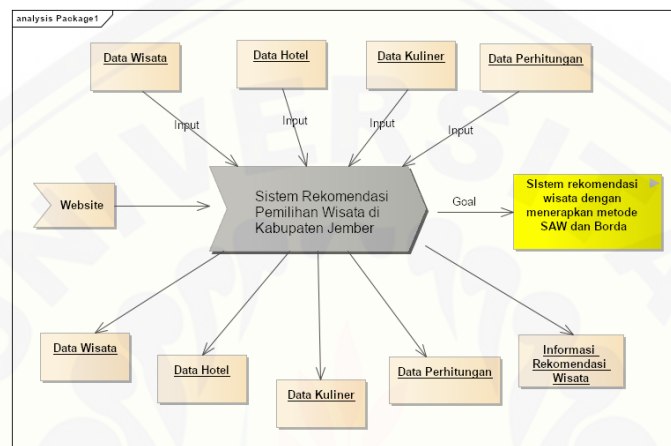
Tahap selanjutnya setelah melakukan analisis sistem yaitu dengan melakukan perancangan sistem atau yang disebut desain sistem. Desain sistem ini meliputi bagan *input*, *proses* dan *output*, *Usecase diagram*, *Skenario*, *Activity diagram*, *Sequence diagram*, *Class diagram*, dan *Entity Relationship Diagram*.

4.2.1. Bagan Input, Proses, Output

Bagan *input*, *proses*, *output* merupakan diagram yang menggambarkan alur proses bisnis yang ada di dalam sistem. Pada Bagan *input*, *proses*, *output* terdapat beberapa point diantaranya yaitu:

1. *Input* : Merupakan semua jenis data yang masuk ke dalam sistem.
2. *Output* : Merupakan data yang dihasilkan atau keluar dari sistem.
3. *Goal* : Merupakan gambaran tujuan dari sistem.
4. *Process* : Proses dari sistem yang bekerja.

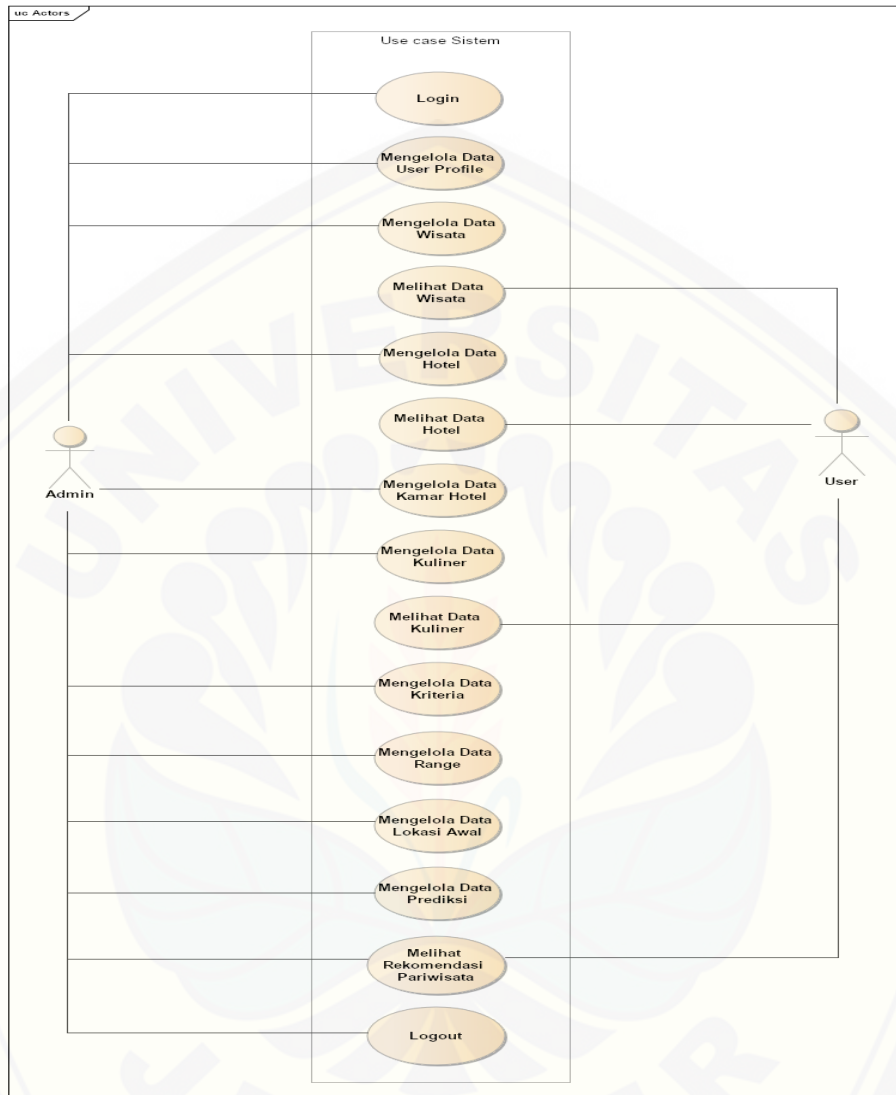
Bagan *input, proses, output* menjelaskan alur yang ada didalam sistem rekomendasi wisata ini, menjelaskan *input, output, process* dan *goals* dalam sistem. Berikut merupakan bagan *input, proses, output* yang ditunjukkan pada gambar 4.1 ini:



Gambar 4.1 *Input, Proses, Output*

4.2.2. Usecase Diagram

Usecase Diagram merupakan bentuk pemodelan yang menggambarkan interaksi antara pengguna dengan sistem rekomendasi wisata ini. Melalui *Usecase* ini, dapat diketahui interaksi apa saja yang dapat dilakukan oleh masing-masing aktor atau pengguna. Berikut merupakan *Usecase diagram* yang akan ditunjukkan pada gambar 4.2.



Gambar 4.2 Usecase Diagram

Usecase diagram pada gambar 4.2 ini mempunyai penjelasan berupa tabel definisi aktor yaitu tabel yang akan menjelaskan tugas dari masing-masing aktor yang dapat dilihat pada tabel 4.3, dan mempunyai penjelasan berupa tabel definisi *usecase* yang akan menjelaskan fungsionalitas dari setiap *usecase* yang dapat dilihat dalam tabel 4.4.

Tabel 4.4 Definisi Aktor

No.	Aktor	Definisi Tugas
1.	Admin	Mengelola data <i>user profile</i> , mengelola data wisata, mengelola data hotel, mengelola data kuliner, mengelola data kriteria, mengelola data range, mengelola data lokasi awal, dan mengelola data prediksi
2.	User	Memilih kriteria wisata yang diinginkan dan untuk melihat hasil rekomendasi wisata yang di rekomendasikan oleh sistem.

Tabel 4.5 Definisi *Usecase*

No.	Usecase	Deskripsi
1.	Mengelola data <i>user profile</i>	<i>Usecase</i> ini merupakan <i>usecase</i> untuk mengelola data <i>user profile</i> dari admin itu sendiri. Mengelola data <i>user profile</i> meliputi melihat dan merubah data.
2.	Mengelola data wisata	<i>Usecase</i> ini merupakan <i>usecase</i> untuk mengelola data wisata yang ada di Kabupaten Jember. Mengelola data wisata

		meliputi melihat, menambah, merubah dan menghapus data wisata.
3.	Melihat data wisata	<i>Usecase</i> ini merupakan <i>usecase</i> untuk melihat informasi data wisata yang ada di Kabupaten Jember.
4.	Mengelola data hotel	<i>Usecase</i> ini merupakan <i>usecase</i> untuk mengelola data hotel yang ada di Kabupaten Jember. Mengelola data hotel meliputi melihat, menambah, merubah dan menghapus data hotel
5.	Melihat data hotel	<i>Usecase</i> ini merupakan <i>usecase</i> untuk melihat informasi data hotel yang ada di Kabupaten Jember.
6.	Mengelola data kamar hotel	<i>Usecase</i> ini merupakan <i>usecase</i> untuk mengelola daftar kamar hotel yang tersedia pada masing-masing hotel. Mengelola data kamar hotel meliputi melihat, menambah, merubah dan menghapus data.
7.	Mengelola data kuliner	<i>Usecase</i> ini merupakan <i>usecase</i> untuk mengelola data kuliner yang ada di Kabupaten Jember. Mengelola data kulner meliputi melihat, menambah, merubah dan menghapus data.
8.	Melihat data kuliner	<i>Usecase</i> ini merupakan <i>usecase</i> untuk

		melihat informasi kuliner yang ada di Kabupaten Jember.
9.	Mengelola data kriteria	<i>Usecase</i> ini merupakan <i>usecase</i> untuk mengelola data kriteria yang dibutuhkan untuk merekomendasikan wisata nantinya. Mengelola data kriteria meliputi melihat, menambah, merubah dan menghapus data.
10.	Mengelola data range	<i>Usecase</i> ini merupakan <i>usecase</i> untuk mengelola data range yang juga merupakan serangkaian data yang diperlukan untuk merekomendasikan wisata. Mengelola data range meliputi melihat, menambah, merubah dan menghapus data.
11.	Mengelola data lokasi awal	<i>Usecase</i> ini merupakan <i>usecase</i> untuk mengelola lokasi awal yang berpotensi menjadi masuknya wisatawan pertama kali di Kabupaten Jember. Mengelola data lokasi awal meliputi melihat, menambah, merubah dan menghapus data.
12.	Mengelola data prediksi	<i>Usecase</i> ini merupakan <i>usecase</i> untuk mengelola data prediksi wisata, data ini berkaitan dengan data kriteria dan data range yang telah dimasukkan sebelumnya. Mengelola data prediksi meliputi melihat,

		menambah, merubah dan menghapus data.
13.	Melihat rekomendasi pariwisata	<i>Usecase</i> ini merupakan <i>usecase</i> untuk melihat informasi hasil rekomendasi wisata, hotel dan kuliner yang ada di Kabupaten Jember.

4.2.3. Skenario Sistem

Skenario merupakan penjelasan alur dari masing-masing *usecase* yang telah dijelaskan pada tabel 4.4 serta penjelasan alur alternatif yang dilakukan oleh masing-masing aktor ketika mengakses sistem. Skenario digambarkan menjadi dua alur yaitu alur aksi aktor dan alur reaksi sistem.

1. Skenario *Login* Sistem

Skenario ini menjelaskan tentang alur kerja dari *login* sistem sebelum aktor admin masuk ke dalam sistem. Penjelasan skenario login ini ditunjukkan pada lampiran A1.

2. Skenario Mengelola Data *User Profile*

Skenario ini menjelaskan tentang alur kerja dari fitur data *user profile* yang dapat mengelola data meliputi menambah, melihat, merubah dan menghapus data *user profile*. Penjelasan skenario mengelola data *user profile* ini ditunjukkan pada lampiran A2 dan tabel 4.5.

3. Skenario Mengelola Data Wisata

Skenario ini menjelaskan tentang alur kerja dari fitur wisata yang dapat mengelola data meliputi menambah, melihat, merubah dan menghapus data wisata. Penjelasan skenario mengelola data wisata ini ditunjukkan pada lampiran A3.

4. Skenario Melihat Data Wisata

Skenario ini menjelaskan tentang alur kerja dari fitur wisata yang hanya dapat melihat informasi wisata yang ada di Kabupaten Jember. Penjelasan skenario melihat data wisata ini ditunjukkan pada lampiran A4.

5. Skenario Mengelola Data Hotel

Skenario ini menjelaskan tentang alur kerja dari fitur hotel yang dapat mengelola data meliputi menambah, melihat, merubah dan menghapus data hotel. Penjelasan skenario mengelola data hotel ini ditunjukkan pada lampiran A5.

6. Skenario Melihat Data Hotel

Skenario ini menjelaskan tentang alur kerja dari fitur hotel yang hanya dapat melihat informasi hotel yang ada di Kabupaten Jember. Penjelasan skenario melihat data hotel ini ditunjukkan pada lampiran A6.

7. Skenario Mengelola Data Kamar Hotel

Skenario ini menjelaskan tentang alur kerja dari fitur kamar hotel yang dapat mengelola data meliputi menambah, melihat, merubah dan menghapus data kamar hotel. Penjelasan skenario mengelola data kamar hotel ini ditunjukkan pada lampiran A7.

8. Skenario Mengelola Data Kuliner

Skenario ini menjelaskan tentang alur kerja dari fitur kuliner yang dapat mengelola data meliputi menambah, melihat, merubah dan menghapus data kuliner. Penjelasan skenario mengelola data kuliner ini ditunjukkan pada lampiran A8.

9. Skenario Melihat Data Kuliner

Skenario ini menjelaskan tentang alur kerja dari fitur kuliner yang hanya dapat melihat informasi kuliner yang ada di Kabupaten Jember. Penjelasan skenario melihat data kuliner ini ditunjukkan pada lampiran A9.

10. Skenario Mengelola Data Kriteria

Skenario ini menjelaskan tentang alur kerja dari fitur kriteria yang dapat mengelola data meliputi menambah, melihat, merubah dan menghapus data kriteria. Penjelasan skenario mengelola data kriteria ini ditunjukkan pada lampiran A10.

11. Skenario Mengelola Data Range

Skenario ini menjelaskan tentang alur kerja dari fitur range yang dapat mengelola data meliputi menambah, melihat, merubah dan menghapus data range. Penjelasan skenario mengelola data range ini ditunjukkan pada lampiran A11.

12. Skenario Mengelola Data Lokasi Awal

Skenario ini menjelaskan tentang alur kerja dari fitur lokasi awal yang dapat mengelola data meliputi menambah, melihat, merubah dan menghapus data lokasi awal. Penjelasan skenario mengelola data lokasi awal ini ditunjukkan pada lampiran A12.

13. Skenario Mengelola Data Prediksi

Skenario ini menjelaskan tentang alur kerja dari fitur data prediksi yang dapat mengelola data meliputi menambah, melihat, merubah dan menghapus data prediksi. Penjelasan skenario mengelola data prediksi ini ditunjukkan pada lampiran A13.

14. Skenario Melihat Rekomendasi Pariwisata

Skenario ini menjelaskan tentang alur kerja dari fitur rekomendasi wisata yang hanya dapat melihat informasi hasil rekomendasi wisata sistem yang ada di Kabupaten Jember. Penjelasan skenario melihat data rekomendasi pariwisata ini ditunjukkan pada lampiran A14.

15. Skenario *Logout* Sistem

Skenario ini menjelaskan tentang alur *logout* atau alur untuk keluar dari sistem. Penjelasan skenario *logout* ini ditunjukkan pada lampiran A15.

Tabel 4.6 Mengelola Data *User Profile*

No. Usecase	UC2
Nama Usecase	Mengelola Data User Profile
Aktor	Admin
Deskripsi Singkat	Aktor dapat melihat dan merubah data user profile
Prakondisi	Dashboard Sistem

Pascakondisi	User Profile
Flow Events	
Normal Flow : Melihat Data User Profile	
Aktor	Sistem
1. Klik User	
	<p>2. Menampilkan Form Halaman User Profile meliputi:</p> <ul style="list-style-type: none"> • Tombol Pilih Gambar (varchar 191) • Nama (varchar 191) • Email (varchar 191) • Password (varchar 191) • Tombol Simpan
Flow Events	
Normal Flow : Merubah Data User Profile	
1. Klik User	<p>2. Menampilkan Form Halaman User Profile meliputi:</p> <ul style="list-style-type: none"> • Tombol Pilih Gambar (varchar 191) • Nama (varchar 191) • Email (varchar 191) • Password (varchar 191) • Tombol Simpan
3. Mengisi Form User Profile	

4. Klik Simpan	
	5. Menampilkan Notifikasi “Data Profile Berhasil Dirubah” dan Menampilkan Form Halaman User Profile
Alternatif Flow : Data Tidak Lengkap	
4. Klik Simpan	
	5. Menampilkan Notifikasi “Harap Lengkapi Data!”
	6. Menampilkan Form Halaman User Profile

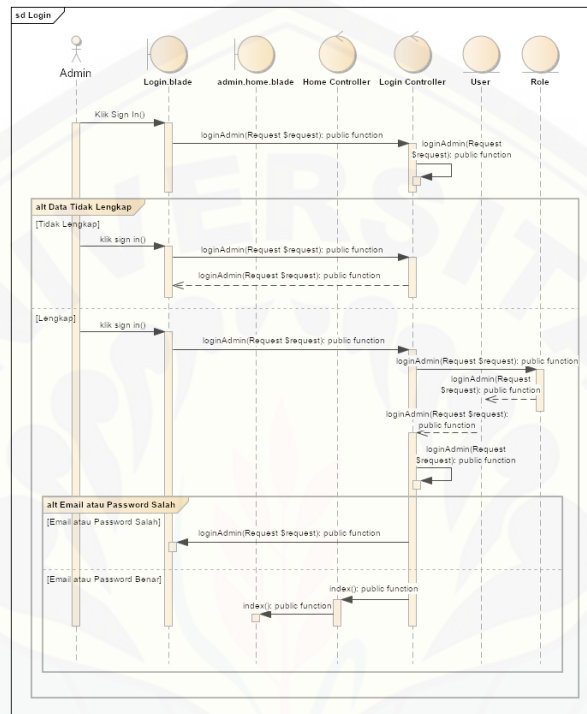
4.2.4. Sequence Diagram

Sequence Diagram merupakan detail gambaran dari skenario yang telah dibuat. *Sequence Diagram* menggambarkan interaksi antar obyek, proses-proses yang ada di dalam sistem baik berupa aliran *input*, trigger sistem, maupun *output* yang dihasilkan. Tujuan dari *sequence diagram* ini agar dapat lebih mengetahui secara detail alur sistem sesuai dengan *requirement* sistem.

1. *Sequence Diagram Login* Sistem

Sequence Diagram Login merupakan penggambaran interaksi antar obyek, menggambarkan fungsi atau *method* dan untuk mengetahui respon dari suatu

kejadian/*event* dari proses *login* didalam sistem. Penjelasan *sequence* ini ditunjukkan pada gambar 4.3 dan pada lampiran B1.



Gambar 4.3 *Sequence Diagram Login* Sistem

2. *Sequence Diagram* Mengelola Data *User Profile*

Sequence Diagram mengelola data *user profile* ini merupakan penggambaran interaksi antar obyek, menggambarkan fungsi atau *method* dan untuk mengetahui respon dari suatu kejadian/*event* dari melihat, menambah, merubah dan menghapus data *user profile*. Penjelasan *sequence* ini ditunjukkan pada lampiran B2.

3. *Sequence Diagram* Mengelola Data Wisata

Sequence Diagram mengelola data wisata ini merupakan penggambaran interaksi antar obyek, menggambarkan fungsi atau *method* dan untuk mengetahui respon dari suatu kejadian/*event* dari melihat, menambah, merubah dan menghapus data wisata. Penjelasan *sequence* ini ditunjukkan pada lampiran B3.

4. *Sequence Diagram* Melihat Data Wisata

Sequence Diagram melihat data wisata ini merupakan penggambaran interaksi antar obyek, menggambarkan fungsi atau *method* dan untuk mengetahui respon dari suatu kejadian/*event* dari melihat data wisata. Penjelasan *sequence* ini ditunjukkan pada lampiran B4.

5. *Sequence Diagram* Mengelola Data Hotel

Sequence Diagram mengelola data hotel ini merupakan penggambaran interaksi antar obyek, menggambarkan fungsi atau *method* dan untuk mengetahui respon dari suatu kejadian/*event* dari melihat, menambah, merubah dan menghapus data hotel. Penjelasan *sequence* ini ditunjukkan pada lampiran B5.

6. *Sequence Diagram* Melihat Data Hotel

Sequence Diagram melihat data hotel ini merupakan penggambaran interaksi antar obyek, menggambarkan fungsi atau *method* dan untuk mengetahui respon dari suatu kejadian/*event* dari melihat data hotel. Penjelasan *sequence* ini ditunjukkan pada lampiran B6.

7. *Sequence Diagram* Mengelola Data Kamar Hotel

Sequence Diagram mengelola data kamar hotel ini merupakan penggambaran interaksi antar obyek, menggambarkan fungsi atau *method* dan untuk mengetahui respon dari suatu kejadian/*event* dari melihat, menambah,

merubah dan menghapus data kamar hotel. Penjelasan *sequence* ini ditunjukkan pada lampiran B7.

8. *Sequence Diagram* Mengelola Data Kuliner

Sequence Diagram mengelola data kuliner ini merupakan penggambaran interaksi antar obyek, menggambarkan fungsi atau *method* dan untuk mengetahui respon dari suatu kejadian/*event* dari melihat, menambah, merubah dan menghapus data kuliner. Penjelasan *sequence* ini ditunjukkan pada lampiran B8.

9. *Sequence Diagram* Melihat Data Kuliner

Sequence Diagram melihat data kuliner ini merupakan interaksi antar obyek, menggambarkan fungsi atau *method* dan untuk mengetahui respon dari suatu kejadian/*event* dari melihat data kuliner. Penjelasan *sequence* ini ditunjukkan pada lampiran B9.

10. *Sequence Diagram* Mengelola Data Kriteria

Sequence Diagram mengelola data kriteria ini merupakan interaksi antar obyek, menggambarkan fungsi atau *method* dan untuk mengetahui respon dari suatu kejadian/*event* dari melihat, menambah, merubah dan menghapus data kriteria. Penjelasan *sequence* ini ditunjukkan pada lampiran B10.

11. *Sequence Diagram* Mengelola Data Range

Sequence Diagram mengelola data range ini merupakan interaksi antar obyek, menggambarkan fungsi atau *method* dan untuk mengerahui respon dari suatu kejadian/*event* dari melihat, menambah, merubah dan menghapus data range. Penjelasan *sequence* ini ditunjukkan pada lampiran B11.

12. *Sequence Diagram* Mengelola Data Lokasi Awal

Sequence Diagram mengelola data lokasi awal ini merupakan interaksi antar obyek, menggambarkan fungsi atau *method* dan untuk mengetahui respon dari suatu kejadian/*event* dari melihat, menambah, merubah dan menghapus data lokasi awal. Penjelasan *sequence* ini ditunjukkan pada lampiran B12.

13. *Sequence Diagram* Mengelola Data Prediksi

Sequence Diagram mengelola data prediksi ini merupakan interaksi antar obyek, menggambarkan fungsi atau *method* dan untuk mengetahui respon dari suatu kejadian/*event* dari melihat, menambah, merubah dan menghapus data prediksi. Penjelasan *sequence* ini ditunjukkan pada lampiran B13.

14. *Sequence Diagram* Melihat Rekomendasi Pariwisata

Sequence Diagram melihat rekomendasi pariwisata ini merupakan interaksi antar obyek, menggambarkan fungsi atau *method* dan untuk mengetahui respon dari suatu kejadian/*event* dari melihat data rekomendasi pariwisata. Penjelasan *sequence* ini ditunjukkan pada lampiran B14.

15. *Sequence Diagram* Logout Sistem

Sequence Diagram *logout* merupakan penggambaran interaksi antar obyek, menggambarkan fungsi atau *method* dan untuk mengetahui respon dari suatu kejadian/*event* dari proses *logout* sistem. Penjelasan *sequence* ini ditunjukkan pada lampiran B15.

4.2.5. Activity Diagram

Activity Diagram merupakan penggambaran dari alur aktivitas dari suatu sistem yang sedang di rancang, menggambarkan bagaimana alur yang pertama kali sistem lalui, *Decision* atau kondisi apa saja yang sistem dapat lakukan dan bagaimana sistem itu berakhir.

1. *Activity Diagram Login Sistem*

Activity Diagram ini menggambarkan alur aktivitas dari proses *login* yang dilakukan oleh actor dan sistem, menggambarkan bagaimana alur pertama kali akan dilalui oleh actor dan sistem, serta *decision-decision* yang ada pada proses *login*. *Activity Diagram* ini ditunjukkan pada gambar 4.4 dan pada lampiran C1.

2. *Activity Diagram Mengelola Data User Profile*

Activity Diagram ini menggambarkan alur aktivitas dari proses mengelola data *user profile* meliputi melihat, menambah, merubah dan menghapus yang dilakukan oleh actor dan sistem, serta *decision-decision* yang ada pada proses *user profile*. *Activity Diagram* ini ditunjukkan pada lampiran C2.

3. *Activity Diagram Mengelola Data Wisata*

Activity Diagram ini menggambarkan alur aktivitas dari proses mengelola data wisata meliputi melihat, menambah, merubah dan menghapus yang dilakukan oleh actor dan sistem, serta *decision-decision* yang ada pada proses fitur wisata. *Activity Diagram* ini ditunjukkan pada lampiran C3.

4. *Activity Diagram Melihat Data Wisata*

Activity Diagram ini menggambarkan alur aktivitas dari proses melihat data wisata yang dilakukan oleh actor dan sistem, serta *decision-decision* yang ada pada proses fitur wisata. *Activity Diagram* ini ditunjukkan pada lampiran C4.

5. *Activity Diagram* Mengelola Data Hotel

Activity Diagram ini menggambarkan alur aktivitas dari proses mengelola data hotel meliputi melihat, menambah, merubah dan menghapus yang dilakukan oleh actor dan sistem, serta *decision-decision* yang ada pada proses fitur hotel. *Activity Diagram* ini ditunjukkan pada lampiran C5.

6. *Activity Diagram* Melihat Data Hotel

Activity Diagram ini menggambarkan alur aktivitas dari proses melihat data hotel yang dilakukan oleh actor dan sistem, serta *decision-decision* yang ada pada proses fitur hotel. *Activity Diagram* ini ditunjukkan pada lampiran C6.

7. *Activity Diagram* Mengelola Data Kamar Hotel

Activity Diagram ini menggambarkan alur aktivitas dari proses mengelola data kamar hotel meliputi melihat, menambah, merubah dan menghapus yang dilakukan oleh actor dan sistem, serta *decision-decision* yang ada pada proses fitur kamar hotel. *Activity Diagram* ini ditunjukkan pada lampiran C7.

8. *Activity Diagram* Mengelola Data Kuliner

Activity Diagram ini menggambarkan alur aktivitas dari proses mengelola data kuliner meliputi melihat, menambah, merubah dan menghapus yang dilakukan oleh actor dan sistem, serta *decision-decision* yang ada pada proses fitur kuliner. *Activity Diagram* ini ditunjukkan pada lampiran C8.

9. *Activity Diagram* Melihat Data Kuliner

Activity Diagram ini menggambarkan alur aktivitas dari proses melihat data kuliner yang dilakukan oleh actor dan sistem, serta *decision-decision* yang ada pada proses fitur kuliner. *Activity Diagram* ini ditunjukkan pada lampiran C9.

10. *Activity Diagram* Mengelola Data Kriteria

Activity Diagram ini menggambarkan alur aktivitas dari proses mengelola data kriteria meliputi melihat, menambah, merubah dan menghapus yang dilakukan oleh actor dan sistem, serta *decision-decision* yang ada pada proses fitur kriteria. *Activity Diagram* ini ditunjukkan pada lampiran C10.

11. *Activity Diagram* Mengelola Data Range

Activity Diagram ini menggambarkan alur aktivitas dari proses mengelola data range meliputi melihat, menambah, merubah dan menghapus yang dilakukan oleh actor dan sistem, serta *decision-decision* yang ada pada proses fitur range. *Activity Diagram* ini ditunjukkan pada lampiran C11.

12. *Activity Diagram* Mengelola Data Lokasi Awal

Activity Diagram ini menggambarkan alur aktivitas dari proses mengelola data lokasi awal meliputi melihat, menambah, merubah dan menghapus yang dilakukan oleh actor dan sistem, serta *decision-decision* yang ada pada proses fitur lokasi awal. *Activity Diagram* ini ditunjukkan pada lampiran C12.

13. *Activity Diagram* Mengelola Data Prediksi

Activity Diagram ini menggambarkan alur aktivitas dari proses mengelola data prediksi meliputi melihat, menambah, merubah dan menghapus yang

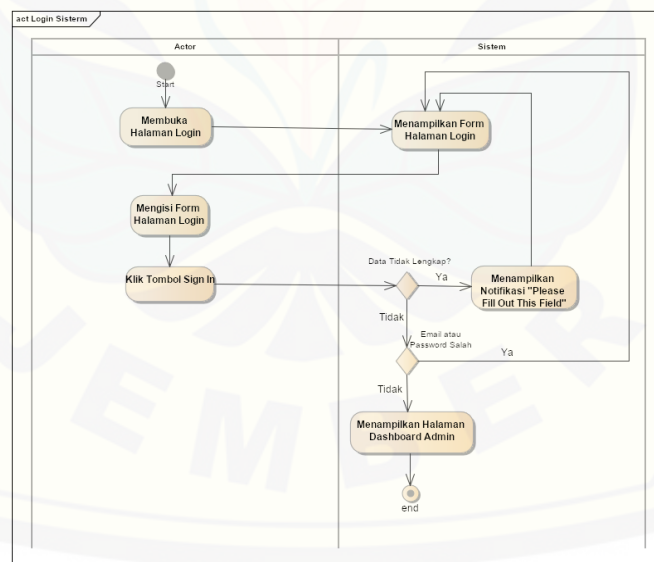
dilakukan oleh actor dan sistem, serta *decision-decision* yang ada pada proses fitur data prediksi. *Activity Diagram* ini ditunjukkan pada lampiran C13.

14. *Activity Diagram* Melihat Rekomendasi Pariwisata

Activity Diagram ini menggambarkan alur aktivitas dari proses melihat rekomendasi pariwisata yang dilakukan oleh actor dan sistem, serta *decision-decision* yang ada pada proses fitur rekomendasi pariwisata. *Activity Diagram* ini ditunjukkan pada lampiran C14.

15. *Activity Diagram* Logout Sistem

Activity Diagram ini menggambarkan alur aktivitas dari proses *logout* sistem yang dilakukan oleh actor dan sistem, serta *decision-decision* yang ada pada proses fitur *logout*. *Activity Diagram* ini ditunjukkan pada lampiran C15.



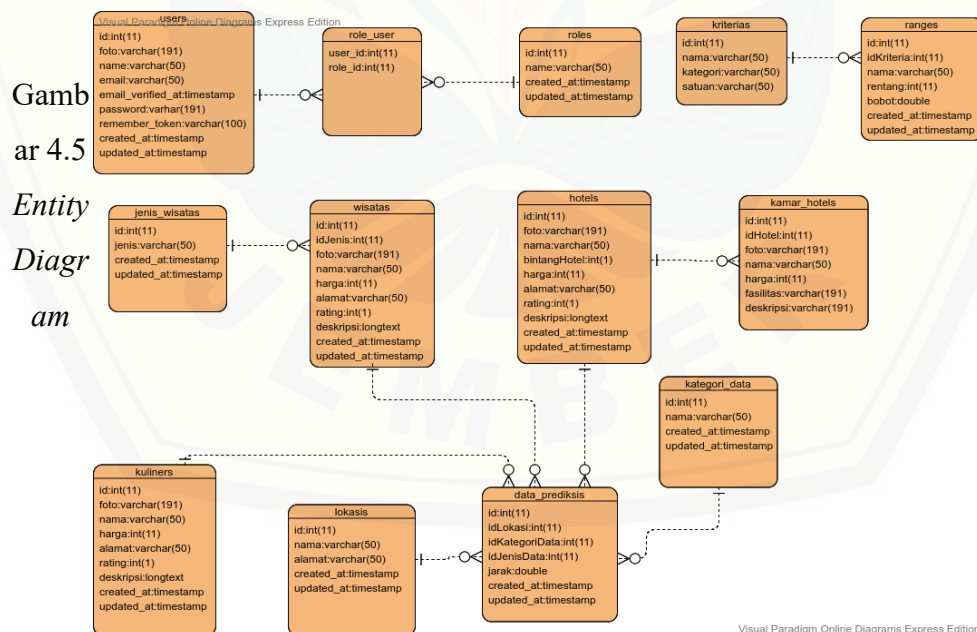
Gambar 4.4 *Activity Diagram* Login Sistem

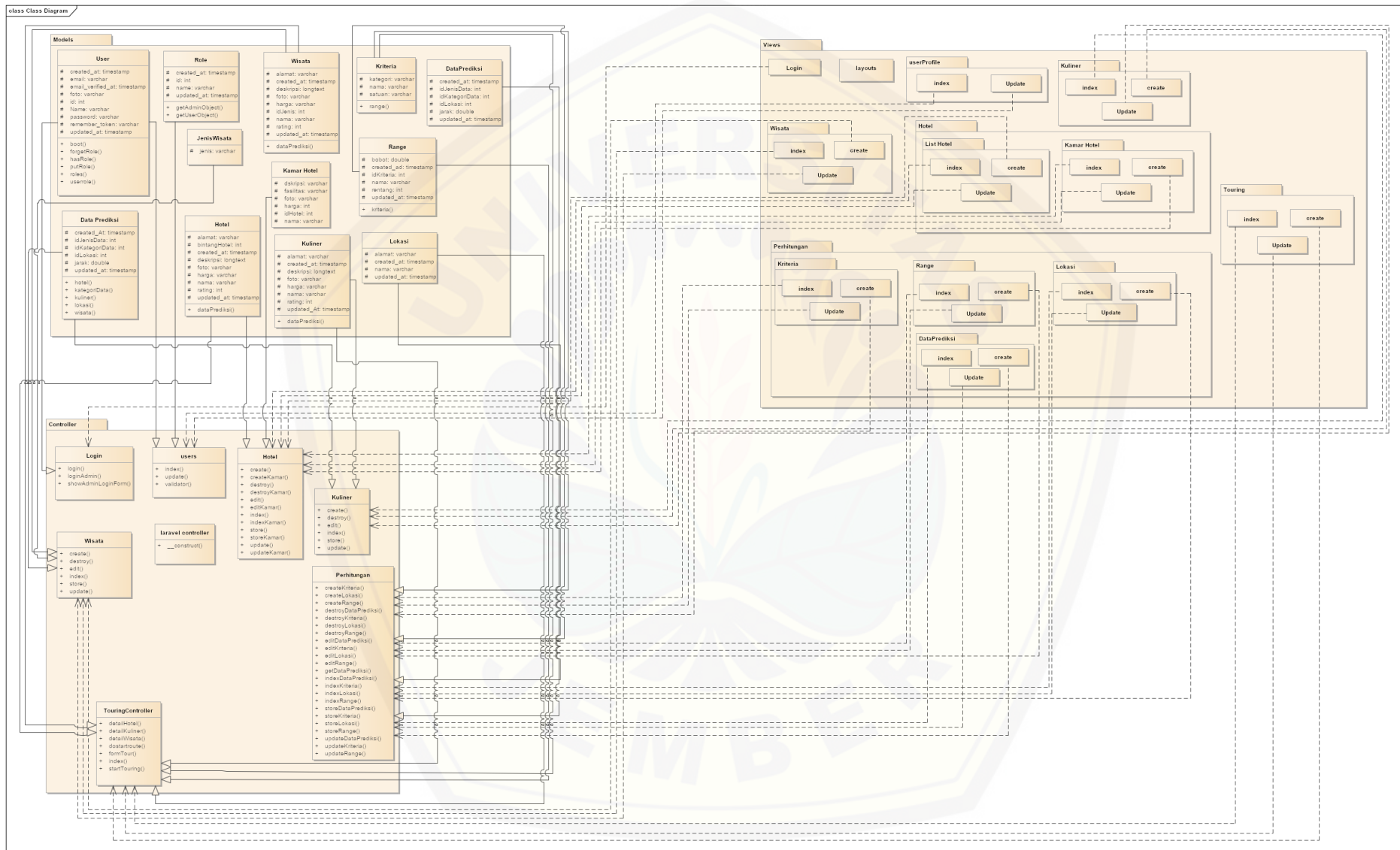
4.2.6. Class Diagram

Class Diagram menggambarkan hubungan antar kelas pada sistem yang akan dibangun. Struktur kelas dibagi menjadi tiga bagian yaitu nama kelas, atribut kelas dan operasi kelas. Nama berfungsi untuk memberikan identitas suatu kelas agar dapat memudahkan untuk mengenali fungsi kelas tersebut, atribut berfungsi untuk mendefinisikan tiap variable yang dimiliki oleh kelas tersebut, sedangkan operasi kelas berisi tentang *method-method* yang ada didalam suatu kelas. *Class Diagram* pada sistem ini ditunjukkan pada gambar 4.5.

4.2.7. Entity Relationship Diagram

Entity Relationship Diagram (ERD) merupakan diagram yang menggambarkan struktur dan hubungan antar *database* dari sistem yang akan dibuat. ERD pada sistem ini ditunjukkan pada gambar 4.6.





Gambar 4.6 Class Diagram

4.3. Pengujian Sistem

Pengujian sistem dilakukan untuk mengevaluasi apakah program dapat dijalankan dengan baik atau tidak, apakah telah sesuai dengan kebutuhan pengguna atau belum. Proses pengujian dilakukan dengan dua pengujian yaitu pengujian *white box* dan pengujian dengan menggunakan *black box*.

4.3.1. Pengujian *White Box*

Pengujian ini menggunakan pengujian *Cyclomatic Complexity* (CC). CC merupakan metode atau matriks yang berfungsi untuk mengetahui kompleksitas logika suatu program. Metode pengujian CC ini memiliki beberapa langkah sebelum melakukannya diantaranya yaitu melakukan *Listing* program, pembuatan diagram grafik alir, menghitung kompleksitas siklomatik dan jalur independen. Berikut ini meruakan rumus untuk menghitung kompleksitas siklomatik dan jalur independen:

$$V(G) = E - N + 2$$

Dimana:

E = Jumlah edge grafik alir

N = Jumlah node grafik alir

Pengujian sistem rekomendasi wisata dengan menggunakan metode SAW dan Borda pada *touringController* dinilai dapat mewakili seluruh alur sistem pada penelitian ini yang akan dijelaskan sebagai berikut:

1. *Listing Program*

```
18
19 class touringController extends Controller{
20     private $wisataRepository, $hotelRepository, $kulinerRepository, $perhitunganRepository;
21
22
23     public function __construct(wisataRepository $wisataRepository, hotelRepository $
        hotelRepository, kulinerRepository $kulinerRepository, perhitunganRepository $
        perhitunganRepository){
24         $this->wisataRepository = $wisataRepository;
25         $this->hotelRepository = $hotelRepository;
26         $this->kulinerRepository = $kulinerRepository;
27         $this->perhitunganRepository = $perhitunganRepository;
28     }
29
```

Gambar 4.7 *Listing Program function __construct*

Pada gambar 4.7, *function __construct* merupakan method khusus yang akan dijalankan secara otomatis ketika object dijalankan. Method construct dapat dikatakan sebagai bentuk pemanggilan atau inisiasi awal untuk dijalankan secara otomatis

```
29
30     public function index(Request $request){
31         $page = $request->get('page', 1);
32         $limit = 9;
33         $wisata = $this->wisataRepository->get($limit);
34         $hotel = $this->hotelRepository->get($limit);
35         $kuliner = $this->kulinerRepository->get($limit);
36         return view('admin.touring.index', compact('wisata', 'hotel', 'kuliner'));
37     }
38
```

Gambar 4.8 *Listing Program function index*

Pada gambar 4.8 *function index* berfungsi untuk memanggil halaman prediksi objek wisata, memanggil beberapa model seperti model wisata, hotel dan kuliner karena didalam halaman prediksi objek wisata memuat data wisata, hotel dan kuliner.

```
38
39     public function detailWisata($id){
40         $detail = Wisata::find($id);
41         return view('admin.touring.detailWisata', compact('detail'));
42     }
43
```

Gambar 4.9 *Listing Program function detailWisata*

Pada gambar 4.9 *function* detailWisata berfungsi untuk memanggil detail objek wisata, ketika tombol lihat detail pada halaman objek wisata di pilih, maka sistem akan menjalankan *function* ini.

```

44     public function detailHotel($id){
45         $detail = Hotel::find($id);
46         return view('admin.touring.detailHotel', compact('detail'));
47     }
48

```

Gambar 4.10 Listing Program *function* detailHotel

Pada gambar 4.10 *function* detailHotel berfungsi untuk memanggil detail objek hotel, ketika tombol lihat detail pada tab list hotel di pilih, maka sistem akan menjalankan *function* ini.

```

48     public function detailKuliner($id){
49         $detail = Kuliner::find($id);
50         return view('admin.touring.detailKuliner', compact('detail'));
51     }
52
53

```

Gambar 4.11 Listing Program *function* detailKuliner

Pada gambar 4.11 *function* detailKuliner berfungsi untuk memanggil detail objek kuliner, ketika tombol lihat detail pada tab list kuliner di pilih, maka sistem akan menjalankan *function* ini.

```

60     public function dostartroute(Request $request){
61         $page = $request->get('page', 1);
62         $limit = 9;
63         $wisata = $this->wisataRepository->get($limit);
64         $hotel = $this->hotelRepository->get($limit);
65         $kuliner = $this->kulinerRepository->get($limit);
66         $lokasi = $this->perhitunganRepository->getLokasi();
67         $range = Range::all();
68
69         $nilaistartroute = $this->startTouring(
70             $request->lokasiAwal,
71             $request->kategori1,
72             $request->kategori2,
73             $request->kategori3,
74             $request->jarak,
75             $request->rentangHarga,
76             $request->rating
77         );
78
79         $currentPage = Paginator::resolveCurrentPage();
80         $col = collect($nilaistartroute);
81         $perPage = 9;
82         $currentPageItems = $col->slice(($currentPage - 1) * $perPage, $perPage->all());
83         $items = new Paginator($currentPageItems, count($col), $perPage);
84         $items->setPath($request->url());
85         $items->append($request->all());
86
87
88         return view('admin.touring.startTour', compact('items', 'lokasi', 'range'));
89     }
90

```

Gambar 4.12 Listing Program *function* dostartroute

Pada gambar 4.12 *function dostartroute* berfungsi untuk memanggil hasil dari perhitungan prediksi objek wisata. *Function* ini akan memanggil sesuai dengan *inputan* yang dimasukkan oleh *user*. Ketika *user* hanya ingin menampilkan data wisata saja, maka akan menampilkan hanya objek wisatanya saja.

```

95
96 public function startTouring($lokasiAwal, $kategori1, $kategori2, $kategori3, $bobotPreferensiJarak, $
97 bobotPreferensiHarga, $bobotPreferensiRating){
98     $rangeJarak = Range::where('idKriteria',1)->get();
99     $rangeHarga = Range::where('idKriteria',2)->get();
100     $rangeRating = Range::where('idKriteria',3)->get();
101
102     if($kategori1 != null && $kategori2 == null && $kategori3 == null){
103         $wisata = DataPrediksi::where('idLokasi', $lokasiAwal)->where('idKategoriData', $kategori1)->get();
104         $matriks = [];
105         $counter = 0;
106
107         foreach($wisata as $wisatas){
108             // normalisasi bobot kriteria
109             foreach($rangeJarak as $jarak){
110                 if($wisatas->jarak <= $jarak->rentang ){
111                     $bobotJarak = $jarak->bobot;
112                 }
113             }
114
115             foreach($rangeHarga as $range){
116                 if($wisatas->wisata()->first()->harga <= $range->rentang){
117                     $bobotHarga = $range->bobot;
118                 }
119             }
120
121             foreach($rangeRating as $rating){
122                 if($wisatas->wisata()->first()->rating <= $rating->rentang){
123                     $bobotRating = $rating->bobot;
124                 }
125             }
126         }
127     }
128

```

Gambar 4.13 Pengecekan Kriteria

```

128
129         // step 1 membuat matriks
130         if($counter <= $wisatas->id){
131             $matriks[$counter][0] = $bobotJarak;
132             $matriks[$counter][1] = $bobotHarga;
133             $matriks[$counter][2] = $bobotRating;
134             $counter++;
135         }
136     }
137
138     // menentukan nilai max jarak
139     $recCountArrayJarak = array();
140     foreach ($matriks as $matrik) {
141         $recCountArrayJarak[] = $matrik['0'];
142     }
143
144     $maxJarak = max($recCountArrayJarak);
145     $minJarak = min($recCountArrayJarak);
146
147     // menentukan nilai min harga
148     $recCountArrayHarga = array();
149     foreach ($matriks as $matrik) {
150         $recCountArrayHarga[] = $matrik['1'];
151     }
152
153     $minHarga = min($recCountArrayHarga);
154
155     // menentukan nilai max rating
156     $recCountArrayRating = array();
157     foreach($matriks as $matrik){
158         $recCountArrayRating[] = $matrik['2'];
159     }
160
161     $maxRating = max($recCountArrayRating);

```



```

201 // Menentukan Nilai V
202 if(count($bobotPreferensiJarak) > 1 ){
203     $nilaiV = [];
204     for($j=0;$j<count($bobotPreferensiJarak);$j++){
205         foreach($matriksR as $matrikss){
206             for($i=0;$i<count($matriks);$i++){
207                 $nilaiV[$j][$i] = ($bobotPreferensiJarak[$j] * $matrikss[0][$i]) + ($bobotPreferensiHarga[
                $j] * $matrikss[1][$i]) + ($bobotPreferensiRating[$j] * $matrikss[2][$i] );
208             }
209         }
210     }
211 }
212
213
214 $idWisata = [];
215 $jarakWisata = [];
216 $namaWisata = [];
217
218 foreach($wisata as $wisataa){
219     $idWisata[] = $wisataa->wisata()->first()->id;
220     $namaWisata[] = $wisataa->wisata()->first()->nama;
221     $jarakWisata[] = $wisataa->jarak;
222 }
223
224
225
226 $tempIdWisata = [];
227 for($j=0;$j<count($nilaiV);$j++){
228     foreach( $idWisata as $origKey => $value ){
229         for($i=0;$i<count($matriks);$i++){
230             $tempIdWisata[$j][$i+1] = $nilaiV[$j][$i];
231         }
232     }
233 }
234
235
236
237
238
239 // Menghitung Borda
240 $duplikat = $indexIdWisata;
241 $sem = 0;
242 $hasilcount = [];
243
244 for($i=0;$i<count($indexIdWisata);$i++){
245     for ($a=1; $a<count($duplikat[$i])+1; $a++) {
246         $sem = $duplikat[$i][$a];
247         $hasilcount[$i][$a][0] = $duplikat[$i][$a];
248         for ($b=1; $b < count($duplikat[$i])+1; $b++) {
249             if($sem == $duplikat[$i][$b]){
250                 $hasilcount[$i][$a][1][$b] = 1;
251             }
252             else{
253                 $hasilcount[$i][$a][1][$b] = 0;
254             }
255         }
256     }
257     sort($hasilcount[$i]);
258 }
259
260 $final_array = [];
261 // $nilaiPeringkat = [];
262 // $bobotBorda = count($final_array);
263
264 for($i=0; $i < count($hasilcount); $i++){
265     for ($j=0; $j < count($hasilcount[0]); $j++) {
266         if($i<count($hasilcount)-1){
267             if($hasilcount[$i][$j][0] == $hasilcount[$i+1][$j][0]){
268                 $final_array[$j][0] = $hasilcount[$i+1][$j][0];
269                 for ($f=1; $f < count($hasilcount[0])+1; $f++) {
270                     $final_array[$j][1][$f] = $hasilcount[$i][$j][1][$f] + $hasilcount[$i+1][$j][1][$f]
271                 };
272             }
273         }
274     }
275 }

```

Gambar 4.14 Melakukan Penentuan Penggunaan Metode

```

299
300 // menentukan bobot peringkat
301 $bobotBorda = count($final_array);
302 $nilaiPeringkat = [];
303
304 for($i=0; $i < count($final_array); $i++){
305     for ($f=1; $f < count($final_array)+1; $f++) {
306         $nilaiPeringkat[$i][1][$f] = ($final_array[$i][1][$f]*$bobotBorda);
307         if($bobotBorda != 1){
308             $bobotBorda--;
309         }else{
310             $bobotBorda = count($final_array);
311         }
312     }
313 }
314
315 for($i=0; $i < count($nilaiPeringkat); $i++){
316     $sumNilaiPeringkat[$i+1] = array_sum($nilaiPeringkat[$i][1]);
317 }
318
319 $totalNilaiPeringkat = array_sum($sumNilaiPeringkat);
320
321 for($i=0; $i < count($nilaiPeringkat); $i++){
322     $bobotNilaiPeringkat[$i+1] = ($sumNilaiPeringkat[$i+1] / $totalNilaiPeringkat);
323 }
324
325 arsort($bobotNilaiPeringkat);
326
327 $arrayIdWisata = 0;
328 foreach($idWisata as $valuee ){
329     $bobotNilaiPeringkat[$valuee] = $idWisata[$arrayIdWisata];
330     $arrayIdWisata ++;
331 }
332
333 foreach($bobotNilaiPeringkat as $temp){
334     $prediksiWisata[] = DataPrediksi::where('idKategoriData', $kategori1)->where('idJenisData',$temp)
->get();

```

Gambar 4.15 Menghitung menggunakan metode Borda

Pada gambar diatas merupakan *function* yang berfungsi untuk melakukan perhitungan baik menggunakan metode SAW maupun Borda, ketika pengguna memasukkan inputan kriteria maka sistem akan melakukan pengecekan apakah hanya menggunakan metode SAW atau dilanjutkan dengan menggunakan metode Borda yang dapat dilihat pada gambar 4.13 dan gambar 4.14. Sistem akan melakukan perhitungan menggunakan metode SAW ketika hanya ada satu *inputan user* kemudian dilanjut dengan perhitungan metode borda yang dapat dilihat pada gambar 4.15. Terakhir, sistem akan menentukan peringkat dari suatu objek wisata berdasarkan kriteria yang diinginkan oleh pengguna yang kemudian hasil peringkat atau hasil perhitungan akan dikembalikan nilainya kedalam method *function dostartroute* (Gambar 4.12).

Listing program dalam pengujian *white box* berfungsi untuk mengetahui fungsi suatu program, percabangan suatu program berada baris ke- berapa yang

nantinya tiap-tiap fungsi pada tiap baris *code* akan di visualisasikan dengan diagram alir yang akan dilakukan pada tahap selanjutnya,

2. Diagram Alir

Diagram alir merupakan bentuk notasi-notasi yang dibuat berdasarkan *listing program* yang berfungsi untuk menggambarkan alur dari sebuah sistem. Dengan adanya diagram alir memudahkan untuk mengetahui jumlah percabangan sebuah *code* dalam suatu *function*. Berikut merupakan penggambaran dari diagram alir :



Gambar 4.16 Diagram Alir *function_construct*

Pada gambar 4.16, merupakan bentuk penggambaran notasi dari *listing program function_construct*. Pada *listing program function_construct*, proses dimulai dari baris 23 dan diakhiri pada baris 28, sehingga bentuk diagram alirnya terlihat seperti gambar diatas. Untuk diagram alir nomor 24-27 merupakan isi dari

proses suatu program yang dilakukan secara berurutan.



Gambar 4.17 Diagram Alir *function_index*

Pada gambar 4.17, merupakan bentuk penggambaran notasi dari *listing program function index*. Pada *listing program function index*, proses dimulai dari baris 30 dan diakhiri pada baris 37, sehingga bentuk diagram alirnya terlihat seperti gambar diatas. Untuk diagram alir nomor 31-36 merupakan isi dari proses suatu program yang dilakukan secara berurutan.



Gambar 4.18 Diagram Alir *function detailWisata*

Pada gambar 4.18, merupakan bentuk penggambaran notasi dari *listing program function detailWisata*. Pada *listing program function detailWisata*, proses dimulai dari baris 39 dan diakhiri pada baris 42, sehingga bentuk diagram alirnya terlihat seperti gambar diatas. Untuk diagram alir nomor 40-41 merupakan isi dari proses suatu program yang dilakukan secara berurutan.



Gambar 4.19 Diagram Alir *function detailHotel*

Pada gambar 4.19, merupakan bentuk penggambaran notasi dari *listing program function detailHotel*. Pada *listing program function detailHotel*, proses dimulai dari baris 44 dan diakhiri pada baris 47, sehingga bentuk diagram alirnya terlihat seperti gambar diatas. Untuk diagram alir nomor 45-46 merupakan isi dari proses suatu program yang dilakukan secara berurutan.



Gambar 4.20 Diagram Alir *function detailKuliner*

Pada gambar 4.20, merupakan bentuk penggambaran notasi dari *listing program function detailKuliner*. Pada *listing program function detailKuliner*, proses dimulai dari baris 49 dan diakhiri pada baris 52, sehingga bentuk diagram alirnya terlihat seperti gambar diatas. Untuk diagram alir nomor 50-51 merupakan isi dari proses suatu program yang dilakukan secara berurutan.

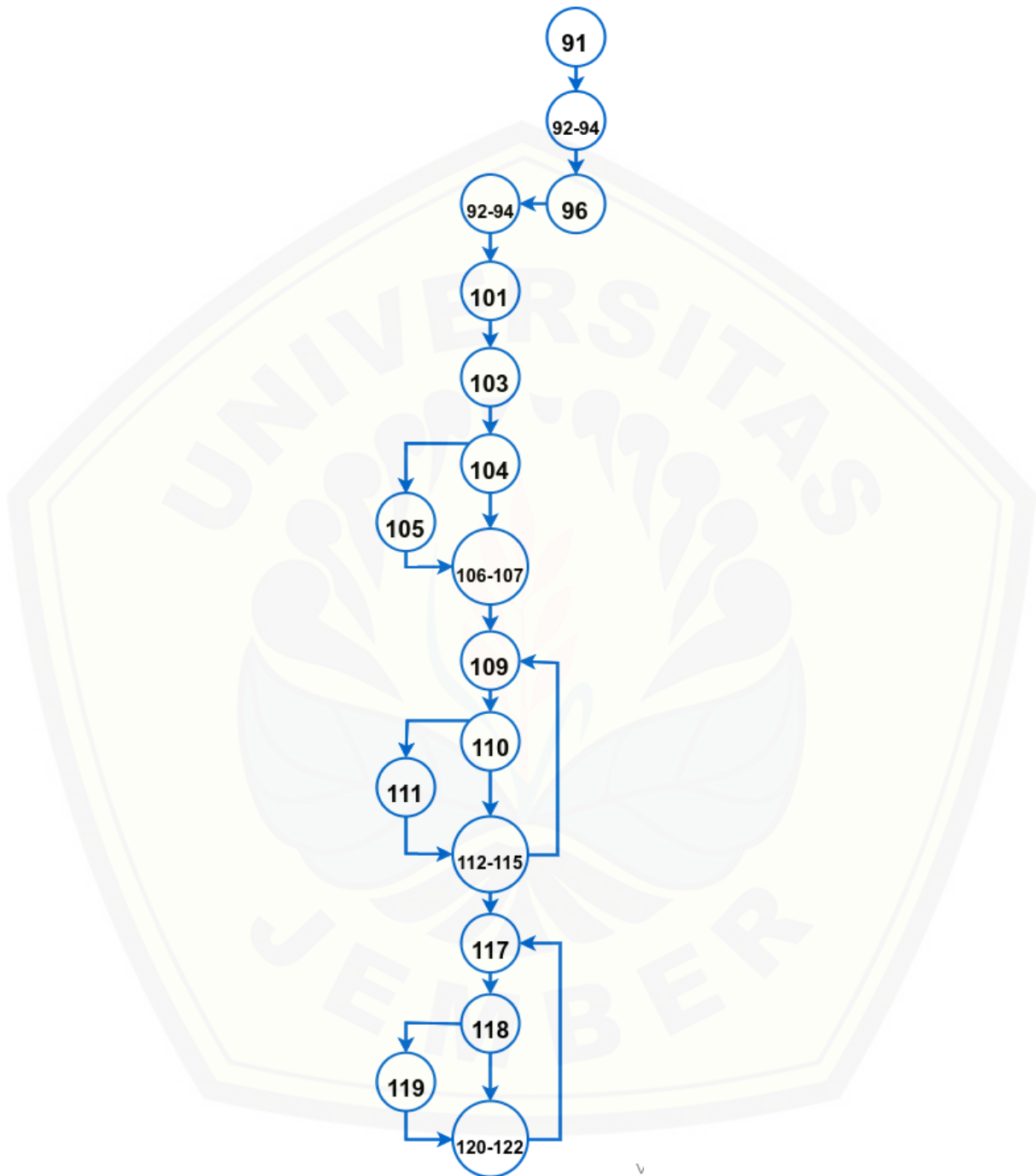


Gambar 4.21 Diagram Alir *function formTour*

Pada gambar 4.21, merupakan bentuk penggambaran notasi dari *listing program function formTour*. Pada *listing program function formTour*, proses dimulai dari baris 54 dan diakhiri pada baris 58, sehingga bentuk diagram alirnya terlihat seperti gambar diatas. Untuk diagram alir nomor 55-57 merupakan isi dari proses suatu program yang dilakukan secara berurutan.

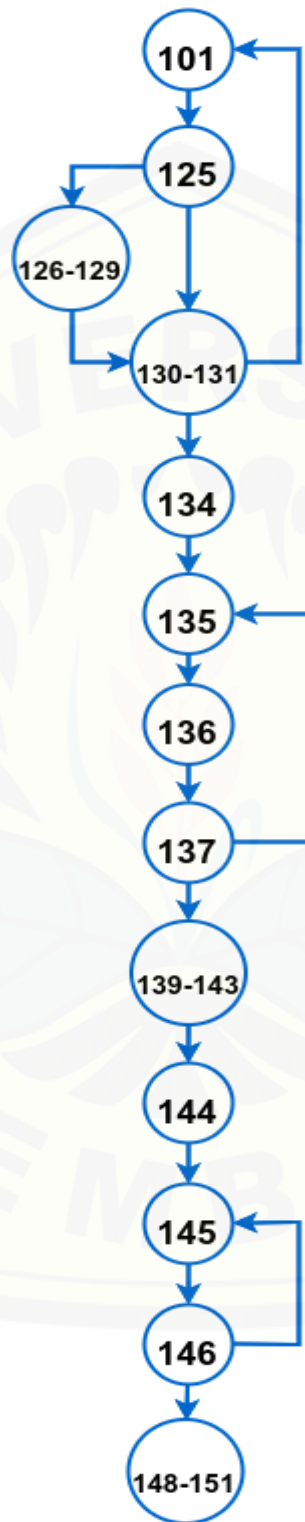


Gambar 4.22 Diagram Alir *function dostartroute*

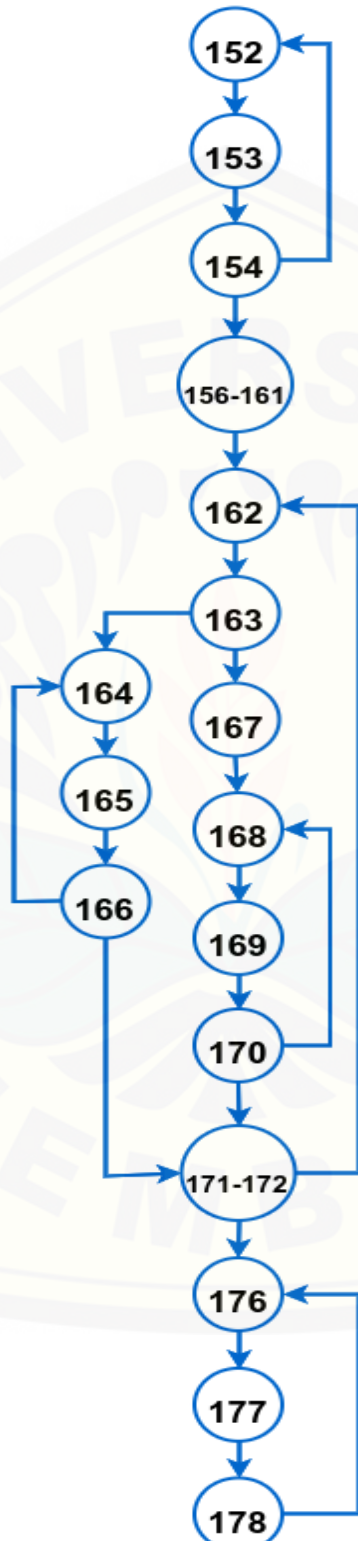


Gambar 4.23 Diagram Alir Pengecekan Kriteria

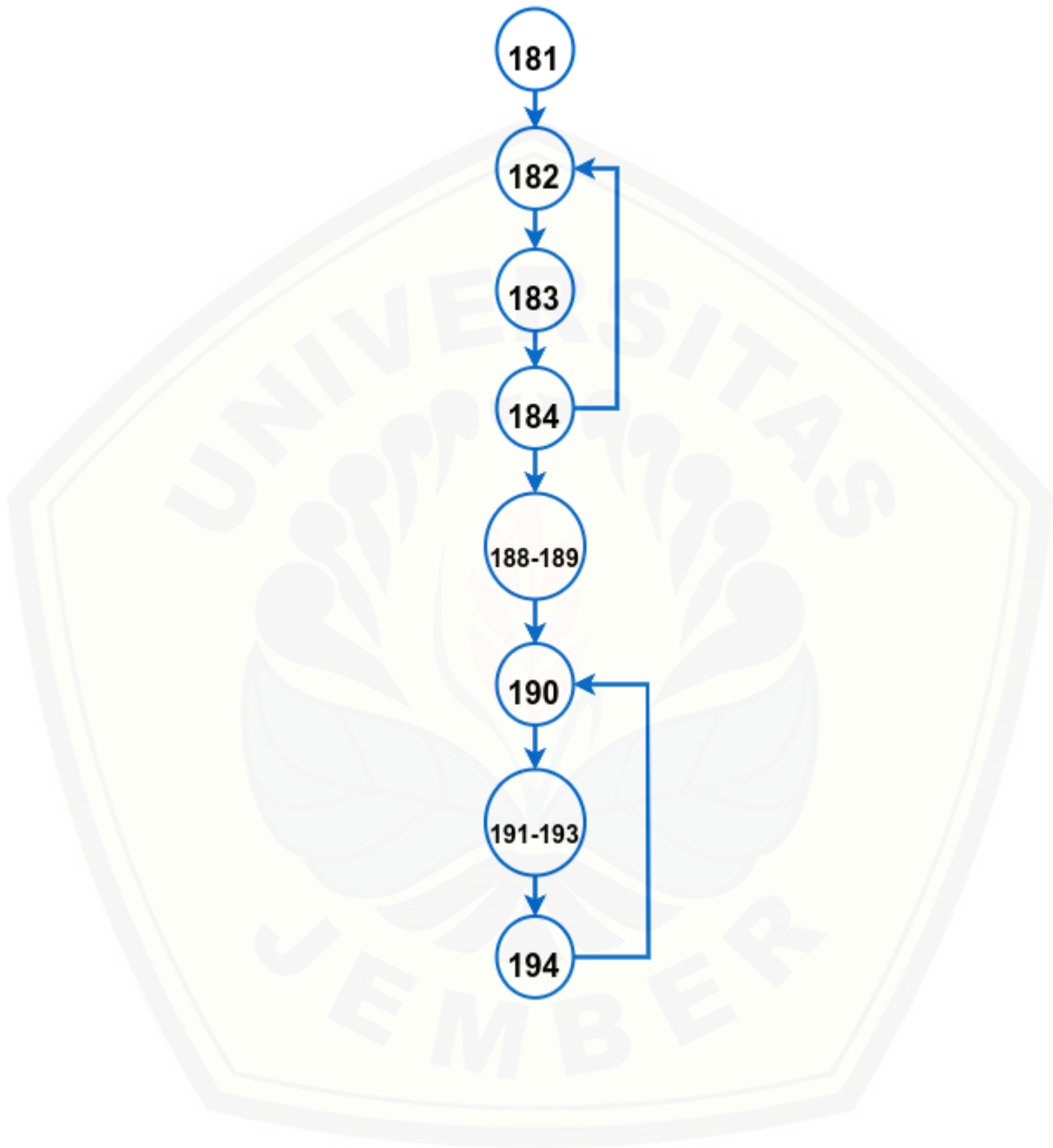
dilanjutkan



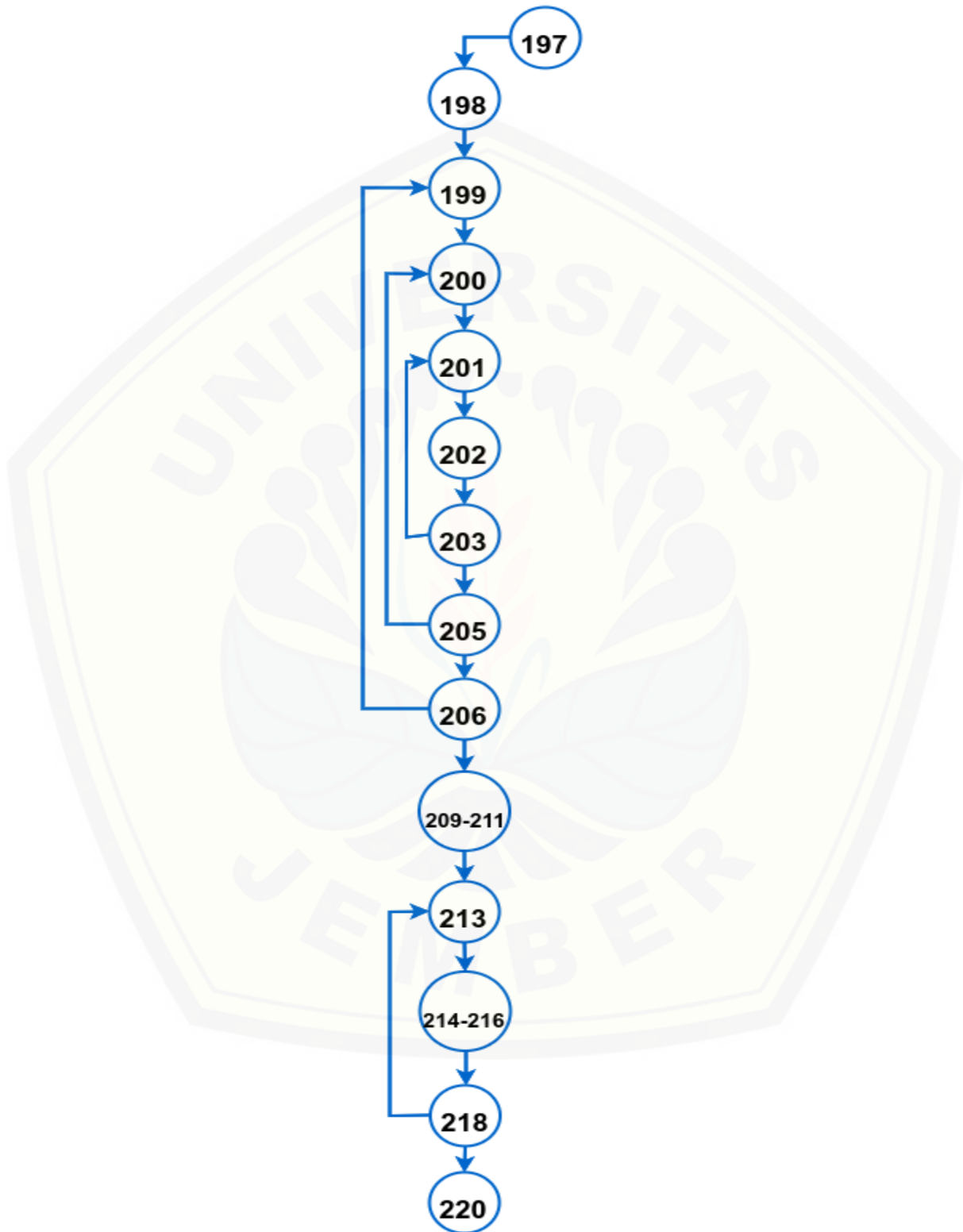
dilanjutkan



dilanjutkan



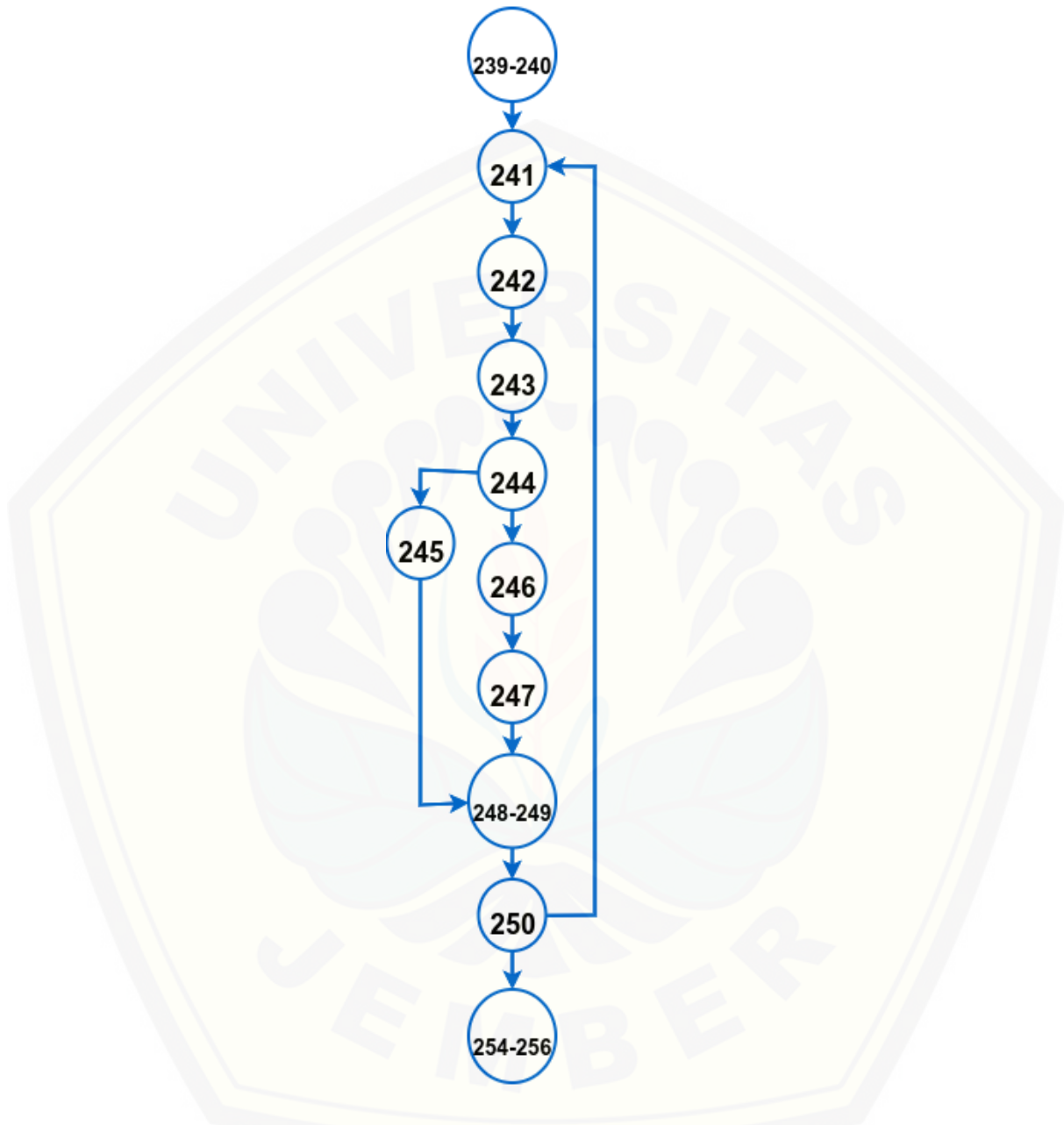
dilanjutkan



dilanjutkan

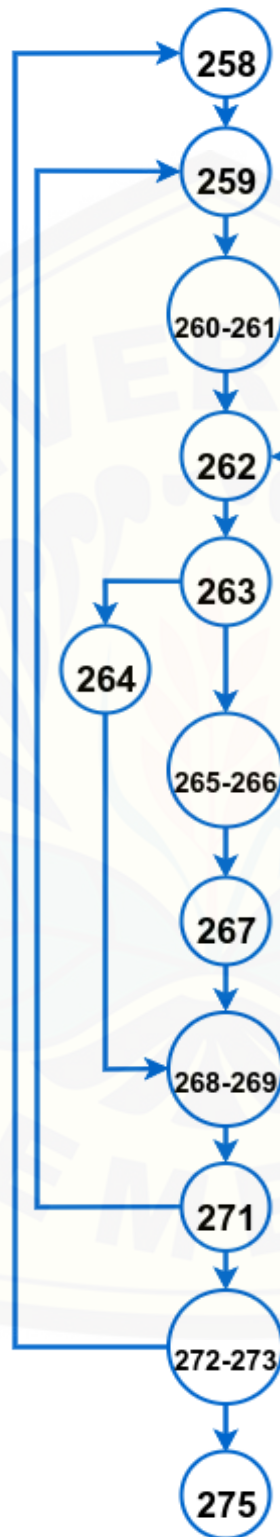


dilanjutkan

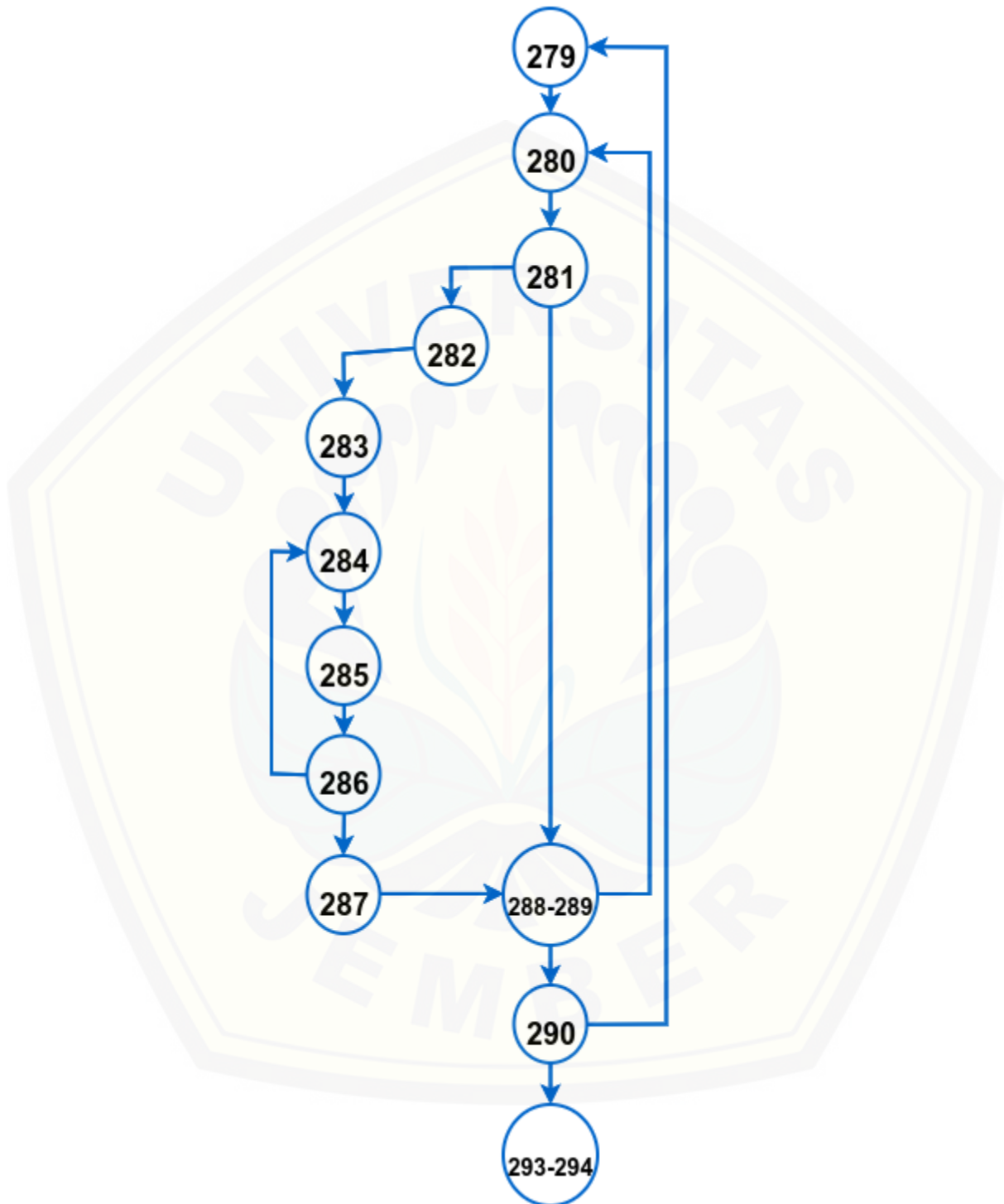


Gambar 4.24 Diagram Alir Penentuan Penggunaan Metode

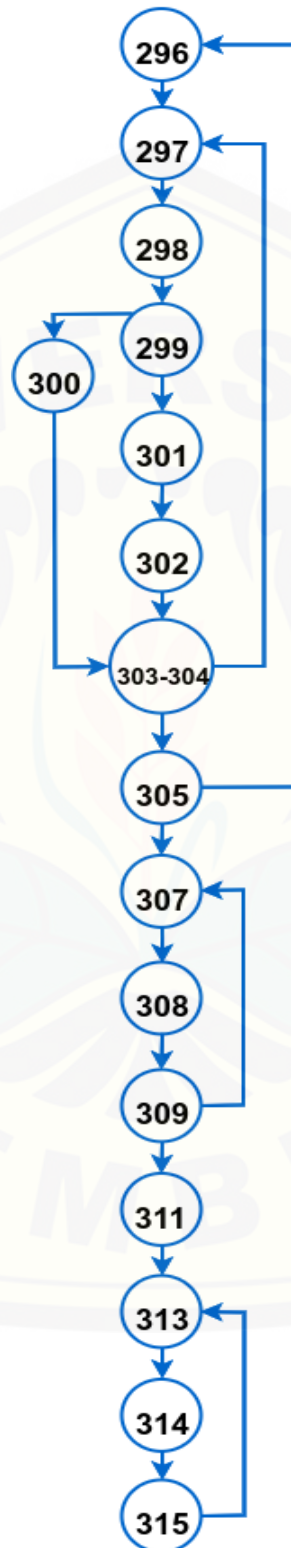
dilanjutkan



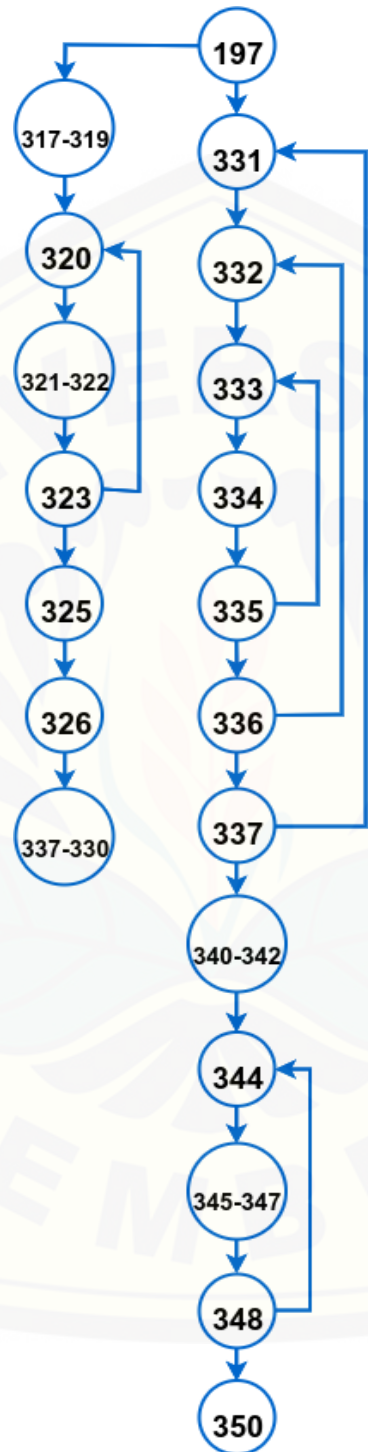
dilanjutkan

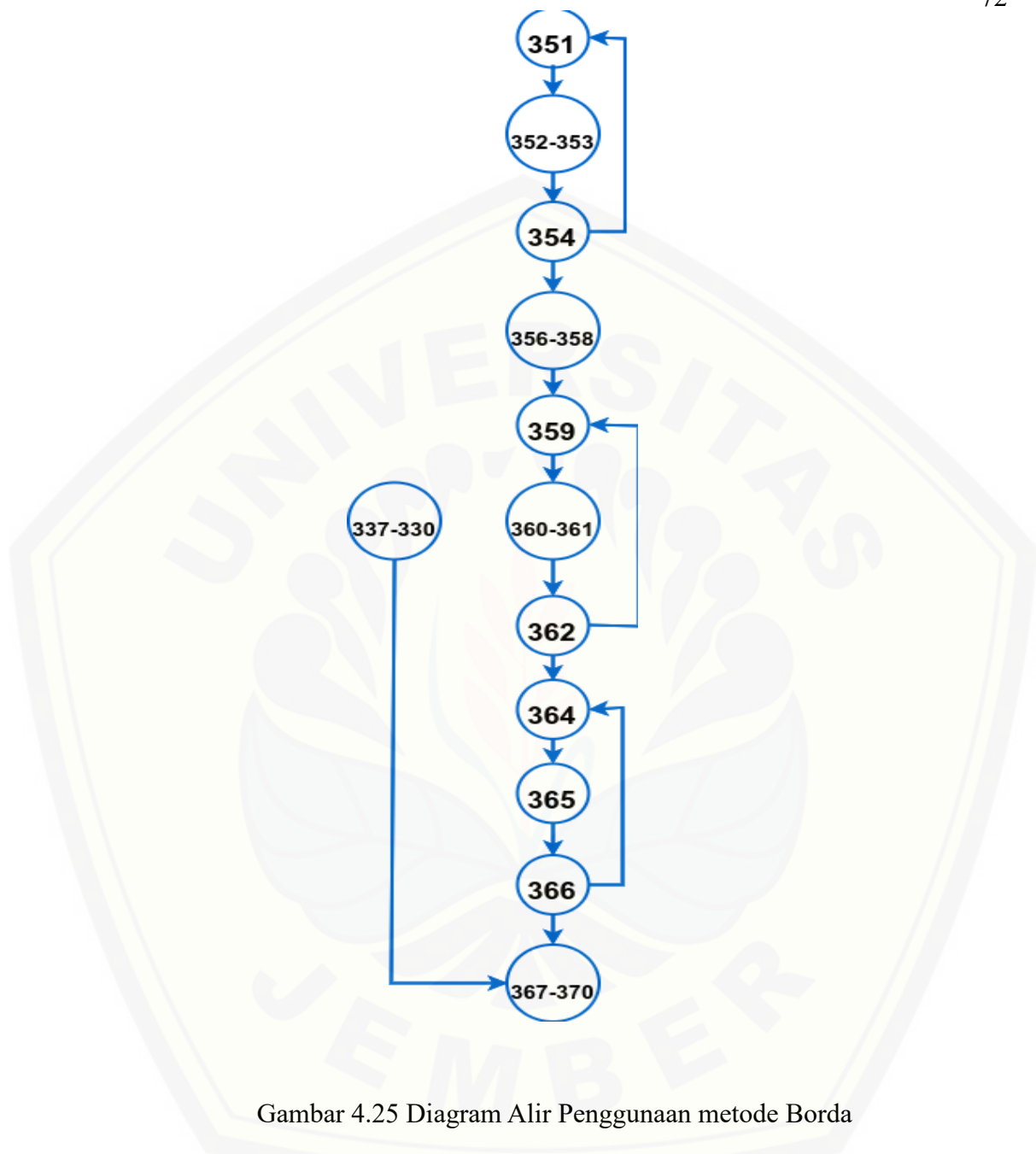


dilanjutkan



dilanjutkan





Pada gambar diatas, merupakan bentuk penggambaran notasi dari *listing program function startTour*. Pada *listing program function startTour*, proses dimulai dari penentuan kriteria pada gambar 4.23 pada gambar tersebut diagram alir nomor 92-96 merupakan isi dari proses suatu program yang dilakukan secara berurutan, untuk diagram alir nomor 96 merupakan awal dari percabangan suatu

kondisi atau kondisi *if-else*, nomor 97-99 merupakan isi dari proses *if-else* yang dilakukan berurutan, nomor 101 merupakan awal dari perulangan *foreach* dan begitu juga seterusnya.

3. *Cyclomati Complexity (CC)*

CC merupakan tahap perhitungan jumlah node, jumlah percabangan node sehingga dapat mengetahui seberapa kompleks *code* yang dapat dibuat.

3.1 *Function construct*

$$V(G) = E - N + 2$$

$$V(G) = 2 - 3 + 2 = 1$$

Function construct memiliki total CC yaitu 1, dengan rincian sebagai berikut: memiliki 2 percabangan node atau bisa disebut dengan jumlah panah yang menuju pada node (E) sedangkan bernilai 3 untuk jumlah node (N) itu sendiri.

3.2 *Function index*

$$V(G) = E - N + 2$$

$$V(G) = 2 - 3 + 2 = 1$$

Function index memiliki total CC yaitu 1, dengan rincian sebagai berikut: memiliki 2 percabangan node atau bisa disebut dengan jumlah panah yang menuju pada node (E) sedangkan bernilai 3 untuk jumlah node (N) itu sendiri.

3.3 *Function detailWisata*

$$V(G) = E - N + 2$$

$$V(G) = 2 - 3 + 2 = 1$$

Function detailWisata memiliki total CC yaitu 1, dengan rincian sebagai berikut: memiliki 2 percabangan node atau bisa disebut dengan jumlah panah yang menuju pada node (E) sedangkan bernilai 3 untuk jumlah node (N) itu sendiri.

3.4 *Function* detailHotel

$$V(G) = E - N + 2$$

$$V(G) = 2 - 3 + 2 = 1$$

Function detailHotel memiliki total CC yaitu 1, dengan rincian sebagai berikut: memiliki 2 percabangan node atau bisa disebut dengan jumlah panah yang menuju pada node (E) sedangkan bernilai 3 untuk jumlah node (N) itu sendiri.

3.5 *Function* detailKuliner

$$V(G) = E - N + 2$$

$$V(G) = 2 - 3 + 2 = 1$$

Function detailKuliner memiliki total CC yaitu 1, dengan rincian sebagai berikut: memiliki 2 percabangan node atau bisa disebut dengan jumlah panah yang menuju pada node (E) sedangkan bernilai 3 untuk jumlah node (N) itu sendiri.

3.6 *Function* formTour

$$V(G) = E - N + 2$$

$$V(G) = 2 - 3 + 2 = 1$$

Function formTour memiliki total CC yaitu 1, dengan rincian sebagai berikut: memiliki 2 percabangan node atau bisa disebut dengan jumlah panah

yang menuju pada node (E) sedangkan bernilai 3 untuk jumlah node (N) itu sendiri.

3.7 *Function dostartroute*

$$V(G) = E - N + 2$$

$$V(G) = 2 - 3 + 2 = 1$$

Function dostartroute memiliki total CC yaitu 1, dengan rincian sebagai berikut: memiliki 2 percabangan node atau bisa disebut dengan jumlah panah yang menuju pada node (E) sedangkan bernilai 3 untuk jumlah node (N) itu sendiri.

3.8 *Function startTouring*

$$V(G) = E - N + 2$$

$$V(G) = 205 - 167 + 2 = 40$$

Function startTouring memiliki total CC yaitu 40, dengan rincian sebagai berikut: memiliki 205 percabangan node atau bisa disebut dengan jumlah panah yang menuju pada node (E) sedangkan bernilai 167 untuk jumlah node (N) itu sendiri.

4. *Test Case*

Test Case merupakan tahap terakhir dari pengujian *white box*, pada tahap ini menjelaskan target yang diharapkan dari suatu *function code*, hasil pengujian sistem, dan nomor jalur node pada diagram alir pada suatu *function*.

Tabel 4.7 *Test Case construct*

<i>Test Case</i>	Jika function construct dijalankan
Target yang Diharapkan	Berhasil memanggil wisata, hotel, kuliner, perhitungan

	repository
Hasil Pengujian	Benar, repository dapat dipanggil
Jalur	23, 24-27, 28

Tabel 4.8 *Test Case* index

<i>Test Case</i>	Jika function index dijalankan
Target yang Diharapkan	Berhasil memanggil model wisata, hotel, kuliner pada wisata, hotel, kuliner repository dan memanggil halaman <code>touring.index</code> untuk menampilkan data wisata, hotel dan kuliner
Hasil Pengujian	Benar, model yang ada pada tiap-tiap repository berhasil dipanggil dan ditampilkan pada halaman <code>touring.index</code>
Jalur	30, 31-36, 37

Tabel 4.9 *Test Case* detailWisata

<i>Test Case</i>	Jika function detailWisata dijalankan
Target yang Diharapkan	Berhasil memanggil wisata model wisata berdasarkan id tertentu untuk ditampilkan pada halaman <code>touring.detailWisata</code>
Hasil Pengujian	Benar, berhasil memanggil wisata model dan menampilkannya pada halaman <code>touring.detailWisata</code>
Jalur	39, 40-41, 42

Tabel 4.10 *Test Case* detailHotel

<i>Test Case</i>	Jika function detailHotel dijalankan
Target yang Diharapkan	Berhasil memanggil hotel model berdasarkan id tertentu untuk ditampilkan pada halaman touring.detailHotel
Hasil Pengujian	Benar, berhasil memanggil hotel model dan menampilkannya pada halaman touring.detailHotel
Jalur	44, 45-46, 47

Tabel 4.11 *Test Case* detailKuliner

<i>Test Case</i>	Jika function detailKuliner dijalankan
Target yang Diharapkan	Berhasil memanggil kuliner model berdasarkan id tertentu untuk ditampilkan pada halaman touring.detailKuliner
Hasil Pengujian	Benar, berhasil memanggil kuliner model dan menampilkannya pada halaman touring.detailKuliner
Jalur	49, 50-51, 52

Tabel 4.12 *Test Case formTour*

<i>Test Case</i>	Jika function formTour dijalankan
Target yang Diharapkan	Berhasil memanggil lokasi model pada perhitungan repository, dan memanggil model range untuk ditampilkan pada touring.formTour
Hasil Pengujian	Benar, berhasil memanggil lokasi dan range model dan

	dapat di tampilkan pada <code>touring.formTour</code>
Jalur	54, 55-57, 58

Tabel 4.13 *Test Case dostartroute*

<i>Test Case</i>	Jika function <i>dostartroute</i> dijalankan
Target yang Diharapkan	Berhasil memanggil model wisata, hotel, kuliner pada wisata, hotel, kuliner repository dan memanggil function <i>startTouring</i> yang akan ditampilkan pada <code>touring.startTour</code>
Hasil Pengujian	Benar, berhasil memanggil lokasi dan range model dan dapat di tampilkan pada <code>touring.formTour</code>
Jalur	60, 61-88, 89

Tabel 4.14 *Test Case startTouring*

<i>Test Case</i>	Jika function <i>startTouring</i> dijalankan
Target yang Diharapkan	Berhasil memanggil <i>request</i> yang terdapat pada function <i>dostartroute</i> dan melakukan perhitungan metode SAW dan Borda serta mengembalikan nilai perhitungan kepada function <i>dostartroute</i>
Hasil Pengujian	Benar, berhasil memanggil semua <i>request</i> pada function

	<i>dostartroute</i> dan berhasil melakukan perhitungan metode SAW dan Borda serta mengembalikan nilai perhitungan kepada <i>function dostartroute</i>
Jalur	91, 92-94, 96, 97-99, 101, 103, 104, 105, 106-107, 109, 110, 111, 112-115, 117, 118, 119, 120-122, 101, 125, 126-129, 130-131, 134, 135, 136, 137, 139-143, 144, 145, 146, 148-151, 152, 153, 154, 156-161, 162, 163, 164, 165, 166, 171-172, 167, 168, 169, 170, 171-172, 176, 177, 178, 181, 182, 183, 184, 188-189, 190, 191-193, 194, 197, 198, 199, 200, 201, 202, 203, 205, 206, 209-211, 213, 214-216, 218, 220, 221, 222, 223, 224, 225, 226, 227, 229, 230, 231, 233, 234, 235, 236, 237, 239-240, 241, 242, 243, 244, 245, 248-249, 246, 247, 248-249, 250, 254-256, 258, 259, 260-261, 262, 263, 264, 268-269, 265-266, 267, 268-269, 271, 272-273, 275, 279, 280, 281, 282, 283, 284, 285, 286, 287, 288-289, 290, 293-294, 296, 297, 298, 299, 300, 303-304, 301, 302, 303-304, 305, 307, 308, 309, 311, 313, 314, 315, 197, 317-319, 320, 321-322, 323, 325, 326, 337-330, 331, 332, 333, 334, 335, 336, 337, 340-342, 344, 345-347, 348, 350, 351, 352-353, 354, 356-358, 359, 360-361, 362, 364, 365, 366, 337-330, 367-370

4.3.2. Pengujian *Black Box*

Pengujian *Black Box* merupakan pengujian fungsionalitas sistem untuk mengetahui apakah input, output, dan proses didalam sistem dapat berjalan sesuai

dengan hasil yang diinginkan. Hasil pengujian sistem ini dapat dilihat pada lampiran D.

