

**IMPLEMENTASI ALGORITMA FUZZY UNTUK PENCARIAN
LOKASI PARKIR YANG IDEAL**

TUGAS AKHIR

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik
Pada jurusan Teknik Informatika Fakultas Sains dan Teknologi
Universitas Islam Negeri Sunan Gunung Djati Bandung

Oleh

Muhamad Adnan Nurus Salam

1167050095



BANDUNG

2023 M / 1444 H

LEMBAR PERSETUJUAN

**IMPLEMENTASI ALGORITMA FUZZY UNTUK PENCARIAN
LOKASI PARKIR YANG IDEAL**

Oleh :

Muhamad Adnan Nurus Salam

1167050095

Menyetujui,

Pembimbing I

Pembimbing II

Undang Syaripudin, M.Kom.
NIP. 197909302009121002

Popon Dauni, ST., M.Kom.
NIDN. 0416118401

**Dekan Fakultas
Sains dan Teknologi**

**Ketua Jurusan
Teknik Informatika**

Prof. Dr. Hasniah Aliah, M.Si.
NIP. 197806132005012014

Dr. Cepy Slamet, M.Kom.
NIP. 198002252011011007

LEMBAR PENGESAHAN

Tugas Akhir yang berjudul IMPLEMENTASI ALGORITMA FUZZY UNTUK PENCARIAN LOKASI PARKIR YANG IDEAL dinyatakan sah dan telah disidangkan dalam sidang MUNAQASYAH Fakultas Sains dan Teknologi, Universitas Islam Negeri Sunan Gunung Djati Bandung pada tanggal 25 Agustus 2023 oleh Majelis Sidang yang terdiri dari:

Bandung, 25 Agustus 2023

Ketua Majelis

Sekretaris Majelis

Undang Syaripudin, M.Kom.

NIP. 197909302009121002

Popon Dauni, ST., M.Kom.

NIDN. 0416118401

Mengetahui,

Penguji I

Penguji II

Jumadi, ST. M.Cs.

NIP. 197805312007101001

Muhammad Deden Firdaus, ST, M.Kom

NIP. 199107202020121010

SURAT PERNYATAAN KARYA SENDIRI

ABSTRAK

Kepemilikan kendaraan menjadi impian banyak keluarga, namun kendala dalam penyimpanan sering mengakibatkan parkir sembarangan di jalanan, mengganggu lalu lintas dan menciptakan masalah kemacetan. Penelitian ini menggali solusi melalui implementasi logika fuzzy untuk mencari lokasi parkir ideal. Algoritma Fuzzy digunakan untuk menentukan lokasi parkir ideal melalui sistem inferensi metode fuzzy. Faktor seperti fasilitas, keamanan, dan jarak menjadi pertimbangan dalam mencari tempat parkir yang nyaman. Model waterfall digunakan untuk pendekatan penelitian, memastikan tahapan yang teratur. Hasil pengujian menunjukkan bahwa algoritma fuzzy efektif digunakan dalam menentukan tempat parkir yang ideal. Kinerja logika fuzzy berjalan dengan baik dan sesuai dengan yang diharapkan, memberikan rekomendasi lokasi parkir yang akurat. Dengan hasil penelitian ini, diharapkan dapat mengurangi praktik parkir liar dan mencari tempat parkir yang aman.

Kata kunci: Algoritma Fuzzy, Android, Penentuan lokasi parkir

ABSTRACT

Maintaining a vehicle is the dream of many families, but problems with storage often result in haphazard parking on the road, disrupting traffic and causing congestion problems. This research explores solutions through the implementation of fuzzy logic to find ideal parking locations. The Fuzzy Algorithm is used to determine the ideal parking location through a fuzzy method inference system. Factors such as facilities, security and distance are considerations when looking for a comfortable parking space. A waterfall model was used to approach the research, ensuring orderly stages. The test results show that the fuzzy algorithm is effectively used in determining the ideal parking spot. Fuzzy logic performance runs well and as expected, providing accurate parking location recommendations. With the results of this research, it is hoped that it can reduce fraudulent parking practices and find safe parking spaces.

Keywords: Fuzzy algorithm, Android, Parking lot finder

KATA PENGANTAR

Assalamualaikum Wr. Wb.,

Dengan rasa syukur Alhamdulillah, atas ke hadirat Allah SWT., Atas karunia-Nya yang tak terhingga. Ini adalah anugerah-Nya yang telah memberikan saya kekuatan, kemampuan, dan kelancaran untuk menyelesaikan tugas akhir yang berjudul "IMPLEMENTASI ALOGIRTMA FUZZY UNTUK PENCARIAN LOKASI PARKIR YANG IDEAL"

Saya juga ingin mengucapkan terima kasih kepada semua individu dan entitas yang telah memberikan dukungan serta kontribusi mereka dalam menyelesaikan tugas akhir ini. Terima kasih kepada:

1. Allah SWT, atas rahmat-Nya yang luar biasa.
2. Nabi Muhammad SAW, sebagai teladan dalam kehidupan kita.
3. Orang tua saya yang selalu memberikan dukungan tanpa henti.
4. Para dosen di Jurusan Teknik Informatika, yang memberikan saran dan masukan berharga tentang tugas akhir ini.
5. Bapak Undang Syaripudin, M.Kom., sebagai pembimbing pertama, dan Ibu Popon Dauni, ST., M.Kom., sebagai pembimbing kedua, atas bimbingan, arahan, dan kesabaran mereka dalam membimbing saya hingga menyelesaikan tugas akhir ini.
6. Bapak Jumadi, ST. M.Cs., sebagai penguji pertama, dan Bapak Muhammad Deden Firdaus, ST., M.Kom., sebagai penguji kedua, atas masukan dan evaluasi yang mereka berikan yang telah membantu meningkatkan kualitas penelitian ini.
7. Teman-teman dan semua pihak yang telah berbagi informasi, memberikan wawasan, dan berkontribusi dalam peningkatan pengetahuan saya.

Akhirnya, saya menerima dengan tulus setiap kritik dan saran untuk perbaikan di masa yang akan datang. Semoga tesis ini dapat memberikan sumbangan yang bermanfaat dalam bidang pendidikan dan teknologi informasi.

Wassalamualaikum Wr. Wb.

Bandung, 25 Agustus 2023

Muhamad Adnan Nur Salam

DAFTAR ISI

LEMBAR PERSETUJUAN	i
LEMBAR PENGESAHAN	ii
SURAT PERNYATAAN KARYA SENDIRI	iii
ABSTRAK	iv
ABSTRACT	v
KATA PENGANTAR	vi
BAB I PENDAHULUAN	1
1.1 Latar Belakang Penelitian	1
1.2 Rumusan Masalah Penelitian	3
1.3 Tujuan Penelitian.....	3
1.4 Batasan Masalah Penelitian.....	3
1.5 Kerangka Pemikiran Penelitian	4
1.6 Metodologi Penelitian	4
1.7 Sistematika Penulisan.....	5
BAB II KAJIAN LITERATUR	6
2.1 Tinjauan Pustaka	6
2.2 Landasan Teori	15
2.2.1 Parkir dan Lahan Parkir	15
2.2.2 Sistem Pendukung Keputusan	16
2.2.3 Algoritma Fuzzy	18
2.2.4 Flutter.....	18
2.2.5 Bahasa Pemrograman Dart	19
2.2.6 MySql.....	19
2.2.7 Waterfall	20
2.2.8 CodeIgniter	20
2.2.9 Rest API.....	21
2.2.10 Postman.....	22
2.2.11 Google Maps API	22
BAB III METODOLOGI PENELITIAN	23
3.1 Analisi Kebutuhan	23
3.1.1 Metode Pengumpulan Data.....	23

3.1.2 Metode Pengembangan.....	23
3.1.3 Kebutuhan Fungsional	24
3.1.4 Kebutuhan Non-Fungsional.....	25
3.1.5 Kebutuhan Perangkat.....	25
3.1.6 Analisis Algoritma Fuzzy	26
3.2 Perancangan.....	28
3.2.1 Perancangan Class Diagram	28
3.2.2 Perancangan Use Case Diagram	29
3.2.3 Perancangan Sequence Diagram.....	32
3.2.4 Perancangan User Interface	33
3.3 Implementasi	35
3.3.1 Implementasi Basis Data	35
3.3.2 Implementasi Algoritma Fuzzy	37
3.3.3 Implementasi User Interface.....	41
3.3.4 Pengujian Sistem.....	44
BAB IV HASIL DAN PEMBAHASAN	48
4.1 Implementasi Algoritma Fuzzy	48
4.2 Kinerja Algoritma Fuzzy.....	52
4.3 Pembahasan Pengujian.....	54
BAB V KESIMPULAN DAN SARAN	55
5.1 Kesimpulan.....	55
5.2 Saran.....	55
DAFTAR PUSTAKA	56

DAFTAR TABEL

Tabel 2.1 Tinjauan Pustaka	11
Tabel 3.1 Kebutuhan Fungsional	25
Tabel 3.2 Kebutuhan Non-Fungsional	25
Tabel 3.3 Definisi Use Case	30
Tabel 3.4 Skenario Use Case 1	30
Tabel 3.5 Halaman Login	44
Tabel 3.6 Halaman Register	45
Tabel 3.7 Halaman Menu Utama	46
Tabel 3.8 Halaman Manage Akun	47
Tabel 3.9 Halaman Register Lokasi	47
Tabel 4.1 Data Penelitian	48
Tabel 4.2 Aturan Fuzzy	50
Tabel 4.3 Hasil Fuzzifikasi	52
Tabel 4.4 Hasil Inferensi	53
Tabel 4.5 Hasil Defuzzifikasi	54

DAFTAR GAMBAR

Gambar 1.1 Kerangka Pemikiran.....	4
Gambar 3.1 Metode Pengembangan Waterfall.....	23
Gambar 3.2 Representasi linier naik.....	27
Gambar 3.3 Representasi linier Turun.....	27
Gambar 3.4.a Class Diagram User.....	28
Gambar 3.4.b Class Diagram Admin 1.....	29
Gambar 3.5 Use Case Diagram.....	29
Gambar 3.6.a Sequence Diagram User.....	32
Gambar 3.6.b Sequence Diagram Admin.....	33
Gambar 3.7 Tampilan Login.....	34
Gambar 3.8 Tampilan Menu Utama.....	34
Gambar 3.9 Tampilan Lokasi.....	35
Gambar 3.10 Login dan Register User.....	41
Gambar 3.11 Tampilan Menu Utama.....	41
Gambar 3.12 Tampilan Deskripsi Lokasi.....	42
Gambar 3.13 Side Bar.....	43
Gambar 3.14 Manage Admin.....	43
Gambar 3.15 Manage Pengguna.....	44

BAB I

PENDAHULUAN

1.1 Latar Belakang Penelitian

Memiliki kendaraan memanglah impian semua keluarga. Kita bisa mengajak semua anggota keluarga kita. Sayangnya, tidak semua pemilik kendaraan mempunyai tempat untuk menyimpan kendaraannya, sampai tidak sedikit orang menggunakan jalanan umum untuk menyimpan kendaraannya dan membuat orang lain susah untuk melewati jalan tersebut bahkan membuat jalanan menjadi macet.

Selain jalanan opsi lain yaitu lapangan atau lahan kosong yang mungkin harus meminta izin terlebih dahulu kepada pemilik lahan, terkadang opsi ini jarang digunakan. Mungkin karena sering digunakan untuk aktivitas warga setempat dan lokasinya yang kurang aman. Beberapa daerah, warga mengubah lahan kosong menjadi tempat parkir untuk umum bahkan ada juga yang membuat garasi untuk disewakan kepada warga yang mempunyai kendaraan agar tidak terjadi parkir liar yang dapat menutupi jalan.

Hampir seluruh kegiatan kini dapat dilakukan secara online, dan berbagai isu muncul sehubungan dengan meningkatnya jumlah kendaraan yang sering diparkir di sisi jalan, sesuai dengan Peraturan Daerah No 5 tahun 2014 mengenai transportasi yang mengatur permasalahan parkir yang mengganggu bahu jalan dan berkontribusi pada terjadinya kemacetan[1].

Seseorang yang memiliki kendaraan dan memarkirkannya dengan tidak tertib di depan rumah dapat dikenai sanksi sesuai dengan Undang-Undang Nomor 22 tahun 2009 tentang Lalu Lintas Angkutan Jalan (LLAJ) Pasal 106 ayat 4. Ayat tersebut berbunyi sebagai berikut: "Setiap orang yang melakukan perbuatan yang mengakibatkan gangguan pada fungsi Rambu Lalu Lintas, Marka Jalan, Alat Pemberi Isyarat Lalu Lintas, fasilitas pejalan kaki, dan alat pengaman pengguna jalan sebagaimana dimaksud dalam pasal 28 ayat 2 dipidana kurungan paling lama 1 bulan atau denda paling banyak Rp 250.000." Artinya, Jika seseorang melakukan parkir yang tidak benar, yang mengganggu fungsi berbagai komponen lalu lintas sesuai dengan ketentuan Pasal 28 ayat 2, mereka bisa dikenai sanksi berupa penjara dengan jangka waktu maksimal satu

bulan atau denda sejumlah Rp 250.000[2].

Masyarakat sering memilih untuk parkir sembarangan di tepi jalan karena beberapa faktor, menurut pernyataan Kepala Dinas Perhubungan DKI Jakarta, M. Akbar. Salah satu alasan utama adalah kemalasan, di mana sebagian orang enggan untuk memarkirkan kendaraannya di gedung parkir yang mungkin lebih jauh dari tujuan mereka. Faktor kedua adalah alasan ekonomi, karena parkir di gedung biasanya lebih mahal daripada parkir di pinggir jalan. Di tepi jalan, membayar parkir hanya Rp 2.000 hingga Rp 3.000, sedangkan di gedung parkir tarifnya bisa progresif dan lebih tinggi jika kendaraan diparkir dalam waktu yang lama. Kedua faktor ini, yaitu kemalasan dan biaya parkir yang lebih rendah di pinggir jalan, menjadi permasalahan utama yang menyebabkan banyak orang memilih parkir sembarangan di tepi jalan daripada menggunakan fasilitas parkir yang disediakan secara resmi. Hal ini dapat menyebabkan gangguan pada fungsi lalu lintas dan menimbulkan kesulitan bagi pengguna jalan lainnya. Oleh karena itu, diperlukan langkah-langkah untuk menghadapi permasalahan parkir yang tidak teratur dengan memberikan pemahaman kepada masyarakat mengenai kepentingan menggunakan tempat parkir yang resmi dan menaati aturan lalu lintas yang telah ditetapkan[3].

Faktor ideal suatu lokasi dapat dilihat dari fasilitas yang tersedia di lokasi, dari luasnya parkir, keamanan, tersedianya bangunan agar kendaraan kita terhindar dari panas dan hujan, dan fasilitas lainnya. Selain itu jarak adalah faktor utama dalam mencari tempat parkir, selain mudah dijangkau kita juga dapat lebih mudah menjaganya.

Pada penelitian ini menggunakan Logika fuzzy, yang merupakan perkembangan dari logika Boolean yang bernilai hanya 0 dan 1, sedangkan logika fuzzy bernilai tidak pasti, samar, dan ketidakpastian atau di antara 0 dan 1. Logika fuzzy memiliki penerapan yang luas dalam teori pengambilan keputusan, kontrol sistem, serta berbagai aspek dalam ilmu manajemen. Salah satu keunggulannya yaitu mampu dalam melakukan penalaran berdasarkan bahasa, sehingga tidak lagi memerlukan persamaan matematika yang rumit untuk merancang sistem pengendalian objek.

Dalam proses pembuatan karya ilmiah, tentu diperlukan pendekatan penelitian tertentu. Dalam penelitian ini, digunakan pendekatan berbentuk model waterfall. Model waterfall memiliki karakteristik khusus di mana setiap tahap dalam proses waterfall harus selesai sebelum melanjutkan ke tahap berikutnya. Jadi kita bisa lebih fokus dalam menyelesaikan penelitian ini. Selain itu kita akan mendapatkan hasil penelitian yang baik dan akurat serta dapat lebih hemat biaya.

Dengan hasil penelitian ini, diharapkan bisa membantu mengurangi praktik parkir liar serta mencari tempat sewa parkir yang aman dan nyaman untuk menyimpan kendaraan kita tanpa kekhawatiran.

Dari permasalahan yang sudah dibahas sebelumnya dapat kita atasi dengan adanya aplikasi pencarian tempat parkir yang ideal. Dengan keamanan dan kenyamanan yang kita perlukan agar dapat dengan tenang menyimpan kendaraan kesayangan kita. Dengan ini, judul dari penelitian ini adalah “Implementasi Algoritma Fuzzy Untuk Pencarian Lokasi Parkir Yang Ideal”.

1.2 Rumusan Masalah Penelitian

Berdasarkan pada latar belakang, maka dapat merumuskan masalah sebagai berikut:

1. Bagaimana mengimplementasikan Logika Fuzzy dalam pencarian lokasi parkir ideal?
2. Bagaimana kinerja Logika Fuzzy pada sistem untuk menentukan tempat parkir ideal?

1.3 Tujuan Penelitian

Dengan mengacu pada perumusan masalah penelitian, maka tujuan yang ingin dicapai dalam penelitian ini adalah:

1. Mengimplementasikan Algoritma Fuzzy dalam mencari lokasi parkir ideal.
2. Mengetahui kinerja Algoritma Fuzzy pada sistem untuk menentukan tempat parkir ideal.

1.4 Batasan Masalah Penelitian

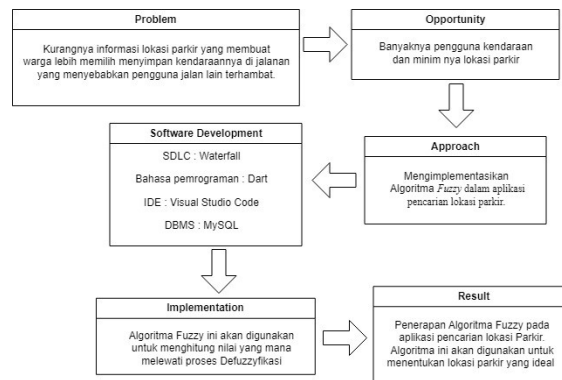
Untuk mencapai tujuan penelitian yang telah dijelaskan, dalam penelitian ini kami menentukan beberapa batasan masalah, di antaranya:

1. Aplikasi ini membahas Informasi lokasi parkir.

2. Aplikasi ini dapat diakses oleh pemilik kendaraan.
3. Ruang lingkup Kota Bandung.
4. Sistem yang dipergunakan untuk pengembangan sistem pendukung keputusan ini mengadopsi pendekatan Fuzzy.
5. Aplikasi ini hanya untuk sistem operasi mobile

1.5 Kerangka Pemikiran Penelitian

Adapun kerangka pemikiran, Gambar 1.1 menggambarkan kerangka pemikiran yang mendasari Aplikasi ini :



Gambar 1.1 Kerangka Pemikiran

Dalam ilustrasi 1.1, terdapat penjelasan mengenai langkah-langkah dalam melakukan penelitian untuk tugas akhir. Ini melibatkan analisis penerapan Algoritma Fuzzy dalam pencarian lokasi parkir, mengingat perkembangan teknologi yang sedang pesat saat ini. Ini memberikan dampak positif yang signifikan, terutama dalam memahami konsep tersebut dan kemudahan dalam penggunaannya di berbagai situasi.

1.6 Metodologi Penelitian

Metode pengumpulan data penelitian ini menggunakan dua tahap dalam proses pengumpulan data, yaitu:

1. Studi Literatur

Pada tahap ini dilakukan pengumpulan data dan menganalisis dari sumber tulisan, karya ilmiah, dan materi yang relevan dengan sistem yang akan dikembangkan.

2. Observasi

Observasi dilakukan dengan cara melakukan peninjauan serta penelitian langsung terhadap permasalahan yang di angkat.

1.7 Sistematika Penulisan

Sistematika penulisan laporan tugas akhir ini mencerminkan pandangan keseluruhan mengenai perangkat lunak/penelitian yang akan dijalankan. Oleh karena itu, susunan penulisan laporan ini adalah sebagai berikut:

BAB I PENDAHULUAN

Bab pertama menjelaskan mengenai latar belakang masalah, perumusan masalah, tujuan penelitian, batasan masalah, metodologi penelitian dan sistematika penulisan.

BAB II KAJIAN LITERATUR

Bab kedua ini menjelaskan tentang tinjauan pustaka dan landasan teori yang mendukung pengembangan Aplikasi pencarian tempat parkir ideal, serta beberapa definisi yang diajukan oleh pakar sebagai dasar dari konsep dalam laporan ini.

BAB III METODOLOGI PENELITIAN

Bab ketiga ini menjelaskan dengan penjelasan tentang langkah-langkah dan teknik yang diterapkan dalam penelitian, merujuk pada model-model proses pengembangan perangkat lunak yang telah ada atau model-model lain yang cocok dengan keperluan dan sifat kajian yang sedang dilakukan.

BAB IV HASIL DAN PEMBAHASAN

Bab keempat, dijelaskan hasil dari pengujian perangkat lunak melalui metode *blackbox* yang bertujuan untuk memastikan perangkat lunak berjalan sesuai perancangan.

BAB V KESIMPULAN DAN SARAN

Bab kelima menjelaskan secara umum hasil penelitian dan memberikan saran untuk perkembangan penelitian di masa depan.

BAB II

KAJIAN LITERATUR

2.1 Tinjauan Pustaka

Beberapa penelitian yang telah dilakukan sebelumnya untuk membuat aplikasi dengan memanfaatkan teknologi yang hasil objeknya lebih nyata. Dalam hal tersebut untuk mengembangkan sistem yang serupa dengan hal tersebut maka perlu adanya studi literature sebagai tahap metode penelitian yang dilakukan. Berikut ini adalah hasil penelitian yang telah dilakukan dan memiliki hubungan dengan penelitian yang akan dilakukan antara lain.

1. Tujuan dari penelitian ini adalah menerapkan metode fuzzy Sugeno untuk memilih rute optimal ke destinasi pariwisata di Surabaya. Penulis, Muizzatul Mukaromah. Metode yang diterapkan adalah pendekatan logika fuzzy versi Sugeno, yang melibatkan fuzzifikasi, implikasi, dan defuzzifikasi untuk mendapatkan jalur optimal dan kurang optimal. Studi ini akan mencakup enam lokasi wisata yang direncanakan untuk dikunjungi, termasuk Tugu Pahlawan, Food Junction Pakuwon, Atlantis Land, Kebun Binatang Surabaya (KBS), Suroboyo Carnival, dan juga Mangrove. Tiga titik awal yang akan dipertimbangkan adalah Balai kota Surabaya, Terminal Surabaya, dan Pelabuhan Tanjung Perak. Kriteria yang dimanfaatkan dalam studi ini mencakup jarak yang harus ditempuh, durasi yang dibutuhkan, dan tingkat kepadatan lalu lintas. Penggunaan kriteria tersebut didasarkan pada adanya keramaian di sepanjang jalur utama kota Surabaya yang dapat mempengaruhi waktu perjalanan ke lokasi wisata. Data yang digunakan dalam studi ini diperoleh dengan membandingkan tiga jalur yang berbeda menggunakan Google Maps. Dari perbandingan ini, akan dipilih rute terbaik yang paling cocok untuk mencapai destinasi wisata di Surabaya. Penelitian ini menunjukkan bahwa penggunaan Logika Fuzzy dapat menghasilkan keputusan yang lebih efisien dalam hal pemilihan jalur terbaik menuju lokasi pariwisata di Surabaya. Pendekatan menggunakan logika fuzzy ini mampu mengatasi kerumitan dalam menentukan jalur terbaik dengan membandingkan tiga opsi jalur yang tersedia dari setiap titik awal ke tujuan wisata, serta mempertimbangkan berbagai faktor yang dapat

memengaruhi perjalanan. Dengan menggunakan metode Sugeno, keputusan dapat diambil dengan mempertimbangkan beberapa faktor, seperti jarak, waktu, dan kepadatan jalan, sehingga memberikan hasil yang lebih tepat dan relevan. Dengan demikian, pengguna metode ini dapat memilih jalur terbaik yang paling sesuai untuk mencapai lokasi wisata di Surabaya[4].

2. Tujuan dari penelitian ini adalah untuk membuat sistem rekomendasi tempat wisata di Kota Malang menggunakan pendekatan berbasis web dengan memanfaatkan teknik fuzzy. Penulis penelitian ini adalah Elys Sugiharto, Istiadi, dan Indra Dharma Wijaya. Sistem berbasis web dengan logika Fuzzy Sugeno telah dikembangkan untuk membantu para wisatawan yang kerap kali merasa kebingungan dalam menentukan pilihan tempat wisata yang cocok dengan preferensi mereka di Kota Malang. Tidak tersedia sarana informasi yang dapat dijangkau dengan mudah mengakibatkan kesulitan bagi para wisatawan dalam mencari informasi terbaru tentang objek wisata di Kota Malang. Oleh karena itu, Sistem ini ditemukan sebagai jawaban dengan mengadopsi pendekatan Fuzzy Sugeno, yang merupakan teknik pengambilan keputusan peringkat berdasarkan toleransi terhadap ketidakpastian dalam data. Sistem ini dikembangkan menggunakan bahasa pemrograman PHP dengan basis data MySQL. Faktor-faktor yang terlibat dalam tahap fuzzifikasi meliputi elemen-elemen seperti fasilitas toilet, pilihan makanan, ruang ibadah, barang-barang oleh-oleh, area taman hiburan, dan lokasi untuk berfoto, dengan setiap elemen memiliki karakteristik keanggotaan yang unik. Hasil Output fuzzy berupa rating bintang 1 hingga 5 untuk setiap tempat wisata yang direkomendasikan. Sistem diuji dengan metode user testing, melibatkan partisipasi dari 5 orang responden. Menurut hasil pengujian, Metode Sugeno untuk perhitungan logika fuzzy terbukti menjadi alat yang efisien dalam membantu pengguna menemukan rekomendasi tempat wisata yang cocok berdasarkan jenis wisata yang diminati dan budget yang diinginkan. Dengan mengacu pada perhitungan yang sudah dilakukan, sistem ini dapat memberikan saran yang tepat dan sesuai. Sistem

rekomendasi ini berhasil diterima dengan baik oleh para pengguna, dan 60% dari partisipan menyatakan sangat setuju (SS) terhadap kesesuaian sistem ini dengan kebutuhan mereka. Dengan adanya sistem rekomendasi ini, diharapkan para wisatawan dapat dengan mudah memilih tempat wisata yang sesuai dengan preferensi mereka, mengoptimalkan waktu perjalanan, dan meningkatkan pengalaman berwisata di Kota Malang[5].

3. Penelitian ini bertujuan untuk menentukan objek-objek wisata yang sesuai di Kota Bandung dengan menggunakan metode Fuzzy Tsukamoto. Penulis penelitian ini adalah Mia Lusmiawati, Erfanti Fatkhiyah, dan Amir Hamzah. Metode yang digunakan dalam penelitian ini adalah metode Fuzzy Tsukamoto. Pendekatan fuzzy logic Tsukamoto digunakan untuk memproses data dan menentukan objek-objek wisata yang paling sesuai dengan beberapa kriteria yang relevan, seperti popularitas, aksesibilitas, fasilitas, dan lain-lain. Penelitian ini bertujuan untuk membantu para wisatawan dalam merencanakan perjalanan mereka di Kota Bandung dengan lebih efisien dan efektif. Dengan menggunakan metode Fuzzy Tsukamoto, diharapkan dapat memberikan rekomendasi objek-objek wisata yang paling relevan dengan preferensi dan kebutuhan para wisatawan. Penelitian ini diharapkan dapat memberikan kontribusi dalam pengembangan teknologi informasi dan pariwisata di Kota Bandung. Metode Fuzzy Tsukamoto yang digunakan dalam penelitian ini dapat dijadikan acuan untuk pengembangan sistem rekomendasi objek wisata di kota-kota lainnya, dengan mempertimbangkan berbagai variabel yang mempengaruhi pengambilan keputusan. Dengan adanya penelitian ini, diharapkan para wisatawan dapat menikmati pengalaman wisata yang lebih menyenangkan dan memuaskan di Kota Bandung, serta dapat lebih memaksimalkan waktu dan sumber daya yang dimiliki dalam perjalanan wisata mereka[6].
4. Penelitian ini bertujuan untuk mencari rute terbaik bagi pemadam kebakaran di Kota Semarang dengan memanfaatkan algoritma Dijkstra yang disesuaikan dengan logika fuzzy sebagai penentu bobot pada graf. Penulis penelitian ini adalah Nanang Nggufon, Rochmad, dan Mashuri.

Metode yang digunakan dalam penelitian ini adalah algoritma Dijkstra yang telah dimodifikasi dengan logika fuzzy sebagai penentu bobot pada graf. Algoritma Dijkstra digunakan untuk mencari rute terpendek dari lokasi pemadam kebakaran ke lokasi yang membutuhkan pertolongan dengan mempertimbangkan beberapa faktor, seperti jarak, waktu tempuh, kepadatan lalu lintas, dan lain-lain, yang diubah menjadi bobot fuzzy. Penelitian ini memiliki tujuan untuk membantu pemadam kebakaran dalam merespons kejadian darurat dengan lebih efisien dan cepat. Dengan menggunakan algoritma Dijkstra yang disesuaikan dengan logika fuzzy, diharapkan dapat memberikan rute terbaik bagi pemadam kebakaran yang mempertimbangkan kondisi lalu lintas dan faktor-faktor lainnya untuk mencapai lokasi kebakaran dengan lebih efektif. Penelitian ini diharapkan dapat memberikan kontribusi dalam pengembangan teknologi dan penerapan ilmu matematika dalam bidang kebakaran dan penanganan darurat di Kota Semarang. Penggunaan algoritma Dijkstra dengan logika fuzzy sebagai penentu bobot pada graf dapat dijadikan acuan untuk pengembangan sistem pencarian rute terbaik dalam situasi darurat lainnya, dengan mempertimbangkan berbagai variabel yang mempengaruhi pengambilan keputusan[7].

5. Penelitian ini bertujuan untuk mengimplementasikan algoritma Fuzzy-Best Fit dalam mengembangkan sebuah sistem cerdas yang dapat membantu pengguna dalam mencari lokasi parkir terbaik. Penulis penelitian ini adalah Usman Nurhasan, Hadi Suyono, dan M. Aziz Muslim. Metode yang diimplementasikan dalam penelitian ini adalah algoritma Fuzzy-Best Fit. Algoritma ini memanfaatkan logika fuzzy untuk memproses data dan memberikan rekomendasi lokasi parkir yang paling sesuai dengan kriteria tertentu, seperti jarak ke tujuan, ketersediaan tempat parkir, dan biaya parkir. Penerapan algoritma ini bertujuan untuk meningkatkan efisiensi dan kenyamanan dalam mencari lokasi parkir bagi pengguna. Dengan sistem cerdas ini, pengguna dapat dengan mudah menentukan lokasi parkir yang ideal sesuai preferensi dan kebutuhan mereka. Pencarian lokasi parkir ideal yang mempertimbangkan berbagai kriteria dapat diatasi dengan

mengkombinasikan dua pendekatan, yakni penggunaan logika Fuzzy bersama dengan algoritma Best Fit. Proses pencarian jalur menggunakan kedua algoritma ini melibatkan perhitungan yang detail. Rekomendasi lokasi parkir terbaik dihasilkan berdasarkan nilai fuzzy output terkecil dan output Best Fit yang sesuai dengan bobot kendaraan, mempertimbangkan kondisi lapangan dan kendaraan. Penerapan algoritma ini memungkinkan tambahan parameter lain, seperti waktu parkir, pendeteksian penempatan dalam slot, dan skala prioritas pengguna. Nilai parkir selalu dinamis dengan pendekatan algoritma ini, sehingga proses dan area parkir yang dipilih dapat berubah sesuai dengan perubahan kondisi dan kepadatan area yang fluktuatif. Untuk meningkatkan efisiensi dan kenyamanan, diperlukan sistem tambahan seperti pendeteksi penempatan dalam slot menggunakan pengolahan citra atau sensor, serta sistem notifikasi yang baik agar informasi dapat diterima oleh pengguna parkir. Pembangunan sistem secara client-server juga disarankan agar penentuan lokasi parkir dapat berjalan dengan cepat, aman, dan real-time. Dengan menggabungkan logika Fuzzy dan algoritma Best Fit serta mempertimbangkan tambahan parameter dan sistem pendukung, penyelesaian permasalahan pencarian lokasi parkir dapat menjadi lebih efektif dan akurat, memastikan pengguna parkir mendapatkan rekomendasi yang optimal dan sesuai dengan kebutuhan mereka[8].

6. Sistem Pendukung Keputusan dalam menentukan Objek Wisata Alam di Daerah Istimewa Yogyakarta dengan Metode Fuzzy Tsukamoto yang ditulis oleh Oktaviani Rahmita Putri, Amir Hamzah, dan Erfanti Fatkhiyah. Membahas tentang pengembangan sistem pendukung keputusan untuk memilih objek wisata alam di Daerah Istimewa Yogyakarta. Dalam penelitian ini, digunakan metode Fuzzy Tsukamoto untuk mengatasi kompleksitas dan ketidakpastian dalam penilaian objek wisata alam. Metode ini menggunakan logika fuzzy untuk menyelesaikan masalah dengan kriteria dan parameter yang tidak jelas. Dengan memanfaatkan kecerdasan buatan, sistem ini dirancang dan dikembangkan menggunakan teknologi dan bahasa pemrograman terbaru. Tujuan dari sistem ini adalah

memberikan hasil yang lebih akurat dan efisien dalam membantu pengguna memilih objek wisata alam yang sesuai dengan preferensi dan kebutuhan mereka di Daerah Istimewa Yogyakarta. Sistem pendukung keputusan pemilihan objek wisata alam di Daerah Istimewa Yogyakarta dengan metode Fuzzy Tsukamoto berhasil dikembangkan dan sesuai dengan kebutuhan pengguna dalam mendapatkan rekomendasi tempat wisata yang optimal tanpa memerlukan proses pendaftaran untuk pengguna biasa[9].

7. Application of The Fuzzy Tsukamoto Method in Determining Household Industry Products yang ditulis oleh Nadra Savira Pasaribu, Jata Tata Hardinata, Hendry Qurniawan. Membahas penggunaan metode logika fuzzy Tsukamoto untuk membantu dalam menentukan atau memprediksi produk yang dihasilkan dari industri rumah tangga. UD. Mie Akwang adalah salah satu perusahaan yang melakukan kegiatan produksi. Produksi yang dilakukan adalah produksi mie pangsit. Permintaan konsumen yang berubah mengikuti perkembangan zaman dan kadang-kadang persediaan yang tidak sesuai menyebabkan UD. Mie Akwang mengalami kesulitan dalam menentukan jumlah produksi mie pangsit yang diproduksi. Masalah yang dihadapi oleh UD. Mie Akwang dapat diatasi dengan menggunakan metode logika fuzzy. Dengan menggunakan logika fuzzy, diharapkan dapat mengatasi masalah yang terjadi di UD. Mie Akwang. Berdasarkan tahapan penelitian yang telah dilakukan pada penerapan algoritma Fuzzy Tsukamoto dalam menentukan jumlah produksi, dapat disimpulkan bahwa metode Fuzzy Tsukamoto terbukti mampu menentukan jumlah produksi dalam industri rumahan berdasarkan jumlah permintaan dan jumlah pasokan[10].

Tabel 2.1 Tinjauan Pustaka

No.	Peneliti	Metode	Hasil
1.	Muizzatul Mukaromah (2019)	Metode Fuzzy Sugeno	Logika fuzzy memberikan keputusan yang lebih optimal terkait pemilihan jalur terbaik menuju lokasi wisata di Surabaya dengan

No.	Peneliti	Metode	Hasil
			<p>membandingkan tiga jalur di setiap titik awal dan lokasi wisata, serta mempertimbangkan beberapa kondisi yang memengaruhi perjalanan. metode Sugeno, keputusan dapat diambil dengan mempertimbangkan beberapa faktor, seperti jarak, waktu, dan kepadatan jalan, sehingga memberikan hasil yang lebih tepat dan relevan.</p>
2.	Elys Sugiharto, Istiadi, Indra Dharma Wijaya (2021)	Metode Fuzzy sugeno	<p>Perhitungan logika fuzzy menggunakan Metode Sugeno efektif dalam membantu pengguna menemukan rekomendasi tempat wisata yang sesuai berdasarkan kategori wisata dan harga yang diinginkan. Melalui perhitungan yang telah dilakukan, sistem ini mampu memberikan rekomendasi yang akurat dan relevan</p>
3.	Mia Lusmiawati, Erfanti Fatkhiyah, Amir Hamzah (2021)	Metode Fuzzy Tsukamoto	<p>Sistem yang dikembangkan menggunakan metode Fuzzy Tsukamoto dapat memberikan rekomendasi atau penilaian terhadap tempat wisata, sehingga memudahkan pengguna dalam memilih</p>

No.	Peneliti	Metode	Hasil
			destinasi wisata yang sesuai dengan preferensi mereka
4.	Nanang Nggufon, Rochmad, Mashuri (2019)	Algoritma Fuzzy	Dengan menggunakan logika fuzzy dan algoritma Dijkstra, dapat dihasilkan rute terbaik yang mengoptimalkan faktor-faktor penting seperti panjang dan kepadatan jalan, memastikan keselamatan dan kecepatan dalam penanganan kebakaran.
5.	Usman Nurhasan, Hadi suyono, M Aziz Muslim (2014)	Algoritma Best Fit dan Logika Fuzzy	Dengan menggunakan gabungan dua metode, yaitu logika Fuzzy dan algoritma Best Fit. Proses pencarian jalur menggunakan kedua algoritma ini melibatkan perhitungan yang detail. Rekomendasi lokasi parkir terbaik dihasilkan berdasarkan nilai fuzzy output terkecil dan output Best Fit yang sesuai dengan bobot kendaraan, mempertimbangkan kondisi lapangan dan kendaraan.

No.	Peneliti	Metode	Hasil
6.	Oktaviani Rahmatia Putri, Amir Hamzah, Erfanti Fatkhiyah (2021)	Metode Fuzzy Tsukamoto	Sistem pendukung keputusan pemilihan objek wisata alam di Daerah Istimewa Yogyakarta dengan metode Fuzzy Tsukamoto berhasil dikembangkan dan sesuai dengan kebutuhan pengguna dalam mendapatkan rekomendasi tempat wisata yang optimal tanpa memerlukan proses pendaftaran untuk pengguna biasa.
7	Nadra Savira Pasaribu, Jata Tata Hardinata, Hendry Qurniawan (2021)	Metode Fuzzy Tsukamoto	Berdasarkan tahapan penelitian yang telah dilakukan pada penerapan algoritma Fuzzy Tsukamoto dalam menentukan jumlah produksi, dapat disimpulkan bahwa metode Fuzzy Tsukamoto terbukti mampu menentukan jumlah produksi dalam industri rumahan berdasarkan jumlah permintaan dan jumlah pasokan
8	Muhamad Adnan Nurus Salam (2023)	Metode Fuzzy Tsukamoto	Hasil penelitian yang telah dilakukan pada penggunaan algoritma Fuzzy Tsukamoto untuk menemukan tempat parkir yang ideal, dapat

No.	Peneliti	Metode	Hasil
			dinyatakan bahwa metode Fuzzy Tsukamoto terbukti efektif dalam menentukan lokasi parkir yang ideal berdasarkan dengan rule yang telah dibuat berdasarkan jarak, jumlah kapasitas dan jumlah disukai

2.2 Landasan Teori

Dalam penelitian ini dibutuhkan beberapa tinjauan pustaka yang dapat mendukung dalam pencarian topik mengenai Implementasi Algoritma Fuzzy Dalam Menentukan Lokasi Parkir Yang Ideal :

2.2.1 Parkir dan Lahan Parkir

Parkir adalah saat kendaraan tidak bergerak dalam waktu yang tidak bersifat sementara. Ini mencakup kendaraan yang berhenti di lokasi tertentu, apakah ada rambu yang mengatur atau tidak, dan tujuan utamanya bukan hanya untuk mengambil atau menurunkan barang atau penumpang. Parkir adalah kebutuhan bagi pemilik kendaraan yang ingin meninggalkan kendaraannya di tempat yang mudah dijangkau. Salah satu bentuk kemudahan yang diinginkan adalah parkir di tepi jalan. Di sini, pola parkir yang umum adalah parkir sejajar atau sudut. Meskipun demikian, tidak selalu memungkinkan untuk parkir di tepi jalan karena kondisi lalu lintas yang tidak memungkinkan hal tersebut[11].

Tempat parkir atau lahan parkir adalah fasilitas yang disediakan untuk parkir kendaraan, baik itu berada di dalam atau di luar badan jalan, yang dapat diakses oleh pemilik kendaraan dan dikelola oleh pemerintah daerah atau pihak ketiga. Fasilitas parkir ini melibatkan area untuk menyimpan kendaraan bermotor dan juga ruang parkir tertutup untuk kendaraan bermotor. Tujuan dari tempat parkir adalah untuk mengatur penempatan kendaraan secara rapi sehingga tidak mengganggu pengendara lain. Jenis-

jenis parkir berdasarkan penempatan meliputi parkir di badan jalan (On Street Parking), yang dilakukan di atas badan jalan dengan mengurangi lebar jalan, namun memiliki kekurangan berupa kepadatan arus lalu lintas, kurangnya sudut pandang pengemudi, dan potensi kecelakaan akibat pengemudi yang berhenti dan keluar secara mendadak. Selain itu, terdapat juga parkir di luar badan jalan (Off Street Parking), yang biasanya dilakukan di lokasi khusus seperti pelataran atau bangunan bertingkat untuk parkir. Parkir juga dapat dibagi menjadi lima berdasarkan status lahan parkir, yaitu parkir umum yang dikelola oleh pemerintah daerah, parkir khusus yang dikelola oleh pihak ketiga, parkir darurat yang hanya tersedia pada waktu-waktu tertentu baik di lahan umum maupun milik swasta, gedung parkir yang berupa bangunan khusus untuk parkir, dan area parkir yang merupakan lahan atau bangunan dengan fasilitas sarana parkir yang dikelola oleh pemerintah daerah atau swasta[12].

2.2.2 Sistem Pendukung Keputusan

Sistem Pendukung Keputusan (SPK) merupakan komponen dalam sistem informasi berbasis komputer yang tergolong dalam sistem manajemen pengetahuan yang berakar pada pengetahuan. Tujuan Sistem Pendukung Keputusan (SPK) adalah meningkatkan kapasitas para pengambil keputusan dengan menyajikan lebih banyak atau lebih optimal opsi keputusan, sambil membantu dalam merinci permasalahan dan situasi yang sedang dihadapi. SPK berperan dalam menghemat waktu, tenaga, dan biaya dalam proses pengambilan keputusan. Dalam ringkasannya, SPK bertujuan untuk meningkatkan hasil yang tepat (*doing the right things*) dan hasil yang efisien (*doing the things right*) dalam proses pengambilan keputusan. Penekanan utama SPK adalah pada peningkatan efektivitas dalam pengambilan keputusan, yang berarti memastikan bahwa keputusan yang diambil sesuai dengan tujuan dan sasaran yang diinginkan. SPK membantu para pengambil keputusan dalam mengidentifikasi alternatif-alternatif yang paling tepat dan relevan dalam menghadapi situasi tertentu. Meskipun efisiensi juga menjadi perhatian, namun efektivitas lebih diutamakan karena tidak cukup hanya melakukan sesuatu dengan efisien,

tetapi harus memastikan bahwa keputusan yang diambil menghasilkan hasil yang diinginkan dan sesuai dengan visi serta misi organisasi.

Dengan demikian, SPK memiliki peran penting dalam membantu pengambil keputusan dalam mencapai hasil yang optimal dan memastikan bahwa sumber daya seperti waktu, tenaga, dan biaya digunakan secara efektif untuk mencapai tujuan organisasi atau instansi[13]. Beberapa karakteristik utama dari sistem pendukung keputusan berdasarkan definisi tersebut adalah:

1. Mendukung proses pengambilan keputusan: Sistem ini difokuskan pada membantu dalam proses pengambilan keputusan dan menekankan pada manajemen berbasis persepsi atau perception management.
2. Interface manusia/mesin: Sistem ini memiliki antarmuka yang memungkinkan pengguna (manusia) tetap memiliki kendali atas proses pengambilan keputusan.
3. Mendukung keputusan untuk pengambilan keputusan dalam situasi yang melibatkan masalah yang sudah terorganisir dengan baik, sebagian terorganisir, dan yang sama sekali tidak terorganisir: Sistem ini mampu mengatasi berbagai jenis masalah, baik yang memiliki struktur yang jelas (terstruktur) maupun masalah yang kompleks dan tidak memiliki struktur yang jelas (tak terstruktur).
4. Kapasitas dialog: "Sistem ini memiliki kemampuan untuk berkomunikasi dengan pengguna dengan tujuan mendapatkan informasi yang relevan sesuai dengan kebutuhan pengguna dalam proses pengambilan keputusan."
5. Integrasi subsistem: Sistem ini terdiri dari subsistem yang bekerja bersinergi dengan lancar, membentuk entitas yang komprehensif dan terpadu.
6. Membutuhkan struktur data komprehensif: Sistem ini memerlukan struktur data yang lengkap dan mampu melayani kebutuhan informasi di semua tingkatan manajemen.

Dengan karakteristik-karakteristik tersebut, sistem pendukung keputusan menjadi alat yang berharga bagi organisasi atau instansi dalam

mengambil keputusan yang tepat dan efisien dalam menghadapi berbagai masalah dan situasi yang dihadapi.

2.2.3 Algoritma Fuzzy

Metode Logika Fuzzy adalah cara yang efisien untuk menghubungkan ruang input ke dalam ruang output. Dalam sistem yang sangat kompleks, Logika Fuzzy menjadi solusi yang lebih praktis karena sistem konvensional sulit untuk mengatur keluaran tunggal dari berbagai masukan yang tidak saling terkait. Untuk menambahkan masukan baru ke dalam sistem fuzzy, hanya perlu menambahkan fungsi keanggotaan dan aturan yang terkait, sedangkan sistem konvensional memerlukan perhitungan ulang keseluruhan fungsi. Konsep dasar dari himpunan fuzzy adalah memperluas jangkauan fungsi karakteristiknya agar mencakup semua nilai bilangan real dalam rentang $[0,1]$. Ini berarti bahwa nilai keanggotaan mengindikasikan bahwa suatu item dalam domain pembicaraan tidak hanya memiliki nilai 0 atau 1, tetapi juga bisa memiliki nilai di antara keduanya. Dengan kata lain, kebenaran suatu item tidak hanya dibatasi menjadi benar atau salah, tetapi dapat memiliki berbagai nilai di antara keduanya[14].

2.2.4 Flutter

Flutter adalah sebuah open-source UI (User Interface) software development kit (SDK) yang dikembangkan oleh Google. Tujuan utama dari Flutter adalah memungkinkan pengembang untuk membuat aplikasi mobile dengan cepat dan mudah yang dapat berjalan di berbagai platform seperti iOS, Android, web, dan desktop dari satu codebase yang sama. Flutter menggunakan bahasa pemrograman Dart yang telah dikembangkan oleh Google, dan menyediakan berbagai komponen dan widget UI yang kaya untuk membangun antarmuka pengguna yang menarik dan responsif. Kelebihan lain dari Flutter adalah kinerjanya yang tinggi dan kemampuannya untuk membuat aplikasi dengan tampilan yang konsisten pada berbagai perangkat. Dengan kemudahan penggunaannya dan dukungan dari komunitas yang aktif, Flutter telah menjadi pilihan populer untuk pengembangan aplikasi mobile modern[15].

2.2.5 Bahasa Pemrograman Dart

Dart merupakan bahasa pemrograman yang diciptakan oleh Google sebagai opsi pengganti JavaScript. Dart menggunakan tipe data statis, yang mengharuskan variabel didefinisikan sebelum digunakan. Bahasa ini dapat berjalan pada berbagai perangkat, dengan menggunakan `dart2js` untuk web (mengubah Dart menjadi JavaScript agar bisa dijalankan di browser), `dart2aot` untuk desktop (mengubah Dart menjadi bahasa mesin), dan Flutter untuk aplikasi mobile. Dart memiliki sintaks yang mudah dipelajari karena serupa dengan bahasa pemrograman lain seperti JavaScript dan Java. Bahasa ini mendukung pemrograman asynchronous, yang memungkinkan program menjalankan kode lain sambil menunggu fungsi asynchronous selesai. Contoh penggunaan sehari-hari adalah ketika aplikasi mengambil data dari web dan menampilkan halaman loading. Selain itu, Dart juga mendukung pemrograman reaktif dengan menggunakan aliran data, yang memungkinkan data untuk diterima lebih dari satu kali, berbeda dengan asynchronous yang hanya menerima data sekali saja. Dart menjadi pilihan populer untuk pengembangan aplikasi karena kemampuannya yang fleksibel, mudah dipahami, dan berjalan pada berbagai platform[15].

2.2.6 MySql

MySQL merupakan sebuah server basis data yang cepat, mendukung banyak pengguna, dan menggunakan bahasa pemrograman SQL (Structured Query Language). MySQL tersedia dalam dua jenis lisensi, yaitu FreeSoftware dan Shareware. MySQL FreeSoftware berlisensi GNU/GPL (General Public License). Sebagai server basis data gratis, MySQL dapat digunakan secara bebas baik untuk keperluan pribadi maupun bisnis tanpa memerlukan pembayaran lisensi. MySQL awalnya dikembangkan oleh Michael Widenius sebagai sebuah database server dan program klien. Sebagai program database server, MySQL berfungsi sebagai server, dan program klien dapat mengakses dan berinteraksi dengan database tersebut. MySQL adalah sistem manajemen basis data relasional (RDBMS) yang menggunakan bahasa SQL untuk melakukan berbagai operasi pengambilan data. Berkat kinerjanya yang tinggi dan kemampuannya yang fleksibel,

MySQL telah menjadi salah satu server basis data yang terkenal dan sering digunakan dalam berbagai aplikasi dan proyek. MySQL memungkinkan pengguna untuk menyimpan, mengelola, dan mengakses data dengan mudah melalui bahasa SQL yang populer dan kuat[16].

2.2.7 Waterfall

Model Waterfall, juga dikenal sebagai "Linear Sequential Model" atau metode aliran sungai, adalah model klasik yang digunakan dalam pengembangan perangkat lunak secara terstruktur. Model ini merupakan salah satu model yang paling umum digunakan dalam dunia desain perangkat lunak dan pertama kali diperkenalkan oleh Winston Royce sekitar tahun 1970. Meskipun dianggap usang oleh beberapa individu, model Waterfall masih tetap menjadi salah satu model yang paling sering digunakan dalam pengembangan perangkat lunak. Pendekatan yang digunakan oleh Model Waterfall adalah berurutan dan sistematis, di mana setiap bagian proyek harus menunggu penyelesaian tahap sebelumnya sebelum dapat melanjutkan ke tahap berikutnya. Model ini dinamakan waterfall karena prosesnya berjalan secara terstruktur seperti air mengalir dari atas ke bawah, satu tahap setelah tahap sebelumnya. Meskipun model Waterfall memiliki keunggulan dalam jaminan kualitas karena setiap tahap diuji sebelum melanjutkan ke tahap berikutnya, namun model ini juga memiliki kelemahan seperti sulit menangani perubahan kebutuhan yang sering terjadi di tengah proyek. Meskipun demikian, model Waterfall masih relevan dan sering digunakan dalam proyek-proyek pengembangan perangkat lunak yang lebih sederhana dan terstruktur[17].

2.2.8 CodeIgniter

Codeigniter adalah sebuah kerangka kerja open source yang menerapkan pola MVC (Model View Controller), yang merupakan pendekatan modern dalam pembuatan aplikasi. Model dalam konsep MVC bertanggung jawab atas pengelolaan basis data, seperti MySQL atau Oracle RDBMS, dan melakukan operasi kueri untuk mengakses serta mengelola data dalam database. Codeigniter adalah kerangka kerja PHP yang dirancang untuk mempermudah pengembang web dalam pembuatan dan

pengembangan aplikasi berbasis web. Salah satu keunggulan utama Codeigniter adalah kinerjanya yang lebih cepat jika dibandingkan dengan kerangka kerja lain[17].

2.2.9 Rest API

Rest API (Representational State Transfer Application Programming Interface) adalah jenis layanan web API yang memanfaatkan permintaan HTTP untuk melaksanakan operasi CRUD (Create, Read, Update, Delete) pada data. Rest API adalah salah satu pendekatan arsitektur yang populer dalam sistem terdistribusi di internet, yang memungkinkan pengembang perangkat lunak berinteraksi dengan aplikasi web secara terstruktur. Rest API didesain agar simpel dan fleksibel, sehingga memungkinkan pengembang untuk membangun aplikasi yang berfungsi di beragam bahasa dan kerangka kerja pemrograman. Ada empat metode HTTP utama yang digunakan dalam Rest API, yakni GET (mengambil data dari server), POST (membuat data baru di server), PUT (memperbarui data yang sudah ada di server), dan DELETE (menghapus data dari server). Rest API mengikuti sejumlah batasan dan prinsip arsitektural, termasuk arsitektur client-server, statelessness, kemampuan caching, sistem berlapis, serta antarmuka yang seragam, yang bertujuan untuk menjamin keandalan dan kemudahan penggunaannya. Rest API menggunakan sumber daya (resource) dan pengenal sumber daya (URI) untuk merepresentasikan data. Setiap sumber daya diidentifikasi oleh URI unik, yang bisa digunakan pengembang untuk berinteraksi dengan sumber daya tersebut melalui metode HTTP. Rest API mempermudah manajemen data dengan konsistensi yang tinggi. Banyak bahasa pemrograman dan kerangka kerja mendukung Rest API, menjadikannya pilihan yang populer dalam pembangunan aplikasi web. Rest API sering digunakan oleh layanan web terkemuka seperti Twitter, Facebook, dan Google Maps, dan dianggap sebagai praktik terbaik dalam pengembangan aplikasi web yang dapat diandalkan, mudah digunakan, dan dapat diperbesar dalam skala aplikasinya[18].

2.2.10 Postman

Postman merupakan sebuah plugin berbasis aplikasi yang berfungsi sebagai REST Client, awalnya dibangun untuk browser Chrome, namun sekarang tersedia juga sebagai aplikasi desktop. Postman menyediakan platform GUI yang kuat untuk mempercepat dan menyederhanakan proses pengembangan API, mulai dari pembuatan API, pengujian, hingga dokumentasi dan berbagi API. Aplikasi ini dapat dijalankan pada sistem operasi Mac, Windows, dan Linux. Dengan menggunakan Postman, pengembang dapat menguji dan mengembangkan API dengan efisien, serta meningkatkan efektivitas dalam proses pengembangan dan pengujian aplikasi berbasis API[19].

2.2.11 Google Maps API

API Google Maps (Application Programming Interface) adalah suatu antarmuka aplikasi yang memberikan akses kepada Google Maps melalui bahasa pemrograman JavaScript. Hal ini memungkinkan penampilan peta Google Maps pada sebuah situs web yang sedang dalam pengembangan. Untuk menggunakan Google Maps API, pengetahuan tentang HTML dan JavaScript diperlukan, sementara peta itu sendiri disediakan oleh Google. Google Maps API merupakan layanan dari Google yang memungkinkan integrasi peta ke dalam situs web menggunakan JavaScript, menyediakan beragam fasilitas dan alat untuk mengubah peta serta menambahkan konten melalui berbagai layanan yang tersedia[20].

BAB III

METODOLOGI PENELITIAN

3.1 Analisis Kebutuhan

Analisis merupakan langkah krusial dalam pengembangan sistem yang bertujuan untuk memperoleh pemahaman mendalam tentang sistem yang akan dibangun. Di bawah ini adalah tahap analisis kebutuhan.

3.1.1 Metode Pengumpulan Data

Metode pengumpulan data penelitian ini menggunakan dua tahap dalam proses pengumpulan data, yaitu:

1. Studi Literatur

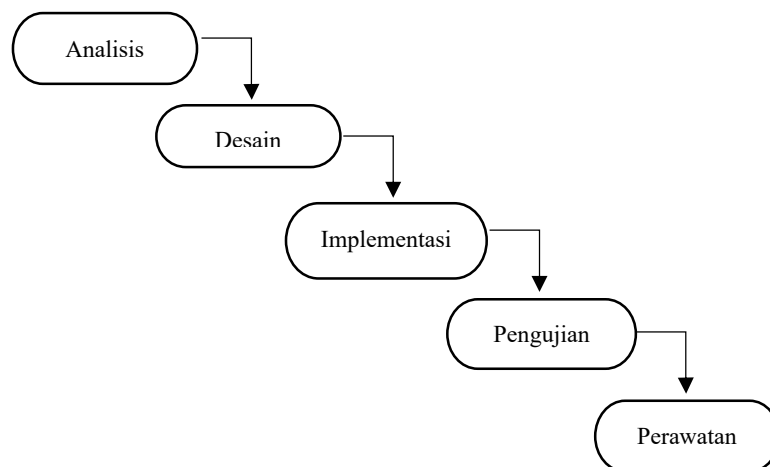
Studi Literatur dilakukan dengan mengumpulkan data dan menganalisis berbagai sumber tulisan, jurnal, dan materi bacaan yang relevan dengan sistem yang akan dikembangkan.

2. Observasi

Observasi yang dilakukan yaitu dengan cara melakukan peninjauan serta penelitian langsung terhadap permasalahan yang di angkat

3.1.2 Metode Pengembangan

Pada penelitian ini, digunakan metode pengembangan perangkat lunak Waterfall. Metode ini bersifat berurutan dan sistematis, dengan pendekatan yang sekuensial dalam pengembangan, yang membuatnya lebih teliti dan mendetail dalam pelaksanaannya. Proses pengembangan perangkat lunak Waterfall terdiri dari serangkaian tahapan sebagai berikut.:



Gambar 3.1 Metode Pengembangan Waterfall

1. Requirement Analysis (Analisis)

Tahap analisis bertujuan untuk mengidentifikasi kebutuhan dari pengguna dan organisasi, serta mengevaluasi situasi yang ada sebelum menerapkan sistem informasi yang baru.

2. System Design (Desain Sistem)

Tahap ini terjadi sebelum proses penulisan kode. Tujuan utamanya adalah memberikan gambaran yang jelas tentang pekerjaan yang harus dilakukan dan bagaimana tampilannya nantinya. Selain itu, tahap ini bertujuan untuk menetapkan spesifikasi rinci untuk semua komponen dalam sistem informasi, termasuk pengguna, perangkat keras, perangkat lunak, jaringan, dan data, serta produk-produk informasi yang akan sesuai dengan hasil analisis sebelumnya.

3. Implementation (Implementasi)

Tahap implementasi adalah langkah di mana perangkat keras dan perangkat lunak diperoleh atau dibangun (melalui proses penulisan kode program), dilakukan pengujian dan peralihan ke sistem yang baru dilaksanakan..

4. Testing (Pengujian),

Pada tahap ini, modul-modul yang telah dibuat digabungkan dan diuji guna memverifikasi kesesuaian perangkat lunak dengan desainnya serta untuk mengidentifikasi masalah atau peluang pengembangan tambahan dalam sistem informasi.

5. Operation & Maintenance (Perawatan)

Tahap perawatan berlangsung setelah sistem informasi telah aktif digunakan. Pada tahap ini, kegiatan meliputi pemantauan proses, evaluasi, dan jika diperlukan, perubahan atau perbaikan.

3.1.3 Kebutuhan Fungsional

Berikut adalah kebutuhan fungsional yang dibutuhkan untuk implementasi algoritma Fuzzy dalam menentukan jarak suatu lokasi.

Tabel 3.1 Kebutuhan Fungsional

Kode	Kebutuhan
F1	Sistem mampu mengimplementasikan algoritma Fuzzy dalam menentukan lokasi parkir yang ideal
F2	Sistem mampu menyimpan data yang dibuat
F3	Sistem mampu menampilkan pencarian lokasi yang di inginkan

3.1.4 Kebutuhan Non-Fungsional

Berikut adalah kebutuhan fungsional yang dibutuhkan untuk implementasi algoritma Fuzzy dalam menentukan jarak suatu lokasi.

Tabel 3.2 Kebutuhan Non-Fungsional

Kode	Parameter	Kebutuhan
NF1	Availability	Sistem dapat diakses kapan pun dibutuhkan
NF2	Reliability	Sistem dapat diandalkan dalam penyediaan lokasi
NF3	Ergonomy	Sistem dapat dimengerti dan digunakan dengan mudah oleh pengguna

3.1.5 Kebutuhan Perangkat

Tujuan dari analisis ini adalah untuk memahami persyaratan atau spesifikasi yang terkait dengan perangkat keras dan perangkat lunak yang akan digunakan selama proses pengembangan sistem.

Perangkat keras yang telah diidentifikasi meliputi:

Prosesor : 1 GHz 32-bit (x86) atau 64-bit (x64) atau lebih tangguh.

RAM : 4 GB atau lebih

Penyimpanan : 512 GB

Adapun perangkat lunak yang telah diidentifikasi terdiri dari:

Sistem Operasi : Windows 10

Bahasa Pemrograman : Dart & PHP

Framework : Flutter & Codeigniter 3

Database : SQLite

Editor Kode : Visual Studio Code

Perancangan Software : REST API

Perangkat Virtual : Android 10
Perancang Mockup : Draw.io
Perancang Model UML : Draw.io

3.1.6 Analisis Algoritma Fuzzy

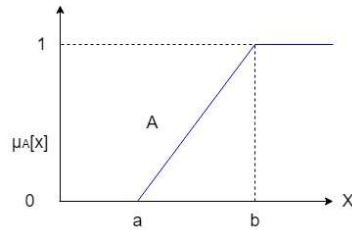
Algoritma Fuzzy digunakan untuk menentukan lokasi parkir ideal melalui sistem inferensi metode fuzzy, yang terdiri dari tiga tahapan utama:

1. Fuzzyfikasi adalah langkah di mana nilai tegas diubah menjadi variabel linguistik dengan menggunakan fungsi keanggotaan yang tersimpan dalam Basis Pengetahuan Fuzzy.
2. Inferensi adalah proses mengubah input menjadi output dengan mengikuti aturan-aturan (IF...THEN) yang ada dalam Basis Pengetahuan Fuzzy.
3. Defuzzifikasi adalah proses mengubah hasil dari setiap inferensi menjadi nilai konkret sebagai output melalui penggunaan fungsi keanggotaan yang telah ditetapkan.

Hasil dari proses ini menghasilkan informasi mengenai lokasi parkir yang dapat dianggap sebagai ideal, berdasarkan beberapa parameter yang telah diolah melalui langkah-langkah tersebut.

Mengenai Representasi Linear, dalam hal ini, pemetaan dari input ke derajat keanggotaannya digambarkan sebagai garis lurus. Terdapat dua varian umum dari representasi linear yang sering diterapkan:

1. Representasi linear naik adalah grafik yang menggambarkan peningkatan derajat keanggotaan himpunan fuzzy seiring perubahan nilai domain. Dalam jenis representasi ini, dimulai dari nilai domain yang memiliki derajat keanggotaan nol [0], grafik bergerak ke arah kanan menuju nilai domain yang memiliki derajat keanggotaan yang lebih tinggi. Secara visual, ini menggambarkan perubahan derajat keanggotaan dari nol menjadi semakin tinggi sepanjang domain nilai tertentu. Contoh representasi linier naik pada gambar.



Gambar 3.2 Representasi linier naik

(a.0) dan (b.1)

Fungsi keanggotaan adalah

$$\frac{y - y_1}{y_2 - y_1} = \frac{x - x_1}{x_2 - x_1}$$

:

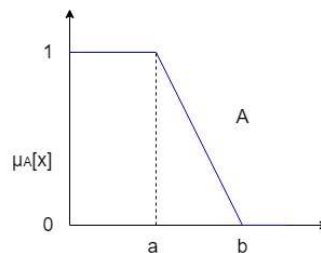
$\mu_A[x]$

$$\frac{y - 0}{1 - 0} = \frac{x - a}{b - a}$$

$$y = \frac{x - a}{b - a}$$

$$= \begin{cases} 0 & y \leq a \\ \frac{x - a}{b - a} & a < x < b \\ 1 & x \geq b \end{cases}$$

2. Representasi linear turun adalah jenis visualisasi yang menggambarkan garis lurus dimulai dari nilai domain dengan derajat keanggotaan tertinggi di sisi kiri, lalu menurun ke arah nilai domain yang memiliki derajat keanggotaan yang lebih rendah. Dalam representasi ini, secara visual, kita dapat melihat bagaimana derajat keanggotaan himpunan fuzzy berkurang saat nilai domain bergerak ke arah kanan.



Gambar 3.3 Representasi linier Turun

(a.1) dan (b.0)

Fungsi keanggotaan adalah :

$$\frac{y - y_1}{y_2 - y_1} = \frac{x - x_1}{x_2 - x_1}$$

$$\frac{y - 1}{0 - 1} = \frac{x - a}{b - a}$$

$$\mu_A[x] = \begin{cases} 1; & x \leq a \\ \frac{b - x}{b - a} & a < x < b \\ 0; & x \geq b \end{cases}$$

$$y - 1 = \frac{-(x - a)}{b - a}$$

$$y = \frac{-x + a + b - a}{b - a}$$

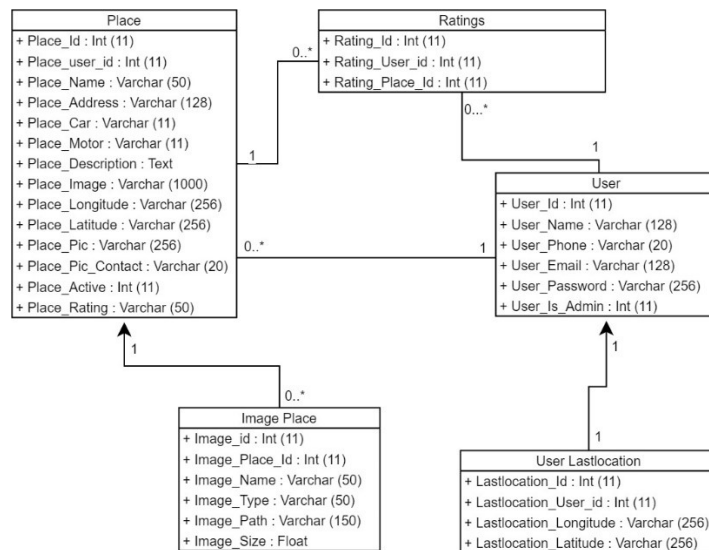
$$y = \frac{b - x}{b - a}$$

3.2 Perancangan

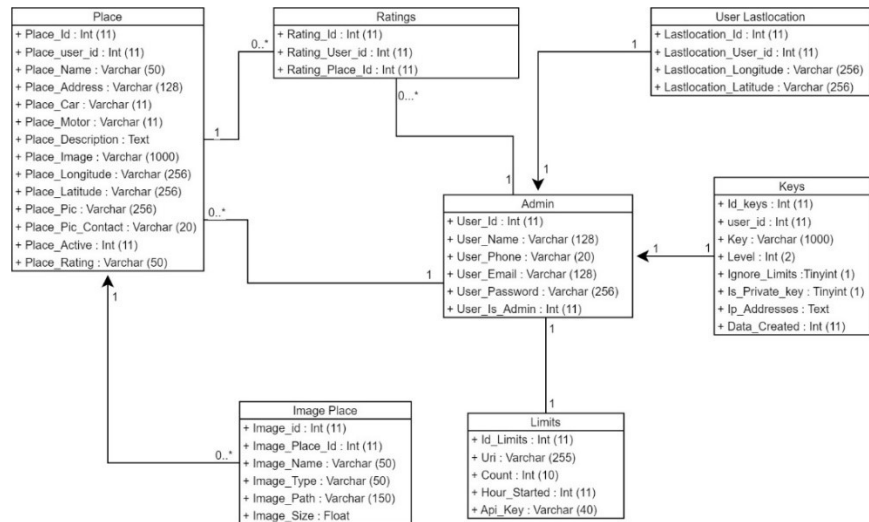
Perancangan dilakukan setelah proses desain telah dilakukan, Perancangan bertujuan untuk mempermudah dalam pembuatan sistem secara terstruktur dan menjadi tolak ukur untuk pembahasan selanjutnya.

3.2.1 Perancangan Class Diagram

Dalam Class Diagram, ini mencakup desain struktur basis data dan juga fungsi atau prosedur yang akan digunakan dalam penelitian ini. Terdapat dua class diagram yang terdiri dari class diagram user dan class diagram admin. Berikut adalah class diagram yang telah dibuat :



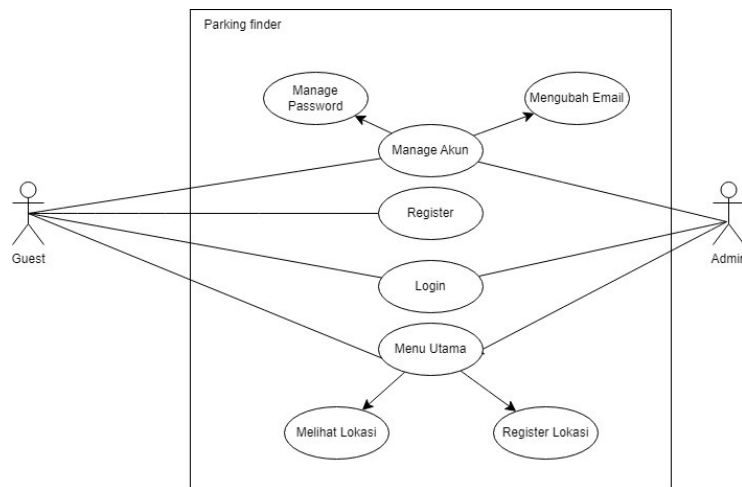
Gambar 3.4.a Class Diagram User



Gambar 3.4.b Class Diagram Admin

3.2.2 Perancangan Use Case Diagram

Pada bagian ini, akan diperlihatkan diagram Use Case yang mengilustrasikan hubungan antara pengguna dan aplikasi.



Gambar 3.5 Use Case Diagram

3.2.2.1 Definisi Use Case

Pada bagian ini, disajikan tabel definisi Use Case yang memberikan penjelasan tentang tugas aktor (user) serta respons yang diharapkan dari sistem dalam setiap interaksi. Berikut adalah tabel yang berisi definisi Use Case.

Tabel 3.3 Definisi Use Case

No	Use Case	Deskripsi
1	Manage Akun	Use Case ini user dapat Manage Akun, user dapat mengubah password dan email, serta mengubah deskripsi lokasi parkir yang dibuat user serta menghapus lokasi jika diperlukan
2	Register	Dalam Use Case ini user diharuskan membuat akun terlebih dahulu sebelum melakukan login
3	Login	Pada Use Case Login, user diperintahkan untuk masukkan email dan password yang telah dibuat sebelumnya
4	Menu Utama	Pada Use Case Menu Utama terdapat lokasi yang telah dibuat oleh user, setelah itu diurutkan menggunakan Algoritma Fuzzy melewati proses perhitungan fuzzyfikasi, inferensi, dan defuzzifikasi.

3.2.2.2 Skenario Use Case

Bagian ini memuat urutan skenario Use Case yang menggambarkan langkah-langkah interaksi antara aktor (pengguna) dan sistem. Skenario ini mencakup detail interaksi yang diharapkan saat aktor memanfaatkan sistem untuk mencapai tujuan tertentu. Informasi tambahan mengenai skenario tersebut akan diberikan dalam tabel-tabel berikut:

A. Skenario Use Case Manage Akun

Tabel 3.4 Skenario Use Case 1

Use Case 1	Memilih Topik
Deskripsi	Use Case ini memungkinkan pengguna untuk manage akun dengan cara mengubah username, email, password, dan deskripsi lokasi
Actor	User
Preconditions	User telah memiliki akun
Postcondition	User telah melakukan Register dan Login

Use Case 1	Memilih Topik
Basic Flow	User berada di halaman menu utama
Alternatif Flow	-

B. Skenario Use Case Register

Tabel 3.5 Skenario Use Case 2

Use Case 1	Memilih Topik
Deskripsi	Use Case ini mengharuskan user melakukan register terlebih dahulu untuk melakukan login
Actor	User
Preconditions	-
Postcondition	User belum memiliki akun
Basic Flow	User berada di halaman login
Alternatif Flow	-

C. Skenario Use Case Login

Tabel 3.6 Skenario Use Case 3

Use Case 1	Memilih Topik
Deskripsi	Use Case ini user melakukan login untuk dapat menggunakan program
Actor	User
Preconditions	User ingin menggunakan program and belum melakukan login
Postcondition	Pengguna sudah berhasil masuk (login) dan dapat melihat list lokasi
Basic Flow	User berada di halaman login
Alternatif Flow	-

D. Skenario Use Case Menu Utama

Tabel 3.7 Skenario Use Case 4

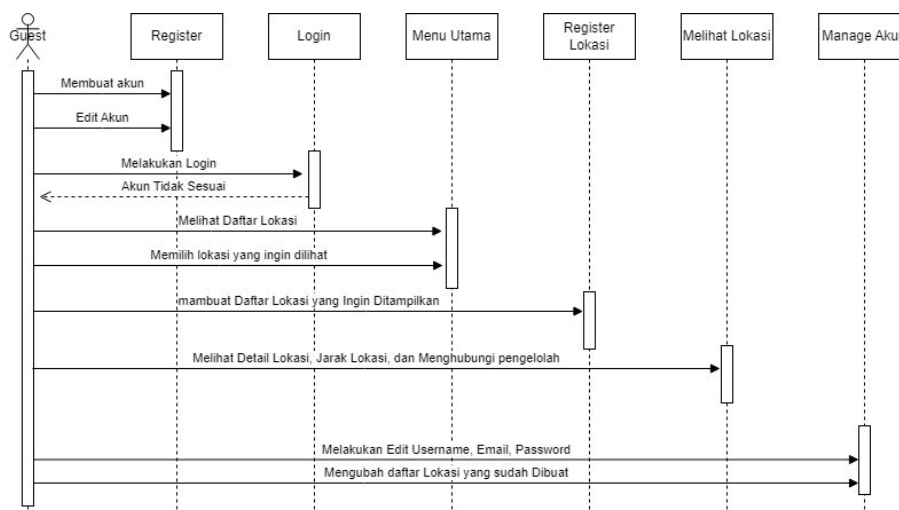
Use Case 1	Memilih Topik
Deskripsi	User melakukan fungsi dari aplikasi yakni melihat lokasi parkir yang ada dan mendaftarkan lokasi yang ingin dikelola
Actor	User & Admin

Use Case 1	Memilih Topik
Preconditions	User telah melakukan login dan ingin melihat lokasi parkir yang berada dalam aplikasi atau mendaftarkan lokasi yang ingin dikelola
Postcondition	User telah melihat tampilan daftar lokasi parkir
Basic Flow	User telah melakukan login
Alternatif Flow	-

Penggunaan tabel-tabel ini dimaksudkan untuk menyampaikan gambaran yang terstruktur tentang bagaimana interaksi antara pengguna dan sistem terjadi dalam berbagai skenario. Hal ini membantu para pemangku kepentingan dan pengembang sistem untuk memahami dengan lebih baik bagaimana sistem akan beroperasi dalam berbagai situasi dan bagaimana sistem dapat memenuhi kebutuhan pengguna dengan efektif dan efisien.

3.2.3 Perancangan Sequence Diagram

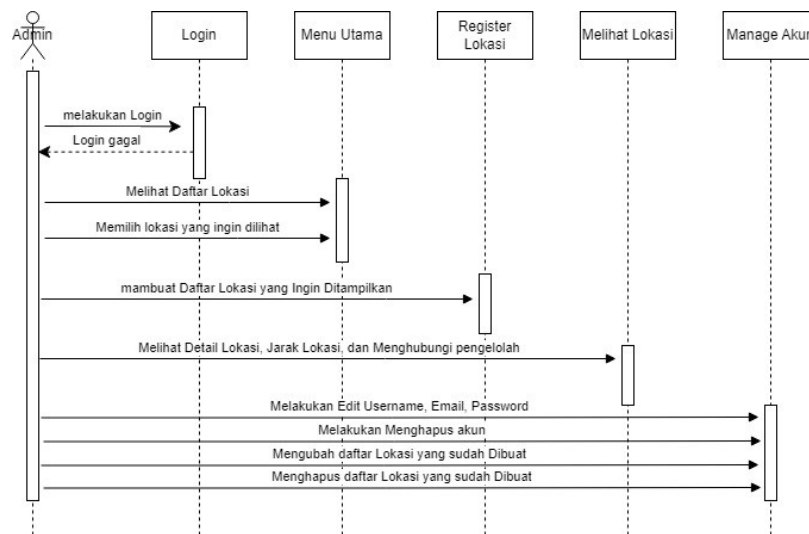
Melalui sequence diagram berikut, akan diilustrasikan interaksi antara pengguna dan sistem dengan proses yang mencakup pemilihan topik pembelajaran, memulai proses pembelajaran, dan juga melakukan evaluasi.



Gambar 3.6.a Sequence Diagram User

Gambar 3.6.a adalah alur bagaimana ilustrasi pengguna berinteraksi di dalam sistem. Ketika awal pengguna membuka aplikasi, akan diperintahkan login terlebih dahulu jika sudah memiliki akun, jika belum pengguna dapat

melakukan register. Setelah itu, pengguna dihadapkan dengan daftar lokasi parkir yang tersedia di menu utama. Pengguna dapat melakukan register untuk menambahkan tempat yang memungkinkan menjadi lahan parkir. Selain itu, pengguna dapat melihat lokasi yang tersedia pada menu utama dan melihat fasilitas yang tersedia, jika diperlukan pengguna dapat menanyakan langsung kepada pengelola parkir melalui whatsapp, dan dapat melacak jalur yang akan dilalui. Pengguna juga dapat mengelola akun yang sudah dibuat.

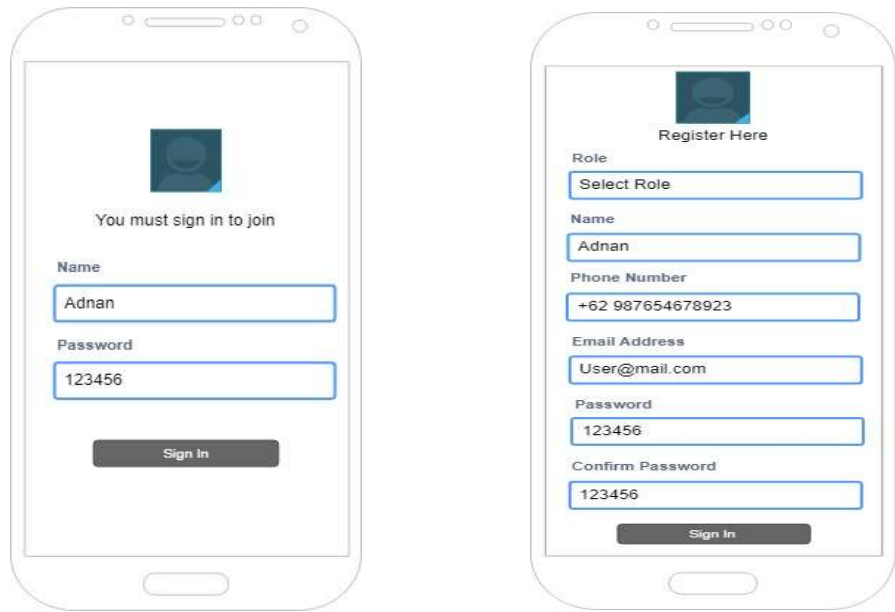


Gambar 3.6.b Sequence Diagram Admin

Gambar 3.6.b terdapat ilustrasi interaksi admin dengan sistem. Perbedaan antara admin dan pengguna yaitu tidak diperlukannya melakukan register karena sudah terdaftar di dalam sistem. Selain itu, admin juga dapat mengelola akun pengguna yang sudah terdaftar. Selibhnya admin dapat melakukan hal yang sama seperti pengguna.

3.2.4 Perancangan User Interface

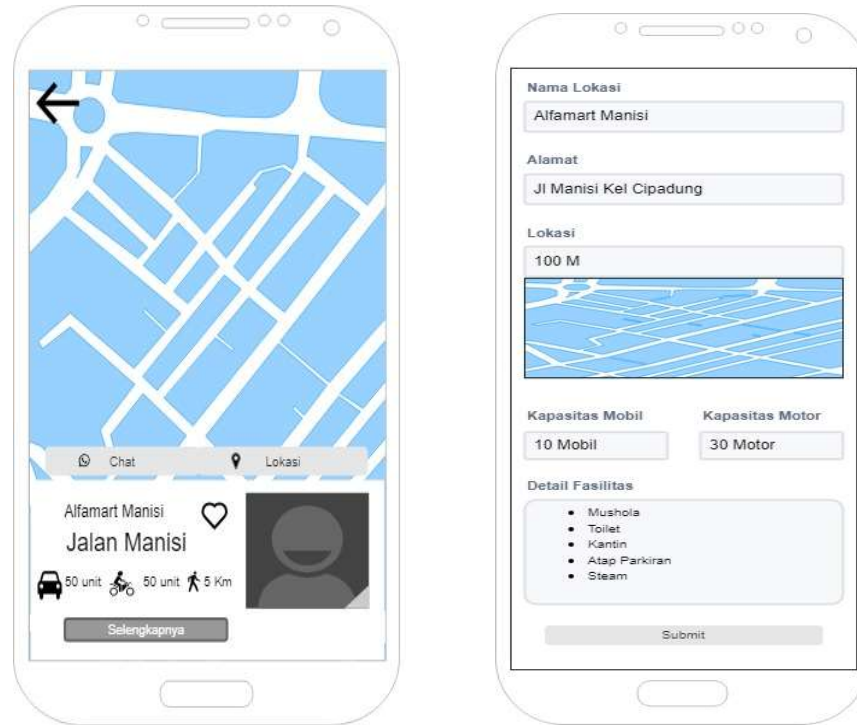
Perancangan ini merupakan gambaran dari aplikasi yang akan dibangun. Dengan adanya perancangan ini diharapkan proses pembangunan aplikasi menjadi terarah dan sesuai kebutuhan, terutama dalam pembangunan tampilan untuk pengguna.



Gambar 3.7 Tampilan Login



Gambar 3.8 Tampilan Menu Utama



Gambar 3.9 Tampilan Lokasi

3.3 Implementasi

Setelah melakukan tahap analisis dan perancangan aplikasi, selanjutnya melakukan tahap Implementasi. Ditahap ini menjelaskan cara-cara implementasi aplikasi sesuai perancangan yang telah dibangun sebelumnya, serta dilakukannya pengujian aplikasi yang sudah dibangun untuk mengetahui kekurangan dan kelayakan aplikasi tersebut.

3.3.1 Implementasi Basis Data

Dalam aplikasi ini terdapat 7 tabel. Berikut query pembuatan tabel-tabel dan kolom jenis SQLite yang ditulis dalam Bahasa Dart:

```
CREATE TABLE `keys` (
  `id` int(11) NOT NULL,
  `user_id` int(11) NOT NULL,
  `key` varchar(1000) NOT NULL,
  `level` int(2) NOT NULL,
  `ignore_limits` tinyint(1) NOT NULL DEFAULT 0,
  `is_private_key` tinyint(1) NOT NULL DEFAULT 0,
  `ip_addresses` text DEFAULT NULL,
  `date_created` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8
COLLATE=utf8_general_ci;

CREATE TABLE `limits` (
```

```

`id` int(11) NOT NULL,
`uri` varchar(255) NOT NULL,
`count` int(10) NOT NULL,
`hour_started` int(11) NOT NULL,
`api_key` varchar(40) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8
COLLATE=utf8_general_ci;

CREATE TABLE `tbl_image_place` (
`image_id` int(11) NOT NULL,
`image_place_id` int(11) NOT NULL,
`image_name` varchar(50) NOT NULL,
`image_type` varchar(50) NOT NULL,
`image_path` varchar(150) NOT NULL,
`image_size` float NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_general_ci;

CREATE TABLE `tbl_place` (
`place_id` int(11) NOT NULL,
`place_user_id` int(11) NOT NULL,
`place_name` varchar(50) NOT NULL,
`place_address` varchar(128) NOT NULL,
`place_car` varchar(11) NOT NULL,
`place_motor` varchar(11) NOT NULL,
`place_description` text NOT NULL,
`place_image` varchar(1000) NOT NULL,
`place_longitude` varchar(256) NOT NULL,
`place_latitude` varchar(256) NOT NULL,
`place_pic` varchar(256) NOT NULL,
`place_pic_contact` varchar(20) NOT NULL,
`place_active` int(11) NOT NULL,
`place_rating` varchar(50) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_general_ci;

CREATE TABLE `tbl_ratings` (
`rating_id` int(11) NOT NULL,
`rating_user_id` int(11) NOT NULL,
`rating_place_id` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_general_ci;

CREATE TABLE `tbl_users` (
`user_id` int(11) NOT NULL,
`user_name` varchar(128) NOT NULL,
`user_phone` varchar(20) NOT NULL,
`user_email` varchar(128) NOT NULL,
`user_password` varchar(256) NOT NULL,

```

```

`user_is_admin` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_general_ci;

CREATE TABLE `tbl_user_lastlocation` (
`lastlocation_id` int(11) NOT NULL,
`lastlocation_user_id` int(11) NOT NULL,
`lastlocation_longitude` varchar(256) NOT NULL,
`lastlocation_latitude` varchar(256) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_general_ci;

```

3.3.2 Implementasi Algoritma Fuzzy

Algoritma Fuzzy digunakan untuk menghitung kapasitas dan rating dengan menggunakan proses Fuzzyfikasi, di mana nilai yang pasti diubah menjadi variabel linguistik. Kemudian, proses Inferensi mengubah input menjadi output dengan mengikuti aturan-aturan (IF...THEN) yang ada di dalam Basis Pengetahuan Fuzzy. Terakhir, proses Defuzzifikasi mengubah hasil dari setiap inferensi menjadi output dengan nilai pasti sesuai dengan fungsi keanggotaan yang telah ditentukan. Berikut adalah tahapan implementasinya.

1. Proses Fuzzifikasi Jarak

```

Program fuzzifikasi_jarak
Deklarasi
nilaiJarak = float
jarakRendah = float
jarakTinggi = float

Algoritma
Input (jarakRendah, jarakTinggi)
If -> nilaiJarak >= jarakTinggi then
nilaiJarakRendah = 0
Else if -> nilaiJarak <= jarakRendah then
nilaiJarakRendah = 1
Else -> nilaiJarakRendah = (jarakTinggi -
nilaiJarak) / (jarakTinggi - jarakRendah)

If -> nilaiJarak >= jarakTinggi then
nilaiJarakRendah = 1
Else if -> nilaiJarak <= jarakRendah then
nilaiJarakRendah = 0
Else -> nilaiJarakRendah = (nilaiJarak -
jarakRendah) / (jarakTinggi - jarakRendah)
Output (jarakRendah, jarakTinggi)

```


Jarak memiliki dua himpunan fuzzy yaitu jarak rendah, jarak tinggi. Pada masing-masing himpunan fuzzy memiliki keanggotaan berbeda tergantung pada nilai input yang diberikan. Himpunan jarak rendah memiliki rentang nilai 0-100, himpunan jarak tinggi memiliki rentang nilai 100-1000.

2. Proses Fuzzifikasi kapasitas

```

Program fuzzifikasi_kapasitas
Deklarasi
nilaiKapasitas = float
kapasitasRendah = float
kapasitasTinggi = float
Algoritma
Input (nilaiKapasitas, kapasitasRendah,
kapasitasTinggi)
If -> nilaiKapasitas >= kapasitasTinggi then
nilaiKapasitasRendah = 0
Else if -> nilaiKapasitas <= kapasitasRendah
then nilaiKapasitasRendah = 1
Else -> nilaiKapasitasRendah = (kapasitasTinggi
- nilaiKapasitas) / (KapasitasTinggi -
kapasitasRendah)

If -> nilaiKapasitas >= kapasitasTinggi then
nilaiKapasitasRendah = 1
Else if -> nilaiKapasitas <= kapasitasRendah
then nilaiKapasitasRendah = 0
Else -> nilaiKapasitasRendah = (nilaiKapasitas -
kapasitasRendah) / (kapasitasTinggi -
kapasitasRendah)
Output (kapasitasRendah, kapasitasTinggi)

```

Kapasitas memiliki dua himpunan fuzzy yaitu kapasitas rendah, kapasitas tinggi. Pada masing-masing himpunan fuzzy memiliki keanggotaan berbeda tergantung pada nilai input yang diberikan. Himpunan kapasitas rendah memiliki rentang nilai 0-40, himpunan kapasitas tinggi memiliki rentang nilai 40-100.

3. Proses Fuzzyfikasi Rating

```

Program fuzzyfikasi_rating
Deklarasi
nilaiRating = float
retingRendah = float
retingTinggi = float

Algoritma

```

```

Input (nilaiRating, ratingRendah, ratingTinggi)
If -> nilaiRating >= RatingTinggi then
nilaiRatingRendah = 0
Else -> if nilaiRating <= ratingRendah then
nilaiRatingRendah = 1
Else -> nilaiRatingRendah = (ratingTinggi -
nilaiRating) / (ratingTinggi - ratingRendah)

If -> nilaiRating >= ratingTinggi then
nilaiRatingRendah = 1
Else -> if nilaiRating <= ratingRendah then
nilaiRatingRendah = 0

Else -> nilaiRatingRendah = (nilaiRating -
ratingRendah) / (ratingTinggi - ratingRendah)
Output (ratingRendah, ratingTinggi)

```

Rating memiliki dua himpunan fuzzy yaitu Rating rendah, Rating tinggi. Pada masing-masing himpunan fuzzy memiliki keanggotaan berbeda tergantung pada nilai input yang diberikan. Himpunan Rating rendah memiliki rentang nilai 0-50, himpunan Rating tinggi memiliki rentang nilai 50-100.

4. Proses Inferensiasi

Fungsi Inferensiasi yang digunakan pada fuzzy Tsukamoto adalah fungsi MIN. Proses perhitungan nilai dari α -predicate dilakukan untuk memperoleh nilai keluaran (Z) menggunakan fungsi MIN

Program Inferensiasi

Deklarasi

```

JarakRendah : float
JarakTinggi : float
KapasitasRendah : float
KapasitasTinggi : float
RatingRendah : float
RatingTinggi : float

```

Algoritma

```

Input (JarakRendah, JarakTinggi,
KapasitasRendah, KapasitasTinggi, RatingRendah,
RatingTinggi)
A1 = min(nilaiJarakTinggi, nilaiKapasitasRendah,
nilaiRatingRendah)
R1 = nilaiIdeal - (nilaiIdeal -
nilaiKurangIdeal) * A1

```

```

A2 = min(nilaiJarakTinggi, nilaiKapasitasRendah,
nilaiRatingTinggi)
R2 = nilaiIdeal - (nilaiIdeal -
nilaiKurangIdeal) * A2
A3 = min(nilaiJarakTinggi, nilaiKapasitasTinggi,
nilaiRatingRendah)
R3 = nilaiIdeal - (nilaiIdeal -
nilaiKurangIdeal) * A3
A4 = min(nilaiJarakTinggi, nilaiKapasitasTinggi,
nilaiRatingTinggi)
R4 = nilaiIdeal - (nilaiIdeal -
nilaiKurangIdeal) * A4
A5 = min(nilaiJarakRendah, nilaiKapasitasRendah,
nilaiRatingRendah)
R5 = nilaiIdeal - (nilaiIdeal -
nilaiKurangIdeal) * A5
A6 = min(nilaiJarakRendah, nilaiKapasitasRendah,
nilaiRatingTinggi)
R6 = nilaiIdeal - (nilaiIdeal -
nilaiKurangIdeal) * A6
A7 = min(nilaiJarakRendah, nilaiKapasitasTinggi,
nilaiRatingRendah)
R7 = nilaiIdeal - (nilaiIdeal -
nilaiKurangIdeal) * A7
A8 = min(nilaiJarakRendah, nilaiKapasitasTinggi,
nilaiRatingTinggi)
R8 = nilaiIdeal - (nilaiIdeal -
nilaiKurangIdeal) * A8
Output (nilaiInferensiasi)

```

5. Defuzzifikasi Kapasitas, Rating dan Jarak

```

Program -> defuzzy {Algoritma untuk proses
defuzzy}
Deklarasi
Minimum [] : float
Jumlah A : float {hasil dari penjumlahan semua a
predikat}
jumlah B : float {hasil kali a predikat dengan z}
hasilDefuzzy : float

Algoritma
Input (minimum 8)
Total A -> (a1 * r1) + (a2 * r2) + (a3 * r3) +
(a4 * r4) + (a5 * r5) + (a6 * r6) + (a7 * r7) +
(a8 * r8)
Total B -> a1 + a2 + a3 + a4 + a5 + a6 + a7 + a8

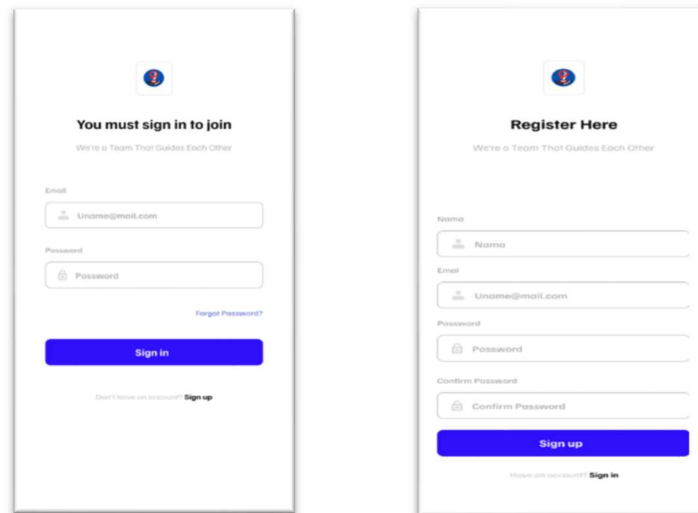
hasilDefuzzy -> Total A / total B
Output (hasilDefuzzy)

```

3.3.3 Implementasi User Interface

Implementasi user interface adalah fase di mana aplikasi yang telah dirancang dibuat menjadi kenyataan. Aplikasi ini disusun sesuai dengan desain yang telah di preparasi sebelumnya. Di bawah ini adalah implementasi antarmuka pengguna dalam penelitian ini.:

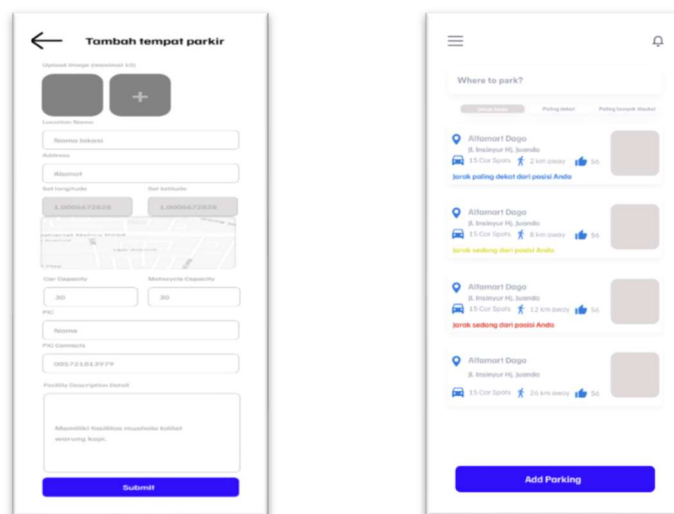
1. Tampilan Login dan Register



Gambar 3.10 Login dan Register User

Tampilan ini Pengguna akan diarahkan untuk melakukan login, jika belum memiliki akun, pengguna dapat melakukan register dan setelah itu bisa melakukan login.

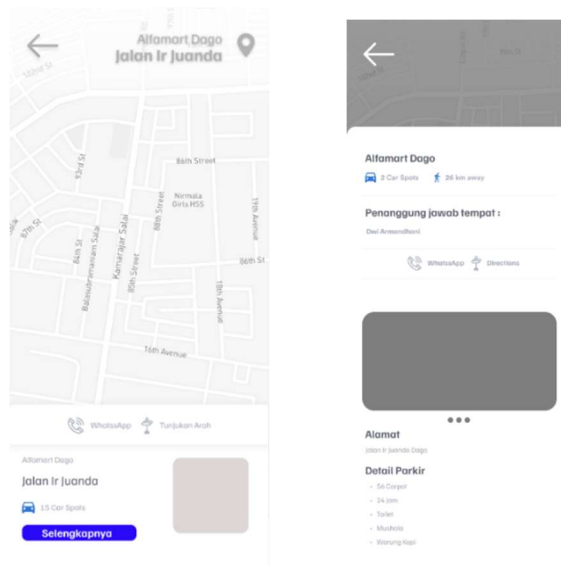
2. Tampilan Menu Utama



Gambar 3.11 Tampilan Menu Utama

Tampilan ini merupakan daftar lokasi yang ada diaplikasi untuk ditampilkan kepada pengguna. Di mana pengguna dapat menjadi pengelola dengan cara mendaftarkan lokasi yang ingin digunakan untuk lokasi parkir. Nantinya lokasi parkir diurutkan berdasarkan lokasi yang disarankan oleh aplikasi menggunakan Algoritma Fuzzy dan dapat juga diurutkan berdasarkan jarak terdekat, jumlah kapasitas dan banyak disukai.

3. Tampilan Deskripsi Lokasi.

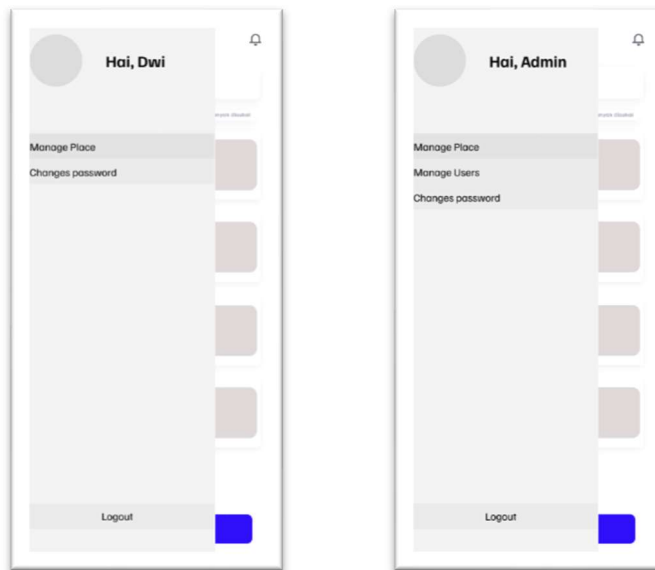


Gambar 3.12 Tampilan Deskripsi Lokasi

Tampilan ini memperlihatkan deskripsi yang tersedia pada lokasi, baik itu fasilitas parkir, alamat tempat parkir, pengelola parkir dan juga nomor telepon pengelola parkir

4. Tampilan Side Bar

Tampilan ini disediakan untuk user dan admin dapat dengan mudah untuk mengubah email, password, dan deskripsi lokasi yang didaftarkan, Jika diperlukan. Dan untuk melakukan logout.



Gambar 3.13 Side Bar

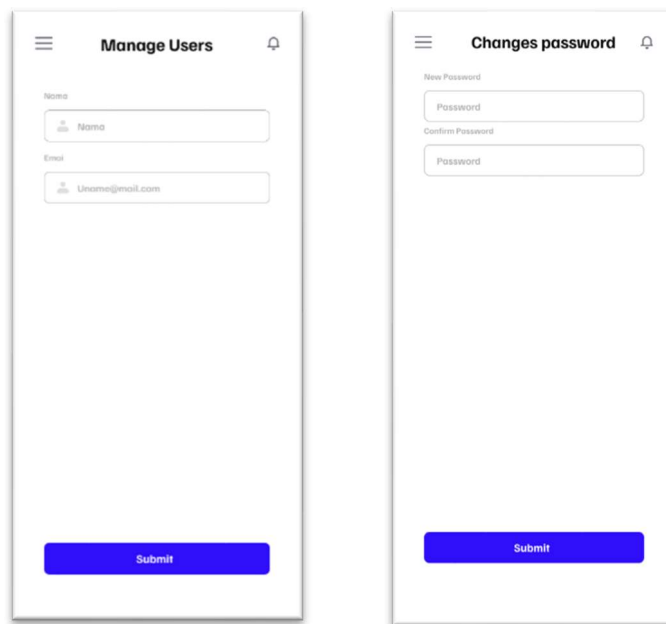
5. Manage Akun

Terdapat menu manage akun. Di mana admin dapat mengelola akun yang pengguna daftarkan dan admin dapat menghapus akun atau mengubah akun.



Gambar 3.14 Manage Admin

Pengguna dapat mengelola akun yang sudah didaftarkan, baik untuk mengelola username, email, dan password.



Gambar 3.15 Manage Pengguna

3.3.4 Pengujian Sistem

Dalam penelitian ini, dilakukan pengujian blackbox terhadap sejumlah tampilan aplikasi. Tujuannya adalah untuk mengonfirmasi bahwa aplikasi berfungsi sesuai dengan susunan tampilan yang telah direncanakan.

Hasil dari pengujian blackbox ini memberikan informasi berharga tentang kualitas aplikasi dan apakah fungsionalitasnya sesuai dengan ekspektasi. Jika terdapat kasus uji yang tidak berhasil, hal ini dapat menunjukkan adanya bug atau masalah dalam implementasi aplikasi yang perlu diperbaiki sebelum diluncurkan secara penuh kepada pengguna. Dengan demikian, pengujian blackbox ini berkontribusi dalam meningkatkan kualitas dan kesesuaian fungsionalitas aplikasi dengan kebutuhan pengguna. Berikut adalah tabel-tabel hasil pengujian *blackbox*.

1. Halaman Login

Tabel 3.5 Halaman Login

ID	Test Case	Hasil yang di harapkan	Respon Sistem	Status pengujian
1	Membuka aplikasi	Menampilkan halaman Login	Halaman Login	Berhasil

ID	Test Case	Hasil yang di harapkan	Respon Sistem	Status pengujian
			dapat ditampilkan	
2	Memasukan email dan password	Email dan password dapat tampil pada layar	Dapat mengisi email dan password	Berhasil
3	Pilih Daftar disini	Menampilkan halaman register	Halaman Register dapat ditampilkan	Berhasil
4	Pilih Login	Dapat masuk ke dalam halaman menu utama	Menu utama dapat ditampilkan	Berhasil

2. Halaman Register

Tabel 3.6 Halaman Register

ID	Test Case	Hasil yang di harapkan	Respon Sistem	Status pengujian
1	Pilih Login di sini	Dapat menampilkan halaman Login	Halaman Login dapat ditampilkan	Berhasil
2	Memasukan data yang diperlukan pada register	Data dapat ditampilkan pada halaman register	Dapat mengisi data register	Berhasil
3	Pilih Register	Register berhasil dan dapat	Data tersimpan pada Database dan dapat	Berhasil

ID	Test Case	Hasil yang di harapkan	Respon Sistem	Status pengujian
		melakukan login	melakukan Login	

3. Halaman Menu Utama

Tabel 3.7 Halaman Menu Utama

ID	Test Case	Hasil yang di harapkan	Respon Sistem	Status pengujian
1	Pilih lokasi yang ingin di lihat	Masuk pada tampilan deskripsi lokasi	Deskripsi lokasi dapat ditampilkan	Berhasil
2	Pilih add parking	Pindah ke halaman register lokasi	Dapat menampilkan halaman register lokasi	Berhasil
3	Memilih tampilan lokasi berdasarkan pilihan yang disediakan	Daftar lokasi dapat menyesuaikan sesuai yang diperintahkan user, baik dari jarak terdekat, banyak disukai, dan lokasi yang disarankan oleh Algoritma Fuzzy	Lokasi dapat ditampilkan berdasarkan perintah yang dilakukan oleh user	Berhasil
4	Pilih menu side bar	Dapat menampilkan menu manage akun	Side bar dapat ditampilkan	Berhasil

4. Halaman Manage Akun

Tabel 3.8 Halaman Manage Akun

ID	Test Case	Hasil yang di harapkan	Respon Sistem	Status pengujian
1	Pilih manage place	Tampilan pindah ke tampilan manage place	Dapat menampilkan manage place	Berhasil
2	Pilih changes Password	Tampilan pindah ke tampilan changes Password	Dapat menampilkan changes password	Berhasil
3	Pilih manage user	Tampilan pindah ke tampilan manage user	Dapat menampilkan manage user	Berhasil
4	Pilih Logout	Akun akan keluar dan Kembali ke tampilan login	Dapat menampilkan login	Berhasil

Tabel 3.9 Halaman Register Lokasi

ID	Test Case	Hasil yang di harapkan	Respon Sistem	Status pengujian
1	Pilih Submit	Data akan tersimpan pada data base dan ditampilkan pada menu utama	Data tersimpan pada Database dan dapat menampilkan menu utama	Berhasil
2	Memasukan data yang dibutuhkan	Data akan tampil pada halaman	Dapat mengisi data register	Berhasil
3	Pilih tanda panah	Tampilan akan Kembali ke halaman sebelumnya	Dapat Kembali kehalaman sebelumnya	Berhasil

BAB IV HASIL DAN PEMBAHASAN

4.1 Implementasi Algoritma Fuzzy

Pengolahan data dilakukan menggunakan algoritma Fuzzy Tsukamoto yang selanjutnya akan diimplementasikan dengan desain program menggunakan Visual Studio Code. Penelitian ini menggunakan data lokasi parkir yang ada di Masjid Raya Al-jabbar seperti terlihat pada tabel 4.1 di bawah ini:

Tabel 4.1 Data Penelitian

Jarak	Kapasitas	Disukai	Nilai Ideal
100 Meter	10 kendaraan	50 disukai	50 nilai ideal
1000 Meter	40 kendaraan	100 disukai	100 nilai ideal

Perhitungan menggunakan algoritma Fuzzy Tsukamoto dalam menentukan lokasi parkir dengan jarak 450 meter, jumlah kapasitas 300 kendaraan dan disukai 76 disukai adalah sebagai berikut:

1. Fuzzyfikasi

Terdapat 4 variabel yang digunakan dan dimodelkan menggunakan fungsi keanggotaan representasi linear. Variabel tersebut adalah:

- a. Jarak, fungsi keanggotaan yang digunakan pada variabel Jarak adalah himpunan Dekat dan Jauh. Himpunannya adalah:

$$\mu_X[Dekat] = \begin{cases} 1 & x \leq 100 \\ \frac{1000 - x}{1000 - 100} & 100 \leq x \leq 1000 \\ 0 & x \geq 1000 \end{cases}$$

$$\mu_X[Jauh] = \begin{cases} 0 & x \leq 100 \\ \frac{x - 100}{1000 - 100} & 100 \leq x \leq 1000 \\ 1 & x \geq 1000 \end{cases}$$

Nilai keanggotaan dari himpunan di atas adalah:

$$\begin{aligned} \mu_x [Dekat] (450) &= (1000-450) /900 \\ &= 0.61 \end{aligned}$$

$$\begin{aligned} \mu_x [Jauh] (450) &= (450-100) /900 \\ &= 0.38 \end{aligned}$$

- b. Kapasitas, fungsi keanggotaan yang digunakan pada variabel Kapasitas adalah himpunan Sedikit, Banyak. Himpunannya adalah:

$$\mu X[\text{Sedikit}] = \begin{cases} 1 & x \leq 10 \\ \frac{40-x}{40-10} & 10 \leq x \leq 40 \\ 0 & x \geq 40 \end{cases}$$

$$\mu X[\text{Banyak}] = \begin{cases} 0 & x \leq 10 \\ \frac{x-10}{40-10} & 10 \leq x \leq 40 \\ 1 & x \geq 40 \end{cases}$$

Nilai keanggotaan dari himpunan di atas adalah:

$$\mu x \text{ Sedikit} (300) = 0$$

$$\mu x \text{ Banyak} (300) = 1$$

- c. Disukai, Fungsi keanggotaan yang digunakan dalam variabel disukai adalah himpunan Sedikit, Banyak. Himpunan tersebut adalah:

$$\mu X[\text{Sedikit}] = \begin{cases} 1 & x \leq 50 \\ \frac{100-x}{100-50} & 50 \leq x \leq 100 \\ 0 & x \geq 100 \end{cases}$$

$$\mu X[\text{Banyak}] = \begin{cases} 0 & x \leq 50 \\ \frac{x-50}{100-50} & 50 \leq x \leq 100 \\ 1 & x \geq 100 \end{cases}$$

Nilai keanggotaan dari himpunan di atas adalah:

$$\mu x [\text{Sedikit}] (76) = (100-76) / 50$$

$$= 0.48$$

$$\mu x [\text{Banyak}] (76) = (76-50) / 50$$

$$= 0.52$$

- d. Nilai Ideal, Fungsi keanggotaan yang digunakan dalam variabel Ideal adalah himpunan Sedikit, Banyak. Himpunan tersebut adalah:

$$\mu X[\text{Kurang Ideal}] = \begin{cases} 1 & x \leq 50 \\ \frac{100-x}{100-50} & 50 \leq x \leq 100 \\ 0 & x \geq 100 \end{cases}$$

$$\mu X[\text{Ideal}] = \begin{cases} 0 & x \leq 50 \\ \frac{x-50}{100-50} & 50 \leq x \leq 100 \\ 1 & x \geq 100 \end{cases}$$

2. Pembentukan Aturan Fuzzy

Dalam penelitian ini digunakan 4 aturan fuzzy yang dapat dilihat pada tabel 4.2 di bawah ini:

Tabel 4.2 Aturan Fuzzy

Rule	Jarak	Kapasitas	Disukai	Nilai Ideal	α -predicate = Min
R1	Jauh	Sedikit	Sedikit	Kurang Ideal	(0.38 ; 0 ; 0.48)
R2	Jauh	Sedikit	Banyak	Kurang Ideal	(0.38 ; 0 ; 0.52)
R3	Jauh	Banyak	Sedikit	Kurang Ideal	(0.38 ; 1 ; 0.48)
R4	Jauh	Banyak	Banyak	Ideal	(0.38 ; 1 ; 0.52)
R5	Dekat	Sedikit	Sedikit	Kurang Ideal	(0.61 ; 0 ; 0.48)
R6	Dekat	Sedikit	Banyak	Ideal	(0.61 ; 0 ; 0.52)
R7	Dekat	Banyak	Sedikit	Kurang Ideal	(0.61 ; 1 ; 0.48)
R8	Dekat	Banyak	Banyak	Ideal	(0.61 ; 1 ; 0.52)

3. Proses Inferensi Berdasarkan α --Predicate

Pada tahap ini, proses perhitungan nilai dari α --predicate dilakukan untuk memperoleh nilai keluaran (Z) menggunakan fungsi MIN. Tahapan-tahapannya adalah sebagai berikut

Rule 1

$$\alpha\text{-predicate} = \text{Min} (0.38 ; 0 ; 0.48)$$

$$5. \quad Z = 0$$

$$0 = \frac{100 - x}{100 - 50} = \frac{100 - x}{50}$$

$$100 - x = 50 * 0$$

$$100 - x = 0$$

$$x = 100 - 0$$

$$x = 100$$

Rule 2

$$\alpha\text{-predicate} = \text{Min} (0.38 ; 0 ; 0.52)$$

$$Z = 0$$

$$0 = \frac{100 - x}{100 - 50} = \frac{100 - x}{50}$$

$$100 - x = 50 * 0$$

$$100 - x = 0$$

$$x = 100 - 0$$

$$6. \quad x = 100$$

Rule 3

$$\alpha\text{-predicate} = \text{Min} (0.38 ; 1 ; 0.48)$$

$$7. \quad Z = 0.38$$

$$0.38 = \frac{100 - x}{100 - 50} = \frac{100 - x}{50}$$

Rule 4

$$\alpha\text{-predicate} = \text{Min} (0.38 ; 1 ; 0.52)$$

$$Z = 0.38$$

$$0.38 = \frac{x - 50}{100 - 50} = \frac{x - 50}{50}$$

$$\begin{aligned}
100 - x &= 50 * 0.38 \\
100 - x &= 19 \\
x &= 100 - 19 \\
x &= 81
\end{aligned}$$

$$\begin{aligned}
0.38 \times 50 &= x - 50 \\
19 &= x - 50 \\
x &= 19 + 50 \\
x &= 69
\end{aligned}$$

Rule 5

$$\alpha\text{-predicate} = \text{Min} (0.61 ; 0 ; 0.48)$$

$$\begin{aligned}
8. \quad Z &= 0 \\
0 &= \frac{100 - x}{100 - 50} = \frac{100 - x}{50} \\
100 - x &= 50 * 0 \\
100 - x &= 0 \\
x &= 100 - 0 \\
9. \quad x &= 100
\end{aligned}$$

Rule 6

$$\alpha\text{-predicate} = \text{Min} (0.61 ; 0 ; 0.52)$$

$$\begin{aligned}
Z &= 0 \\
0 &= \frac{x - 50}{100 - 50} = \frac{x - 50}{50} \\
0 \times 50 &= x - 50 \\
0 &= x - 50 \\
x &= 0 + 50 \\
x &= 50 \\
10.
\end{aligned}$$

Rule 7

$$\alpha\text{-predicate} = \text{Min} (0.61 ; 1 ; 0.48)$$

$$\begin{aligned}
11. \quad Z &= 0.48 \\
0.48 &= \frac{100 - x}{100 - 50} = \frac{100 - x}{50} \\
100 - x &= 0.48 \times 50 \\
100 - x &= 24 \\
x &= 100 - 24 \\
12. \quad x &= \underline{76}
\end{aligned}$$

Rule 8

$$\alpha\text{-predicate} = \text{Min} (0.61 ; 1 ; 0.52)$$

$$\begin{aligned}
Z &= 0.52 \\
0.52 &= \frac{x - 50}{100 - 50} = \frac{x - 50}{50} \\
0.52 \times 50 &= x - 50 \\
26 &= x - 50 \\
x &= 26 + 50 \\
x &= 76 \\
13.
\end{aligned}$$

4. Defuzzifikasi

Proses defuzzifikasi dari perhitungan sebelumnya, yaitu

$$Z = \frac{(0 * 100) + (0 * 100) + (0.38 * 81) + (0.38 * 69) + (0 * 100) + (0 * 50) + (0.48 * 76) + (0.52 * 76)}{0 + 0 + 0.38 + 0.38 + 0 + 0 + 0.48 + 0.52}$$

$$Z = \frac{(0) + (0) + (30.78) + (26.22) + (0) + (0) + (36.48) + (39,52)}{1.76}$$

$$Z = \frac{133}{1.76}$$

$$Z = 75,56$$

Dari hasil proses defuzzifikasi di atas, dapat disimpulkan bahwa untuk menentukan suatu Lokasi dapat dikatakan Ideal, jika diketahui bahwa jarak 450 meter, Kapasitas adalah 300 spot dan disukai adalah 76 disukai, maka nilai ideal nya adalah 75,56.

4.2 Kinerja Algoritma Fuzzy

Pada bagian ini, disajikan temuan dari pengujian blackbox terhadap implementasi algoritma fuzzy dalam menentukan lokasi parkir. Hasil pengujian ditampilkan dalam tabel yang mencakup perhitungan fuzzifikasi, inferensi, dan defuzzifikasi dari implementasi algoritma. Berikut adalah tabel-tabel yang memperlihatkan hasil implementasi algoritma Fuzzy:

1. Proses Fuzzifikasi

Hasil proses Fuzzyfikasi yang didapat pada pengujian blackbox. Diambil 5 sampel dari lokasi yang tersedia disekitar Masjid Raya Al Jabbar. Adapun hasilnya sebagai berikut:

Tabel 4.3 Hasil Fuzzifikasi

No	Lokasi	Jarak Terdekat	Jarak terjauh	Kapasitas Rendah	Kapasitas Tinggi	Disukai Rendah	Disukai Tinggi
1	Parkir HMS 45	0.61	0.38	0	1	0.48	0.52
2	Rest Area Parkir Aljabar	0.48	0.52	0	1	0.96	0.04
3	Sekolah Kuttab Al Fatih Bandung	0.71	0.28	0	1	1	0
4	Parkir Aljabar Warga Rw 03	0.67	0.32	0	1	0.76	0.24
5	Parkir Motor	0.79	0.2	1	0	1	0

	Posyandu Mawar Rw 04								
--	----------------------------	--	--	--	--	--	--	--	--

2. Proses Inferensi

Setelah didapat hasil fuzzyfikasi. Tahap selanjutnya proses inferensi. Di mana tahapan ini melewati rule dalam pembentukan aturan fuzzy yang telah dibuat sebelumnya.

Tabel 4.4 Hasil Inferensi

No	Lokasi	Nilai R1	Nilai R2	Nilai R3	Nilai R4	Nilai R5	Nilai R6	Nilai R7	Nilai R8
1	Parkir HMS 45	100	100	80.5	69.4	100	50	76	76
2	Rest Area Parkir Aljabar	100	100	71.66	52	100	50	78.33	52
3	Sekolah Kuttab Al Fatih Bandung	100	100	85.55	50	100	50	64.44	50
4	Parkir Aljabar Warga Rw 03	100	100	83.88	62	100	50	66.11	62
5	Parkir Motor Posyandu Mawar Rw 04	90	100	100	50	60	50	100	50

3. Proses Defuzzifikasi

Setelah proses fuzzyfikasi dan inferensi selesai. Proses selanjutnya yaitu defuzzyfikasi.

Tabel 4.5 Hasil Defuzzifikasi

No	Lokasi	Total A	Total B	Hasil
1	Parkir HMS 45	134.33	1.77	75.56
2	Rest Area Parkir Aljabar	78.71	1.08	72.88
3	Sekolah Kuttab Al Fatih Bandung	70.54	1	70.54
4	Parkir Aljabar Warga Rw 03	101.59	1.47	68.64
5	Parkir Motor Posyandu Mawar Rw 04	66	1	66

4.3 Pembahasan Pengujian.

Tabel-tabel di atas menggambarkan hasil pengujian implementasi algoritma fuzzy dalam menentukan lokasi parkir. Tabel pertama menunjukkan nilai masukan dan fungsi keanggotaan yang digunakan dalam proses fuzzyfikasi. Tabel kedua memperlihatkan aturan inferensi yang diterapkan berdasarkan nilai keanggotaan dari variabel masukan. Tabel ketiga menggambarkan hasil akhir dari proses defuzzifikasi yang menentukan lokasi parkir berdasarkan hasil inferensi.

Dari hasil yang didapat, tiga diantaranya adalah lokasi parkir yang ideal. Parkir HMS 45 menjadi yang teratas dengan hasilnya 75.56 dan Parkir Motor Posyandu Mawar Rw 04 menjadi yang terkecil dengan hasilnya 66.

Hasil penelitian yang telah dilakukan pada penggunaan algoritma Fuzzy Tsukamoto untuk menemukan tempat parkir yang ideal, dapat dinyatakan bahwa metode Fuzzy Tsukamoto terbukti efektif dalam menentukan lokasi parkir yang ideal berdasarkan dengan rule yang telah dibuat berdasarkan jarak, jumlah kapasitas dan jumlah disukai.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Menurut hasil penelitian yang dilaksanakan dalam studi berjudul “Implementasi algoritma Fuzzy untuk pencarian lokasi parkir yang ideal ”, dapat diambil sebuah kesimpulan yang mampu menjawab perumusan masalah yang telah ditentukan sebelumnya, yaitu sebagai berikut:

1. Penelitian ini berhasil menerapkan algoritma fuzzy dalam mencari tempat parkir yang ideal.
2. Hasil pengujian menunjukkan bahwa algoritma fuzzy efektif digunakan dalam menentukan tempat parkir yang ideal. Kinerja logika fuzzy berjalan dengan baik dan sesuai dengan yang diharapkan.
3. Sistem ini memungkinkan penilaian apakah lokasi parkir tersebut dikatakan layak direkomendasikan atau tidak, menggunakan pendekatan metode Fuzzy.
4. Sistem ini akan menampilkan lokasi parkir dengan jarak 1-kilometer dari lokasi yang dipilih.

5.2 Saran

Hasil dari penelitian yang telah dilaksanakan menyediakan beberapa saran yang dapat dipertimbangkan dalam pelaksanaan penelitian selanjutnya. Beberapa dari saran-saran tersebut mencakup:

1. Perlunya penambahan alat untuk mendeteksi ada atau tiadanya mobil pada parkir. Perlu menggunakan alat yang menggunakan sensor *Passive Infra Red (PIR)* yang mana sensor dapat memberikan informasi kapasitas yang tersedia pada lokasi parkir, agar dapat memberikan informasi secara real time dan lebih efisien dalam mencari lokasi parkir.
2. Diperlukan suatu sistem notifikasi yang berfungsi dengan efisien untuk memastikan informasi dapat diterima dengan efektif oleh pengguna parkir.

DAFTAR PUSTAKA

- [1] A. Hidayat and F. Piliang, "Rancang Bangun Sistem Informasi Penyewaan Lahan Parkir Berbasis Web Gis," *J. Sist. Inf. dan Sains Teknol.*, vol. 1, no. 1, pp. 1–9, 2019, doi: 10.31326/sistek.v1i1.320.
- [2] Almadinah Putri Brilian, "Catat! Ini Sanksi dan Denda buat Pemilik Mobil yang Parkir di Jalan Depan Rumah," *detikProperti*, 2023. <https://www.detik.com/properti/berita/d-6826936/catat-ini-sanksi-dan-denda-buat-pemilik-mobil-yang-parkir-di-jalan-depan-rumah>.
- [3] detikNews, "Ini Penyebab Maraknya Parkir Liar di Jakarta: Males dan Mahal," *detikNews*, 2014. <https://news.detik.com/berita/d-2623592/ini-penyebab-maraknya-parkir-liar-di-jakarta-males-dan-mahal>
- [4] M. Mukaromah, "Penerapan Metode Fuzzy Sugeno Untuk Menentukan Jalur Terbaik Menuju Lokasi Wisata Di Surabaya," *J. Mat. Sains dan Teknol.*, vol. 20, no. 2, pp. 95–101, 2019, doi: 10.33830/jmst.v20i2.187.2019.
- [5] E. Sugiharto, I. -, and I. D. Wijaya, "Sistem Rekomendasi Tempat Wisata di Malang Raya Dengan Metode Fuzzy Berbasis Web," *J. Apl. Dan Inov. Ipteks "Soliditas,"* vol. 4, no. 1, p. 8, 2021, doi: 10.31328/js.v4i1.1731.
- [6] M. Lusmiawati, E. Fatkhiyah, and A. Hamzah, "Penentuan Objek Wisata Kota Bandung Menggunakan Metode Fuzzy Tsukamoto," *J. Scr.*, vol. 9, no. 2, pp. 142–151, 2021.
- [7] P. Bobot and P. Graf, "Purwanto, Y., Purwitasari, A., & Wibowo, A. W. (2005). Implementasi dan Analisis Algoritma Pencarian Rute Terpendek di Kota Surabaya. Jurnal Penelitian dan Pengembangan TELEKOMUNIKASI, 10 (2) 94-101. Taufiq, M. (2015). Men.pdf," vol. 8, no. 1, pp. 40–49, 2019.
- [8] U. Nurhasan, H. Suyono, and M. Muslim, "Sistem Cerdas Pencarian Lokasi Parkir Terbaik Dengan Algoritma Fuzzy-Best Fit," *J. EECCIS*, vol. 8, no. 1, p. pp.7-12, 2014.
- [9] O. R. Putri, A. Hamzah, and E. Fatkhiyah, "Sistem Pendukung Keputusan Pemilihan Objek Wisata Alam Di Daerah Istimewa Yogyakarta Dengan Metode Fuzzy Tsukamoto," *J. Scr.*, vol. 9, no. 1, pp. 74–83, 2021.
- [10] N. S. Pasaribu, J. T. Hardinata, and H. Qurniawan, "Application of The Fuzzy Tsukamoto Method in Determining Household Industry Products," *J.*

- Artif. Intell. Eng. Appl.*, vol. 1, no. 1, pp. 71–75, 2021, doi: 10.59934/jaiea.v1i1.57.
- [11] A. Oktaviani, D. Sarkawi, and A. Priadi, “Perancangan Sistem Parkir Pada Gedung Menara Palma Jakarta,” *Petir*, vol. 12, no. 2, pp. 231–241, 2019, doi: 10.33322/petir.v12i2.532.
- [12] Ramadhika Dwi Poetra, “BAB II Tinjauan Pustaka BAB II TINJAUAN PUSTAKA 2.1. 1–64,” *Gastron. ecuatoriana y Tur. local.*, vol. 1, no. 69, pp. 5–24, 2019.
- [13] W. Setiyaningsih, *Konsep Sistem Pendukung Keputusan*, vol. 1. 2015.
- [14] A. . dan Y. . L. Rindengan, *Sistem Fuzzy*. 2019.
- [15] S. Tjandra and G. S. Chandra, “Pemanfaatan Flutter dan Electron Framework pada Aplikasi Inventori dan Pengaturan Pengiriman Barang,” *J. Inf. Syst. Hosp. Technol.*, vol. 2, no. 02, pp. 76–81, 2020, doi: 10.37823/insight.v2i02.109.
- [16] C. Shah, “MySQL,” *A Hands-On Introd. to Data Sci.*, pp. 187–206, 2020, doi: 10.1017/9781108560412.008.
- [17] M. Ridwan, T. H. Sinaga, and M. Elsera, “Penerapan Framework Codeigniter Dalam Perancangan Aplikasi Manajemen Iuran Perumahan Griya Mandiri,” *Djtechno J. Teknol. Inf.*, vol. 3, no. 1, pp. 49–58, 2022, doi: 10.46576/djtechno.v3i1.2196.
- [18] Ratna Patria, “Rest API adalah: Perbedaan Rest API dan Restful AP,” *DomaiNesia*, 2023. <https://www.domainesia.com/berita/rest-api-adalah/#:~:text=Salah satu perbedaan Rest API,semua batasan dan panduan tersebut.>
- [19] M. Laya and B. S. Arifin, “Web Service Processor sebagai Penghubung Sistem Kiosk Medicom dengan SIM RS Kanker Dharmais,” *Multinetics*, vol. 3, no. 2, pp. 49–56, 2017, doi: 10.32722/multinetics.v3i2.1129.
- [20] S. Utomo and M. A. Hamdani, “Sistem Informasi Geografis (SIG) Pariwisata Kota Bandung menggunakan Google Maps API dan PHP,” *J. Teknol. Inf. dan Komun.*, vol. 11, no. 1, pp. 1–9, 2021.

LAMPIRAN

1. Potongan Kode