

BAB II

TINJAUAN PUSTAKA

2.1 Sistem

Sistem berasal dari bahasa Latin (*systēma*) dan bahasa Yunani (*sustēma*) adalah sekumpulan unsur atau elemen yang saling berkaitan dan saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan. Sistem merupakan sebuah objek yang dikaji atau dipelajari, dimana memiliki karakteristik tertentu atau spesifikasi tersendiri. Sistem mendefinisikan pengertian dari sistem sebagai berikut : Suatu sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen, atau variable-variabel yang terorganisasi, saling berinteraksi, saling tergantung satu sama lain dan terpadu. Sistem juga merupakan kumpulan elemen-elemen saling terkait dan bekerja sama untuk memproses masukan (*input*) yang ditujukan kepada sistem tersebut dan mengolah masukan tersebut sampai menghasilkan keluaran (*output*) yang diinginkan.

Sistem adalah setiap sesuatu terdiri dari obyek-obyek, atau unsur-unsur, atau komponen-komponen yang bertata kaitan dan bertata hubungan satu sama lain, sedemikian rupa sehingga unsur-unsur tersebut merupakan satu kesatuan pemrosesan atau pengolahan yang tertentu. Secara sederhana menerangkan bahwa sistem dapat diartikan sebagai suatu kumpulan himpunan dari unsur, komponen atau variabel-variabel yang terorganisasi, saling berinteraksi, saling tergantung satu sama lain dan terpadu. Pengembangan sistem (*system development*) dapat berarti dapat menyusun suatu sistem yang baru untuk menggantikan sistem yang lama secara keseluruhan atau memperbaiki sistem yang telah ada. Pembangunan sistem adalah sekumpulan aktivitas yang menggambarkan secara rinci bagaimana sistem akan berjalan. Hal itu bertujuan untuk menghasilkan produk perangkat lunak yang sesuai dengan kebutuhan *user*. Dari pendapat yang dikemukakan di atas dapat disimpulkan bahwa sistem adalah suatu kumpulan atau kelompok dari elemen atau komponen yang saling berhubungan atau saling berinteraksi dan saling bergantung satu sama lain untuk mencapai tujuan tertentu (Musthofa et al., 2022).

2.2 Sistem Pendukung Keputusan

Sistem pendukung keputusan adalah sistem penghasil informasi yang ditujukan pada suatu masalah tertentu yang harus dipecahkan oleh manager dan dapat membantu manager dalam pengambilan keputusan. Sistem pendukung keputusan merupakan bagian tak terpisahkan dari totalitas sistem organisasi keseluruhan. Suatu sistem organisasi mencakup sistem fisik, sistem keputusan dan sistem informasi. Sistem pendukung keputusan adalah sebagai sistem basis komputer yang terdiri dari tiga komponen yang saling berinteraksi, sistem bahasa (mekanisme untuk memberikan komunikasi antara pengguna dan komponen sistem pendukung keputusan lain), sistem pengetahuan (respositori pengetahuan domain masalah yang ada pada sistem pendukung keputusan atau sebagai data atau sebagai prosedur), dan sistem pemrosesan masalah (hubungan antara dua komponen lainnya terdiri dari satu atau lebih kapabilitas manipulasi masalah umum yang diperlukan untuk pengambilan keputusan).

Konsep SPK ditandai dengan sistem interaktif berbasis komputer yang membantu pengambilan keputusan. Dengan memanfaatkan data dan model untuk menyelesaikan masalah yang bersifat tidak terstruktur dan semi terstruktur. Sistem pendukung keputusan didefinisikan sebagai sistem berbasis komputer yang terdiri dari komponen-komponen yang saling berinteraksi, yaitu : sistem bahasa, sistem pengetahuan, dan sistem pemrosesan masalah. Sistem Pendukung Keputusan (SPK) bukan merupakan alat pengambilan keputusan, melainkan merupakan sistem yang membantu pengambil keputusan dengan melengkapi mereka dengan informasi dari data yang telah diolah dengan relevan dan diperlukan untuk membuat keputusan tentang suatu masalah dengan lebih cepat dan akurat. SPK ditujukan untuk membantu para pengambil keputusan untuk memecahkan masalah semi dan atau tidak terstruktur dengan fokus menyajikan informasi yang nantinya bisa dijadikan sebagai bahan alternatif pengambilan keputusan yang terbaik (Sintawati, 2022).

Secara umum, sistem pendukung keputusan adalah sistem yang berusaha mengadopsi pengetahuan manusia ke komputer, agar komputer dapat menyelesaikan masalah seperti yang biasa dilakukan oleh para ahli. Sistem pendukung keputusan memberikan nilai tambahan pada teknologi untuk membantu

dan menangani era reformasi yang semakin canggih (Nasution & KHAIRUNA, 2017). Bagi seorang ahli sendiri, sistem ini akan membantu aktivitasnya sebagai seorang asisten yang sangat berpengalaman. Sistem pendukung keputusan mengkombinasikan kaidah-kaidah penarikan kesimpulan (inference rules) dengan basis pengetahuan tertentu yang diberikan oleh satu atau lebih pakar dalam bidang tertentu (Zufria & Santoso, 2021).

2.3 Pengertian Jalan

Jalan merupakan sarana transportasi darat yang sangat penting bagi masyarakat dalam memperlancar perekonomian dan kebudayaan antar daerah di Indonesia. Dengan kondisi jalan yang baik akan memudahkan masyarakat dalam mengadakan kegiatan sosial lainnya. Seiring kenaikan perekonomian masyarakat, kondisi jalan yang dilalui oleh volume lalu lintas yang tinggi dan berulang-ulang akan mempengaruhi kondisi konstruksi jalan, dan mengakibatkan terjadinya penurunan kualitas jalan tersebut, sehingga berdampak pada keamanan, kenyamanan, dan kelancaran dalam berlalu lintas. Prasurvei yang dilakukan terhadap kondisi lapangan tentang kerusakan jalan terjadi karena sering dilintasi kendaraan berat bermuatan lebih (Overloading) atau yang melebihi batas beban maksimum, meningkatkan beban pada permukaan jalan oleh roda kendaraan sehingga dapat mempercepat kerusakan pada struktur jalan yang membuat kecepatan kendaraan tersebut menurun dan waktu tempuh akan terasa semakin lama serta dapat memungkinkan mengancam keselamatan pengendara (kecelakaan). Dampak langsung kekasaran jalan yang buruk diantaranya, seperti lapisan permukaan yang diakibatkan kinerja jalan menurun, memberikan tekanan pada struktur kendaraan, dan menurunkan tingkat kenyamanan bagi pengguna jalan. Untuk menjaga supaya tidak terjadinya kerusakan pada badan jalan mulai dari jalan retak, bekas roda, hingga jalan berlubang dimana-mana, maka butuh dilakukan penelitian awal untuk kondisi permukaan jalan dengan melakukan survei visual dengan cara menganalisis kerusakan jenis dan tingkat kerusakannya (Aptarila et al., 2020).

2.4 Logika Fuzzy

Konsep logika fuzzy pertama kali diperkenalkan oleh Professor Lotfi A. Zadeh dari Universitas California, pada bulan Juni 1965. Fuzzy secara bahasa

diartikan sebagai kabur atau samar-samar. Fuzzy merupakan suatu nilai yang dapat bernilai benar atau salah secara bersamaan. Namun seberapa besar nilai kebenaran dan kesalahannya tergantung pada derajat keanggotaan yang dimilikinya. Derajat keanggotaan dalam fuzzy memiliki rentang nilai 0 (nol) hingga 1 (satu). Hal ini berbeda dengan himpunan tegas yang memiliki nilai 1 atau 0 (ya atau tidak). Logika fuzzy digunakan untuk menterjemahkan suatu besaran yang diekspresikan menggunakan bahasa (linguistik), misalkan besaran kecepatan laju kendaraan yang diekspresikan dengan pelan, agak cepat, cepat, dan sangat cepat. Dan logika fuzzy menunjukkan sejauh mana suatu nilai itu benar dan sejauh mana suatu nilai itu salah. Tidak seperti logika tegas, suatu nilai hanya mempunyai 2 kemungkinan yaitu merupakan suatu anggota himpunan atau tidak. Derajat keanggotaan 0 (nol) artinya nilai bukan merupakan anggota himpunan dan 1 (satu) berarti nilai tersebut adalah anggota himpunan. (Sovina & Harahap, 2022)

Logika fuzzy adalah alat matematika yang digunakan untuk mengatasi ketidakpastian dan ambiguitas dalam pengambilan keputusan. Logika fuzzy memperluas logika boolean yang menggunakan nilai kebenaran "benar" atau "salah" menjadi himpunan fuzzy dengan nilai keanggotaan yang dapat berada di antara 0 dan 1. Ini memungkinkan penanganan variabel yang memiliki tingkat ketidakpastian atau tingkat keanggotaan yang tidak dapat dinyatakan secara biner.

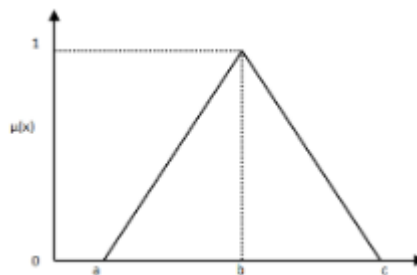
Logika fuzzy adalah logika yang dapat digunakan untuk menganalisis masalah yang mengandung ketidakpastian, salah satu contohnya proses prediksi. Logika ini dianggap mampu untuk memetakan suatu input ke dalam suatu output tanpa mengabaikan faktor-faktor yang ada. Logika fuzzy sering digunakan untuk mengekspresikan suatu nilai yang diterjemahkan dalam bahasa (linguistic), misalnya untuk mengekspresikan suhu dalam ruangan apakah ruangan tersebut dingin, hangat, atau panas. Logika fuzzy adalah suatu cara yang tepat untuk memetakan suatu ruang input dalam suatu ruang output dan memiliki nilai yang berkelanjutan. (Djarat, Widiastuti, & Sihotang, 2019)

Fuzzy Logic merupakan metode pemecahan masalah dalam memetakan suatu permasalahan yang berupa masukan atau input ke dalam suatu ruang keluaran atau output atau hasil. Nilai input yang dimasukan bakalan dirubah menjadi sebuah

nilai linguistik. Perubahan tersebut terjadi pada fungsi keanggotaan. Fungsi keanggotaan yang digunakan adalah representasi kurva segitiga dan representasi kurva trapesium (Kurniadi, Nuraeni, & Jaelani, 2022)

Adapun bentuk umum dari fungsi keanggotaan yang digunakan adalah sebagai berikut:

1. Representasi Kurva Segitiga



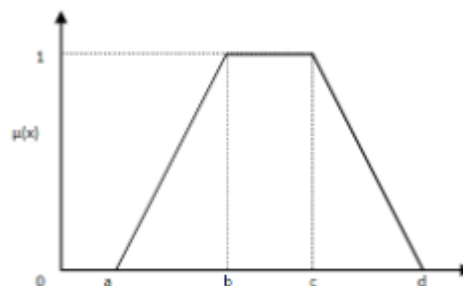
Gambar 1.1 Representasi Kurva Segitiga

(Sumber : Kurniadi et al., 2022)

Fungsi Keanggotaan:

$$\mu[x] = \begin{cases} 0; & x \leq a \text{ atau } x \geq c \\ (x - a)/(b - a); & a \leq x < b \\ (b - x)/(c - b); & b \leq x \leq c \end{cases} \dots\dots\dots(2.1)$$

2. Representasi Kurva Trapesium



Gambar 1.2 Representasi Kurva Trapesium

(Sumber : Kurniadi et al., 2022)

Fungsi Keanggotaan:

$$\mu[x] = \begin{cases} 0; & x \leq a \text{ atau } x \geq d \\ (x - a)/(b - a); & a \leq x \leq b \\ 1; & b \leq x \leq c \\ (d - x)/(d - c); & x \geq c \end{cases} \dots\dots\dots(2.2)$$

2.5 *Fuzzy Tahani*

Fuzzy Tahani adalah metode pengambilan keputusan yang dikembangkan oleh Tahani pada tahun 2000. Metode ini menggabungkan logika fuzzy dengan konsep pengambilan keputusan multi-kriteria untuk menentukan prioritas berdasarkan variabel input yang beragam. Analisis Fuzzy Tahani melibatkan tahap fuzzifikasi untuk mengubah variabel input menjadi variabel linguistik, pengembangan himpunan fuzzy yang menggambarkan tingkat kondisi fisik, formulasi aturan fuzzy untuk menghubungkan himpunan fuzzy, inferensi fuzzy untuk mengambil keputusan fuzzy, dan defuzzifikasi untuk mengubah hasil fuzzy menjadi nilai konkret.

Fuzzy Tahani adalah salah satu cabang dari logika fuzzy, yang merupakan salah satu metode fuzzy yang menggunakan basis data standar. Tahani mendeskripsikan suatu metode pemrosesan query fuzzy, dengan didasarkan atas manipulasi bahasa yang dikenal dengan nama SQL (Structured Query Language), sehingga model Fuzzy Tahani sangat tepat digunakan dalam proses pencarian data yang tepat dan akurat. Model Tahani ini masih menerapkan relasi kueri standar, akan tetapi model ini menggunakan teori dari himpunan fuzzy dalam mendapatkan informasi pada kuerinya (Permana & Lelah, 2020). Fuzzy model tahani digunakan karena penerapannya menggunakan database sebagai penyimpanan data yaitu model tahani yang dirasa database lebih kuat. Pada fuzzy model Tahani, alur proses sudah dirancang dengan beberapa tahapan yang memudahkan dalam perancangan dan penghitungan menggunakan basis data. Tahapan Logika Fuzzy model Tahani yaitu, menggambarkan fungsi/derajat keanggotaan, Fuzzifikasi, dan membuat rekomendasi keputusan. Contoh dari penerapan Fuzzy Tahani adalah sebagai berikut:

1. Perancangan Domain Fuzzy

Dari hasil wawancara, maka diperoleh batas dari himpunan fuzzy untuk setiap variabel himpunan. Domain untuk masing-masing himpunan fuzzy pada penelitian ini bisa dilihat pada tabel 2.1.

Tabel 1.1 Pembentukan Domain *Fuzzy*

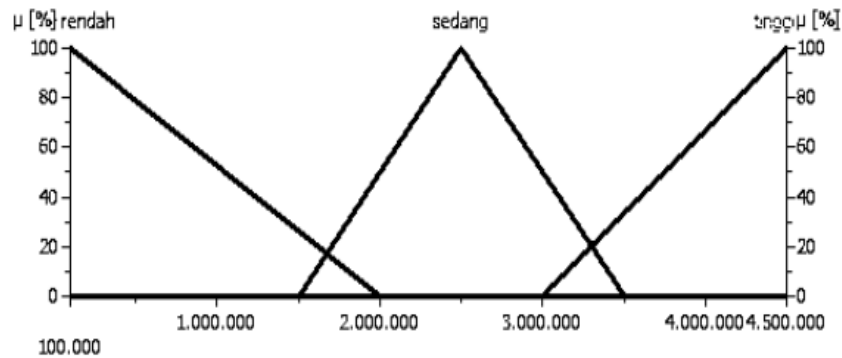
Variabel	Himpunan	Domain
Penghasilan	Rendah	[1.000.000-2000.000]
	Sedang	[1.500.000-2.500.000-3.500.000]
	Tinggi	[3.000.000-4.500.000]
Tanggungan	Sedikit	[0-5]
	Banyak	[3-8]
Usia	Muda	[25-35]
	Parubaya	[30-40-50]
	Tua	[45-60]
Luas tanah	Sempit	[0-30]
	Luas	[20-50]

(Sumber: Permana & Lelah, 2020)

Kemudian berdasarkan batasan fuzzy tersebut, maka fungsi keanggotaan dari setiap variabel dapat direpresentasikan dalam bentuk representasi kurva linear dan segitiga sebagai berikut. Representasi linear merupakan pemetaan dari inputan ke derajat keanggotaannya yang digambarkan sebagai sebuah garis lurus. Ada dua keadaan himpunan fuzzy linear, yaitu representasi linear naik dan turun. Sedangkan representasi segitiga pada dasarnya merupakan gabungan antara dua garis linear naik dan turun.

2. Fungsi Keanggotaan Variabel Penghasilan

Variabel penghasilan terbagi menjadi tiga himpunan fuzzy, yaitu tinggi, sedang dan rendah. Representasi dari variabel penghasilan terlihat pada gambar 2.3.



Gambar 1.3 Representasi Variabel Penghasilan

(Sumber: Permana & Lelah, 2020)

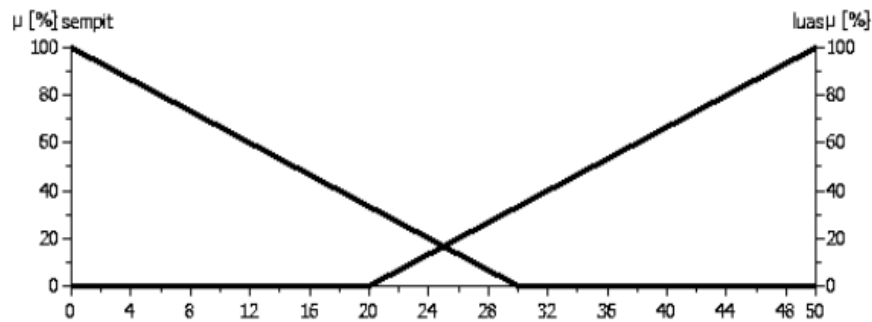
Fungsi keanggotaan dari representasi variabel penghasilan di atas, maka dapat dirumuskan dalam rumus sebagai berikut ini:

$$\begin{aligned} \mu[x]_{\text{rendah}} &= \begin{cases} 1; & x \leq 100.000 \\ \frac{2.000.000 - x}{2.000.000 - 100.000}; & 100.000 < x < 2.000.000 \\ 0; & x \geq 2.000.000 \end{cases} \\ \mu[x]_{\text{sedang}} &= \begin{cases} 0; & x \leq 1.500.000 \text{ atau } x \geq 3.500.000 \\ \frac{x - 1.500.000}{2.500.000 - 1.500.000}; & 1.500.000 \leq x \leq 2.500.000 \\ \frac{3.500.000 - x}{3.500.000 - 2.500.000}; & 2.500.000 \leq x \leq 3.500.000 \\ 1; & x = 2.500.000 \end{cases} \\ \mu[x]_{\text{tinggi}} &= \begin{cases} 0; & x \leq 3.000.000 \\ \frac{x - 3.000.000}{4.500.000 - 3.000.000}; & 3.000.000 \leq x \leq 4.500.000 \\ 1; & x \geq 4.500.000 \end{cases} \end{aligned} \quad (2.3)$$

3. Fungsi Keanggotaan Variabel Luas Tanah

Variabel luas tanah terbagi menjadi dua himpunan fuzzy, yaitu sempit dan luas. Representasi dari variabel luas tanah terlihat pada gambar

2.4.



Gambar 1.4 Representasi Variabel Luas Tanah

(Sumber : Permana & Lelah, 2020)

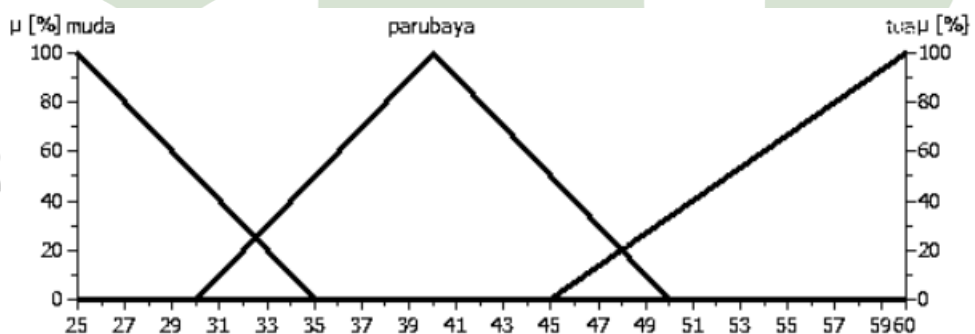
Fungsi keanggotaan dari representasi variabel luas tanah di atas, maka dapat dirumuskan dalam rumus sebagai berikut ini:

$$\mu[x]_{sempit} = \begin{cases} 1; & x \leq 0 \\ \frac{30 - x}{30 - 0}; & 0 \leq x \leq 30 \\ 0; & x \geq 30 \end{cases}$$

$$\mu[x]_{luas} = \begin{cases} 0; & x \leq 20 \\ \frac{x - 20}{50 - 20}; & 20 \leq x \leq 50 \\ 1; & x \geq 50 \end{cases} \dots\dots\dots(2.4)$$

4. Fungsi Keanggotaan Variabel Usia

Variabel usia terbagi menjadi tiga himpunan fuzzy, yaitu tua, parubaya dan muda. Representasi dari variabel usia terlihat pada gambar 2.5.



Gambar 1.5 Representasi Variabel Usia

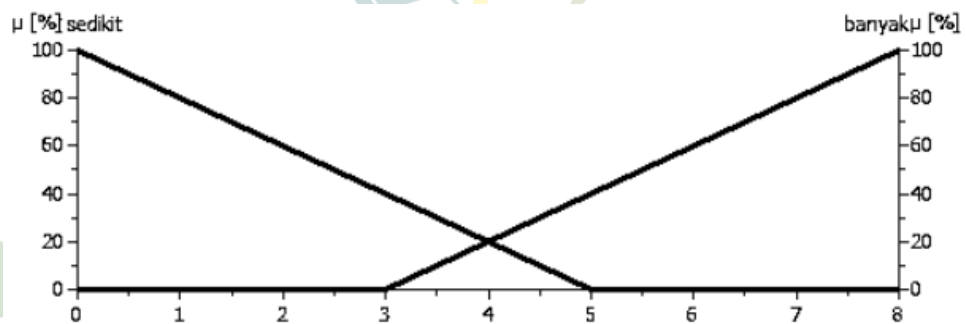
(Sumber : Permana & Lelah, 2020)

Fungsi keanggotaan dari representasi variabel usia di atas, maka dapat dirumuskan dalam rumus berikut ini:

$$\begin{aligned}
\mu[x]muda &= \begin{cases} 1; & x \leq 25 \\ \frac{35-x}{35-25}; & 25 \leq x \leq 35 \\ 0; & x \geq 35 \end{cases} \\
\mu[x]parubaya &= \begin{cases} 0; & x \leq 30 \text{ atau } x \geq 50 \\ \frac{x-30}{40-30}; & 30 \leq x \leq 40 \\ \frac{50-x}{50-40}; & 40 \leq x \leq 50 \\ 1; & x = 40 \end{cases} \\
\mu[x]tua &= \begin{cases} 0; & x \leq 45 \\ \frac{x-45}{60-45}; & 45 \leq x \leq 60 \\ 1; & x \geq 60 \end{cases}
\end{aligned}
\tag{2.5}$$

5. Fungsi Keanggotaan Variabel Tanggungan

Variabel tanggungan terbagi menjadi dua himpunan fuzzy, yaitu banyak dan sedikit. Representasi dari variabel tanggungan terlihat pada gambar 2.6.



Gambar 1.6 Representasi Variabel Tanggungan

(Sumber : Permana & Lelah, 2020)

Fungsi keanggotaan dari representasi variabel tanggungan di atas, maka dapat dirumuskan dalam rumus sebagai berikut ini:

$$\begin{aligned}
\mu[x]sedikit &= \begin{cases} 1; & x \leq 0 \\ \frac{5-x}{5-0}; & 0 \leq x \leq 5 \\ 0; & x \geq 5 \end{cases} \\
\mu[x]banyak &= \begin{cases} 0; & x \leq 5 \\ \frac{x-3}{8-3}; & 5 \leq x \leq 8 \\ 1; & x \geq 8 \end{cases}
\end{aligned}
\tag{2.6}$$

6. Inferensi Rule

Pada peneliti menentukan rule dari logika fuzzy dengan tujuan menyatakan relasi antara variabel input dan output, dengan menggunakan operator zadeh. Rule yang terbentuk sebanyak 36 sebagai berikut:

Tabel 1.2 Aturan *Fuzzy*

No	Rule
1	IF penghasilan = rendah AND tanggungan = sedikit AND usia = muda AND luas tanah = sempit THEN prasejahtera
2	IF penghasilan = rendah AND tanggungan = sedikit AND usia muda AND luas tanah = luas THEN prasejahtera
3	IF penghasilan = rendah AND tanggungan = sedikit AND usia = parubaya AND luas tanah = sempit THEN prasejahtera
4	IF penghasilan = rendah AND tanggungan = sedikit AND usia = parubaya AND luas tanah = luas THEN prasejahtera
5	IF penghasilan = rendah AND tanggungan = sedikit AND usia = tua AND luas tanah = sempit THEN keluarga tidak mampu
6	IF penghasilan = rendah AND tanggungan = sedikit AND usia = tua AND luas tanah = luas THEN keluarga tidak mampu
7	IF penghasilan = rendah AND tanggungan = banyak AND usia = muda AND luas tanah = sempit THEN prasejahtera
8	IF penghasilan = rendah AND tanggungan = banyak AND usia = muda AND luas tanah = luas THEN prasejahtera
9	IF penghasilan = rendah AND tanggungan = banyak AND usia = paruhbaya AND luas tanah = sempit THEN keluarga tidak mampu
10	IF penghasilan = rendah AND tanggungan = banyak AND usia = parubaya AND luas tanah = luas THEN keluarga tidak mampu
11	IF penghasilan = rendah AND tanggungan = banyak AND usia = tua AND luas tanah = sempit THEN keluarga tidak mampu
12	IF penghasilan = rendah AND tanggungan = banyak AND usia = tua AND luas tanah = luas THEN keluarga tidak mampu
13	IF penghasilan = sedang AND tanggungan = sedikit AND usia = muda AND luas tanah = sempit THEN prasejahtera

No	Rule
14	IF penghasilan = sedang AND tanggungan = sedikit AND usia = muda AND luas tanah = luas THEN prasejahtera
15	IF penghasilan = sedang AND tanggungan = sedikit AND usia = parubaya AND luas tanah = sempit THEN prasejahtera
16	IF penghasilan = sedang AND tanggungan = sedikit AND usia = parubaya AND luas tanah = luas THEN prasejahtera
17	IF penghasilan = sedikit AND tanggungan = sedikit AND usia = tua AND luas tanah = sempit THEN prasejahtera
18	IF penghasilan = sedang AND tanggungan = sedikit AND usia = tua AND luas tanah = luas THEN prasejahtera
19	IF penghasilan = sedang AND tanggungan = banyak AND usia = muda AND luas tanah = sempit THEN prasejahtera
20	IF penghasilan = sedang AND tanggungan = banyak AND usia = muda AND luas tanah = luas THEN prasejahtera
21	IF penghasilan = sedang AND tanggungan = banyak AND usia = parubaya AND luas tanah = sempit THEN keluarga tidak mampu
22	IF penghasilan = sedang AND tanggungan = banyak AND usia = parubaya AND luas tanah = luas THEN keluarga tidak mampu
23	IF penghasilan = sedang AND tanggungan = banyak AND usia = tua AND luas tanah = sempit THEN keluarga tidak mampu
24	IF penghasilan = sedang AND tanggungan = banyak AND usia = tua AND luas tanah = luas THEN keluarga tidak mampu
25	IF penghasilan = tinggi AND tanggungan = sedikit AND usia = muda AND luas tanah = sempit THEN sejahtera
26	IF penghasilan = tinggi AND tanggungan = sedikit AND usia = muda AND luas tanah = luas THEN sejahtera
27	IF penghasilan = tinggi AND tanggungan = sedikit AND usia = parubaya AND luas tanah = sempit THEN sejahtera

No	Rule
28	IF penghasilan = tinggi AND tanggungan = sedikit AND usia = parubaya AND luas tanah = luas THEN sejahtera
29	IF penghasilan = tinggi AND tanggungan = sedikit AND usia = tua AND luas tanah = sempit THEN sejahtera
30	IF penghasilan = tinggi AND tanggungan = sedikit AND usia = tua AND luas tanah = luas THEN sejahtera
31	IF penghasilan = tinggi AND tanggungan = banyak AND usia = muda AND luas tanah = sempit THEN sejahtera
32	IF penghasilan = tinggi AND tanggungan = banyak AND usia = muda AND luas tanah = luas THEN sejahtera
33	IF penghasilan = tinggi AND tanggungan = banyak AND usia = parubaya AND luas tanah = sempit THEN prasejahtera
34	IF penghasilan = tinggi AND tanggungan = banyak AND usia = parubaya AND luas tanah = luas THEN prasejahtera
35	IF penghasilan = tinggi AND tanggungan = banyak AND usia = tua AND luas tanah = sempit THEN prasejahtera
36	IF penghasilan = tinggi AND tanggungan = banyak AND usia = tua AND luas tanah = luas THEN prasejahtera

(Sumber : Permana & Lelah, 2020)

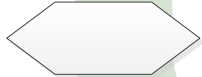
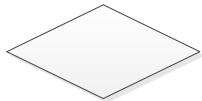
2.6 Flowchart

Flowchart adalah penggambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program. Bagan alir (*flowchart*) adalah bagan (*chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika. Bagan alir digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi. Ada lima macam bagan alir, di antaranya:

1. Bagan Alir Sistem (*system flowchart*) merupakan bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem.
2. Bagan Alir Dokumen (*document flowchart*) disebut juga bagan alir formulir (*form flowchart*) merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusan-tembusannya.

3. Bagan Alir Skematik (*schematic flowchart*) merupakan bagan alir yang menggambarkan prosedur di dalam sistem dengan menggunakan simbol-simbol bagan alir sistem dan gambar-gambar komputer serta peralatan lainnya yang digunakan oleh sistem.
4. Bagan Alir Program (*program flowchart*) merupakan bagan yang menjelaskan secara rinci langkah-langkah dari proses program.
5. Bagan Alir Proses (*process flowchart*) merupakan bagan alir yang banyak digunakan di teknik industri untuk menggambarkan proses dalam suatu prosedur. (Verawati & Liksha, 2018)

Tabel 1.3 Simbol-simbol Flowchart

Simbol	Nama	Fungsi
	Terminator	Permulaan/akhir program
	Garis Alir (Flow Line)	Arah aliran program
	Preparation	Proses inisialisasi/pemberian harga awal
	Process	Proses perhitungan/proses pengolahan data
	Input/output Data	Proses input/output data, parameter, informasi
	Predefine Process	Permulaan sub program/proses menjalankan sub program
	Decision	Perbandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya

Simbol	Nama	Fungsi
	On Page Connector	Penghubung bagian-bagian flowchat yang berada pada satu halaman
	Off Page Connector	Penghubung bagian-bagian flowchat yang berada pada halaman berbeda

(Sumber : Verawati & Liksha, 2018)

2.7 Pengertian UML

Unified Modelling Language (UML) adalah sebuah bahasa yang berdasarkan grafik atau gambar untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumentasian dari sebuah sistem pengembangan *software* berbasis OO (*Object-Oriented*). UML sendiri juga memberikan standar penulisan sebuah sistem *blue print*, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema *database*, dan komponen-komponen yang diperlukan dalam sistem. (Arif, 2020) Alat bantu yang digunakan dalam perancangan berorientasi objek berdasarkan UML adalah sebagai berikut:

2.7.1 Use Case Diagram

Use Case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *Use Case* digunakan untuk mengetahui fungsi apa saja yang berhak menggunakan fungsi-fungsi tersebut. Simbol-simbol yang ada pada *Use Case* dapat dilihat pada tabel 2.2. (Malius et al., 2021)

Tabel 1.4 Use Case Diagram

No.	Nama	Simbol	Keterangan
1	Actor		Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan use case.
2	Dependency		Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (independent) akan

No.	Nama	Simbol	Keterangan
			mempengaruhi elemen yang tidak mandiri independent)
3	Generalization	←	Hubungan dimana objek anak (descendent) berbagai perilaku dan struktur data dari objek yang ada di atasnya objek induk (ancestor)
4	Include	→	Menspesifikasikan bahwa use case sumber secara eksplisit
5	Extend	←	Menspesifikasikan bahwa use case target memperluas perilaku dari use case sumber pada suatu titik yang diberikan
6	Association	—	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7	System	○	Menspesifikasikan data paket yang menampilkan system secara terbatas

(Sumber : Malius et al., 2021)

Dari tabel 2.4 terdapat beberapa simbol yang digunakan dalam pembuatan *Use Case* pada penelitian ini yang pertama simbol *actor* yang menggambarkan orang yang akan menggunakan sistem, simbol *Use Case* unit-unit yang mewakili *menu-menu* dari sistem yang akan dibuat, yang ketiga adalah simbol asosiasi yang merupakan penghubung komunikasi antara aktor dengan *user*. (Malius et al., 2021)

2.7.2 Activity Diagram

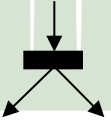
Activity diagram atau diagram aktivitas menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau *menu* yang ada pada perangkat lunak, yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan yang dilakukan aktor, jadi aktivitas

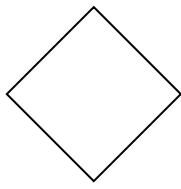
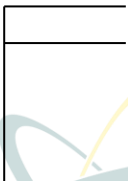
dilakukan oleh sistem. Diagram aktivitas banyak digunakan untuk mendefinisikan hal-hal berikut:

1. Rancangan proses bisnis di mana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
2. Pengelompokan tampilan dari sistem di mana setiap aktivitas dianggap memiliki rancangan antar muka tampilan.
3. Rancangan pengujian di mana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya. (Malius et al., 2021)

Simbol-simbol yang digunakan dalam *activity diagram* yaitu:

Tabel 1.5 Activity Diagram

No.	Nama	Simbol	Keterangan
1	Start Point		Diletakkan pada pojok kiri atas dan merupakan awal aktivitas
2	End Point		Akhir aktivitas
3	Activities		Menggambarkan suatu proses atau kegiatan bisnis
4	Fork		Digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu
5	Join		Digunakan untuk menunjukkan adanya dekomposisi

No.	Nama	Simbol	Keterangan
6	Decision Points		Menggambarkan pilihan untuk pengambilan keputusan, <i>true</i> atau <i>false</i>
7	Swimlane		Pembagian <i>activity</i> diagram untuk menunjukkan siapa melakukan apa

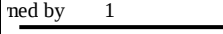
(Sumber : Malius et al. ; 2021)

2.7.3 Class Diagram

Class diagram adalah diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. *Class* adalah kumpulan objek-objek dengan dan yang mempunyai struktur umum, behavior umum, relasi umum, dan semantic/kata yang umum. *Class-class* ditentukan/ditemukan dengan cara memeriksa objek-objek dalam *sequence diagram* dan *collaboration diagram*. Sebuah *class* digambarkan seperti sebuah bujur sangkar dengan tiga bagian ruangan. Simbol dan keterangan *class diagram* dapat dilihat pada tabel 2.4. (Malius, Apriyanto, & Dani, 2021)

Tabel 1.6 Class Diagram

No.	Nama	Simbol	Keterangan
1	Dependency		Penggunaan dependency digunakan untuk menunjukkan operasi pada suatu class yang menggunakan class yang lain.
2	Class	Nama Class +atribut +atribut	Class adalah blok-blok pembangun pada pemrograman berorientasi objek. class digambarkan sebagai sebuah kotak yang terbagi atas 3 bagian.

No.	Nama	Simbol	Keterangan
3	Association		Sebuah asosiasi merupakan sebuah relationship paling umum antara 2 class dan dilambangkan oleh sebuah garis yang menghubungkan antara 2 class. Garis ini bisa melambangkan tipe-tipe relationship dan juga dapat menampilkan hukum-hukum multiplisitas pada sebuah relationship.
4	Composition		Jika sebuah class tidak bisa berdiri sendiri dan harus merupakan bagian dari class yang lain, maka class tersebut memiliki relasi Composition terhadap class tempat dia bergantung tersebut.

2.8 PHP (*Hypertext Preprocessor*)

Bahasa pemrograman PHP merupakan bahasa pemrograman untuk membuat *website* yang bersifat *server-side scripting*. PHP bersifat dinamis. PHP dapat dijalankan pada berbagai macam sistem operasi seperti Windows, Linux, dan Mac Os. Selain Apache, PHP juga mendukung beberapa *web server* lain, seperti Microsoft ISS, Caudium, dan PWS. PHP dapat memanfaatkan *database* untuk menghasilkan halaman *web* yang dinamis. Sistem manajemen *database* yang sering digunakan bersama PHP adalah MYSQL. Namun, PHP juga mendukung sistem manajemen Database Oracle, Microsoft Access, Interbase, d-ase, dan PostgreSQL. PHP adalah salah satu bahasan pemrograman skrip yang dirancang untuk membangun aplikasi *web*. Ketika dipanggil dari *web browser*, program yang ditulis dengan PHP akan di-parsing di dalam *web server* oleh interpreter PHP dan

diterjemahkan ke dalam dokumen HTML, yang selanjutnya akan ditampilkan kembali *web server*. Karena pemrosesan program PHP dilakukan didalam lingkungan *web browser*, PHP dikatakan sebagai bahasa sisi server (*server-side*). Oleh sebab itu, seperti yang telah dikemukakan sebelumnya, kode PHP tidak akan terlihat pada saat user memilih perintah “*View Source*” pada *web browser* yang mereka gunakan. Penulisan scrip PHP diawali dengan tanda lebih kecil (<) dan diakhiri tanda lebih besar (>). Ada cara untuk menuliskan script PHP, yaitu:

```
<? Scrip_php ?>
<? Php Scrip_php?>
<% Script_php %>
<Script laguange= “ php” > Script_php</Script>
```

Pemisah antara instruksi adalah titik koma (;) dan untuk membuat atau menambahkan komentar penulisannya adalah: /* komentar */, # komentar # komentar. Cara penulisan dibedakan menjadi *Embedded script* dan *non-Embedded script*. (Muhammad Saed Novendri, Ade Saputra, 2019)

1. *Embedded Script*

```
<html>
  <head>
    <title> PHP dengan Embedded Script </title>
  </head>
  <body>
    <?php echo “hallo_world!” ?>
  </body>
</html>
```

2. *Non-Embedded Script*

```
<html>
  <head>
    <title> PHP dengan Embedded Script </title>
  </head>
```

```
<body>
  <?php echo "hallo_world!" ?>
</body>
</html>
```

2.9 XAMPP

XAMPP adalah perangkat lunak gratis yang mendukung banyak sistem operasi dan merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai *server* yang terdiri sendiri (*localhost*), yang terdiri atas program Apache HTTP Server, MySQL *database*, dan penterjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP dan Perl. Program ini tersedia dalam GNU *General Public Lisensi* dan bebas, merupakan *web server* yang mudah digunakan yang dapat melayani tampilam halaman *web* yang dinamis. Mengenal bagian XAMPP yang biasa digunakan pada umumnya:

1. HTDOC adalah *folder* tempat meletakkan berkas-berkas yang akan dijalankan, seperti berkas PHP, HTML dan skrip lain.
2. PhpMyAdmin merupakan bagian untuk mengelola basis data MySQL yang ada dikomputer. Untuk membukanya, buka *browser* lalu ketikkan alamat `http://localhost/phpMyAdmin`, maka akan muncul halaman phpMyAdmin.
3. Panel yang berfungsi untuk mengelola layanan (*service*) XAMPP. Seperti menghentikan (*stop*) layanan, ataupun memulai (*start*). (Matusea, 2021)

2.10 MySQL

Pada perkembangannya, MYSQL disebut juga SQL yang merupakan singkatan dari *Structured Query Language*. SQL merupakan bahasa terstruktur yang khusus digunakan untuk mengolah *database*. SQL pertama kali didefinisikan oleh *American National Standards Institute* (ANSI) pada tahun 1986. MYSQL adalah sebuah sistem manajemen *database* yang bersifat *open source*. MYSQL merupakan sistem manajemen *database* yang bersifat relational. Artinya, data yang

dikelola dalam *database* yang akan diletakkan pada beberapa tabel yang terpisah sehingga manipulasi data akan jauh lebih cepat. MYSQL dapat digunakan untuk mengelola database mulai dari yang kecil sampai dengan yang sangat besar.

SQL juga dapat diartikan sebagai antar muka standar untuk sistem manajemen relasional, termasuk sistem yang beroperasi pada komputer pribadi. SQL memungkinkan seorang pengguna untuk mengetahui dimana lokasinya, atau bagaimana informasi tersebut disusun. SQL lebih mudah digunakan dibandingkan dengan bahasa pemrograman, tetapi rumit dibandingkan *software* lembar kerja dan pengolah data. Sebuah pernyataan SQL yang sederhana dapat menghasilkan set permintaan untuk informasi yang tersimpan pada komputer yang berbeda diberbagai lokasi yang tersebar, sehingga membutuhkan waktu dan sumber daya komputasi yang banyak. SQLinte dapat digunakan untuk ivestigasi interaktif, atau pembuatan laporan adhoc atau disisipkan dalam program aplikasi.

SQL juga merupakan bahasa pemrograman yang dirancang khusus untuk mengirimkan suatu perintah *query* (pengaksesan data berdasarkan pengalamatan tertentu) terhadap sebuah *database*. Kebanyakan *software database* mengimplementasikan SQL secara sedikit berbeda, tapi seluruh *database* SQL mendukung *subset* standar yang ada. Jadi, SQL adalah permintaan yang melekat pada suatu *database* atau SMBD tertentu. Dengan kata lain, SQL adalah perintah atau bahasa yang melekat di dalam SMBD. Sebagai suatu bahasa permintaan, SQL didukung oleh SMBD, seperti MySQL Server, MySQL, PostgreSQL, Interbase, dan Oracle. Selain itu SQL juga didukung oleh *database* bukan *server*, seperti MS Access maupun Paradox (Novendri & Saputra, 2019)

2.11 Database

Database merupakan sekumpulan dari data yang terintegrasi. Sistem *Database* merupakan sistem komputerisasi yang bertujuan untuk menyimpan sejumlah data sehingga memudahkan pengguna untuk mendapatkan dan mengupdate informasi sesuai dengan kebutuhan Informasi yang disimpan dalam *Database* dapat berupa *text* maupun angka. Salah satu penggunaan *Database* dalam kehidupan sehari hari adalah pada aplikasi *software* pembayaran oleh kasir *supermarket*. Ketika melakukan pembayaran dikasir, kasir tersebut melakukan

pekerjaan dengan cara menempelkan *bar-code* di salah satu sensor *infrared*. Kemudian muncul harga dari barang tersebut dalam komputer. Sehingga kasir tersebut tidak perlu mengetik ulang barang yang yang dibeli oleh pembeli tersebut.

Aplikasi pembayaran oleh kasir merupakan salah satu contoh dari penggunaan *Database* yang terintegrasi oleh komputer. Dalam aplikasi komputer tersebut telah terekam data-data barang beserta harga, sehingga kasir dengan mudah mendapatkan data barang serta saat melakukan input barang, maka *Database* tersebut secara otomatis ter-*update*. Salah satu keuntungan dari *Database* adalah dapat mempersingkat waktu pekerjaan. Sebuah *Database* dapat menampung data yang kompleks yang dapat digunakan secara internal maupun eksternal. *Software* yang digunakan oleh suatu organisasi untuk mengendalikan *Database* disebut dengan *Database Manegement System* atau dapat disingkat dengan DBMS. DBMS merupakan suatu kumpulan program yang memudahkan pengguna untuk membuat dan mengelola *Database*. Data yang ada pada DBMS dapat dimanipulasi atau dikelola dengan menggunakan *query*, dimana dengan *query* tersebut pengguna dapat mendapatkan data spesifik yang dibutuhkan, serta dapat meng-*update* informasi dalam data-data tersebut. (Mardiono, Fil'aini, & Didin, 2019)

2.12 Visual Studio Code

Visual Studio Code (disingkat VSCode) adalah perangkat lunak penyunting kode-sumber buatan Microsoft untuk Linux, macOS, dan Windows. Visual Studio Code menyediakan fitur seperti penyorotan sintaksis, penyelesaian kode, kutipan kode, merefaktor kode, pengawakutuan, dan Git. Microsoft merilis sumber kode Visual Studio Code di repositori GitHub dengan lisensi MIT (Code-OSS), sedangkan biner yang dibangun oleh Microsoft tidak dirilis dengan lisensi MIT dan merupakan perangkat lunak berpaten. Visual Studio Code pertama kali diperkenalkan di tanggal 29 April 2015 oleh Microsoft di konferensi Build 2015.

Visual Studio Code adalah Software yang sangat ringan, namun kuat editor kode sumbernya yang berjalan dari desktop. Muncul dengan built-in dukungan untuk JavaScript, naskah dan Node.js dan memiliki array beragam ekstensi yang tersedia untuk bahasa lain, termasuk C ++, C #, Python, dan PHP. Hal ini didasarkan sekitar Github ini Elektron, yang merupakan versi cross-platform dari Atom

komponen kode-editing, berdasarkan JavaScript dan HTML5. Editor ini adalah fitur lengkap lingkungan pengembangan terpadu (IDE) dirancang untuk pengembang yang bekerja dengan teknologi cloud yang terbuka Microsoft. (Nurdiandto, 2021)

2.13 Penelitian Terdahulu

Penelitian ini dilaksanakan berdasarkan beberapa penelitian terdahulu. Beberapa penelitian terdahulu yang dijadikan acuan pada penelitian ini dapat dilihat pada tabel 2.5.

Tabel 1.7 Penelitian Terdahulu

No	Peneliti	Judul	Kesimpulan
1	Setiawan, Y.E. (2020)	Sistem Pendukung Pengambilan Keputusan Rekrutmen Guru Menggunakan Logika Fuzzy Tahani. BAREKENG : Jurnal Ilmu Matematika dan Terapan. Sinta 3.	Hasil penelitian ini telah menunjukkan bahwa implementasi logika fuzzy Tahani untuk sistem pendukung pengambilan keputusan perekrutan guru dapat digunakan. Karena sistem pendukung pengambilan keputusan dengan menggunakan logika fuzzy tahani adalah mempertimbangkan kriteria-kriteria secara menyeluruh. Jadi peneliti merekomendasikan untuk menerapkan fuzzy tahani sebagai sistem pendukung pengambilan keputusan yang melibatkan berbagai kriteria.

No	Peneliti	Judul	Kesimpulan
2	Asrori, M., & Falani, A. Z. (2019)	Implementasi Penentuan Pemberian Tunjangan Pendidik & Tenaga Kependidikan Berbasis Fuzzy Database Model Tahani. Insand Comtech : Information Science and Computer Technology Journal. Sinta 2.	Sistem pendukung keputusan yang dibuat berbasis web menggunakan bahasa pemrograman PHP dan database MySQL sehingga bisa diakses oleh lebih dari satu user. Sistem yang dibuat menggunakan metode fuzzy database model tahani sehingga bisa mengolah data dan relasinya untuk mendapatkan query. Pemberian tunjangan pada sistem ini berdasarkan usia, masa kerja, TMT sertifikasi dan gaji sesuai dengan ketentuan pihak sekolah.
3	Sari Rochman, E. M., et. al. (2020)	Pemilihan Tempat Wisata dengan Fuzzy Tahani Berbasis Aplikasi. Pekommas. Sinta 2.	Kesimpulan yang didapatkan dari sistem pendukung keputusan ini adalah dengan kriteria jumlah pengunjung, jarak dari pusat kota, dan review objek wisata, sistem mampu memberikan rekomendasi objek wisata.

No	Peneliti	Judul	Kesimpulan
			Berdasarkan uji fungsionalitas, semua fungsi sistem bekerja dengan baik.
4	Sianturi, F., et. al. (2020)	Penerapan Metode Fuzzy Model Tahani Dalam Sistem Pendukung Keputusan Kenaikan Jabatan. InfoTekJar: Jurnal Nasional Informatika dan Teknologi Jaringan. Sinta 3.	Proses kenaikan jabatan yang berjalan di Kantor Kelurahan Tualang Perbaungan selama ini masih bersifat manual dan memakan waktu yang lama untuk menentukan pegawai yang layak naik jabatan. Sistem pendukung keputusan kenaikan jabatan ini dapat menjadi solusi untuk memberikan rekomendasi dalam menentukan pegawai yang layak untuk naik jabatan secara efektif dan efisien.
5	Sikumbang, H., Indrianto, I., & Siregar, G. S. (2021).	Rancang Bangun Sistem Monitoring Pada Bendungan Dengan Menggunakan Metode Fuzzy Tahani Berbasis Mikrokontroler. PETIR : Jurnal Pengkajian dan Penerapan Teknik Informatika. Sinta 3.	Rancang bangun sistem monitoring ini ditujukan untuk petugas dan koordinator sup ciliwung bendung katulampa sebagai sistem monitoring ketinggian air pada bendungan. Untuk membuat atau

No	Peneliti	Judul	Kesimpulan
			membangun sistem monitoring pada penelitian ini dilakukan menggunakan metode prototypedan metode logika fuzzy tahani.
6	Yadnya, I. P. A. D. D., Gandhiadi, G. K., & Harini, L. P. I. (2019).	Aplikasi Fuzzy Model Tahani Dalam Penentuan Pemilihan Spesifikasi Tablet PC. Jurnal Teknik Informatika. Sinta 3.	Data tablet yang disimulasikan yaitu Tablet Samsung Galaxy Tab S memiliki nilai rekomendasi sebesar 0.7647 dengan kriteria Layar LCD Sedang dan Memory Internal Sedang dan Processor Tinggi yang ditampilkan pada tabel pencarian tablet sesuai dengan penghitungan nilai fire strength yang dilakukan secara manual. Sehingga hasil rancangan aplikasi fuzzy model Tahani telah mampu menghasilkan rekomendasi sesuai dengan kriteria yang diinputkan pengguna.
7	Waluyo, B., Taufiq, G., &	Model Fuzzy Tahani Untuk Pencarian Pelanggaran Pelanggan	Logika Fuzzy Tahani dapat diterapkan untuk pencarian pelanggaran

No	Peneliti	Judul	Kesimpulan
	Handrianto, Y. (2021).	Dalam Pemakaian Tenaga Listrik. SATIN-Sains dan Teknologi Informasi. Sinta 3.	pelanggan dalam pemakaian tenaga listrik dengan menggunakan beberapa kriteria tertentu. Dengan menggunakan logika Fuzzy Tahani, penilaian lebih akurat dan mengurangi subjektifitas dalam pemilihan golongan pelanggaran pelanggan serta mempermudah dalam memilih golongan pelanggaran.
8	Nasution, V. M., & Prakarsa, G. (2020).	Optimasi Produksi Barang Menggunakan Logika Fuzzy Metode Mamdani. Jurnal Media Informatika Budidarma. Sinta 3.	Optimasi produksi barang menggunakan Logika Fuzzy metode mamdani, maka dapat diambil kesimpulan bahwa dengan dibangunnya aplikasi ini dapat membantu perusahaan menentukan jumlah produksi yang sesuai dengan permintaan konsumen dan dengan mengaplikasikan Logika Fuzzy metode Mamdani sebagian besar permintaan

No	Peneliti	Judul	Kesimpulan
			di Salman Collection terpenuhi dan lebih optimal dibandingkan dengan sistem lama atau jumlah yang di produksi oleh perusahaan. Perhitungan dengan menggunakan aplikasi ini semakin lebih akurat dan tidak memakan waktu lama dan pihak lebih mudah dalam mengambil keputusan untuk jumlah barang yang akan diproduksi. Selain itu perancangan aplikasi ini mempermudah pihak perusahaan untuk memprediksi 5 tahun produksi barang secara optimal berdasarkan permintaan dari tahun-tahun sebelumnya.
9	Nurhayati, S., & Immanudin, I. (2019).	Penerapan Logika Fuzzy Mamdani Untuk Prediksi Pengadaan Peralatan Rumah Tangga Rumah Sakit. Komputika: Jurnal	Logika fuzzy mamdani dapat digunakan untuk prediksi pengadaan peralatan rumah tangga rumah sakit berdasarkan jumlah stok peralatan yang ada dan jumlah

No	Peneliti	Judul	Kesimpulan
		Sistem Komputer. Sinta 3.	kondisi peralatan yang rusak. Logika fuzzy juga dapat digunakan karena tingkat kebenaran mencapai 81,1 %.
10	Setia, B. (2019).	Penerapan Logika Fuzzy pada Sistem Cerdas. Jurnal Sistem Cerdas. Sinta 3.	Logika Fuzzy dapat diterapkan dengan baik dengan sistem cerdas dikarenakan logika Fuzzy mudah dimengerti, sangat fleksibel, memiliki toleransi terhadap data-data yang tidak tepat, mampu memodelkan fungsi-fungsi non linear yang sangat kompleks serta dapat membangun dan mengaplikasikan pengalaman para pakar secara langsung tanpa harus melalui proses pelatihan.