

BAB II

LANDASAN TEORI

2.1 Kajian Terhadap Penelitian Yang Terkait Sebelumnya

Berikut ulasan beberapa penelitian terkait yang menjadi referensi pada penelitian ini : Nasution dalam jurnal yang berjudul : “Penerapan *Weighted Sum Model* (WSM) Dalam Penentuan Peserta Jaminan Masyarakat” pada judul ini menggunakan metode *Weighted Sum Model* sebagai hasil penelitian bahwa metode *Weighted Sum Model* sangat cocok untuk memberikan hasil penentuan peserta jaminan kesehatan masyarakat dan metode WSM dapat membantu pengambil keputusan untuk menghasilkan suatu keputusan yang terbaik dari beberapa alternatif.

Berikut ulasan beberapa penelitian terkait yang menjadi referensi pada penelitian ini : Nurhayati dan Immanudin dalam jurnal yang berjudul : “Penerapan Logika *Fuzzy Mamdani* untuk Prediksi Pengadaan Peralatan Rumah Tangga Rumah Sakit” pada judul ini menggunakan metode *Fuzzy Mamdani* bertujuan untuk menganalisis apakah logika *Fuzzy Mamdani* dapat digunakan untuk memprediksi pengadaan peralatan rumah tangga rumah sakit.

2.2 *Decision Support System*

2.2.1 Definisi *Decision Support System* (DSS)

Menurut (Hamdhani et al., 2015) *Decision Support System* atau sistem pendukung keputusan adalah sebuah sistem yang mampu memberikan kemampuan dalam memecahkan suatu masalah dan memberikan solusi suatu masalah tersebut.

Decision Support System mampu memecahkan masalah dengan memberikan informasi atau usulan suatu keputusan, DSS dapat didefinisikan sebagai sebuah sistem yang mendukung kerja seorang manajer dalam memberikan keputusan pada suatu masalah.

Pembuatan keputusan merupakan fungsi utama seorang manajer atau administrasi. Kegiatan pembuatan keputusan meliputi pengidentifikasian masalah. Pencarian alternatif penyelesaian masalah, evaluasi dari alternatif-alternatif tersebut dan pemilihan alternatif keputusan yang terbaik. Kemampuan seorang manajer dalam membuat keputusan dapat ditingkatkan apabila mengetahui dan menguasai teori dan teknik pembuatan keputusan. Dengan peningkatan kemampuan manajer dalam pembuatan keputusan diharapkan dapat meningkatkan kualitas kerja dan efisiensi waktu dalam menentukan keputusan.

DSS atau sistem pendukung keputusan sebagai sistem yang digunakan untuk mendukung dan membantu pihak manajemen melakukan pembuatan keputusan pada kondisi semi terstruktur dan tidak terstruktur, Pada dasarnya konsep DSS hanyalah sebatas pada kegiatan membantu untuk melakukan penilaian pada suatu keputusan masalah. Konsep DSS ditandai dengan sistem interaktif berbasis komputer yang membantu mengambil keputusan memanfaatkan data dan model keputusan untuk menyelesaikan masalah-masalah dan dirancang untuk menunjang seluruh tahapan pembuatan keputusan, yang dimulai dari tahapan mengidentifikasi masalah, memilih data yang relevan, menentukan pendekatan yang digunakan dalam proses pembuatan keputusan sampai pada kegiatan mengevaluasi pemilihan alternatif.

2.2.2 Karakteristik *Decision Support System* (DSS)

Beberapa karakteristik yang membedakan DSS dengan sistem informasi lain adalah sebagai berikut:

1. DSS dirancang untuk membantu pengambilan keputusan dalam memecahkan masalah yang sifatnya semi terstruktur atau tidak terstruktur dengan menambahkan kebijaksanaan manusia dan informasi komputerisasi.
2. Proses pengolahannya, sistem pendukung keputusan mengkombinasikan penggunaan model-model analisi dengan teknik pemasukkan data konvensional serta fungsi-fungsi pencari atau pemeriksa informasi.
3. Sistem pendukung keputusan dapat digunakan atau dioperasikan dengan mudah oleh orang-orang yang tidak memiliki dasar kemampuan pengoperasian komputer.
4. Sistem pendukung keputusan dirancang dengan menekankan pada aspek fleksibilitas serta kemampuan adaptasi yang tinggi sehingga mudah disesuaikan dengan berbagai perubahan lingkungan yang terjadi dan kebutuhan *user*.

2.3 *Weighted sum model* (WSM)

Menurut (Nasution et al., 2017) Metode *Weighted Sum Model* (WSM) merupakan metode yang sangat umum, dan banyak diterapkan untuk membantu pengambil keputusan dalam mengambil suatu keputusan. WSM merupakan salah satu metode yang paling sederhana dan mudah dipahami penerapannya.

Metode *Weighted Sum Model* (WSM) merupakan bagian dalam metode MCDM (*Multi-Criteria Decision Making*) dalam mengevaluasi nilai pada setiap

alternatif. Dan metode ini sangat umum dan banyak diterapkan untuk membantu pengambil keputusan dalam mengambil suatu keputusan. Ini adalah metode yang sering digunakan pada permasalahan dimensi tunggal.

Jika terdapat m alternatif dan n kriteria, maka alternatif terbaik dapat dirumuskan sebagai berikut:

$$A_i^{WSM} - score = \sum_{j=1}^n w_j X_{ij}, \text{ for } i = 1, 2, 3, \dots \dots \dots (1)$$

Dimana:

n = jumlah kriteria

w_j = bobot dari setiap kriteria

X_{ij} = nilai matrik x

Nilai A_i yang paling besar merupakan alternatif yang terpilih.

2.4 Logika *Fuzzy*

Menurut (Hosseinzadeh et al. 2011) “Logika *Fuzzy* adalah konsep yang tepat untuk menangani masalah *nonlinear*, waktu yang beravariasi dan sistem adaptif. Logika *Fuzzy* ini memungkinkan penggunaan nilai-nilai linguistik dari variabel dan hubungan tidak tepat untuk perilaku sistem modeling. Logika *Fuzzy* sering digunakan pada sistem cerdas dalam memilah proses untuk mendeteksi cacat dalam penerapannya”.

Menurut (Siddique 2014) “Salah satu aplikasi yang paling terkenal dari logika *Fuzzy* adalah *Fuzzy inference system* (FIS). Ada tiga jenis dasar FIS yang

telah banyak digunakan dalam berbagai aplikasi kontrol, yaitu FIS *Mamdani*, *Sugeno*, dan *Tsukamoto*. Perbedaan antara tiga FIS ini terletak pada konsekuensi dari aturan *Fuzzy* mereka, agregasi dan prosedur *defuzzifikasi*”.

Logika *Fuzzy* adalah cabang dari sistem kecerdasan buatan (*Artificial Intelligent*) yang mengemulasi kemampuan manusia dalam berfikir ke dalam bentuk algoritma yang kemudian dijalankan oleh mesin. Algoritma ini digunakan dalam berbagai aplikasi pemrosesan data yang tidak dapat direpresentasikan dalam bentuk biner. Logika *fuzzy* menginterpretasikan statement yang samar menjadi sebuah pengertian yang logis.

Metode-metode untuk strategi ini adalah:

1. Metode keanggotaan maximum (*maxmembership*)
2. Metode pusat luas (*Center of Area, CoA*).
3. Metode keanggotaan maksimum rata-rata (*Meanmax Membership* atau *Middle-of-Maxima*)

Ada beberapa alasan mengapa orang menggunakan logika *Fuzzy*, antar lain (Kusumadewi and Purnomo 2010):

1. Konsep logika *Fuzzy* mudah dimengerti. Konsep matematis yang mendasari penalaran *Fuzzy* sangat sederhana dan mudah dimengerti.
2. Logika *Fuzzy* sangat fleksibel.
3. Logika *Fuzzy* memiliki toleransi terhadap data yang tidak tepat.
4. Logika *Fuzzy* mampu memodelkan fungsi nonlinier yang sangat kompleks.
5. Logika *Fuzzy* dapat membangun dan mengaplikasikan pengalaman para pakar secara langsung tanpa harus melalui proses pelatihan.

6. Logika *Fuzzy* dapat bekerjasama dengan teknik kendali secara konvensional.
7. Logika *Fuzzy* didasarkan pada bahasa alami.

Ada beberapa tahap yang harus diketahui untuk menghitung nilai di dalam logika *Fuzzy*, yaitu:

1. Fungsi keanggotaan
2. *Fuzzyfikasi*
3. Operasi himpunan *fuzzy*
4. *Deffuzifikasi*

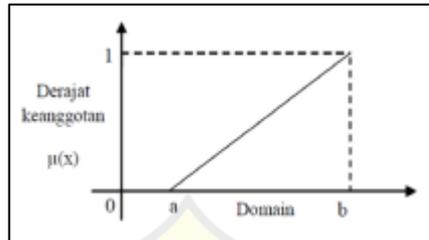
2.4.1 Fungsi Keanggotaan

Menurut (Hosseinzadeh et al. 2011) “Fungsi keanggotaan didefinisikan sebagai representasi grafis dari besarnya keikutsertaan setiap *input* (masukan). Fungsi keanggotaan ini menghubungkan bobot dengan masing-masing *input* yang diproses, mendefinisikan tumpang tindih di antara *input*, dan akhirnya menentukan respon *output* (keluaran). Aturan menggunakan nilai keanggotaan *input* sebagai *factor* pembobotan untuk menentukan pengaruhnya terhadap set *output Fuzzy* kesimpulan hasil akhir”.

Menurut (Ross 2010) Jika fungsi keanggotaan tidak sesuai, maka hasil yang didapat akan jauh berbeda dari diharapkan. Fungsi keanggotaan ini merupakan tahap awal dan paling penting untuk menuju tahap *fuzzifikasi*. *Fuzzifikasi* merupakan tahap dimana nilai *inputan* yang berupa nilai crisp (tegas) diubah menjadi nilai *Fuzzy*.

A. Representasi Linear Naik

Pada representasi linear, pemetaan *input* ke derajat keanggotaannya digambarkan sebagai sebuah garis lurus sebagai berikut:



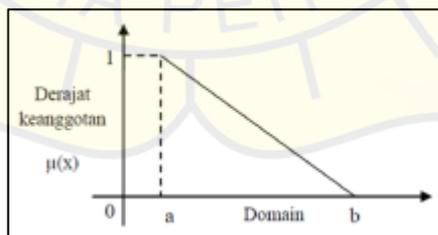
Gambar 2. 1 Representasi linear naik

Fungsi keanggotaan representasi linear naik yang dapat dilihat pada persamaan (1):

$$\mu[x] = \begin{cases} 0; & x \leq a \\ \frac{x - a}{b - a}; & a \leq x \leq b \\ 1; & x \geq b \end{cases} \quad (1)$$

B. Representasi Linear Turun

Representasi ini merupakan kebalikan dari representasi linear naik. sebagai berikut:



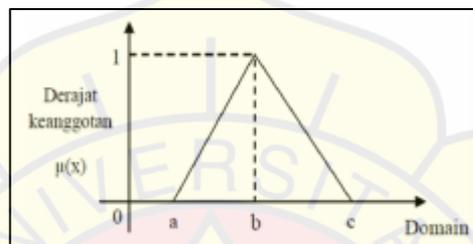
Gambar 2. 2 Representasi linear turun

Fungsi keanggotaan representasi linear turun yang dapat dilihat pada persamaan (2):

$$\mu[x] = \begin{cases} 1; & x \leq a \\ \frac{b-x}{b-a}; & a \leq x \leq b \\ 0; & x \geq b \end{cases} \quad (2)$$

C. Representasi Kurva Segitiga

Representasi kurva segitiga pada dasarnya merupakan gabungan antara linear naik dan linear turun sebagai berikut:



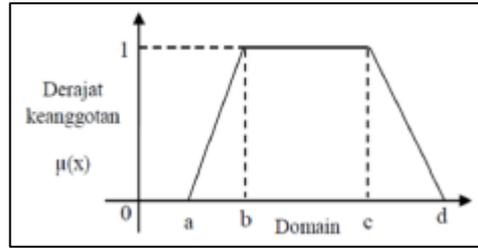
Gambar 2. 3 Representasi kurva segitiga

Fungsi keanggotaan representasi kurva segitiga yang dapat dilihat pada persamaan (3):

$$\mu[x] = \begin{cases} 0; & x \leq a \text{ or } x \geq c \\ \frac{x-a}{b-a}; & a \leq x \leq b \\ \frac{c-x}{c-b}; & b \leq x \leq c \end{cases} \quad (3)$$

D. Representasi Kurva Trapesium

Representasi kurva trapesium memiliki domain lebih luas dari representasi kurva segitiga sebagai berikut:



Gambar 2. 4 Representasi kurva trapesium

Fungsi keanggotaan representasi kurva trapesium yang dapat dilihat pada persamaan (4):

$$\mu[x] = \begin{cases} 0; & x \leq a \text{ or } x \geq d \\ \frac{x-a}{b-a}; & a \leq x \leq b \\ 1; & b \leq x \leq c \\ \frac{d-x}{d-c}; & c \leq x \leq d \end{cases} \quad (4)$$

2.4.2 Fuzzy mamdani

Metode *mamdani* atau yang biasa disebut dengan metode *Max-Min* diperkenalkan oleh *Ebrahim Mamdani* pada tahun 1975 (Setiawan & Subanar 2010). Langkah-langkah dalam menggunakan *Fuzzy Mamdani* untuk proses klasifikasi adalah:

1. Melakukan fungsi implikasi.
2. Melakukan komposisi aturan.
3. Penegasan (*defuzzy*).

2.5 Unified Modelling Language (UML)

“UML (*Unified Modelling Language*) adalah salah standar bahasa yang banyak digunakan didunia industri untuk mendefinisikan *Requirement*, membuat

analisis dan desain, serta menggambarkan arsitektur dalam pemograman berorientasi objek” Rosa dan Shalahudin (2018:137).

Unified Modeling Language (UML) muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. UML merupakan bahasa visual untuk pemodelan dan komunikasi dalam pemodelan dan teks-teks pendukung (Sukamto & Shalahuddin, 2018).

Tujuan Penggunaan UML adalah:

- a. Memberikan bahasa pemodelan yang bebas dari berbagai Bahasa pemrograman dan proses rekayasa.
- b. Menyatukan praktek-praktek terbaik yang terdapat dalam pemodelan.
- c. Memberikan model yang siap pakai, Bahasa pemodelan visual yang ekspresif untuk mengembangkan dan saling menukar model dengan mudah dan dimengerti secara umum.

UML bisa juga berfungsi sebagai sebuah (*blue print*) cetak biru karena sangat lengkap dan detail. Dengan cetak biru ini maka akan bias diketahui informasi secara detail tentang koding program atau bahkan membaca program dan menginterpretasikan kembali ke dalam bentuk diagram (*reverse engineering*).

2.5.1 Use case Diagram

Use Case Diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih *actor* dengan *system* informasi yang akan dibuat. Secara kasar *use case*

digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah *system* informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut *actor* dan *use case*:

1. *Actor* merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi, jadi meskipun simbol dari *actor* adalah gambar orang, tapi *actor* belum tentu merupakan orang.
2. *Use Case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antara unit atau *actor*.

2.5.2 Activity Diagram

Menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah *system* atau proses bisnis. Yang perlu diperlihatkan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan *actor*, jadi aktivitas yang dapat dilakukan oleh sistem. Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut:

1. Rancangan proses bisnis dimana setiap urusan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
2. Urutan atau pengelompokkan tampilan dari sistem atau user *Interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
3. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujiannya yang dimiliki kelas yang diinstansinya menjadi objek itu.

2.5.3 *Sequence Diagram*

Sequence Diagram menggambarkan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antar objek.

2.5.4 *Deployment Diagram*

Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan (*run-time*). Memuat simpul-simpul beserta komponen-komponen yang ada didalamnya. *Deployment Diagram* berhubungan erat dengan diagram komponen dimana diagram ini memuat satu atau lebih komponen-komponen. Diagram ini sangat berguna saat aplikasi kita berlaku sebagai aplikasi yang dijalankan pada banyak mesin (*distributing computing*).



TEKNOLOGI INFORMASI

UNIVERSITAS DARMA PERSADA