

BAB II

LANDASAN TEORI

2.1 Sistem Pendukung Keputusan

Konsep SPK (Sistem Pendukung keputusan) pertama dinyatakan oleh Michael S. Scott Morton pada awal tahun 1970an (1971) dengan artian Management Decision System. “Sistem Pendukung Keputusan (SPK) atau *Decision Support System* (DSS) adalah salah satu komponen sistem yang sering digunakan untuk mendukung pengambilan keputusan jangka pendek, menengah, dan panjang.”.(Umami et al., n.d.) Sejumlah dalam lingkungan yang terorganisasi maupun tidak terstruktur, di mana pengambilan keputusan tidak begitu jelas metode ini digunakan untuk mendukung pengambilan keputusan.

“Menurut Prof. Dr. Prajudi Atmosudirjo, SH keputusan kesimpulan dari tahap kognitif suatu permasalahan atau isu dengan pertanyaan mengenai langkah apa yang mesti diambil untuk menyelesaikan situasi tersebut dengan memilih opsi yang berbeda.”.(Ridwan & Rahman, 2019)

Menurut Kusriani dalam Rinianty dan Sukardi (2018:49), “Sistem informasi yang menawarkan manipulasi data, pemodelan, dan informasi disebut sistem pembuat keputusan. Pembuatan keputusan semi-terstruktur dan tak terstruktur dapat dibantu dengan penggunaan sistem pendukung keputusan”.

Berdasarkan beberapa pengertian diatas dapat diartikan bahwa SPK bukanlah alat pengganti untuk pengambilan keputusan sebaliknya ini adalah alat yang memberi para manajer dan pembuat keputusan lainnya pertimbangan lebih ketika membuat keputusan sehingga memungkinkan mereka untuk memecahkan berbagai situasi terorganisasi semi-terstruktur dan tidak terstruktur.

2.2 Komponen-komponen Sistem Pendukung Keputusan

Suatu SPK memiliki tiga sistem utama yang memastikan kemampuan teknik SPK (Arna farizah ,2022), yaitu sebagai berikut :

1. Sistem manajemen basis data (Database Management Subsystem) perbedaan diantara murni data untuk SPK dan bukan-SPK. Sebagai hasil dari ketergantungan yang besar dari proses pengambilan keputusan pada sumber data eksternal seperti statistik ekonomi khususnya pada tingkat manajemen puncak DSS memiliki sumber data yang "lebih kaya" daripada non-DSS, yang mengharuskan data berasal dari sumber internal dan eksternal. Langkah kedua melibatkan pengambilan data dari aslinya yang besar. DSS memerlukan prosedur ekstraksi dan sistem manajemen basis data (DBMS) dengan fleksibilitas administratif yang memadai untuk memungkinkan penambahan dan penghapusan yang cepat.
2. Sistem manajemen model (Model Management Subsystem) salah satu alat mengintegrasikan data dan model-model keputusan. Hal ini dapat dicapai dengan mengintegrasikan dan memfasilitasi komunikasi lintas model serta dengan memperkenalkan model keputusan klaster yang menggunakan basis data sebagai mekanisme komunikasi ke sistem informasi.

3. Sistem perangkat lunak penyelenggara dialog (Dialog Generation and Management Subsystem) pengguna terminal dan sistem perangkat lunak merupakan bagian dari sistem dialog. Terdapat tiga divisi dalam subsistem dialog yaitu:

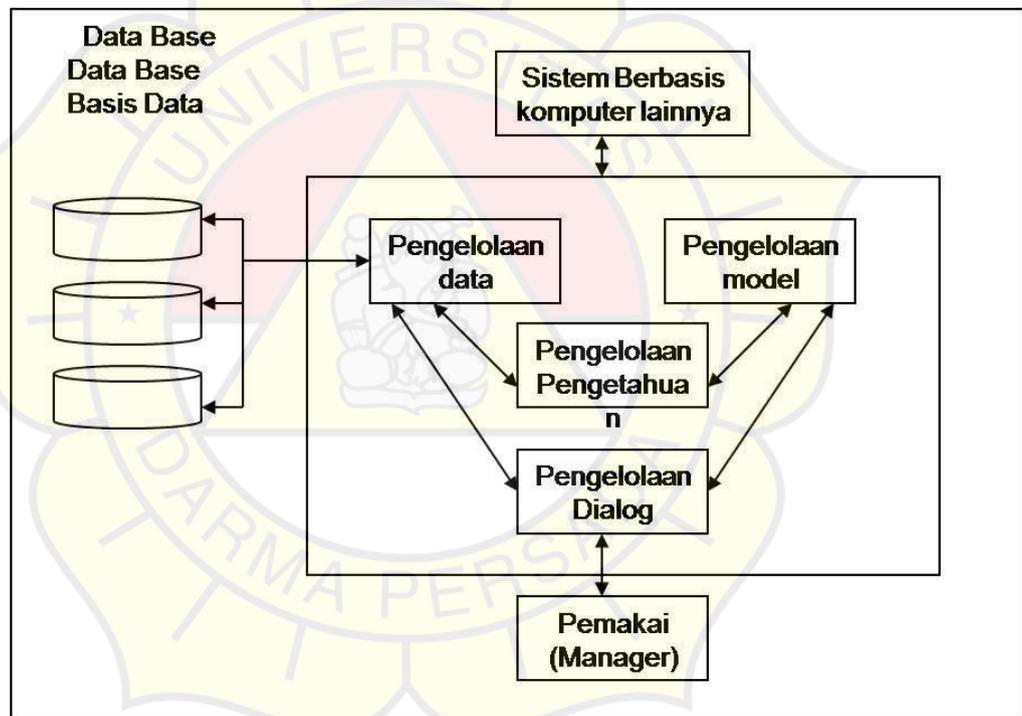
a. Bahasa aksi membahas tentang yang dapat dilakukan pengguna saat berinteraksi dengan sistem. Keyboard, layar sentuh, joystick, perintah suara, dan opsi lainnya disertakan dalam ini.

b. Bahasa tampilan atau persentasi pengetahuan penting bagi pengguna. Pilihan termasuk printer, layar, tampilan, grafik, warna, plotter, keluaran suara, dan banyak lagi disertakan dalam bahasa tampilan.

c. Bahasa pengetahuan membahas yang mesti dipahami oleh pengguna agar sistem dapat berfungsi dengan baik. Pemikiran pengguna kartu referensi atau panduan, manual, dan sebagainya semuanya dapat mencakup basis pengetahuan.

4.) Sistem manajemen berbasis pengetahuan (Knowledge Management Subsystem) subsistem ini dapat menuju setiap sistem lain atau berfungsi dibagian yang berdiri sendiri. Lebih jauh lagi sistem ini dapat menawarkan kecerdasan untuk memperluas pemahaman

kepada si pengambil keputusan. DBMS, MBMS, dan antarmuka pengguna adalah tiga komponen utama yang harus dimiliki sistem pendukung keputusan. Namun komponen manajemen berbasis pengetahuan tidak diperlukan. Namun sistem pendekatan berbasis pengetahuan memiliki beberapa keunggulan karena memberikan inteligensi bagi 3 (tiga) komponen utama tersebut. Pengguna dapat dilihat sebagai salah satu komponen DSS di samping empat komponen lainnya seperti halnya semua sistem informasi manajemen.



Gambar 2. 1 Model Konseptual SPK(Arbian, 2017)

2.1 Karakteristik Dan Kemampuan Sistem Pendukung Keputusan

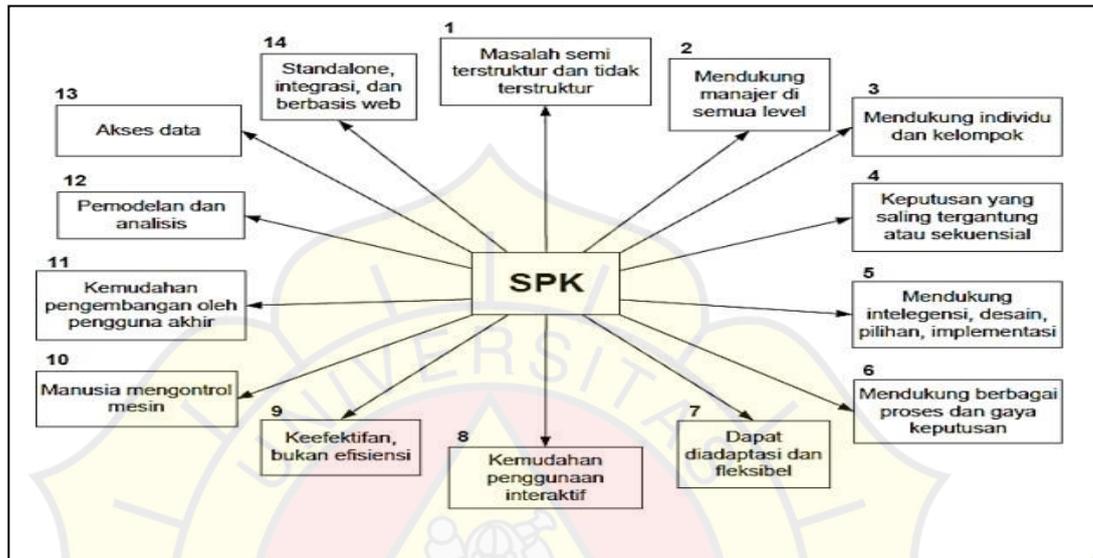
Karakteristik dan kemampuan sebuah sistem pendukung keputusan (Pratiwi, 2020) sebagai berikut :

1. Sistem pendukung keputusan memberikan dukungan untuk para pembuat keputusan dengan menggabungkan data teknologi dan penilaian manusia terutama dalam skenario semi-terstruktur dan tidak terstruktur.
2. Menyediakan dukungan untuk tingkat manajer mulai dari manajer sampai eksekutif.
3. Menyediakan situasi yang kurang terstruktur yang melibatkan sekelompok orang memerlukan partisipasi banyak orang dari berbagai area.
4. Sistem pendukung keputusan memberi dukungan independen atau keputusan yang berjalan.
5. Semua tahapan proses pengambilan keputusan didukung oleh sistem pendukung keputusan: intelijen, desain, pilihan, dan implementasi.
6. Sistem pendukung keputusan membuat banyak proses dan gaya pembuatan keputusan.
7. Sistem pendukung keputusan bersifat adaptif terhadap waktu artinya para pengambil keputusan semestinya harus menanggapi keadaan yang berubah dan menyesuaikan sistem pembuat keputusan agar cukup fleksibel bagi konsumen untuk menambah, menghapus, menggabungkan, memodifikasi, dan mengatur ke elemen dasarnya.
8. Sistem pendukung keputusan seharusnya mudah saat berinteraksi dengan mesin dan pengguna merasa nyaman berkat fitur-fitur seperti kemudahan penggunaan, fleksibilitas, kemampuan penggunaan grafis tinggi dan bahasa yang mudah dipahami yang semuanya berkontribusi pada

peningkatan kemanjuran sistem pembuat keputusan.

9. Keputusan memberikan efektifitas pembuatan pilihan pada hal ketepatan waktu dan kualitas dibandingkan dengan biaya pengambilan keputusan atau waktu komputasi.
10. Pembuat keputusan dapat mengontrol tahapan-tahapan pembuatan keputusan yang dibuat selama tahap intelijen pilihan serta implementasi serta keputusan yang dibuat dengan bantuan sistem pembuat keputusan dimaksudkan agar membantu para pemilih keputusan dan tidak untuk merubah mereka.
11. Memungkinkan pengguna membuat sistem dasar mereka sendiri. Pakar sistem informasi dapat membantu dalam pembangunan sistem yang besar.
12. Sistem pendukung keputusan keputusan memilih model-model sederhana atau buatan konumen untuk mengevaluasi skenario yang melibatkan keputusan. Kemampuan untuk membuat model memungkinkan pengujian berbagai strategi dalam berbagai pengaturan. Akses ke berbagai sumber data, format, dan jenis termasuk kedalaman sistem informasi geografis dalam orientasi objek didukung oleh sistem pendukung keputusan.

4. Sistem pendukung keputusan mengarah bahkan pada tingkat pembelajaran tingkat lanjut pada sistem ini memiliki komponen pengetahuan yang memungkinkannya memberikan jawaban yang efektif dan efisien terhadap berbagai situasi rumit.



13. **Gambar 2. 2** Karakteristik dan Kapabilitas Spk(Pratiwi, 2020)

2.2 Langkah-Langkah Pengambilan Keputusan

Proses pendukung keputusan dimulai dengan fase cerdas, yang menguji realitas dan mengidentifikasi masalah. Berikutnya adalah fase desain berbasis model yang menjelaskan suatu sistem berdasarkan norma dan standar yang dikumpulkan untuk menilai alternatif tindakan yang telah ditemukan. Langkah selanjutnya adalah memilih, yang mencakup target penyelesaian model. Tahap terakhir adalah implementasi, saat efektivitas sistem dalam menyelesaikan masalah saat ini dinilai. (Turban, 1988).

2.3 Tujuan Sistem Pendukung Keputusan

Menurut Nofriansyah (2014:4), sistem pendukung keputusan memiliki beberapa tujuan, yaitu:

- 1.) Menolong dalam membuat pemilihan tentang masalah terstruktur.
- 2.) Memberikan dukungan atas perhatian atasan dan tidak dimaksudkan untuk menggubah peran atasan tersebut.
- 3.) Memilih untuk membuat keputusan yang lebih efektif daripada yang lebih efisien.
- 4.) Kecepatan pengambil keputusan dapat memproses data dalam jumlah besar dengan cepat dan terjangkau karena kecepatan komputasi komputer.
- 5.) Peningkatan produktivitas membentuk kelompok pengambil keputusan terutama kelompok yang beranggotakan para ahli bisa memakan biaya yang sangat besar.

2.4 Karakteristik dan Nilai Guna Sistem

Menurut Turban (1995) beberapa karakteristik yang membedakannya adalah:

1. Sistem pendukung keputusan dibuat dalam pengambilan keputusan saat menangani tantangan tidak sesuai dengan aturannya.
2. Dalam proses pembuatan sistem pendukung keputusan mengintegrasikan metode entri data tradisional fungsi pencarian informasi dan teknik analitis.
3. Sistem pendukung keputusan dibuat sedemikian rupa sehingga dapat cukup digunakan bagi seseorang yang tidak memiliki pengetahuan komputer yang luas. Karena itu model interaktif merupakan metode yang sering dipakai

4. Sistem pendukung keputusan tingkat adaptasi dan fleksibilitas yang tinggi diprioritaskan dalam desainnya agar mudah disesuaikan dengan perubahan lingkungan dan kebutuhan pengguna.

2.5 Pemrograman Aplikasi

2.7.1 HTML

Rio Astamal (dalam Masykur, 2016:95) menyatakan HTML (*Hypertext Markup Language*) sekelompok simbol atau tag yang ditulis di berkas yang seharusnya ditampilkan oleh peramban web di suatu halaman. Tag ini memberi instruksi kepada peramban tentang cara menampilkan seluruh halaman web kepada pengguna.

2.7.2 JavaScript

Bahasa skrip yang dikenal sebagai JavaScript pertama kali diciptakan oleh Netscape. JavaScript dapat digunakan untuk meningkatkan kecerdasan tampilan web. Satu-satunya instrumen yang dibutuhkan oleh pengguna untuk menjalankan javascript adalah browser dengan fungsionalitas javascript yang aktif. Dimungkinkan untuk menggabungkan pembuatan Javascript ke dalam halaman HTML yang menunjukkan bahwa Javascript adalah bahasa pemrograman sisi klien. Mengingat bahwa proses browser pengguna memengaruhi proses Javascript:

1. <html>
2. <body>
3. <script language =”javascript”>
4. Document write(“ini dihasilkan oleh javascript”);
5. </script>

6. `</body>`

7. `</html>`

2.7.3 PHP

Hypertext Preprocessor atau php merupakan script open source yang sering dipakai oleh programmer pada mengembangkan website. Rasmus Lerdorf adalah pembuat pertama kali pada tahun 1994. Php mempunyai aturan yang ditulis dengan tag : `<?php & ?>` atau `<? &?>` atau `<script language="php"> & </script>` atau `<% & %>`. Titik koma (;) akan digunakan untuk mengakhiri setiap kalimat (perintah). Saat membuat pengenal secara manual (menggunakan variabel, konstanta, fungsi, dll.) nama pengenal harus peka huruf besar/kecil namun pengenal bawaan PHP tidak peka huruf besar/kecil menurut Solichin (dalam Masykur, 2016:95). Cara kerja php sebenarnya sama dengan Python dan JavaScript, yang membedakan adalah Python untuk komunikasi sisi server (backend). Sementara JavaScript bisa digunakan frontend dan backend.

2.7.4 Bootstrap

Bootstrap merupakan front-end framework sangat bagus dan membantu dalam pembuatan situs web. Bootstrap menawarkan HTML, CSS, dan Javascript yang digunakan dan dikembangkan. Sebuah kerangka kerja agar membuat desain web responsif disebut Bootstrap.

Ini berarti bahwa ukuran layar browser baik desktop, tablet, atau perangkat seluler akan disesuaikan dengan tampilan web yang dibuat oleh bootstrap. Kita memiliki kemampuan untuk mengaktifkan atau menonaktifkan fitur ini sesuai keinginan.

Sehingga kita dapat membuat web hanya tampilan desktop saja dan situs yang kami buat tidak menyesuaikan tampilannya agar sesuai dengan layar saat ditampilkan oleh browser seluler. Kami dapat membuat situs web statis dan dinamis dengan bootstrap .(Effendy, 2016:24).

Keunggulan dari menggunakan Bootstrap adalah kerangka ini dibangun menggunakan Less, teknik CSS dasar dan mudah digunakan. Selain itu, less lebih fleksibel serta canggih daripada CSS standar. Pengembang dapat mengakses fungsi dan data tentang warna, variabel, dan aktivitas penggunaan dengan cepat menggunakan Less.

2.7.5 Database

Menurut Mukhamad Masrur (2016), Basis data adalah kumpulan berkas data terkait yang telah diatur sedemikian rupa sehingga mudah diakses dan dimanipulasi. Lingkungan sistem basis data menekankan pada data yang tidak terkait dengan aplikasi yang menggunakannya.

Penggunaan database pada komputer dilakukan dengan menggunakan table-table. Untuk membedakan satu set data dari yang lain data tersebut tetap dibagi menjadi beberapa bagian dalam tabel-tabel ini. Kategori data diperlukan dalam tabel basis data agar dapat digunakan sebagai kunci untuk mengidentifikasi data di dalam tabel tersebut. Data kunci harus berbeda dari satu data ke data lainnya. Data kunci sering disebut dengan *Primary Key*.

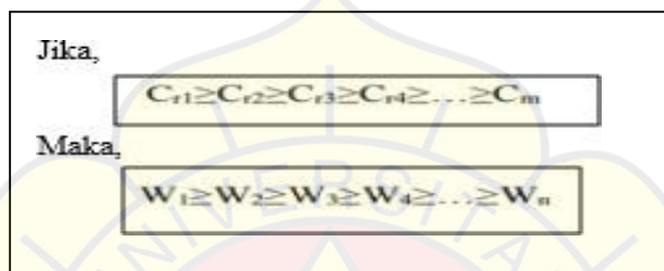
2.7.6 MySQL

MySQL adalah suatu sistem manajemen database bersifat terbuka (*Open Source*). Maksudnya terbuka siapa pun dapat mengunduh dan menggunakan MySQL secara gratis termasuk versi biner dan kode program aslinya. Para pencipta MySQL Michael "Monty" Widenius, David Axmark, dan Allan Larson, memulai sejarah basis data tersebut pada tahun 1995. Di Swedia mereka bertiga memulai bisnis bernama Mysql. (Lukman,2015:148). Perangkat lunak basis data ini banyak digunakan karena banyak manfaatnya oleh para profesional untuk membuat sebuah proyek. Banyak program komputer yang dibangun dalam berbagai bahasa pemrograman dapat mengakses basis data MySQL berkat tersedianya fasilitas API (Application Programming Interface) yang dimiliki oleh MySQL.

2.6 Algoritma Sistem

2.8.1 Metode *Rank Order Centroid (ROC)*

Ide dasar metode ini adalah mengalokasikan bobot sesuai dengan signifikansi kriteria. (Khalida dkk., 2021) terkadang dikonstruksi sebagai berikut: "Kriteria 1 lebih penting daripada kriteria 2, yang mana lebih penting daripada kriteria 3," dan seterusnya, hingga kriteria ke-n..



Gambar 2. 3 Rumus ROC 1 (Al Farosa et al., 2022)

Gambar 2.3 menjelaskan bahwa jika C1 atau kriteria lebih penting dari C2 maka dari itu W1 atau bobot lebih penting dari W2 tersebut, jika k memiliki nilai tinggi dari banyak kriteria, maka:

$$W_1 = \frac{1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{k}}{k}$$
$$W_2 = \frac{0 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{k}}{k}$$
$$W_3 = \frac{0 + 0 + \frac{1}{3} + \dots + \frac{1}{k}}{k}$$
$$W_k = \frac{0 + \dots + 0 + \frac{1}{k}}{k}$$

Gambar 2. 4 Rumus ROC 2 (Al Farosa et al., 2022)

Gambar 2.4 menjelaskan bahwa jika k atau kriteria adalah jumlah dari kriteria bobotnya maka tergambar pada rumus diatas.

2.8.2 Metode *Weight Aggregated Sum Product Assesment (WASPAS)*

WASPAS metode yang dapat meminimalkan kesalahan atau memaksimalkan estimasi untuk memilih nilai terbesar dan terendah. Oleh karena itu (WASPAS) adalah gabungan khusus dari teknik MCDM yang sudah mapan khususnya *Weighted Product Model (WPM)* dan *Weighted Sum Model (WSM)*." Pada awalnya elemen matriks keputusan harus dinormalisasi secara linier menggunakan dua persamaan.”(Marbun et al., 2018)

Tujuan utama pendekatan MCDM adalah memilih pilihan terbaik di antara beberapa opsi ketika ada persyaratan yang saling bersaing. Dalam makalah ini ada upaya untuk memberikan bukti tentang kesesuaian dan penerapan teknik MCDM yang baru-baru ini yaitu metode penilaian jumlah agregat berbobot. (Faiz Zamzah Amirillah Hasibuan & Sianturi, 2022)

Langkah proses perhitungan menerapkan metode WASPAS, yaitu:

1. Buat sebuah matriks keputusan

Matriks keputusan merupakan struktur dari sebuah keputusan hal ini sering ditunjukkan dalam tabel perbandingan antara beberapa pilihan untuk membuat keputusan yang telah ditetapkan untuk melakukan pemilihan alternatif mana yang sudah atau pun belum memenuhi kriteria.

$$X = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{11} & \dots & x_{2n} \\ \dots & \dots & \dots & \dots \\ x_{m1} & x_{m1} & \dots & x_{mn} \end{bmatrix}$$

Gambar 2. 5 Rumus Waspas 1(Damanik & Lase, 2019)

2. Melakukan normalisasi terhadap matrik x

Pada tahap ini jika nilai maksimal dan minimal sudah ditentukan maka persamaan menjadi sebagai berikut :

$$\begin{array}{l} \text{Kriteria Benefit} \\ \bar{x}_{ij} = \frac{x_{ij}}{\max_i x_{ij}} \dots \\ \text{Kriteria Cost} \\ \bar{x}_{ij} = \frac{\min_i x_{ij}}{x_{ij}} \dots \end{array}$$

Gambar 2. 6 Rumus Waspas 2(Damanik & Lase, 2019)

3. Menghitung nilai Qi

Jika sudah selesai menentukan nilai nilai kedua dilanjutkan ketahap pengambilan keputusan yang dimana memakai rumus yang dibawah ini

$$Q_i = 0,5 \sum_{j=1}^n X_{ij}w_j + 0,5 \prod_{j=1}^n (x_{ij})^{w_j} \dots$$

Dimana :

- Q_i =Nilai dari Q ke i
- $X_{ij}w_j$ = Perkalian nilai X_{ij} dengan bobot (w_j)
- 0,5 = Ketetapan

Gambar 2. 7 Rumus Waspas 3(Damanik & Lase, 2019)

Artinya:

i = Kriteria keuntungan (*Benefit*)

i = Kriteria *Cost* (Biaya)

X_{ij} = Nilai atribut yang dimiliki dari setiap kriteria

Maxi (X_{ij}) = Nilai besar dari kriteria

Mini (X_{ij}) = Nilai kecil dari kriteria

Benefit = Nilai yang besar diambil

Cost = Nilai yang kecil diambil

Dimana: Q_i = Nilai dari Q ke I

X_{ijw} = Perkalian nilai X_{ij} dengan bobot (W)

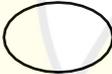
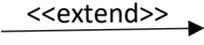
0.5 = Ketetapan pada bilangan

2.7 Pemodelan UML

“*Unified Modeling Language* (UML) adalah bahasa pemodelan grafis untuk mendefinisikan, membuat, dan mendokumentasikan semua artefak sistem perangkat lunak bukan suatu metode. Dengan menggunakan sistem ini seseorang berharap agar menentukan komponen apa saja yang berada dalam ruang lingkup sistem yang sedang dibahas dan bagaimana satu sistem berhubungan dengan sistem lain baik eksternal maupun subsistemik. “Unified Modeling Language (UML)” adalah pemrograman berbasis grafis yang digunakan untuk menggambarkan dan menampilkan sistem pengembangan perangkat lunak berorientasi objek.” Saya dapat menyimpulkan dari penjelasan sebelumnya bahwa sistem perangkat lunak ditentukan dan divisualisasikan menggunakan Unified Modeling Language (UML) bahasa pemodelan grafis.

2.9.1 Use Case Diagram

Diagram use case merupakan model untuk keseharian (behavior) sistem informasi yang sedang dibangun. Use case menjelaskan bagaimana satu atau beberapa aktor berinteraksi dengan sistem informasi yang akan dikembangkan. Use case secara umum digunakan untuk menentukan fungsi apa saja yang ada dalam suatu sistem dan siapa yang berwenang untuk menggunakannya.” (Yunahar, 2018). Berikut adalah simbol-simbol yang ada pada diagram use case:

Simbol	Keterangan
	Aktor: Mewakili peran orang, yang lain, atau alat untuk berkomunikasi dengan use case
	Use case : Abstraksi dan interaksi antara sistem dan aktor
	Association : Abstraksi dari penghubung antara actor untuk dapat berpartisipasi dengan use case
	Menunjukkan bahwa suatu use case seluruhnya merupakan fungsionalitas dari use case lainnya.
	Menunjukkan suatu use case merupakan tambahan fungsional dari use case lainnya jika suatu kondisi tersebut terpenuhi

Gambar 2. 8 Simbol Use Case Diagram (Yunahar,2018)

2.9.2 Activity Diagram

Diagram aktivitas menunjukkan cara kerja sistem atau proses bisnis dan apa saja tindakannya. Penting untuk diingat bahwa diagram aktivitas menggambarkan tindakan sistem, bukan tindakan aktor, artinya sistem mampu menjalankan aktivitas.” (Yunahar, 2018). Simbol-simbol yang digunakan dalam activity diagram sebagai berikut:

Simbol	Keterangan
	Simbol start untuk menjalankan awal dari suatu proses
	Simbol stop untuk menyatakan akhir dari suatu proses
	Simbol decision digunakan untuk menyatakan kondisi dari suatu proses
	Simbol action menyatakan aksi yang dilakukan dalam suatu aritektur sistem

Gambar 2. 9 Simbol Activity Diagram(Yunahar,2018)

2.9.3 Sequence Diagram

Sequence diagram adalah salah satu alat yang paling banyak digunakan untuk mengembangkan sistem informasi berorientasi objek

adalah diagram sekuens. dan menunjukkan bagaimana object berinteraksi. Diagram sequence adalah alat yang digunakan dalam pengembangan sistem, menurut definisi ini.” (Yunahar, 2018).

Simbol	Nama Komponen	Keterangan
	Actor	Actor berperan untuk berkomunikasi dalam sistem
	Message	Digambarkan dengan anak panah horizontal antara activation. Message mengidentifikasi komunikasi antara objek-objek
	Object	Merupakan instance dari sebuah class dan dituliskan tersusun secara horizontal
	Lifeline	Mengindikasikan keberadaan objek dalam batas waktu Notasi untuk lifeline adalah sebuah garis putus-putus vertical yang ditarik dari sebuah object
	Activation	Dinotasikan sebagai sebuah kotak segi empat digambarkan pada sebuah lifeline mengidentifikasi sebuah objek yang akan melakukan aksi

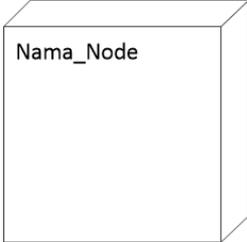
Gambar 2. 10 Simbol Sequence Diagram(Yunahar,2018)

2.9.4 Deployment Diagram

Menurut Sukamto dan Shalahudin (2015) mengemukakan bahwa “Komponen-komponen yang dikonfigurasi dalam proses eksekusi aplikasi ditampilkan dalam diagram penyebaran. Berikut ini dapat dijelaskan menggunakan diagram penyebaran :

1. Sistem tambahan yang menggambarkan rancangan device, node dan hardware.
2. Sistem client dan sistem server.
3. Sistem terdistribusi.
4. Rekayasa ulang aplikasi.

Menurut Hendini (2016) Deployment Diagram digunakan untuk menjelaskan secara spesifik susunan komponen-komponen infrastruktur sistem. Dengan demikian, dapat dikatakan bahwa diagram penyebaran adalah diagram yang menunjukkan bagaimana suatu sistem dikonfigurasi.

Simbol	Keterangan
	Package merupakan sebuah bungkus dari satu atau lebih node
	Biasa mengacu pada perangkat keras(Hardware), perangkat lunak yang tidak dibuat sendiri(software), jika di dalam node disertakan rancangan maka komponen yang telah diikutsertakan harus sesuai dengan komponen yang telah didefinisikan sebelumnya pada diagram komponen.

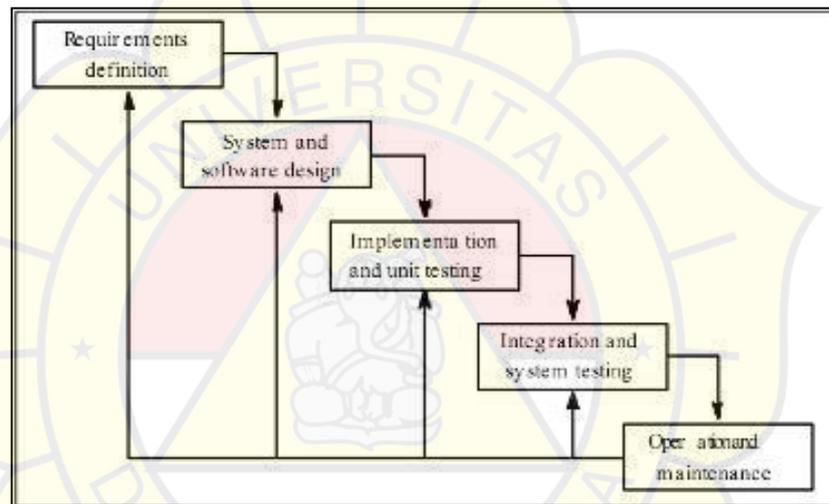
Ketergantungan 	Kebergantungan antar node, arah panah mengarah pada node yang dipakai.
Lnk 	Relasi antar node

Gambar 2. 11 Simbol Deployment Diagram (*Yunahar,2018*)



2.8 Metodologi Pengembangan Sistem

Metode Waterfall adalah suatu proses pengembangan perangkat lunak berlanjut. Teknik air terjun terdiri dari langkah-langkah berikut dalam urutan pengembangan: persyaratan (analisis kebutuhan), desain sistem, pengkodean dan pengujian, implementasi program, dan pemeliharaan. (Chrisantus Trisianto. 2018. Jurnal Teknologi Informasi: Penggunaan metodel waterfall untuk pengembangan sistem monitoring dan evaluasi pembangunan pedesaan).



Gambar 2. 12 Metode Waterfall (Trisianto,2018)

Adapun penjelasan urutan dari tahapan-tahapan yang dimiliki metodologi waterfall adalah sebagai berikut:

2.10.1. Requirement (Analisa Kebutuhan)

Persyaratan sistem dianalisis pada langkah ini. Pada tahap ini, data dapat dikumpulkan melalui desain, wawancara, atau tinjauan pustaka

Sebisa mungkin data dari pengguna untuk membangun sistem komputer yang dapat menjalankan tugas yang diinginkan pengguna dokumen persyaratan pengguna, atau data yang berkaitan dengan keinginan pengguna dalam mengembangkan sistem, akan dibuat pada tahap ini. Makalah ini akan berfungsi sebagai panduan saat menerjemahkan analisis sistem ke dalam bahasa pemrograman.

2.10.2 Design System (Design Sistem)

Sebelum pengkodean apa pun dilakukan, proses desain akan mengubah persyaratan menjadi desain perangkat lunak yang dapat diperkirakan struktur data, arsitektur perangkat lunak, representasi antarmuka, dan aspek prosedural (algoritmik) merupakan area utama yang menjadi perhatian dalam prosedur ini. Hasil dari langkah ini akan berupa dokumen yang dikenal sebagai persyaratan perangkat lunak. Programmer akan memanfaatkan dokumen ini untuk menjalankan tugas pembuatan sistem mereka.

2.10.3. Coding & Testing (Program/Implementation)

Coding merupakan penerjemahan design dalam bahasa yang bisa dikenali oleh komputer. Dilakukan oleh seorang programmer yang akan menginterpretasikan pesanan yang diminta oleh pengguna. Tahap pengembangan sistem yang sebenarnya ada pada titik ini. dalam artian bahwa pada titik ini, penggunaan komputer akan dioptimalkan. Pengujian akan dilakukan pada sistem yang dikembangkan sebelumnya setelah kode selesai. Pengujian dilakukan untuk mengidentifikasi kelemahan sistem sehingga dapat dievaluasi.

2.10.4 Penerapan/Pengujian Program (Integration & Testing)

Dapat dikatakan bahwa ini adalah langkah terakhir dalam menciptakan sebuah sistem. Pengguna menggunakan sistem yang telah dibuat setelah melalui analisis, desain, dan pengkodean.

2.10.5 Pemeliharaan (Operation & Maintenance)

Perangkat lunak yang susah disampaikan kepada pelanggan pasti akan mengalami perubahan. Modifikasi ini mungkin merupakan hasil dari bug, kebutuhan perangkat lunak untuk membiasakan diri dilingkungan baru (periferal atau sistem operasi), atau permintaan dari klien untuk kemajuan fungsional.

