

BAB II TINJAUAN PUSTAKA

Bab ini berisikan hal-hal yang bermanfaat bagi penulis dalam merancang dan mengembangkan aplikasi web sistem pendukung pengambilan keputusan pemilihan saham terbaik dengan menggunakan metode AHP (*Analytical Hierarchy Process*).

2.1 Pencapaian Terdahulu

Penelitian ini mengacu pada berbagai referensi dari penelitian-penelitian sebelumnya menjadi dasar dan rujukan untuk pengembangan penelitian ini.

Tabel 2. 1 Referensi Pencapaian Terdahulu

No	Peneliti (Tahun)	Judul	Hasil	Publikasi
1	Edison (2022)	Implementasi Metode Analytical Hierarchy Process Untuk Mendukung Pengambilan Keputusan Dalam Pemilihan Saham	Hasil penelitian menunjukkan bahwa AHP dapat membantu investor dalam pengambilan keputusan pemilihan atau seleksi saham terbaik secara efektif dan konsisten. Dengan menggunakan AHP, penelitian ini memberikan informasi mengenai prioritas kriteria, sub-kriteria, dan alternatif saham yang dapat membantu investor dalam pengambilan keputusan.	Journal of Digital Ecosystem for Natural Sustainability (JoDENS)
<p>Diferensiasi :</p> <p>Fokus sub sektor yang berbeda, peneliti saat ini menggunakan sub sektor farmasi, sedangkan penelitian ini menggunakan sub sektor makanan sebagai fokus penelitiannya</p>				
2	Rudolfo Rizki Damanik, Leonardy Khanady (2021)	Implementasi Metode Analytic Hierarchy Process Untuk Pemilihan Saham Terbaik Berbasis Website	Penelitian ini bertujuan untuk membantu investor dalam memilih saham terbaik dengan menggunakan metode Analytic Hierarchy Process (AHP) melalui pendekatan sistem pendukung keputusan berbasis web. Dari hasil penelitian, pendekatan metode AHP memberikan hasil berupa saham terbaik, yang dapat membantu investor dalam menentukan saham terbaik sesuai dengan	Jurnal PSDKU Sistem Informasi UPH Kampus Medan

			penilaian mereka sendiri.	
<p>Diferensiasi :</p> <p>Penelitian saat ini menyajikan subkriteria dengan lebih lengkap dan mendetail sedangkan peneliian ini hanya menyajikan kriteria saja</p>				
3	Muhammad Saiful Munir, Panca Rahardiyanto (2023)	Sistem Pendukung Keputusan Pemilihan Investasi Saham Syariah dalam Indeks JII 70 Menggunakan Metode AHP Beserta Perhitungan MoneyManagement	<p>Penelitian ini bertujuan untuk membantu investor pemula dalam memilih perusahaan mana yang cocok untuk dipilih dengan menggunakan metode AHP yang didukung dengan beberapa laporan keuangan perusahaan. Selain itu, penelitian ini juga menambahkan fitur Money Management untuk memudahkan investor dalam mengalokasikan dana yang ingin diinvestasikan sesuai dengan tujuannya. Dengan metode AHP, laporan keuangan perusahaan yang tadinya standar dan tidak terstruktur berubah menjadi terstruktur dan terarah sehingga investor pemula yang menggunakan aplikasi ini dapat memperoleh gambaran dalam memilih saham terbaik berdasarkan hasil nilai peringkat tertinggi dan juga nilai saham terbaik. Selain itu, investor dengan aplikasi ini mendapatkan bantuan simulasi pengelolaan uang dengan mengedepankan teknik averaging agar return yang</p>	Jurnal Sprit

			didapat bisa maksimal.	
<p>Diferensiasi :</p> <p>Penelitian saat ini menggunakan ROA, ROE, DER, PBV dan PER sebagai Kriterianya sedangkan peneliian ini menggunakan NPM, PER, PBV, EPS</p>				
4	Romindo (2021)	Analisa Penentuan Saham Terbaik Menggunakan	Hasil penelitian menunjukkan bahwa berdasarkan metode AHP, saham CPIN dinilai	Jurnal Media Informatika Budidarma

		<p>Metode Analytic Hierarchy Process (AHP)</p>	<p>sebagai saham terbaik dengan Priority Vector sebesar 0.6536, diikuti oleh saham JFPA dengan Priority Vector 0.2013, dan saham MAIN dengan Priority Vector 0.1452. Penelitian ini bertujuan untuk membantu investor dalam memilih saham terbaik dengan menggunakan metode AHP, yang merupakan sistem pendukung keputusan yang membantu dalam pengambilan keputusan berdasarkan analisis fundamental. Penelitian ini memberikan kontribusi dalam pengembangan sistem pendukung keputusan untuk memudahkan analisis dan pemilihan saham terbaik bagi para investor</p>	
<p>Diferensiasi : Fokus sub sektor yang berbeda, peneliti saat ini menggunakan sub sektor farmasi, sedangkan penelitian ini menggunakan sub sektor lain sebagai fokus penelitiannya.</p>				

5	Muhammad Hilmi Fauzi dan Riffa Haviani Laluma (2023)	Penerapan Metode AHP Dan Promethee Pada Sistem Pendukung Keputusan Dalam Pemilihan Investasi Saham Tembakau	Hasil penelitian menunjukkan bahwa penerapan metode AHP dapat digunakan untuk menentukan bobot kriteria dalam pemilihan investasi saham tembakau, sementara metode Promethee memberikan rekomendasi peringkat alternatif terhadap perusahaan tembakau. Dengan demikian, penelitian ini memberikan kontribusi dalam membantu para investor untuk membuat keputusan investasi yang lebih baik dalam pemilihan saham tembakau.	Bulletin Information Systems Vol. 1, No. 1, July 2023
<p>Diferensiasi :</p> <p>Fokus sub sektor yang berbeda, peneliti saat ini menggunakan sub sektor farmasi, sedangkan penelitian ini menggunakan saham tembakau sebagai fokus penelitiannya</p>				

2.2 Tinjauan Teoritis

Penelitian ini didasari oleh beberapa teori dan konsep yang relevan dengan permasalahan yang akan dikaji. Tinjauan teoritis bertujuan untuk memberikan landasan pemahaman yang kuat bagi peneliti dalam menganalisis masalah dan menarik kesimpulan

2.2.1 Saham

Saham merupakan instrumen keuangan yang mendefinisikan bagian kepemilikan dari seorang atau entitas kepada perusahaan tertentu. Pemegang saham memiliki hak dan saluran tertentu, misalnya hak atas dividen dan hak suara pada rapat pemegang saham. Saham diperdagangkan di pasar modal sehingga menciptakan nilai dan likuiditas bagi investor.

Saham memiliki peran integral dalam keberlangsungan perusahaan dan pasar modal. Perusahaan dapat mengumpulkan modal melalui

penawaran saham perdana (IPO) untuk mendukung operasional dan pertumbuhan. Bagi investor, saham dapat menjadi instrumen investasi dengan peluang keuntungan melalui kenaikan harga saham dan dividen. Saham juga memberikan hak suara dalam keputusan perusahaan.

Ada dua jenis saham utama: saham biasa dan saham preferen. Saham biasa memberikan hak suara pada rapat pemegang saham dan dividen variabel. Saham preferen memiliki prioritas dibandingkan dividen, namun biasanya tidak memiliki hak suara sebanyak saham biasa. Saat menganalisis struktur pemegang saham dan peringkat harga saham, penting untuk memahami karakteristik keduanya.

2.2.2 Teori Keuangan Dasar

Dalam upaya memahami perilaku pasar saham dan fondasi pengambilan keputusan investasi, beberapa teori keuangan dasar sangat krusial. Teori-teori ini membantu menjelaskan bagaimana pasar saham beroperasi dan memberikan wawasan tentang faktor-faktor yang mempengaruhi keputusan investasi. Tinjauan teoritis ini mencakup:

1. *Efficient Market Hypothesis (EMH)*

Efficient Market Hypothesis (EMH), yang pertama kali dikemukakan oleh Eugene Fama pada tahun 1970, menyatakan bahwa harga saham selalu mencerminkan semua informasi yang tersedia, dan saham selalu diperdagangkan pada nilai wajarnya di pasar. Ini berarti bahwa dalam pasar yang efisien, tidak mungkin bagi investor untuk secara konsisten memperoleh pengembalian pasar yang lebih tinggi tanpa mengambil risiko yang lebih tinggi. EMH dibagi menjadi tiga bentuk:

- Bentuk Lemah: Semua informasi historis pasar sudah tercermin dalam harga saham.
- Bentuk Semi-kuat: Semua informasi yang tersedia untuk publik sudah tercermin dalam harga saham.
- Bentuk Kuat: Semua informasi, termasuk informasi internal, sudah tercermin dalam harga saham.

2. *Portfolio Theory*

Teori portofolio, yang dikembangkan oleh Harry Markowitz pada tahun 1952, mengusulkan konsep diversifikasi untuk mengurangi risiko. Teori ini menekankan pentingnya memilih kombinasi aset dalam portofolio investasi untuk meminimalkan risiko dan memaksimalkan pengembalian. Dalam konteks ini, risiko dan pengembalian portofolio tidak hanya bergantung pada performa aset individual, tetapi juga pada hubungan korelasi antara return aset tersebut. Teori ini membantu investor dalam mengelola portofolio saham mereka dengan lebih efektif dan mengoptimalkan keseimbangan antara risiko dan pengembalian.

3. **Konsep *Risk and Return***

Konsep risiko dan pengembalian adalah inti dari pengambilan keputusan investasi. Risiko diartikan sebagai kemungkinan bahwa hasil investasi akan berbeda dari yang diharapkan, termasuk kemungkinan kehilangan sebagian atau seluruh investasi. Pengembalian adalah keuntungan atau kerugian dari investasi selama periode tertentu, yang biasanya dinyatakan sebagai persentase. Hubungan antara risiko dan pengembalian adalah inti dari keputusan investasi, di mana aset dengan risiko yang lebih tinggi diharapkan memberikan pengembalian yang lebih tinggi untuk mengkompensasi risiko tersebut. Investor harus mengevaluasi toleransi risiko mereka sendiri dan kebutuhan pengembalian sebelum membuat keputusan investasi.

Mempelajari dan memahami teori-teori keuangan dasar ini memberikan landasan bagi investor untuk membuat keputusan yang lebih informasional dan terstruktur, memanfaatkan kerangka kerja yang telah diuji untuk memaksimalkan potensi keberhasilan investasi mereka.

2.2.3 **Kriteria Penilaian**

Dalam analisis saham, kriteria keuangan merupakan alat esensial untuk mengevaluasi potensi dan kinerja sebuah perusahaan. Berikut adalah rincian kriteria yang digunakan dalam analisis saham, disertai dengan kutipan dari literatur yang relevan:

1. *Price to Earnings Ratio (PER)*

Price Earning Ratio (PER) mengukur seberapa banyak pasar bersedia membayar untuk setiap unit pendapatan perusahaan. Menurut Damodaran (2019), "PER yang tinggi dapat menunjukkan ekspektasi pasar terhadap pertumbuhan pendapatan yang lebih besar di masa depan, sementara PER yang rendah mungkin menunjukkan penilaian pasar bahwa saham tersebut undervalued atau perusahaan tersebut menghadapi tantangan signifikan" (p. 112). Rasio ini dihitung dengan membagi harga pasar per saham dengan laba per saham (EPS).

2. *Price to Book Value (PBV)*

Price to Book Value (PBV) adalah rasio yang membandingkan harga pasar saham dengan nilai bukunya per saham, yang menawarkan wawasan tentang bagaimana pasar menilai nilai intrinsik perusahaan. Menurut Haskel and Westlake (2019), "PBV yang lebih besar dari satu bisa mengindikasikan bahwa perusahaan dihargai tinggi oleh pasar karena prospek pertumbuhan yang baik, sedangkan PBV yang lebih kecil dari satu mungkin menunjukkan bahwa saham dianggap undervalued" (p. 88).

3. *Debt to Equity Ratio (DER)*

Debt to Equity Ratio (DER) adalah ukuran leverage finansial yang menunjukkan proporsi dana yang diperoleh melalui utang versus ekuitas. Menurut Smith and Walter (2020), "DER yang tinggi menunjukkan bahwa perusahaan memiliki tingkat leverage finansial yang lebih besar, yang dapat meningkatkan risiko keuangan, sedangkan DER yang rendah menunjukkan ketergantungan yang lebih kecil pada utang dalam struktur pembiayaan" (p. 231).

4. *Return on Equity (ROE)*

Return on Equity (ROE) mengukur seberapa efektif manajemen dalam menggunakan ekuitas investor untuk menghasilkan pendapatan. Menurut Arnold (2020), "ROE yang tinggi merupakan indikator kunci dari efektivitas manajemen dalam menghasilkan laba dan nilai tambah bagi pemegang saham" (p. 275).

5. *Return on Assets (ROA)*

Return on Assets (ROA) mengukur efisiensi perusahaan dalam menggunakan asetnya untuk menghasilkan laba. According to Moyer, McGuigan, and Kretlow (2021), "ROA yang tinggi menandakan bahwa perusahaan menggunakan asetnya dengan efisien untuk menghasilkan laba, yang berarti perusahaan dapat memaksimalkan hasil dari investasi asetnya" (p. 149).

2.2.4 Sistem Pendukung Keputusan

Decision Support System/Sistem pendukung keputusan adalah sistem informasi yang dirancang khusus untuk mendukung proses pengambilan keputusan dengan menggunakan data, model, dan berbagai teknik analisis. Sistem ini bertujuan untuk memecahkan masalah yang kompleks dengan memberikan informasi yang akurat dan relevan kepada pengguna. SPK dapat digunakan di berbagai bidang seperti bisnis, manajemen, dan kesehatan.

Tujuan utama SPK adalah untuk meningkatkan efektivitas dan efisiensi proses pengambilan keputusan. *Decision Support System* memungkinkan pengambil keputusan menganalisis data lebih dalam dan lebih cepat untuk membuat keputusan yang lebih baik dalam waktu yang lebih singkat. Selain itu, *Decision Support System* meminimalkan risiko kesalahan dalam proses dan memungkinkan pengambilan keputusan yang lebih andal. (Sarwandi et al., 2023).

Menurut Lubis et al. (2022), Sistem pendukung keputusan memiliki

komponen-komponen berikut:

1. *Data Management*

Ini melibatkan pengelolaan basis data yang memiliki informasi relevan dengan situasi tertentu dan dikelola oleh *software* yang disebut sistem manajemen basis data.

2. *Model Management*

Suatu bentuk paket perangkat lunak yang mencakup berbagai model keuangan, statistik, ilmu manajemen, atau model kuantitatif lainnya yang memungkinkan dilakukannya analisis terperinci. Selain itu, perangkat lunak manajemen terkait juga disertakan.

3. *User Interface*

Bagian ini merupakan media interaksi antara pengguna dengan sistem, dimana pengguna memberikan masukan yang dapat diproses oleh sistem untuk mengambil keputusan.

4. *Knowledge-Based Subsystem*

Ini adalah komponen yang mendukung semua subsistem lainnya atau dapat berfungsi sebagai elemen yang berdiri sendiri.

Adapun metode-metode yang dapat digunakan dalam sistem pendukung keputusan, di antaranya sebagai berikut:

1. *Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS)*

Metode *Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS)* adalah metode untuk pengambilan keputusan multi-kriteria. Teknik ini membantu memilih alternatif terbaik dari berbagai opsi berdasarkan kriteria tertentu, dengan prinsip bahwa alternatif terbaik adalah yang paling dekat dengan solusi ideal positif dan paling jauh dari solusi ideal negatif (Uzun et al., 2021).

2. *Simple Additive Weighting (SAW)*

SAW, atau metode penjumlahan terbobot, adalah metode yang menghitung total nilai bobot dari kinerja setiap alternatif pada semua kriteria. Proses ini memerlukan normalisasi matriks keputusan (X) ke dalam skala yang memungkinkan perbandingan antar nilai

alternatif (Ibrahim, 2020).

3. *Analytical Hierarchy Process* (AHP)

AHP adalah teknik pengambilan keputusan yang memecah masalah menjadi kriteria spesifik dalam struktur hierarki. Proses ini memungkinkan pembandingan berpasangan antara kriteria, sehingga bobot kepentingan dari masing-masing kriteria dapat diperoleh secara lebih akurat (Hananto et al., 2021).

2.2.5 **Analytical Hierarchy Process**

Analytical Hierarchy Process (AHP) dikembangkan oleh Profesor Thomas L. Saaty sebagai alat bantu pengambilan keputusan untuk memecahkan masalah multi-kriteria yang kompleks dengan membaginya menjadi struktur hierarki. AHP adalah sebuah metode yang memungkinkan pemecahan situasi kompleks dan tidak terstruktur menjadi beberapa elemen yang diatur dalam bentuk hierarki. Metode ini menggunakan penilaian subjektif untuk menilai tingkat kepentingan relatif setiap variabel, dan menentukan variabel mana yang memiliki prioritas tertinggi dalam memengaruhi hasil dari situasi tersebut (Parhusip, 2019).

Hierarki sendiri merupakan representasi dari masalah yang kompleks dalam bentuk struktur berlapis-lapis. Pada lapisan pertama terdapat tujuan utama, kemudian diikuti oleh lapisan faktor, kriteria, sub-kriteria, hingga lapisan terakhir yang mencakup berbagai alternatif solusi.

Menurut Marsono (2020), penggunaan AHP terdapat tiga prinsip dasar, yaitu:

1. Dekomposisi (*Decomposition*)

Pada tahap ini, masalah yang kompleks dipecah menjadi bagian-bagian kecil yang disusun dalam bentuk hierarki. Tujuannya adalah untuk menguraikan masalah dari tingkat yang lebih umum ke yang lebih spesifik. Dalam bentuk dasarnya, struktur ini memfasilitasi perbandingan antara tujuan, kriteria, dan alternatif pada berbagai tingkatan. Setiap kelompok alternatif dapat diperinci lebih lanjut, sehingga mencakup lebih banyak kriteria. Pada tingkatan tertinggi dari hierarki ini, terdapat "tujuan utama" yang hanya terdiri dari satu elemen.

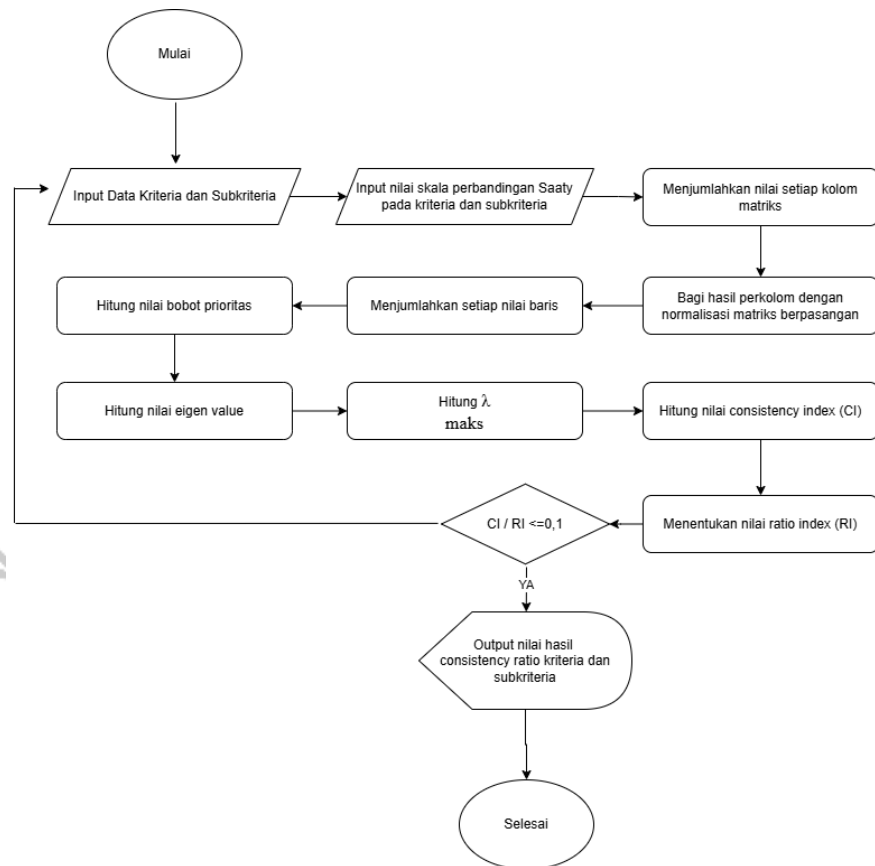
Di tingkat berikutnya, beberapa elemen dapat muncul sebagai "kriteria," di mana elemen-elemen ini saling dibandingkan satu sama lain, biasanya memiliki tingkat kepentingan yang hampir setara. Jika ada perbedaan signifikan antar elemen, maka dibuat sub-kriteria pada tingkatan yang lebih rendah untuk memperjelas perbedaan tersebut.

2. Perbandingan penilaian/pertimbangan (*Comparative Judgments*)

Pada tahap ini, setiap elemen dalam hierarki dibandingkan satu per satu untuk menentukan tingkat kepentingan relatifnya. Hasil dari perbandingan ini disajikan dalam bentuk angka pada skala tertentu. Proses perbandingan berpasangan ini, jika digabungkan, akan menghasilkan prioritas dari setiap elemen berdasarkan perhitungan vektor eigen.

3. Sintesa Prioritas (*Priority Synthesis*)

Proses sintesis prioritas melibatkan pengalihan antara nilai prioritas lokal suatu elemen dengan nilai prioritas dari kriteria yang ada di tingkat di atasnya, kemudian menjumlahkan hasil tersebut ke setiap elemen yang dipengaruhi oleh kriteria tersebut. Output dari proses ini adalah nilai prioritas global yang kemudian digunakan untuk memberikan bobot pada nilai prioritas lokal dari elemen di tingkat terbawah, sesuai dengan kriteria yang ada.



Gambar 2. 1 Tahapan Proses AHP

Gambar 2.1 merupakan tahapan-tahapan proses AHP secara detail (Parhusip,2019), berikut penjelasannya

1. Input Data Kriteria dan Subkriteria

Proses dimulai dengan memasukkan data yang mencakup kriteria dan subkriteria yang relevan untuk keputusan yang akan diambil. Ini adalah langkah awal untuk membangun struktur hierarki yang akan digunakan dalam analisis AHP.

2. Input Nilai Skala Perbandingan Saaty pada Kriteria dan Subkriteria

Selanjutnya, nilai skala perbandingan Saaty diinput untuk setiap kriteria dan subkriteria. Skala ini berkisar dari 1 hingga 9, di mana angka tersebut mewakili tingkat kepentingan relatif antara dua elemen dalam hierarki.

3. Menunjukkan Nilai Setiap Kolom Matriks

Nilai dari setiap kolom matriks perbandingan dijumlahkan. Ini adalah langkah penting untuk menormalisasi matriks dan mempersiapkan perhitungan bobot prioritas.

4. Menghitung Nilai Bobot Prioritas

Setelah matriks dinormalisasi, bobot prioritas dihitung dengan membagi setiap elemen matriks dengan total kolomnya. Bobot ini menunjukkan tingkat kepentingan relatif dari setiap kriteria dan subkriteria dalam konteks keputusan keseluruhan.

5. Menghitung Nilai Eigen Value

Eigen value dihitung sebagai produk dari matriks perbandingan dengan vektor bobot prioritasnya. Langkah ini menghasilkan nilai λ maks, yang digunakan untuk menguji konsistensi perbandingan yang dilakukan.

6. Menghitung λ maks

λ maks dihitung sebagai rata-rata nilai eigen yang diperoleh dari proses sebelumnya. Nilai ini penting untuk menentukan apakah perbandingan yang dibuat konsisten atau tidak.

7. Menghitung Consistency Index (CI)

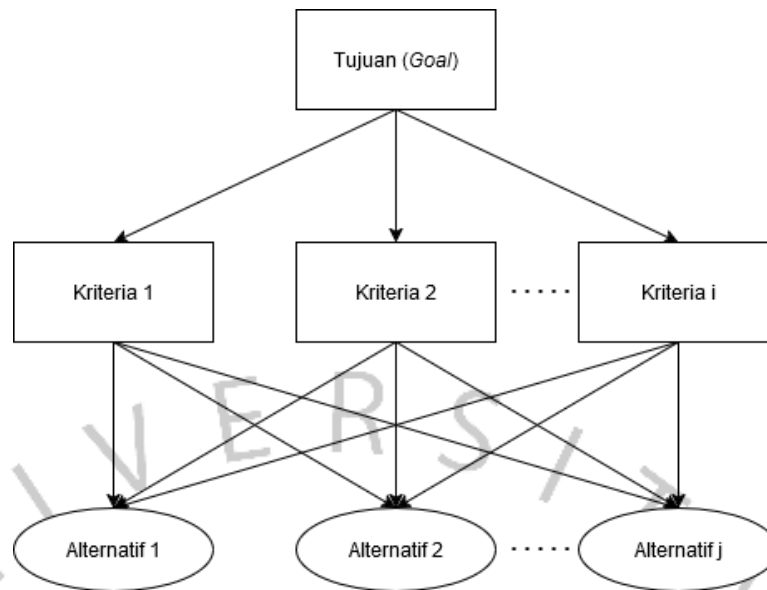
Consistency Index (CI) dihitung menggunakan rumus $(\lambda \text{ maks} - n) / (n - 1)$, di mana n adalah jumlah kriteria atau elemen dalam matriks. CI membantu menilai apakah perbandingan yang dilakukan secara matematis konsisten.

8. Menghitung Consistency Ratio (CR)

Consistency Ratio (CR) dihitung sebagai CI dibagi dengan Random Index (RI) yang sesuai. RI merupakan nilai yang ditentukan berdasarkan ukuran matriks dan digunakan sebagai benchmark untuk menilai konsistensi. Jika $CR \leq 0.1$, maka perbandingan dianggap konsisten dan hasilnya dapat diandalkan.

9. Output Nilai Hasil Consistency Ratio

Tahap akhir adalah output nilai CR dari semua kriteria dan subkriteria. Jika nilai CR menunjukkan konsistensi, maka analisis AHP dianggap valid dan dapat digunakan untuk membuat keputusan yang berinformasi dan objektif.



Gambar 2. 2 Struktur Hirarki AHP

Menurut Putra & Diana (2021), terdapat beberapa langkah dalam penyelesaian menggunakan metode AHP, yaitu memaparkan masalah dan solusi yang diusulkan, serta menyusun struktur hierarki untuk analisis lebih lanjut.

10. Menjelaskan situasi masalah dan memberikan solusi, kemudian menyusun struktur hierarki.
11. Memberikan bobot prioritas pada elemen-elemen dengan melakukan perbandingan berpasangan menggunakan skala 1-9 (membandingkan setiap elemen secara berpasangan sesuai dengan kriteria yang telah ditetapkan). Selanjutnya, hasil perbandingan ini disajikan dalam bentuk matriks dengan bilangan desimal.

Tabel 2. 2 Skala Penilaian AHP

Intensitas Kepentingan	Keterangan
1	Kedua elemen sangat penting.
3	Elemen yang satu sedikit lebih penting dari elemen lainnya.
5	Elemen yang satu lebih penting dari elemen lainnya.
7	Elemen yang satu sangat penting dari elemen lainnya.
9	Elemen yang satu mutlak sangat penting dari elemen lainnya.
2,4,6,8	Nilai-nilai antardua nilai pertimbangan yang berdekatan.
Kebalikan	Jika aktivitas i mendapat satu angka dibandingkan dengan aktivitas j, maka j memiliki nilai kebalikan dibanding i.

12. Melakukan sintesis perbandingan berpasangan, agar mendapatkan masing-masing prioritas. Setelah itu, nilai tersebut dimasukkan kedalam matrik. Bobot nilai metode AHP dihitung dengan langkah berikut:

- a. Memaparkan matrik kedalam angka desimal.
- b. Mengalikan matrik tersebut dengan matriks nya sendiri.
- c. Jumlahkan hasil dari proses perkalian matriks.
- d. Menjumlahkan matrik normalisasi (baris), kemudian membagi tiap jumlah baris dengan nilai akhir baris. Hasil rata-rata nilai tersebutdisebut dengan *eigenvector*.
- e. Resume nilai *eigenvector*.

13. Mengukur konsistensi perhitungan *Consistency Index* (CI) dengan rumus:

$$CI = (\lambda \text{ maks} - n) / n - 1$$

Dimana,

CI: *Consistency Index*

λ Maks: Eigen value

N: Banyaknya kriteria

14. Kemudian menghitung *Consistency Ratio* (CR) dengan rumus:

$$CR = CI/RI$$

Dimana,

CR: *Consistency Ratio*

CI: *Consistency Index*

RI: *Random Index*

Berikut tabel *Random Index* (RI):

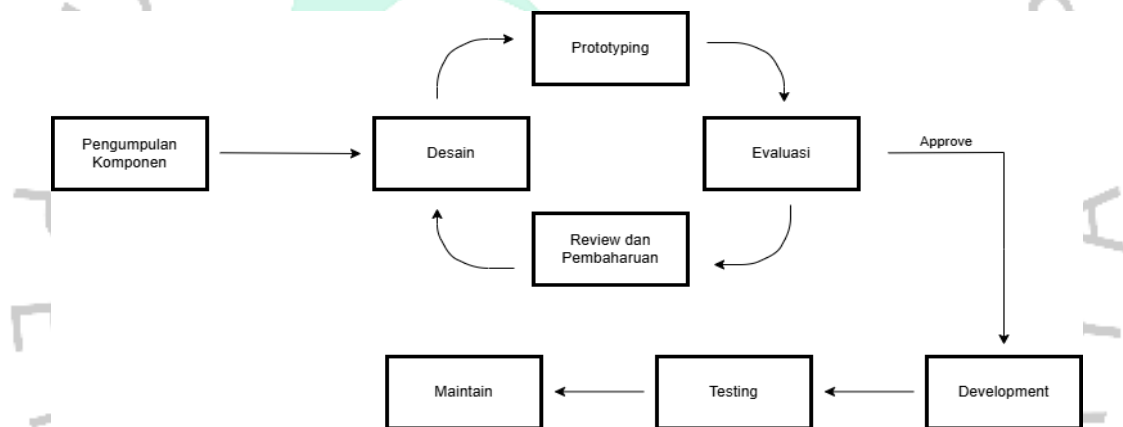
Tabel 2. 3 *Random Index* (RI)

Ordo Matriks	Random Index (RI)
1	0.00
2	0.00
3	0.58
4	0.9

5	1.12
6	1.24
7	1.32
8	1.41
9	1.45
10	1.49

Apabila hasil dari *Consistency Ratio* (CR) > 10% atau 0.1, maka kuesioner harus diulang kembali. Jika hasil *Consistency Ratio* (CR) ≤ 0.1 , maka hasil perhitungan sudah dapat diputuskan benar.

2.2.6 Metode Pengembangan Sistem



Gambar 2. 3 Metodologi Prototipe

Gambar ini menggambarkan metodologi prototipe yang sering digunakan dalam proses pengembangan sistem atau perangkat lunak. Berikut adalah penjelasan dari masing-masing langkah dalam diagram tersebut (Pressman, R. S. (2019)):

1. Pengumpulan Komponen

Langkah awal ini melibatkan identifikasi kebutuhan sistem. Pengembang mengumpulkan informasi, data, dan komponen-komponen yang diperlukan untuk memahami apa yang harus dibangun.

2. Desain

Berdasarkan hasil pengumpulan komponen, pengembang membuat rancangan awal sistem atau perangkat lunak. Desain ini digunakan sebagai dasar untuk membangun prototipe.

3. Prototyping

Prototipe awal dari sistem dikembangkan berdasarkan desain yang telah dibuat. Prototipe ini merupakan model sederhana atau versi awal dari sistem untuk menguji konsep dan mendapatkan umpan balik.

4. Evaluasi

Prototipe yang telah dibuat dievaluasi oleh pengguna atau tim proyek untuk menentukan apakah memenuhi kebutuhan dan harapan. Jika prototipe tidak sesuai, akan dilakukan proses perbaikan.

5. Review dan Pembaruan

Berdasarkan hasil evaluasi, prototipe diperbarui atau dimodifikasi agar lebih mendekati kebutuhan pengguna. Langkah ini memungkinkan pengembang untuk meningkatkan kualitas sistem sebelum tahap pengembangan penuh.

6. Approve (Persetujuan)

Setelah prototipe disempurnakan dan mendapat persetujuan, proyek masuk ke tahap pengembangan (development).

7. Development

Sistem atau perangkat lunak dibangun secara penuh berdasarkan prototipe akhir yang telah disetujui. Pengembangan ini mencakup pengkodean, integrasi, dan dokumentasi.

8. Testing

Setelah pengembangan selesai, sistem diuji untuk memastikan bahwa semua fungsi berjalan sesuai spesifikasi dan tidak ada kesalahan.

9. Maintain (Pemeliharaan)

Setelah sistem selesai dan digunakan, dilakukan proses pemeliharaan untuk memastikan bahwa sistem tetap bekerja dengan baik, memperbaiki bug, dan melakukan pembaruan jika diperlukan.

2.2.7 Contoh Implementasi Penggunaan AHP

1. Implementasi dalam bidang Pendidikan

Implementasi metode Analytical Hierarchy Process (AHP) dalam penentuan penerima bantuan Program Indonesia Pintar (PIP) di SD Aek Nabara Tonga merupakan studi kasus yang menarik (Siregar & Hendrik (2023)). Dalam kasus ini, AHP digunakan untuk memastikan bahwa bantuan PIP diberikan kepada siswa yang paling memerlukannya berdasarkan kriteria yang telah ditentukan seperti kepemilikan kartu Program Keluarga Harapan (PKH), Surat Keterangan Tidak Mampu (SKTM), status yatim/piatu, nilai rapor, dan penghasilan orang tua.

Proses AHP dimulai dengan pembangunan hierarki yang mencakup tujuan utama, kriteria, dan alternatif penerima bantuan. Perbandingan berpasangan dilakukan untuk menentukan bobot relatif antar kriteria menggunakan skala Saaty, yang memungkinkan penilaian objektif terhadap pentingnya setiap kriteria.

Tabel 2. 4 *Matrix Perbandingan Berpasangan*

	C5	C4	C3	C2	C1
C5	1	0.33	0.16	0.20	0.16
C4	3	1	0.20	0.25	0.20
C3	6	5	1	0.50	0.33
C2	5	4	2	1	0.50
C1	6	5	3	2	1
Total	21	15.33	6.36	3.95	2.19

Selanjutnya, Melakukan normalisasi kriteria dengan mengacu pada rumus persamaan

Tabel 2. 5 Normalisasi Kriteria

	C5	C4	C3	C2	C1
C5	0.048	0.022	0.025	0.051	0.073
C4	0.143	0.065	0.031	0.063	0.091
C3	0.286	0.326	0.175	0.127	0.151
C2	0.238	0.261	0.314	0.253	0.228
C1	0.286	0.326	0.472	0.506	0.457

Selanjutnya, Menghitung eigen vector dengan mengacu pada rumus persamaan.

Tabel 2. 6 Eigen Vektor

	C5	C4	C3	C2	C1	Eigen Vector
C5	0.048	0.022	0.025	0.051	0.073	0.044
C4	0.143	0.065	0.031	0.063	0.091	0.079
C3	0.286	0.326	0.175	0.127	0.151	0.213
C2	0.238	0.261	0.314	0.253	0.228	0.259
C1	0.286	0.326	0.472	0.506	0.457	0.409

Selanjutnya, Menentukan nilai *Consistency Index* (CI) dan *Consistency Ratio* (CR)

$$W_i = (2.19 * 0.409) + (3.95 * 0.259) + (6.36 * 0.213) + (15.33 * 0.079) + (21 * 0.044) = 5.40851$$

$$CI = \frac{w_i - n}{n-1} = \frac{5.40851-5}{5-1} = 0.1021$$

$$IR = 1.12 \text{ (jumlah kriteria ada 5)}$$

$$CR = \frac{CI}{IR} = \frac{0.1021}{1.12} = 0.0912$$

Rasio konsistensi perhitungan (CR), yaitu 0.0912 dimana CR < 0.1, disetujui berdasarkan temuan yang diperoleh. Hasilnya, setiap kriteria diberi bobot sebagai berikut.

Tabel 2. 7 Bobot Kriteria

Kriteria	Keterangan	Bobot	Jenis
C1	PKH	0.409	Benefit
C2	SKTM	0.259	Benefit
C3	Yatim/Piatu	0.213	Benefit
C4	Nilai Rapor	0.079	Benefit
C5	Penghasilan Ortu	0.044	Benefit

Selanjutnya, Menghitung matriks normalisasi dengan menggunakan

rumus. didapatkan hasil sebagai berikut.

Tabel 2. 8 *Perhitungan Alternatif*

Alternatif	C1	C2	C3	C4	C5
S1	0	1	0	0.4	0.42
S2	1	1	0	0.87	0.25
S3	0	0	0	1	0
S4	1	1	0	0	0.21
S5	0	1	0	0.93	1
S6	0	1	1	0.8	0.25
S7	1	0	0	0.53	0.58
S8	1	1	0	0.6	0.08
S9	0	0	0	0.4	0.42
S10	1	1	1	0.93	0.42

Dari hasil akhir perhitungan menggunakan metode MAUT dan AHP, maka siswa yang mendapatkan nilai total tertinggi yaitu alternatif S10 yang bernama Riski Pinaldi Nasution dengan nilai total 0.9490, sedangkan siswa dengan nilai total terendah yaitu alternatif S9 yang bernama Muhammad Akbar Soleh dengan nilai 0.0127.

Tabel 2. 9 *Perangkingan Alternatif*

Alternatif	Nilai Utilitas Akhir	Rangking
S1	0.2717	8
S2	0.7219	2
S3	0.079	9
S4	0.6693	4
S5	0.3662	7
S6	0.5148	5
S7	0.4341	6
S8	0.6878	3
S9	0.0127	10
S10	0.9490	1

2. Implementasi dalam bidang Industri

Implementasi Metode *Analytical Hierarchy Process* (AHP) dalam pemilihan pemasok daging sapi dilakukan oleh Restoran Nominomi Delight di Jakarta. Penelitian yang dilakukan oleh Fadhilah dan Wiranthi (2021) bertujuan untuk menilai dan menentukan urutan prioritas pemasok berdasarkan kriteria seperti kualitas, biaya, waktu pengiriman, fleksibilitas, responsivitas, solusi, dan hubungan.

Studi ini menggunakan tujuh kriteria utama dengan tiga puluh delapan subkriteria untuk mengevaluasi empat pemasok daging sapi. Proses AHP dimulai dengan pembangunan hierarki yang meliputi tujuan, kriteria, subkriteria, dan alternatif. Setiap elemen hierarki dinilai melalui perbandingan berpasangan menggunakan skala 1-9 yang diusulkan oleh Saaty. Berdasarkan perhitungan, bobot masing-masing kriteria ditentukan dan konsistensi dari penilaian diuji.

Tabel 2. 10 *Bobot dan Prioritas*

Kriteria	Bobot	Prioritas
<i>Quality</i>	0,371	1
<i>Cost</i>	0,256	2
<i>Delivery</i>	0,101	3
<i>Flexibility</i>	0,070	5
<i>Responsiveness</i>	0,069	6
<i>Solutive</i>	0,078	4
<i>Relationship</i>	0,055	7

Selanjutnya, Berikut Hasil Pengolahan Horizontal Alternatif Pemasok pada Subkriteria dari Kriteria *Quality*

Tabel 2. 11 *Hasil Perhitungan Kriteria Quality*

Kriteria	Subkriteria	Alternatif Pemasok			
		Indoguna	Santori	Amino	Sukanda
<i>Quality</i>	Warna merah dan masih segar	0,437	0,136	0,089	0,338
	Pemotongan <i>mincebeef</i> menggunakan mesin giling	0,384	0,107	0,139	0,371
	Pemotongan <i>Slice beef</i> menggunakan mesin slice	0,453	0,085	0,107	0,355
	Ukuran slice beef setebal 1 mm	0,446	0,104	0,089	0,362
	Lemak daging sesuai permintaan	0,391	0,169	0,108	0,331
	Daging tidak keras	0,381	0,188	0,093	0,338
	Pemasok memiliki label Halal resmi dari MUI	0,250	0,250	0,250	0,250

Hasil Pengolahan Horizontal Alternatif Pemasok pada Subkriteria dari Kriteria *Cost*

Tabel 2. 12 *Hasil Perhitungan Kriteria Cost*

Kriteria	Subkriteria	Alternatif Pemasok			
		Indoguna	Santori	Amino	Sukanda
Cost	Harga daging Stabil	0,298	0,141	0,082	0,479
	Harga daging lebih murah dari harga pasar	0,315	0,199	0,104	0,382
	Biaya pengiriman lebih murah	0,317	0,210	0,168	0,395
	Biaya cincang lebih murah	0,347	0,138	0,228	0,228
	Harga sudah termasuk biaya kirim atau gratis ongkir	0,395	0,170	0,146	0,289
	Ada pengurangan harga atau diskon pada <i>quantity</i> tertentu	0,250	0,250	0,250	0,250

Hasil Pengolahan Horizontal Alternatif Pemasok pada Subkriteria dari Kriteria *Delivery*

Tabel 2. 13 Hasil Perhitungan Kriteria *Delivery*

Kriteria	Subkriteria	Alternatif Pemasok			
		Indoguna	Santori	Amino	Sukanda
<i>Delivery</i>	Ketepatan waktu pengiriman H+2 dari waktu order	0,344	0,106	0,117	0,433
	Pengiriman menggunakan alat pendingin / <i>Chiller</i>	0,346	0,227	0,115	0,313
	Pengemasan daging dengan baik	0,388	0,110	0,107	0,396
	Ketepatan jumlah timbangan <i>mince beef</i>	0,413	0,122	0,111	0,354
	Ketepatan jumlah	0,402	0,121	0,111	0,366

Hasil Pengolahan Horizontal Alternatif Pemasok pada Subkriteria dari Kriteria *Solutive*

Tabel 2. 14 Hasil Perhitungan Kriteria *Solutive*

Kriteria	Subkriteria	Alternatif Pemasok			
		Indoguna	Santori	Amino	Sukanda
<i>Solutive</i>	Pemasok memiliki truck pengantar, <i>chiller</i> , dan pengemudi	0,294	0,098	0,127	0,481
	Pemasok berbagi informasi libur, promo harga, dan produk baru	0,302	0,118	0,124	0,456
	Pemasok membantu restoran memenuhi kebutuhan dadakan	0,354	0,092	0,117	0,437
	Pemasok bertanggung jawab ketika salah kirim pesanan	0,294	0,124	0,115	0,469
	Pemasok turut memberi ide pada restoran saat keadaan sulit	0,309	0,112	0,111	0,468

Hasil Pengolahan Horizontal Alternatif Pemasok pada Subkriteria dari Kriteria *Flexibility*

Tabel 2. 15 Hasil Perhitungan Kriteria *Flexibility*

Kriteria	Subkriteria	Alternatif Pemasok			
		Indoguna	Santori	Amino	Sukanda
<i>Flexibility</i>	Menerima perubahan jumlah order <i>slice beef</i>	0,378	0,104	0,120	0,398
	Menerima perubahan jumlah order <i>mince beef</i>	0,364	0,128	0,116	0,391
	Menerima perubahan waktu kirim <i>slice beef</i>	0,440	0,100	0,116	0,344
	Menerima perubahan waktu kirim <i>mince beef</i>	0,424	0,124	0,117	0,335
	Rentang waktu dari pemesanan ke pengiriman pendek	0,297	0,108	0,137	0,458

Hasil Pengolahan Horizontal Alternatif Pemasok pada Subkriteria dari Kriteria *Responsiveness*

Tabel 2. 16 Hasil Perhitungan Kriteria *Responsiveness*

Kriteria	Subkriteria	Alternatif Pemasok			
		Indoguna	Santori	Amino	Sukanda
<i>Responsiveness</i>	Cepat dalam merespon permintaan <i>slice beef</i>	0,369	0,110	0,118	0,403
	Cepat dalam merespon permintaan <i>mince beef</i>	0,298	0,104	0,117	0,481
	Tanggap merespon pengaduan ketidaksesuaian orderan	0,357	0,092	0,153	0,398
	Tanggap merespon ketidakpuasan konsumen	0,354	0,119	0,122	0,405
	Pemasok memiliki banyak SDM yang kompeten	0,306	0,135	0,115	0,443

Hasil Pengolahan Horizontal Alternatif Pemasok pada Subkriteria dari Kriteria *Relationship*

Tabel 2. 17 Hasil Perhitungan Kriteria *Relationship*

Kriteria	Subkriteria	Alternatif Pemasok			
		Indoguna	Santori	Amino	Sukanda
<i>Relationship</i>	Pemasok memberi informasi dengan akurat	0,292	0,134	0,107	0,467
	Pemasok memberikan pelayanan terbaik dengan maksimal	0,302	0,117	0,102	0,479
	Komunikasi dengan pemasok terjalin dengan baik	0,282	0,098	0,081	0,539
	Pemasok 1 bulan sekali mengajak bertemu dan berdiskusi	0,289	0,117	0,105	0,489
	Pemasok peduli dengan kendala restoran	0,295	0,099	0,098	0,507

Hasil dari penelitian menunjukkan bahwa PT Sukanda Djaya mendapat prioritas tertinggi sebagai pemasok daging sapi, berdasarkan bobot total dari evaluasi AHP. Pendekatan ini memungkinkan restoran untuk membuat keputusan yang informasional dan objektif dalam pemilihan pemasok, yang sangat penting untuk menjaga kualitas dan keberlanjutan pasokan daging sapi.

2.2.8 Unified Modelling Language (UML)

Unified Modelling Language (UML) adalah suatu alat yang digunakan untuk memvisualisasikan dan mendokumentasikan hasil analisis dan desain yang mengandung sintaksis dalam memodelkan sistem secara visual. UML juga merupakan sekumpulan konvensi pemodelan yang digunakan untuk mendefinisikan atau menggambarkan sistem perangkat lunak yang terkait dengan objek (Ronald, Yunita, & Yuliana, 2022). Pada proses pengembangan perangkat lunak, UML membantu dalam merencanakan dan merancang solusi yang efektif dan efisien. Hal ini memungkinkan pengembang untuk mengidentifikasi kebutuhan, merancang struktur, dan mengantisipasi potensi masalah atau konflik sebelum implementasi sebenarnya dimulai.

UML terdiri dari berbagai jenis diagram yang masing-masing menggambarkan aspek khusus dari suatu sistem. Beberapa diagram UML yang sering digunakan meliputi *use case diagram*, *activity diagram*, *class diagram*, dan *sequence diagram*.

2.2.5.1 Use Case Diagram

Use case diagram adalah representasi model untuk tingkah laku sistem informasi yang akan dikembangkan. Use case menjelaskan interaksi antara satu atau lebih aktor dengan sistem informasi yang sedang dibangun. Use case digunakan untuk mengidentifikasi fungsi-fungsi yang ada dalam sistem informasi dan untuk menentukan siapa yang memiliki hak untuk menggunakan fungsi-fungsi tersebut (Simatupang & Sianturi, 2019).

Berikut fungsi dari *use case diagram*:

- a) Menunjukkan berbagai aksi atau interaksi yang mungkin terjadi

antar aktor dan sistem.

- b) Memberikan representasi visual tentang fungsionalitas utama yang akandisediakan oleh sistem.
- c) Menggambarkan bagaimana aktor berinteraksi dengan berbagai *usecase* di dalam sistem.
- d) Membantu dalam berkomunikasi antara pengembang, pemangku kepentingan, dan tim proyek tentang kebutuhan dan fungsionalitassistem.

Komponen-komponen pada *use case diagram*:

- a) Sistem

Komponen ini menetapkan batasan sistem dengan melibatkan aktor yang menggunakan sistem tersebut. Sistem diberi label yang sesuai, namun, umumnya tidak diwakili secara visual karena fokus lebih pada desain diagram.

- b) *Actor*

Actor adalah entitas atau pengguna yang berinteraksi dengan sistem pada *use case diagram*. *Actor* dapat direpresentasikan sebagai manusia atau objek tergantung pada konteks sistem yang sedang dibuat.

- c) *Use Case*

Use case adalah tindakan atau kegiatan yang dapat dilakukan oleh aktorterdhadap sistem. Elemen ini umumnya diilustrasikan dalam bentuk oval.

2.2.5.2 *Activity Diagram*

Activity diagram mengilustrasikan sejumlah alur kegiatan dalam sistem yang sedang direncanakan. Diagram ini menjelaskan bagaimana setiap alur dimulai, potensi keputusan yang dapat terjadi, menggambarkan proses paralel yang mungkinginterjadi dalam beberapa eksekusi, dan menunjukkan bagaimana kegiatan-kegiatan tersebut akan berakhir (Sandfreni, Ulum & Azizah, 2021). Tujuan dari *activity diagram* adalah merekam dinamika perilaku sistem dengan

menggambarkan alur pesan dari satu aktivitas ke aktivitas lainnya.

Berikut fungsi dari *activity diagram*:

- a) Memberikan gambaran yang komprehensif, jelas, dan terstruktur tentang serangkaian aktivitas.
- b) Menyajikan urutan proses dalam sistem yang akan dilaksanakan.
- c) Merupakan diagram khusus yang memodelkan tahapan proses berdasarkan satu atau beberapa *use case*.
- d) Mendukung penguraian algoritma sekuensial yang memiliki tingkat kompleksitas.
- e) Menganalisis penggunaan sistem termasuk langkah-langkah atau keputusan yang harus diambil beserta momentumnya.
- f) Membantu dalam pemodelan aplikasi dengan sistem pemrosesan paralel.
- g) Berfungsi sebagai medium untuk menelusuri kebutuhan bisnis pada tingkat yang lebih mendalam.

Komponen-komponen pada *activity diagram*:

- a) *Initial State* (Keadaan Awal)
Initial state merujuk pada titik awal suatu alur kerja dalam *activity diagram* dan dalam satu diagram hanya terdapat satu keadaan awal.
- b) *Final State* (Keadaan Akhir)
Final state merupakan titik akhir dari alur kerja dalam suatu *activity diagram* dan satu *activity diagram* dapat memiliki lebih dari satu keadaan akhir.
- c) *Activity* (Aktivitas)
Activity merujuk pada pekerjaan atau tindakan yang dilakukan dalam alur kerja suatu sistem.
- d) *Decision* (Keputusan)
Decision digunakan untuk menggambarkan kondisi yang memungkinkan terjadinya percabangan atau transisi yang berbeda, memastikan bahwa alur kerja dapat beralih ke jalur yang berbeda.
- e) *Merge* (Menggabungkan)

Merge berfungsi untuk menggabungkan kembali jalur alur kerja yang sebelumnya dipisahkan oleh keputusan.

f) *Transition* (Transisi)

Transition mengindikasikan perpindahan dari satu aktivitas ke aktivitas berikutnya dalam alur kerja.

g) *Synchronization* (Sinkronisasi)

Synchronization menunjukkan lokasi dalam diagram di mana aktivitas-aktivitas yang berjalan secara paralel dapat dijalankan bersama-sama. Terdapat dua komponen dalam *synchronization*, yaitu *fork* dan *join*. *Fork* digunakan untuk memisahkan perilaku menjadi aktivitas-aktivitas paralel, sementara *join* digunakan untuk menggabungkan kembali aktivitas-aktivitas tersebut.

2.2.5.3 *Class Diagram*

Class diagram mengilustrasikan struktur sistem dalam hal definisi kelas- kelas yang akan dibuat untuk konstruksi sistem (Simatupang & Sianturi, 2019). Diagram ini juga menggambarkan aturan dan tanggung jawab entitas yang mengatur perilaku sistem. Berikut fungsi dari *class diagram*:

- a) Memberikan representasi visual tentang struktur dan komposisi kelas- kelas dalam sistem perangkat lunak. Setiap kelas diilustrasikan dengan nama kelas dan atribut-atributnya.
- b) Menggambarkan hubungan dan asosiasi antar kelas, baik itu hubungan *one-to-one*, *one-to-many*, atau *many-to-many*. Hal ini membantu dalam memahami bagaimana objek-objek berinteraksi satu sama lain.
- c) Menunjukkan hierarki warisan (*inheritance*) antar kelas dan hubungan polimorfisme. Ini membantu menggambarkan bagaimana kelas-kelas dapat mewarisi sifat-sifat atau perilaku dari kelas lain.
- d) Menyajikan atribut-atribut dan metode-metode dari setiap kelas, membantu dalam memahami struktur data dan fungsionalitas yang dimiliki oleh kelas tersebut.

- e) Memberikan pemodelan objek yang memberikan pandangan konseptual tentang objek-objek yang akan dibuat dalam sistem. *Class diagram* membantu dalam merencanakan dan merancang struktur kelas secara keseluruhan.
- f) Digunakan selama fase analisis dan perancangan sistem untuk mengidentifikasi dan menyusun komponen-komponen utama serta hubungan antar mereka.
- g) Membantu dalam menjaga konsistensi antara tim pengembang dengan memberikan pandangan visual yang dapat dipahami bersama tentang struktur kelas dan interaksi antar mereka.
- h) Menyediakan dokumentasi yang berguna untuk memahami arsitektur sistem dan membantu dalam pengembangan, pemeliharaan, dan perubahan sistem pada tahap-tahap selanjutnya.

Komponen-komponen pada *class diagram*:

- a) *Class* (Kelas)
Class adalah elemen dasar dalam *class diagram* yang mewakili suatu tipe objek atau entitas dalam sistem.
- b) *Attribute* (Atribut)
Attribute adalah properti atau karakteristik dari suatu kelas yang menggambarkan data yang dimiliki oleh objek-objek dari kelas tersebut.
- c) *Method* (Metode)
Method adalah fungsi atau operasi yang dapat dilakukan oleh suatu kelas. *Method* mendefinisikan perilaku atau tindakan yang dapat dijalankan oleh objek.
- d) *Association* (Asosiasi)
Association menggambarkan hubungan antar dua atau lebih kelas dalam sistem. Ini menunjukkan bagaimana objek dari satu kelas berinteraksi atau terkait dengan objek dari kelas lain.
- e) *Composition* (Komposisi)
Composition adalah tipe asosiasi khusus yang menunjukkan bahwa objek dari satu kelas adalah bagian integral dari objek dari

kelas lain. Jika objek induk dihapus, maka objek bagian juga akan dihapus.

f) *Inheritance* (Warisan)

Inheritance menunjukkan hubungan hierarki antar kelas, di mana suatukelas dapat mewarisi atribut dan metode dari kelas lain.

g) *Aggregation* (Agregasi)

Aggregation adalah tipe asosiasi yang menunjukkan hubungan keseluruhan dan bagian antara dua kelas. Objek dalam hubungan *aggregation* tetap dapat eksis ketika objek keseluruhan dihapus.

h) *Abstract Class* (Kelas Abstrak)

Abstract class adalah kelas yang tidak dapat diinstansiasi dan sering berisi metode-metode abstrak yang harus diimplementasikan oleh kelasturunannya.

2.2.5.4 *Sequence Diagram*

Sequence Diagram menggambarkan perilaku objek dalam suatu *use case* dengan menjelaskan rentang waktu hidup objek dan pesan-pesan yang dikirimkan dan diterima antar objek (Simatupang & Sianturi, 2019). Tujuan utama dari pembuatan *sequence diagram* adalah untuk memahami rangkaian kejadian yang dapat menghasilkan output yang diinginkan. Selain itu, tujuan dari *sequence diagram* ini sebanding dengan diagram aktivitas, yang mencakup penggambaran alur kerja dari suatu aktivitas dan kemampuan untuk merinci aliran data dengan lebih rinci, termasuk data atau perilaku yang diterima atau dikirimkan.

Berikut fungsi dari *sequence diagram*:

- a) Memberikan representasi visual tentang urutan kejadian atau interaksi antar objek dalam suatu skenario atau *use case*.
- b) Mendeskripsikan rentang waktu hidup objek dalam sistem, mulai dari penciptaan objek hingga penghancuran atau selesai digunakan.
- c) Menggambarkan pesan-pesan yang dikirim dan diterima antar objek dalam urutan waktu tertentu, memperlihatkan bagaimana

objek berinteraksi satu sama lain.

- d) Menunjukkan koordinasi dan sinkronisasi antar objek dalam suatu skenario atau *use case*, membantu dalam pemahaman alur logika dan pemrosesan sistem.
- e) Membantu mengidentifikasi potensi kesalahan atau ketidakjelasan dalam desain atau logika sistem dan memfasilitasi pemecahan masalah.
- f) Digunakan untuk memodelkan logika bisnis atau proses dalam suatu skenario, membantu pengembang dan pemangku kepentingan memahami bagaimana sistem berperilaku.
- g) Menyediakan dokumentasi visual yang dapat dipahami dengan mudah untuk tim pengembang dan pemangku kepentingan, memfasilitasi komunikasi yang efektif dalam pengembangan perangkat lunak.

Komponen-komponen pada *sequence diagram*:

- a) *Actor* (Aktor)
Actor menggambarkan individu pengguna yang berinteraksi dengan sistem dan berada di luar lingkup sistem tersebut. Dalam *sequence diagram*, simbol *stick figure* umumnya dipakai untuk merepresentasikan aktor.
- b) *Activation Box* (Kotak Aktivasi)
Activation box menunjukkan periode waktu ketika objek sedang aktif atau menjalankan suatu operasi. *Activation box* ditempatkan di atas *lifeline* dan membantu memvisualisasikan durasi aktivitas objek.
- c) *Lifeline* (Waktu Hidup Objek)
Lifeline menggambarkan rentang waktu hidup objek dalam suatu skenario. Ini adalah garis vertikal yang mengindikasikan waktu objek aktif dalam interaksi.
- d) Objek
Objek adalah entitas yang berpartisipasi dalam interaksi. Objek direpresentasikan oleh nama kelas atau instance dari kelas tertentu, tergantung pada apakah objek tersebut merupakan kelas

atau instance.

e) *Message* (Pesan)

Message menggambarkan komunikasi atau interaksi antar objek.

Message diberi label dengan notasi panah dan dapat mencakup informasi tambahan seperti parameter atau nilai kembalian.

2.2.9 Metode Pengembangan Sistem Berbasis *Website*

Website atau situs web merupakan media yang terdiri dari banyak halamanyang saling terhubung melalui hyperlink. Situs web berfungsi untuk menyampaikan informasi dalam berbagai bentuk seperti teks, gambar, video, suara, animasi, atau kombinasi dari semuanya. Karakteristik utama dari situs web adalah adanya halaman-halaman yang saling terkait, dilengkapi dengan domain sebagai alamat dan hosting yang berfungsi sebagai penyimpanan data. Situs web dapat diakses melalui jaringan internet menggunakan peramban seperti Chrome, Mozilla Firefox, Internet Explorer, Opera, dan lain-lain.

Situs web atau aplikasi berbasis web biasanya dibangun menggunakan bahasa pemrograman seperti *Hypertext Preprocessor* (PHP) dan *Active Server Pages* (ASP), yang dipadukan dengan *Hypertext Markup Language* (HTML), *Cascading Style Sheet* (CSS), dan JavaScript (Elgamar, 2020).

2.2.6.1 *Hypertext Markup Language* (HTML)

Hypertext Markup Language (HTML) adalah suatu bahasa markah (*markup language*) yang digunakan untuk membuat dan mendesain struktur halaman web. HTML memberikan kerangka dasar atau struktur dasar untuk menyusun konten dalam sebuah halaman web. Dokumen HTML adalah dokumen teks yang dapat diedit menggunakan editor teks apapun dan disimpan dengan ekstensi file .html.

Dokumen HTML terdiri dari berbagai elemen yang dikelilingi oleh tag, yang dimulai dengan simbol “<” dan diakhiri dengan simbol “>” (Sari & Abdilah, 2019). HTML terdiri dari

berbagai elemen yang membangun struktur dasar halaman web, seperti elemen paragraf, heading, daftar, tautan, gambar, tabel, dan formulir. Elemen-elemen ini memungkinkan desainer web untuk mengatur konten secara terstruktur dan mudah diakses melalui browser. Selain itu, HTML sering digunakan bersama dengan CSS (*Cascading Style Sheets*) dan JavaScript untuk memperbaiki tampilan dan meningkatkan interaktivitas halaman web. CSS berfungsi mengatur gaya dan tata letak elemen HTML, sementara JavaScript menambahkan fungsi dinamis ke halaman web, seperti respons terhadap interaksi pengguna dan manipulasi konten secara real-time.

2.2.6.2 Hypertext Preprocessor (PHP)

Hypertext Preprocessor atau PHP adalah bahasa pemrograman sisi server yang dirancang khusus untuk pengembangan web. PHP dapat disisipkan ke dalam kode HTML dan dieksekusi di server web, menghasilkan konten dinamis yang kemudian ditampilkan di browser pengguna. Umumnya, PHP digunakan untuk membuat halaman web yang dapat berinteraksi dengan basis data, mengelola formulir, menghasilkan konten dinamis, serta menangani berbagai tugas pemrograman sisi server lainnya (Al Hadi, 2024).

Selain itu, PHP juga memiliki dukungan yang kuat untuk berbagai jenis basis data, seperti MySQL, yang memungkinkan pengguna untuk mengakses dan mengelola data secara efisien. Berikut kelebihan dari bahasa pemrograman PHP:

1. PHP dieksekusi di server, bukan di sisi klien. Hal ini output yang dihasilkan oleh PHP adalah HTML, yang kemudian dikirim ke browser pengguna. Kelebihan ini memberikan kontrol yang lebih baik atas sumber daya server dan memungkinkan pengembang untuk menyembunyikan implementasi detail dari pengguna.
2. PHP relatif mudah dipelajari, terutama bagi mereka yang sudah memiliki dasar pemrograman web. Syntax PHP mirip dengan C

dan Java, yang membuatnya mudah dipahami oleh banyak pengembang.

3. PHP bersifat open source dan gratis. Hal ini memudahkan pengembang untuk mengakses kode sumber, melakukan modifikasi, dan berbagi dengan komunitas.
4. PHP dapat terintegrasi dengan berbagai sistem basis data populer seperti MySQL, PostgreSQL, Oracle, dan banyak lainnya. Kemampuan ini sangat penting untuk mengembangkan aplikasi web yang memerlukan penyimpanan dan pengelolaan data.
5. PHP kompatibel dengan berbagai sistem operasi seperti Windows, Linux, Unix, dan macOS. Selain itu, mendukung berbagai server web seperti Apache, Nginx, dan IIS.

2.2.6.3 MySQL

MySQL adalah sistem manajemen basis data relasional (RDBMS) yang dirancang untuk menyimpan, mengelola, dan mengelola data dalam format tabel yang terstruktur. MySQL menggunakan bahasa SQL (*Structured Query Language*) sebagai cara untuk berinteraksi dengan basis data, memungkinkan pengguna untuk membuat, mengakses, memperbarui, dan menghapus data dengan mudah. Pada MySQL, setiap tabel memiliki baris yang mewakili entitas dan kolom yang mewakili atribut atau karakteristik dari entitas tersebut. MySQL menawarkan kemampuan akses yang cepat, efisiensi dalam penyimpanan data, serta mendukung transaksi yang memiliki peran sentral dalam pengembangan sistem informasi (Maulana, 2023).

Berikut adalah fitur utama dari MySQL:

1. MySQL mendukung prinsip ACID (*Atomicity, Consistency, Isolation, Durability*) yang memastikan keandalan dan konsistensi transaksi.
2. MySQL mendukung berbagai jenis penyimpanan tabel seperti MyISAM, InnoDB, MEMORY,

dan lainnya.

3. MySQL memungkinkan penggunaan indeks untuk meningkatkan kecepatan pencarian dan pengurangan waktu eksekusi kueri.
4. MySQL menyediakan alat-alat administrasi yang efisien seperti MySQLWorkbench untuk memudahkan pengelolaan basis data, membuat dan mengelola skema, dan menjalankan kueri.
5. MySQL mendukung pembatasan data dan triger yang memungkinkan pengembang untuk menentukan aturan perilaku tertentu pada tingkat basis data.

2.2.6.4 Visual Studio Code

Visual Studio Code merupakan editor kode sumber gratis yang dapat dijalankan pada perangkat desktop dengan sistem operasi Windows, MacOS, dan Linux yang dikembangkan oleh Microsoft. Visual Studio Code dapat digunakan untuk membuat dan mengedit sumber kode menggunakan berbagai bahasa pemrograman, termasuk Node.js, TypeScript, dan JavaScript. Selain itu, aplikasi ini juga kompatibel dengan bahasa dan lingkungan runtime lainnya seperti .NET, Java, Python, dan PHP dalam hal visualisasi.

Berikut adalah kelebihan Visual Studio Code:

1. Visual Studio Code mendukung banyak bahasa pemrograman termasuk JavaScript, TypeScript, Python, Java, C#, HTML, CSS, dan banyak bahasa lainnya.
2. Visual Studio Code memiliki ekosistem ekstensi yang besar dan aktif. Pengguna dapat menyesuaikan dan memperluas fungsionalitas editor dengan menginstal ekstensi untuk bahasa pemrograman, alat pengembangan, tema, dan lainnya.
3. Visual Studio Code memiliki integrasi Git yang terpadu, memungkinkan pengembang untuk melakukan operasi Git seperti commit, push, pull, dan lainnya langsung dari editor.
4. Visual Studio Code menyediakan terminal terpadu yang memungkinkan pengembang menjalankan perintah langsung dari

editor tanpa perlu beralih ke terminal eksternal.

5. Visual Studio Code memiliki *debugger* terintegrasi yang mendukung berbagai bahasa pemrograman. Hal ini memudahkan pengembang untuk mengidentifikasi dan memperbaiki *bug* dalam kode mereka.

2.2.10 Metode Pengujian Sistem

Untuk menjamin kualitas dan kesesuaian sistem perangkat lunak yang dikembangkan sesuai dengan spesifikasi yang telah ditetapkan, diperlukan proses pengujian. Pengujian yang digunakan adalah *Black Box* dan *White Box*.

2.2.7.1 Pengujian *Black Box*

Pengujian *Black Box* (Kotak Hitam) adalah metode pengujian perangkat lunak di mana tester tidak memerlukan pengetahuan internal atau rincian implementasi dari sistem yang diuji. Pengujian *Black Box* difokuskan pada *input* dan *output* dari program, tanpa mempertimbangkan rincian proses aplikasi (Nurfauziah & Jamaliyah, 2022). Pengujian *Black Box* ini, perangkat lunak dianggap sebagai "kotak hitam" di mana *input* diberikan dan *output* dinilai. Pengujian dengan menggunakan metode *Black Box Testing* memiliki tujuan untuk memverifikasi apakah perangkat lunak atau aplikasi yang diuji memenuhi standar kelayakan, sesuai dengan spesifikasi, dan memenuhi kebutuhan pengguna. Proses ini bertujuan untuk mencegah timbulnya masalah yang dapat terjadi pada penggunaan perangkat lunak atau aplikasi, sehingga ke depannya dapat dihindari potensi masalah yang dapat muncul akibat penggunaan tersebut.

2.2.7.2 Pengujian *White Box*

Pengujian *White Box* adalah metode pengujian yang berfokus pada internal sistem, yaitu pada sumber kode program. Tujuan dari pengujian *white box* adalah sebagai alat untuk menguji kompleksitas dari kode program (Dhaifullah, Salsabila, & Yaqin, 2022). Bagi

pengembang perangkat lunak, pengujian *white box* sangat penting untuk menilai seberapa kompleks suatu kode. Selain itu, pengujian *whitebox* juga dapat digunakan sebagai validasi untuk memastikan apakah sumber kode mengikuti desain, sesuai dengan kebutuhan fungsional, dan bebas dari kerentanan.

Menurut Nurfauziah & Jamaliyah (2022), berikut terdapat beberapa kelebihan dari *white box testing*:

1. Meningkatkan akurasi dalam implementasi perangkat lunak.
2. Mempermudah identifikasi kesalahan atau bug dalam perangkat lunak yang sebelumnya tidak terdeteksi.
3. Memudahkan proses pengujian secara menyeluruh, mengurangi kemungkinan terjadinya kesalahan pada kode.
4. Mengurangi potensi kesalahan atau bug karena pengujian dapat dilakukan sebelum perangkat lunak diluncurkan.