

BAB II

TINJAUAN PUSTAKA

Bab ini berisi hal-hal yang menjadi referensi bagi penulis dalam merancang dan mengembangkan aplikasi system pendukung keputusan pemilihan kendaraan listrik menggunakan metode *Simple Additive Weighting* (SAW).

2.1 Pencapaian Terdahulu

Dalam melakukan penelitian ini, penulis menggunakan beberapa referensi yang dijadikan acuan penulis dalam melakukan penelitian, yang diperoleh dari penelitian-penelitian yang sudah dilakukan terdahulu. Metode *Simple Additive Weighting* (SAW) ini banyak digunakan dalam penelitian-penelitian mengenai system pendukung keputusan (DSS), seperti penelitian “Sistem Pendukung Keputusan Pemilihan Unit Kendaraan Bermotor Dengan Metode *Simple Additive Weighting* (SAW)” (Fathul Ghina, dkk 2022) yang dipublikasi melalui jurnal OKTAL : Jurnal Ilmu Komputer dan *Science*. Pada penelitian ini dapat disimpulkan Metode *Simple Additive Weighting* (SAW) sederhana untuk sistem pendukung keputusan pemungutan suara. Produk mesin ini membuat aplikasi berguna dan menunjukkan hasil nilai yang sangat baik untuk unit mesin Honda ADV 150 CBS-ISS memberikan nilai 80, sehingga merupakan produk sepeda motor Honda ADV 150 CBS-ISS. rekomendasi produk yang tepat dan kriteria pelanggan penciptaan nilai paling tinggi.

Selain itu hasil dari penelitian “Sistem Pendukung Keputusan Pemilihan Mobil Bekas Dengan Metode AHP Dan SAW Pada Nava Sukses Motor” (Irwan Stiadi, 2019) yang merancang sistem untuk membantu konsumen dalam memilih mobil bekas, dan dipublikasikan Jurnal String Vol. 3 No. 3 April 2019. Sistem ini dapat membantu para calon konsumen dalam memilih mobil bekas pada *showroom* Nava Sukses Motor. Di mana hasil dari penelitian ini menyimpulkan bahwa sistem keputusan dengan metode AHP dan SAW bisa memilih mobil bekas dengan cepat dan akurat sesuai dengan kriteria terkonfirmasi Nava Motor *Showroom*. hasil

pengujian model System pendukung keputusan menggunakan uji koherensi proporsional (rasio koherensi) terhadap pemilihan mobil bekas memberikan nilai CR 0,024, jadi bisa dibilang begitu bahwa kriteria evaluasi sudah konsisten karena kurang dari 0,10.

Pencapaian terdahulu selanjutnya yaitu “Sistem Pendukung Keputusan Pemilihan Sepeda Motor Matic terbaik Pada Honda Mitra Cemerlang Dengan Metode *Simple Additive Weighting* (SAW)” (Ridarmin, Amat Sofyan, M.Wahyu.Irwansyah, 2019), yang merancang sistem pendukung keputusan pemilihan sepeda motor pada hondatama mitra cemerlang. Penelitian ini menyimpulkan bahwa aplikasi yang dibuat untuk mempermudah dan mempercepat pengambilan keputusan pemilihan sepeda motor berhasil dibuat dengan menggunakan metode *Simple Additive Weighting* (SAW).

Selain mengambil referensi dari pembuatan sistem pendukung keputusan (DSS) dengan metode *Simple Additive Weighting* (SAW), penulis juga mengambil referesi teori tentang pengembangan kendaraan listrik yang berjudul “Kajian pengembangan kendaraan listrik di Indonesia: prospek dan hambatannya” (Victor Tulus Pangapoi Sidabutar, 2020). Di mana hasil penelitian ini dapat disimpulkan pengembangan industri kendaraan listrik di Indonesia dalam jangka panjang ataupun pendek masih sulit dilakukan, hal ini dikarenakan :

- (1) Tingginya biaya dalam pengembangan kendaraan listrik.
- (2) Indonesia belum memiliki standar dalam memproduksi kendaraan listrik.
- (3) Indonesia belum memiliki infrastruktur pembangkit listrik yang mendukung jika diterapkan secara nasional karena kebutuhan listrik akan meningkat secara nasional.

Referensi teori selanjutnya yaitu referensi mengenai teorti dampak yang ditimbulkan oleh kendaraan bermotor konvensional dengan judul penelitian.” Efek Rumah Kaca Oleh Kendaraan Bermotor” (Alfi Kurnia, dan Sudarti, 2021) yang menarik kesimpulan bahwa. Kendaraan bermotor merupakan sumber emisi. Kelebihan kendaraan bermotor dapat menyebabkan emisi berlebih. Pesatnya jumlah kendaraan bermotor merupakan akibat dari kebutuhan dan tuntutan hidup manusia. Emisi mobil juga merupakan sumber utama pencemaran lingkungan karena menimbulkan pencemaran udara yang tidak baik bagi kesehatan manusia.

Berdasarkan beberapa penelitian yang sudah dilakukan terdahulu, maka penulis memilih untuk membangun sebuah sistem pendukung keputusan (DSS), dengan menggunakan metode *Simple Additive Weighting* (SAW). Dikarenakan metode SAW yang sederhana dan efektif untuk memudahkan calon konsumen dalam memilih kendaraan listrik.

2.2 Tinjauan Teoritis

Landasan teori merupakan teori-teori yang sudah dipelajari oleh penulis yang bertujuan mendukung pengetahuan penulis dalam melakukan penelitian. Dalam pembahasan kali ini, premis hipotetis merupakan bagian yang membahas tentang strategi dan teknik yang dilakukan pencipta dalam memimpin eksplorasi.

2.2.1 Mobil Listrik

Mobil listrik, sebuah inovasi yang membentuk masa depan transportasi, memberikan solusi yang efisien dan ramah lingkungan dalam penggunaan sumber daya energi. Konsep ini bekerja pada sistem motor listrik, meliputi motor listrik, pengontrol daya, dan baterai. Motor listrik merupakan bagian penting yang bertugas mengubah energi listrik menjadi gerak mekanis yang menggerakkan roda mobil. Selain itu, baterai juga berfungsi sebagai akumulator energi, menyediakan daya yang dibutuhkan motor dan membuka jalan bagi mobilitas berkelanjutan. Salah satu keunggulan utama mobil listrik adalah tidak mengeluarkan gas buang saat digunakan, sehingga memberikan solusi efektif terhadap masalah pencemaran udara dan dampak gas rumah kaca (Smith & Johnson, 2021).

Selain itu, keberlanjutan energi juga menjadi prioritas, dengan penggunaan sumber daya terbarukan sebagai sarana untuk mengisi ulang baterai (Smith & Johnson, 2021). Untuk mengoptimalkan kinerja baterai, teknologi baterai lithium-ion sering diterapkan dan dikelola melalui sistem manajemen baterai (BMS). Perkembangan teknologi baterai juga berdampak pada peningkatan jangkauan kendaraan listrik, sehingga mengurangi kekhawatiran akan keterbatasan jangkauan. Mendukung infrastruktur pengisian cepat merupakan faktor kunci dalam mengintegrasikan mobil listrik ke dalam kehidupan sehari-hari.

Menyediakan stasiun pengisian daya yang mudah diakses akan meningkatkan kenyamanan dan kepraktisan penggunaan kendaraan listrik. Aspek ekonomi juga ikut berperan karena total biaya kepemilikan (TCO) yang cenderung lebih rendah untuk mobil listrik. Hal ini didukung dengan biaya operasional dan pemeliharaan yang lebih efisien (Wang & Wu, 2019). Pada tataran kebijakan, insentif pemerintah seperti insentif perpajakan dan kebijakan yang berpihak pada mobil listrik berperan penting dalam mendorong penggunaan kendaraan ramah lingkungan (Wang & Wu, 2019).

2.2.2 Sistem Pendukung Keputusan

Sistem pendukung keputusan (DSS) memanfaatkan berbagai teori dan konsep untuk membantu membuat keputusan dengan lebih efektif. Inilah landasan teori yang mendukung pengembangan SPK:

- (1) Teori pengambilan keputusan: Teori ini mencakup dua pendekatan utama: pendekatan rasional dan pendekatan behavioral. Pendekatan rasional menekankan analisis dan perhitungan logis, sedangkan pendekatan behavioral mempertimbangkan faktor psikologis dan sosial yang mempengaruhi pengambilan keputusan.
- (2) Model keputusan: Model keputusan memberikan kerangka kerja untuk memahami langkah-langkah dalam proses pengambilan keputusan. Ini mencakup langkah-langkah definisi masalah, pengumpulan data, pemodelan, serta pemilihan dan implementasi Solusi.
- (3) Analisis multi-kriteria: Pendekatan ini membantu mengevaluasi alternatif berdasarkan beberapa kriteria. Metode AHP (*Analytical Hierarchy Process*) & TOPSIS (*Ideal Solution Prioritization Technique*) merupakan contoh metode analisis multi kriteria.

Ada banyak metode berbeda yang dapat digunakan dalam SPK, tergantung konteks dan kebutuhan pengambil keputusan. Beberapa metode yang populer antara lain:

- (1) AHP (*Analytic Hierarchy Process*): AHP adalah metode yang memungkinkan pengambil keputusan mengambil keputusan berdasarkan hierarki dengan membandingkan dan mengevaluasi kriteria dan alternatif bertahap.

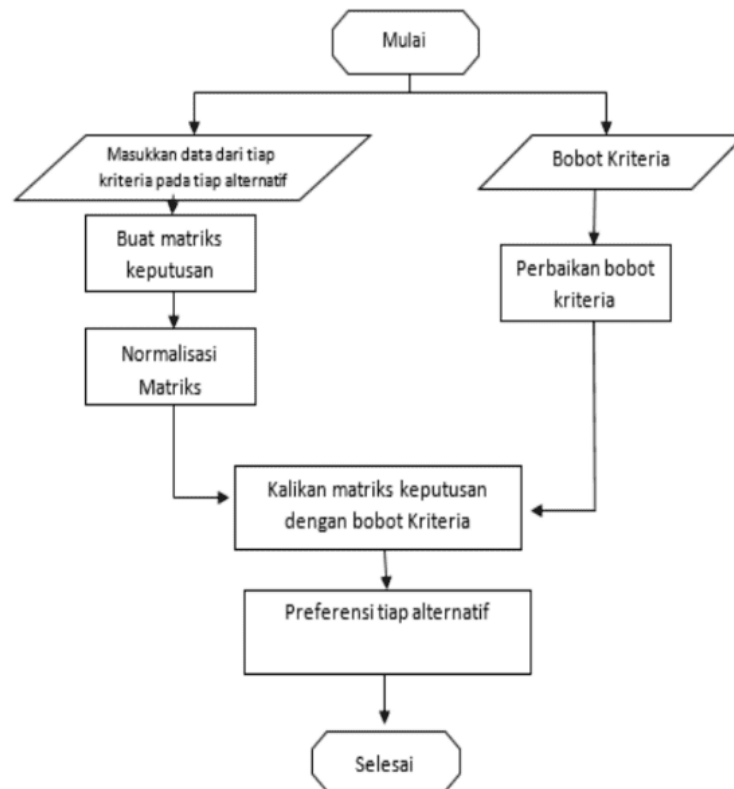
(2) TOPSIS (Teknik Penyortiran Topsis Berdasarkan Kesamaan dengan Solusi Ideal): TOPSIS menentukan solusi alternatif terbaik dengan membandingkan jarak relatif solusi alternatif terhadap solusi ideal, dan anti solusi ideal.

Simple Additive Weighting (SAW): Metode SAW memberikan bobot pada setiap kriteria dan menghitung nilai total untuk setiap alternatif. Solusi dengan nilai total tertinggi dianggap sebagai solusi terbaik.

Simple Additive Weighting (SAW)

Untuk pengambilan keputusan multi-baseline, metode *Simple Additive Weighting (SAW)* merupakan pendekatan langsung yang lugas. Teknik ini digunakan untuk membantu pembeli dalam memilih pilihan yang paling ideal dari standar penentuan yang telah ditentukan sebelumnya. Tujuan mendasar dari teknik SAW adalah untuk menghasilkan serangkaian pilihan berdasarkan standar tertentu. kaitannya dengan sistem pendukung keputusan, SAW bertujuan untuk menyederhanakan proses pengambilan keputusan dengan memberi bobot pada setiap kriteria dan menghitung nilai total dari setiap alternatif (Widodo, B. Sistem Pendukung Keputusan).

Metode SAW menggunakan pendekatan penjumlahan untuk mendapatkan alternatif terbaik. Nilai alternatif dihitung dengan cara mengalikan nilai kriteria dengan bobot kriteria kemudian menjumlahkan hasil perkalian dari masing-masing alternatif. Bobot kriteria menunjukkan tingkat kepentingan kriteria tersebut dan biasanya ditentukan oleh pengambil keputusan atau ahli yang bertanggung jawab.



Gambar 2.1 Flowchart Metode SAW (Dindo Sarwono, 2020, Vol. 20 No. 01)

Langkah-langkah proses SAW adalah sebagai berikut:

- (1) Tentukan kriteria yang relevan dan pilih opsi yang akan dievaluasi.
- (2) Normalisasikan setiap kriteria pada skala yang sama, misalnya skala 0-1 atau 1-5.

Berikut adalah rumus perhitungan normalisasi pada metode SAW :

$$r_{ij} = \begin{cases} \frac{x_{ij}}{\max_i x_{ij}} & \text{Jika } j \text{ merupakan atribut } \textit{benefit} \text{ (keuntungan)} \\ \frac{\min_i x_{ij}}{x_{ij}} & \text{Jika } j \text{ merupakan atribut } \textit{cost} \text{ (biaya)} \end{cases}$$

Keterangan :

r_{ij} = Hasil perhitungan normalisasi data kriteria yang ada

\max_{ij} = Nilai maksimal dari kriteria yang ada

Min_{ij}	= Nilai minimal dari kriteria yang ada
x_{ij}	= Nilai bobot dari baris dan kolom yang ada pada tabel data.
<i>Benefit</i>	= Kriteria keuntungan (semakin tinggi nilainya semakin baik)
<i>Cost</i>	= Kriteria biaya (semakin rendah nilainya semakin baik)

- (3) Mengetahui bobot kriteria yang digunakan dalam pengambilan keputusan, pada penelitian ini nilai bobot kriteria yang digunakan merupakan nilai yang *diinputkan* pengguna.
- (4) Kalikan skor kriteria dengan bobot kriteria masing-masing alternatif untuk menemukan nilai preferensi yang sesuai dengan data nilai bobot yang *diinputkan* oleh pengguna *website*.
- (5) Jumlahkan hasil perkalian kriteria dan bobot kriteria tiap alternatif.
- (6) Urutkan pilihan dari tertinggi ke terendah untuk mendapatkan pilihan terbaik.

Langkah-langkah algoritma yang diberlakukan pada sistem pendukung Keputusan metode SAW ini adalah sebagai berikut :

- (1) Inisialisasi data:
 - A. Matriks keputusan (X) adalah $m \times n$. Di mana m adalah banyaknya pilihan dan n adalah banyaknya kriteria.
 - B. Bobot kriteria (W) adalah $1 \times n$, yang menunjukkan pentingnya setiap kriteria.
- (2) Normalisasi matriks keputusan:

Normalisasikan nilai-nilai matriks keputusan agar mempunyai skala yang seragam dan tidak ada perbedaan satuan antar kriteria.
- (3) Hitung nilai total:

Hitung nilai total setiap pilihan dengan mengalikan nilai matriks keputusan yang dinormalisasi dengan bobot kriteria dan menjumlahkan hasilnya.
- (4) Peringkat Alternatif:

Urutkan alternatif berdasarkan nilai keseluruhan (tertinggi ke terendah).
- (5) Keluaran:

Alternatif dengan nilai total tertinggi merupakan solusi terbaik.

Perhitungan SAW membantu secara dinamis dengan memposisikan opsi berdasarkan ukuran kebutuhan yang telah ditentukan sebelumnya (Muhammad Rizky Ramadhan, Muhammad Khairul Nizam, Mesran. 2021) Metode SAW sangat mudah diimplementasikan dan dapat memberikan hasil yang memuaskan dalam pengambilan keputusan multikriteria. Namun metode ini memiliki beberapa kelemahan, seperti. Sensitif terhadap perubahan pembobotan kriteria dan tidak memperhatikan hubungan antar kriteria. Oleh karena itu, penggunaannya harus hati-hati, dan dapat dikombinasikan dengan metode lain untuk membuat keputusan yang lebih akurat dan tepat.

2.2.3 Metodologi Pengembangan Sistem (*Sequential linear*)

Metode pengembangan sistem sekuensial linier (*Sequential linear*), atau yang biasa dikenal dengan metode air terjun (*Waterfall*), adalah pendekatan pengembangan sistem perangkat lunak yang mengikuti urutan linier langkah demi langkah. Metodenya terstruktur dengan jelas mulai dari tahap perencanaan hingga tahap pemeliharaan (Sumantri, R Bagus Bambang , Willy Setiawan, and Deny Nugroho Triwibowo. 2022)

Tujuan utama dari metodologi *sequential linear* adalah untuk menyediakan kerangka terstruktur untuk pengembangan sistem yang memungkinkan pengendalian yang baik pada setiap fase. Dengan pendekatan ini, penulis berharap dapat mencapai kejelasan spesifikasi, pengembangan yang terorganisir, dan mengurangi risiko gangguan perubahan selama proses pengembangan (Sumantri, R Bagus Bambang , Willy Setiawan, and Deny Nugroho Triwibowo. 2022) Langkah-langkah dalam metode *sequential linear* adalah sebagai berikut :

- (1) Identifikasi persyaratan dan tujuan sistem.
Membuat rencana pengembangan, menentukan sumber daya dan waktu yang dibutuhkan.
- (2) Analisis Mengumpulkan kebutuhan pengguna dan Desain struktur sistem.
- (3) Desain sistem terperinci berdasarkan hasil analisis. Dan menentukan arsitektur sistem.
- (4) Implementasi Proses pembuatan sistem sesuai desain yang dibuat.

- (5) Pengujian sistem untuk memastikan bahwa sistem beroperasi sesuai spesifikasi.
- (6) Pemeliharaan dan perbaikan bug atau penyesuaian berdasarkan perubahan persyaratan.

Metode *Waterfall* memberikan kejelasan dan pengendalian yang baik dalam pengembangan sistem, namun kurang responsif terhadap perubahan kebutuhan yang terjadi pada saat proses pengembangan sistem (Boehm, B. W. *A Spiral Model of Software Development and Enhancement*).

2.2.4 Metode Pengembangan Sistem

UML (*Unified Modeling Language*) adalah bahasa standar untuk pelaporan, perencanaan, dan pembuatan kerangka pemrograman. UML memberikan dokumentasi dan aturan yang luas untuk mendemonstrasikan berbagai bagian kerangka kerja, misalnya struktur kelas, kolaborasi antar item, dan proses bisnis.. UML berarti mampu menjelaskan sesuatu proses pada sistem. simbol ada untuk membantu penulis menjelaskan hasil yang diharapkan. Simbol-simbol yang dimaksud disebut notasi dan proses (Rafel Limantoro, Dedy Prasetya Kristiad, 2021). UML dapat digunakan dalam berbagai metodologi pengembangan sistem. Diagram yang umum digunakan antara lain diagram aktivitas (*Activity Diagram*), diagram urutan (*Sequence Diagram*), dan diagram kelas

Use case diagram atau Diagram kasus pemanfaatan didasarkan pada statistik. Bagan ini menunjukkan bermacam-macam kasus penggunaan kerangka kerja dan penghibur. Grafik ini penting untuk menyusun dan menampilkan kerangka perilaku yang dibutuhkan dan diantisipasi klien (Rafel Limantoro, Dedy Prasetya Kristiad, 2021). Kasus pemanfaatan dapat menggambarkan jenis hubungan antara klien kerangka kerja dan kerangka kerja.

Kemampuan bagan kasus pemanfaatan adalah:

- a) Berguna untuk menampilkan aliran operasi dalam sistem secara berurutan.
- b) Mampu menggambarkan proses bisnis bahkan menunjukkan urutan operasi dalam proses tersebut.
- c) Sebagai jembatan atau penghubung antara produsen dan pembeli untuk menggambarkan system.

Komponen-komponen yang ada pada *use case diagram* :

a) Sistem

Berfungsi untuk membatasi *use case* dari interaksi luar system.

b) *Actor*

Actor berfungsi untuk memberikan gambaran hubungan dengan system.

c) *Use Case*

Kasus penggunaan adalah bagian gambaran praktis dari kerangka kerja. sehingga pembeli dan produsen dapat saling mengenal dan mempelajari proses pembangunan sistem.

Activity Diagram

Activity Diagram merupakan grafik yang dapat menunjukkan siklus yang terjadi dalam kerangka. Aliran siklus dalam kerangka digambarkan ke atas. Grafik tindakan merupakan penyempurnaan dari kasus pemanfaatan yang memiliki perkembangan latihan. Bagan tindakan bersifat dinamis. Secara umum Grafik pergerakan adalah jenis bagan keadaan luar biasa yang menunjukkan aliran yang dimulai dari satu tindakan lalu ke tindakan berikutnya dalam suatu kerangka (Rafel Limantoro, Dedy Prasetya Kristiad, 2021).

Fungsi dari *activity diagram* adalah :

a) Menggambarkan urutan kegiatan dalam sebuah proses.

b) Dalam dunia bisnis biasanya digunakan untuk modeling (menunjukkan urutan proses bisnis).

c) Proses yang mudah dipahami di seluruh sistem.

d) Metode ini adalah metode desain terstruktur yang mirip dengan Flowchart maupun *Data Flow Diagram* (DFD).

e) Mengetahui operasi *operator*/pengguna berdasarkan kasus/diagram penggunaan yang dibuat sebelumnya.

Komponen-komponen yang ada pada *Activity diagram* :

a) *Start Point* atau *Initial State* (Titik Mulai/Status Awal)

Start Point adalah lingkaran hitam kecil. Biasanya digunakan untuk menunjukkan keadaan awal, fungsi atau titik dari suatu diagram aktivitas.

b) *Activity* (Aktivitas)

Activity adalah tindakan yang dilakukan atau terjadi dalam sistem. Biasanya dimulai dengan "kata kerja" dari tindakan yang sedang dikerjakan.

c) *Decision* atau Percabangan

Decision atau Percabangan adalah titik atau titik yang menunjukkan keadaan dengan kemungkinan transisi lain. Ini diperlukan ketika ada banyak pilihan atau jalur alternatif dalam sistem.

d) *Synchronization*

Sinkronisasi dibagi menjadi 2 bagian yaitu *fork* dan *join*. *Fork* digunakan untuk membagi perilaku menjadi operasi paralel atau tindakan. *Join* digunakan untuk menghubungkan aktivitas ke aktivitas paralel.

e) *Merge*

Bergabung dengan aliran yang dipartisi menjadi beberapa bagian.

f) *Swimlanes*

Bagilah grafik pergerakan menjadi beberapa bagian dan garis untuk memisahkan kewajiban item yang memainkan tindakan tersebut.

g) *Transition*

Kemampuan untuk menunjukkan latihan langsung dan sebelumnya.

h) Notasi akhir (*end state*)

Dokumentasi akhir digunakan untuk menunjukkan akhir dari suatu interaksi.

Sequence Diagram

Sequence diagram atau bagan pengelompokan adalah grafik yang digunakan untuk memahami dan mengukur kerja sama antar objek dalam suatu kerangka. Selain itu, grafik pengaturan juga menunjukkan pesan atau perintah yang dikirim dan waktu pelaksanaannya. Biasanya, objek yang berhubungan dengan proses operasi disusun dari kiri ke kanan.

Alasan mendasar pembuatan grafik pengelompokan adalah untuk mengetahui rangkaian kejadian yang dapat memberikan hasil yang ideal. Selain itu, garis suksepsi memiliki alasan yang mirip dengan diagram pergerakan. Misalnya menggambarkan proses kerja suatu tindakan, dan dapat menggambarkan aliran informasi secara lebih rinci, termasuk informasi atau tindakan yang didapat atau dikirimkan..

Komponen-komponen yang ada pada *Sequence diagram* :

a) aktor

Bagian utama adalah aktor. Bagian ini menggambarkan klien yang berada di luar kerangka dan berinteraksi dengan kerangka tersebut. Dalam susunan grafis, penghibur biasanya disapa dengan simbol gambar tongkat.

b) *Activation box*

Berikutnya adalah kotak inisiasi. Bagian dari kotak pembuatan ini menunjukkan waktu yang diharapkan untuk menyelesaikan item tersebut. Semakin lama hal ini bertahan, semakin lama pula proses pemberlakuannya. Bagian ini dialamatkan dengan bentuk persegi panjang.

c) *Lifeline*

Kemudian datanglah lingkungan sehari-hari. Bagian-bagian ini dialamatkan dengan garis-garis yang dioles. *Lifeline* biasanya terdiri dari kotak berisi objek dengan deskripsi fungsinya.

d) obyek

Bagian selanjutnya adalah obyek. Bagian item ini digambarkan sebagai wadah yang berisi nama obyek dan garis bawah. Biasanya, objek berfungsi untuk mendokumentasikan perilaku sistem.

e) *Messages*

Pesan merupakan barang akhir. Bagian ini menggambarkan korespondensi antar objek. Pesan-pesan ini biasanya muncul satu per satu di baris pemberitahuan awal. Bagian dari pesan yang dikirim oleh baut. Inti dari grafik suksesi terletak pada bagian yang membatasi dan pesan ini.

Class Diagram

Diagram kelas adalah garis besar struktur UML yang menggambarkan secara gamblang konstruksi dan penggambaran kelas setiap item, kualitas, strategi, dan hubungan. Statis seperti pada bagan kelas tidak memahami apa yang terjadi ketika kelas-kelas terhubung satu sama lain, namun tetap memahami apa hubungan tersebut.

Diagram kelas ini memiliki beberapa kemampuan, kemampuan utamanya adalah menggambarkan kerangka kerja yang mendasarinya. Elemen lain dicatat di bawah ini :

- a) Menunjukkan struktur sistem dengan jelas.
- b) Memberikan pemahaman yang unggul mengenai garis besar program.
- c) Dapat digunakan untuk pemeriksaan bisnis dan tampilan kerangka sisi bisnis.
Mampu memberikan garis besar tentang suatu framework atau pemrograman dan hubungan-hubungan yang ada di dalamnya.

2.2.5 Metode Pengembangan Sistem Berbasis *Website*

Website adalah kumpulan halaman atau laporan terkait yang dapat diakses melalui Web dan diperkenalkan dalam desain *hypertext*. Halaman beranda, halaman kontak, atau halaman produk adalah contoh halaman multimedia yang biasanya berisi teks, gambar, video, atau elemen multimedia lainnya. Situs dibuat menggunakan dialek pemrograman web seperti HTML, CSS, dan *JavaScript*. Pada awalnya, situs dibuat khusus untuk memperkenalkan konten tertentu, namun dalam jangka panjang situs juga digunakan untuk tujuan bisnis seperti belanja internet dan periklanan..

Situs ini dapat diakses oleh siapa saja yang memiliki koneksi web dan program, misalnya *Google Chrome*, *Mozilla Firefox*, atau *Safari*. Ada berbagai jenis situs web termasuk situs web statis yang berisi konten tetap dan situs web dinamis yang mengadaptasi konten berdasarkan interaksi pengguna. Selain itu, ada juga *website* responsif yang dapat menyesuaikan tata letak dan ukuran halaman dengan perangkat yang digunakan pengguna, seperti komputer *desktop*, laptop, tablet, atau *smartphone*.

Website dapat digunakan untuk berbagai tujuan, termasuk menyediakan informasi, berinteraksi dengan pengguna, mengiklankan dan menjual produk, menyediakan layanan dan dukungan, atau memposting dan berbagi informasi di antara komunitas tertentu. Situs web juga dapat menjadi alat penting untuk membangun merek dan meningkatkan *visibilitas online*.

HTML

HTML (*HyperText Markup Language*) adalah bahasa *markup* yang digunakan untuk membuat situs dan konten berbasis web lainnya. HTML digunakan untuk menyusun arsip web, menentukan desain dan penataan, serta menambahkan komponen media interaktif seperti gambar, video, dan suara ke dalam halaman.

Dalam HTML, struktur laporan dikoordinasikan menggunakan label yang dibungkus dalam bagian persegi (<>). Pada suatu halaman web, setiap tag HTML dapat menampilkan teks, gambar, link, tabel, atau elemen multimedia lainnya. Setiap tag HTML memiliki tujuan tertentu. HTML juga mendukung atribut seperti ukuran, warna, dan jenis *font* yang memberikan informasi tambahan tentang tag tertentu. Selain tag dan atribut, HTML juga menggunakan elemen seperti paragraf, judul, dan daftar untuk menyediakan struktur dan tata letak dokumen web. Struktur dokumen web yang dibuat dari HTML dapat diatur dengan berbagai cara menggunakan CSS (*Cascading Style Sheets*) dan *JavaScript*.

HTML adalah salah satu kemajuan utama dalam pembuatan situs dan konten online. Oleh karena itu, sangat penting untuk memahami dan menguasai dasar-dasar HTML untuk membuat situs dapat diakses dan dimanfaatkan dengan baik oleh klien. Saat ini, bentuk HTML yang paling umum digunakan adalah HTML 5, yang memberikan elemen baru untuk membuat halaman lebih *intuitif* dan *responsif*.

PHP

PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman yang biasanya digunakan untuk membuat aplikasi web yang dinamis dan intuitif. PHP berjalan di server dan digunakan untuk membuat substansi dinamis muncul di halaman web.

PHP memungkinkan pembuat situs web untuk menambahkan fitur interaktif ke situs web seperti formulir, sistem masuk, dan basis data. Bahasa pemrograman ini memiliki tanda baca seperti bahasa pemrograman C dan digunakan oleh sebagian besar desainer web karena kebaikan dan kemudahan pencampurannya dengan HTML.

Beberapa kelebihan dari PHP adalah:

- (1) PHP gratis dan open source

- (2) Mendukung banyak sistem operasi, seperti Windows, Linux dan macOS
- (3) Dapat berintegrasi dengan berbagai *database*, seperti MySQL, Oracle dan PostgreSQL
- (4) Dapat digunakan untuk membangun aplikasi web dan API
- (5) Mudah dipelajari dan digunakan oleh pemula

Bagaimanapun, PHP juga memiliki beberapa kekurangan, misalnya masalah keamanan dan kelemahan. Oleh karena itu, diperlukan pemahaman yang baik tentang keamanan web dan praktik terbaik untuk membuat aplikasi PHP yang aman dan solid.

MySql

MySQL, sistem manajemen basis data relasional sumber terbuka, sering digunakan dalam pembuatan perangkat lunak dan aplikasi web. MySQL adalah salah satu kerangka kerja manajemen kumpulan data (RDBMS) yang paling terkenal dan paling banyak digunakan di dunia. MySQL awalnya dikembangkan oleh MySQL Stomach Muscle (saat ini diklaim oleh *Prophet Enterprise*) pada tahun 1995 dan dikirimkan di bawah izin open source.

MySQL menggunakan SQL (*Organized Question Language*) untuk mengontrol informasi dalam kumpulan data sosial. MySQL dapat diinstal dan dijalankan pada berbagai sistem operasi termasuk Windows, Linux, dan macOS. MySQL juga menjunjung banyak dialek pemrograman, seperti PHP, Java, dan Python, dan dapat digunakan dalam berbagai macam kegunaan, seperti aplikasi web, aplikasi area kerja, dan permainan.

Beberapa fitur utama dari MySQL adalah:

- (1) Mendukung banyak jenis tabel dan tipe data
- (2) Mampu menangani *database* besar dan kinerja cepat
- (3) Mendukung transaksi ACID (*Atomicity, Consistency, Isolation, Durability*)
- (4) Memiliki alat pemantauan dan manajemen yang kuat
- (5) Memiliki fitur keamanan dan autentikasi yang dapat dikonfigurasi
- (6) MySql bersifat *open source* dan gratis untuk digunakan

MySQL digunakan secara luas baik oleh perusahaan besar maupun proyek sumber terbuka di seluruh dunia. Oleh karena itu, MySQL adalah keterampilan

yang penting untuk dikuasai dalam perangkat lunak dan pengembangan aplikasi web.

Visual Studio Code

Visual Studio Code adalah pengelola kode sumber lintas tahap yang dibuat oleh *Microsoft*. *Visual Studio Code* atau biasa disebut *Versus Code* memiliki elemen berbeda yang memudahkan desainer dalam membuat kode, misalnya pemecahan masalah, potongan kode, pemenuhan otomatis, dan koordinasi dengan *Git*. *Versus Code* dapat digunakan untuk membuat kode dalam berbagai dialek pemrograman, seperti *JavaScript*, *Python*, *PHP*, *Ruby*, dan lain-lain. Selain itu, *Versus Code* juga dapat diubah dengan berbagai topik dan penambahan yang meningkatkan fungsi dan kegunaannya. *Visual Studio Code* adalah salah satu editor kode sumber paling terkenal dan umumnya digunakan oleh desainer di mana pun. *Versus Code* adalah pemrograman sumber terbuka dan dapat diunduh secara gratis dari situs *Microsoft*.

XAMPP

XAMPP adalah paket produk yang digunakan untuk membangun dan menjalankan kondisi pengembangan web lokal. XAMPP menggabungkan berbagai aplikasi dan bagian yang diperlukan untuk membangun dan menjalankan aplikasi web, misalnya server web Apache, bahasa pemrograman PHP, dan basis informasi MySQL.

XAMPP dapat diterapkan pada banyak kerangka kerja, termasuk Windows, Linux, dan macOS, dan dapat digunakan untuk mengembangkan aplikasi web yang mbingungkan dan dinamis. Dengan XAMPP, pengembang dapat menguji aplikasi web mereka di lingkungan lokal sebelum memublikasikannya ke server langsung. XAMPP sangat berguna untuk pengembang web karena dapat mempercepat waktu pengembangan dan memungkinkan pengembang menguji aplikasi web mereka secara lokal sebelum menyebarkannya ke server langsung.

2.2.6 Metode Pengujian Sistem

Untuk menjamin kerangka yang dibuat berjalan sesuai keinginan dan target eksplorasi ini. Penulis mencoba *framework* tersebut menggunakan dua teknik pengujian, yaitu :

Pengujian *Black-box*

Pengujian kotak hitam atau *black box testing* adalah metode pengujian perangkat lunak yang berfokus pada pengujian fungsionalitas sistem tanpa memperhatikan detail internal dari operasi sistem. Pendekatan ini memandang perangkat lunak sebagai "kotak hitam" yang dapat diuji *input* dan *output* tanpa memperhatikan logika internal atau detail implementasi. Pengujian *black box* dilakukan dengan memasukkan rangkaian masukan ke dalam sistem dan mengamati keluarannya. Pengujian ini dimaksudkan untuk memastikan bahwa sistem beroperasi sesuai spesifikasi dan dapat memberikan keluaran yang diharapkan pada berbagai kondisi masukan.

Dalam pengujian kotak hitam, tidak diperlukan pengetahuan mendetail tentang operasi sistem, sehingga pengujian dapat dilakukan oleh orang tanpa pengetahuan teknis yang tinggi. Namun pengujian *black box* memiliki kelemahan yaitu sulit untuk menemukan kesalahan pada bagian internal sistem, seperti algoritma atau kinerja sistem. Pengujian *black box* dapat dilakukan dengan menggunakan beberapa teknik, seperti pengujian batas, pengujian pengguna, pengujian *use case*, pengujian fungsional, dan pengujian non-fungsional. Setelah pengujian selesai, hasil pengujian kotak hitam akan memberikan informasi tentang kesalahan atau cacat sistem, yang kemudian dapat diperbaiki oleh tim pengembangan perangkat lunak.

Pengujian *White-Box*

Pengujian kotak putih, juga dikenal sebagai *white box testing*, adalah metode pengujian perangkat lunak yang memeriksa struktur *internal* dan *implementasi* sistem yang diuji. Dalam jenis pengujian ini, pengujian dilakukan dengan akses ke kode sumber sistem, arsitektur, dan dokumen lainnya. Pengujian kotak putih melibatkan pengujian unit, integrasi, dan sistem untuk memastikan bahwa setiap

bagian dari sistem bekerja dengan baik dan sesuai dengan spesifikasi yang telah ditentukan. Oleh karena itu, pengujian *white box* memerlukan pengetahuan teknis yang mendalam tentang sistem dan bahasa pemrograman yang digunakan. Metode ini berguna untuk menemukan kemungkinan kesalahan logika dan kinerja dalam sistem, serta memastikan bahwa kode ditulis sesuai dengan standar pemrograman yang berlaku.

