

BAB II

KAJIAN PUSTAKA

A. Kajian Teoritis

1. Rancang Bangun

Rancang bangun merupakan suatu proses yang dilakukan untuk merencanakan, merancang, dan mengembangkan sebuah sistem guna menyelesaikan permasalahan tertentu sesuai dengan kebutuhan pengguna. Rancang bangun tidak hanya berfokus pada pembuatan sistem, tetapi juga melibatkan identifikasi permasalahan dan kebutuhan pengguna, perancangan solusi, implementasi, serta pengujian terhadap sistem yang telah dikembangkan.

Menurut (Desmonda Supriatna, Kasim and Rabbani, 2025), proses rancang bangun sistem informasi dapat dilakukan secara terstruktur melalui tahapan analisis kebutuhan sistem, perancangan sistem, implementasi, dan pengujian. Penerapan tahapan tersebut digunakan untuk menghasilkan sistem terkomputerisasi yang mampu mengatasi permasalahan pada proses manual serta mendukung pengelolaan data agar menjadi lebih cepat, akurat, dan terintegrasi.

Menurut (Amanda and Firdonsyah, 2021), rancang bangun sistem informasi berbasis web dapat digunakan sebagai solusi terhadap permasalahan pengelolaan data yang masih dilakukan secara manual. Pengembangan sistem dilakukan dengan mengidentifikasi kebutuhan, merancang dan mengimplementasikan sistem, serta melakukan pengujian

untuk memastikan fungsi sistem berjalan sesuai dengan yang direncanakan. Hasil penerapan sistem informasi dapat membantu meningkatkan efisiensi pengelolaan data, mempercepat proses administrasi, serta meningkatkan akurasi dan keamanan informasi.

Menurut (Ramdhani and Thantawi, 2024), rancang bangun sistem informasi merupakan proses pembangunan sistem berbasis web yang dilakukan secara terstruktur untuk menyelesaikan permasalahan pengelolaan data pada suatu instansi. Proses tersebut melibatkan analisis kebutuhan, perancangan sistem, implementasi, dan pengujian sehingga menghasilkan sistem yang mampu meningkatkan efektivitas pengelolaan informasi bagi pengguna.

(Sosial *et al.*, 2024) menjelaskan bahwa rancang bangun sistem informasi manajemen dilakukan untuk menciptakan solusi terkomputerisasi yang mampu membantu pengguna dalam mengelola data secara lebih efisien dan terintegrasi. Melalui proses rancang bangun, sistem yang sebelumnya dilakukan secara manual dapat diubah menjadi sistem yang terkomputerisasi sehingga mampu meningkatkan kualitas pelayanan dan produktivitas.

Dalam penelitian ini, rancang bangun digunakan sebagai proses pengembangan Sistem Pendukung Keputusan Pemilihan Program Studi perguruan tinggi yang bersifat universal. Sistem yang dibangun bertujuan untuk membantu siswa dalam menentukan program studi yang sesuai dengan kemampuan akademik, minat, dan bakat yang dimiliki melalui penerapan metode *Simple Additive Weighting* (SAW), dengan SMA Negeri 4 Kota

Madiun sebagai lokasi studi kasus penelitian.

Berdasarkan beberapa pendapat tersebut, dapat disimpulkan bahwa rancang bangun merupakan proses sistematis yang dilakukan untuk merancang, mengembangkan, dan mengimplementasikan suatu sistem guna menyelesaikan permasalahan tertentu sehingga dapat menghasilkan solusi yang efektif, efisien, dan sesuai dengan kebutuhan pengguna.

2. Sistem Pendukung Keputusan (SPK)

Sistem Pendukung Keputusan (SPK) merupakan sistem berbasis komputer yang dirancang untuk membantu proses pengambilan keputusan dalam menyelesaikan permasalahan semi-terstruktur maupun tidak terstruktur. Sistem ini memanfaatkan data dan model tertentu untuk membantu pengambil keputusan dalam memperoleh informasi yang dapat digunakan sebagai dasar dalam menentukan keputusan. Selain menyajikan data, SPK juga dapat memberikan informasi yang lebih mendalam melalui berbagai teknik analisis sehingga proses pengambilan keputusan dapat dilakukan secara lebih efektif dan efisien.

Menurut (Abdul 'Afwa and Sulisty Nugroho, 2026), Sistem Pendukung Keputusan merupakan sistem berbasis komputer yang membantu proses pengambilan keputusan dengan memanfaatkan data, model analisis, serta metode evaluasi yang terstruktur. SPK tidak berfungsi untuk menggantikan peran manusia dalam menentukan keputusan, melainkan memberikan informasi yang relevan dan alternatif terbaik dari berbagai

pilihan yang tersedia. Selain itu, SPK memiliki kemampuan dalam mengolah dan menganalisis data secara cepat sehingga dapat mendukung pengambilan keputusan yang lebih objektif dan terukur.

Sementara itu, (Vianingrum and Zulherry, 2026) menjelaskan bahwa tujuan utama Sistem Pendukung Keputusan adalah meningkatkan efisiensi dalam proses pengambilan keputusan melalui penyediaan analisis data yang lebih baik dan cepat. Dalam penerapannya, SPK tidak hanya digunakan untuk menyajikan data, tetapi juga dapat mengotomatisasi proses pengolahan data, melakukan analisis, serta menghasilkan rekomendasi berbasis data yang dapat membantu pengguna dalam menentukan keputusan..

3. Metode Simple Additive Weighting (SAW)

Menurut (Martin and Yudhistira, 2026), metode *Simple Additive Weighting* (SAW) merupakan salah satu metode dalam sistem pendukung keputusan yang termasuk dalam kategori *Multi Attribute Decision Making* (MADM). Metode ini sering dikenal sebagai metode penjumlahan terbobot karena konsep dasarnya adalah mencari penjumlahan terbobot dari rating kinerja setiap alternatif pada semua kriteria. Inti dari metode SAW adalah mengalikan nilai kinerja setiap alternatif dengan bobot masing-masing kriteria setelah nilai alternatif tersebut dinormalisasi.

Adapun tahapan perhitungan dalam metode SAW adalah sebagai berikut:

a. Menentukan Kriteria dan Bobot

Langkah pertama adalah menentukan kriteria-kriteria (C) yang akan dijadikan acuan dalam pengambilan keputusan beserta bobot kepentingan (W) masing-masing kriteria. Bobot setiap kriteria dinyatakan dalam nilai antara 0 hingga 1, di mana total seluruh bobot bernilai 1 (atau 100%).

b. Pembentukan Matriks Keputusan

Setelah kriteria ditentukan, dibentuk matriks keputusan X yang berisi nilai rating kinerja setiap alternatif (A_i) pada setiap kriteria (C_j). Matriks keputusan X didefinisikan sebagai berikut:

$$X = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{pmatrix}$$

Keterangan:

- 1) x_{ij} = nilai rating kinerja alternatif ke-i pada kriteria ke-j
- 2) m = jumlah alternatif
- 3) n = jumlah kriteria

c. Normalisasi Matriks Keputusan

Setiap nilai pada matriks keputusan dinormalisasi menggunakan rumus yang berbeda tergantung pada jenis kriteria, yaitu kriteria **benefit** (semakin besar nilainya semakin baik) dan kriteria **cost** (semakin kecil nilainya semakin baik).

Rumus normalisasi SAW adalah sebagai berikut (Kusumadewi et al., 2006):

$$r_{ij} = \begin{cases} \frac{x_{ij}}{\max_i(x_{ij})} & \text{jika } j \text{ adalah kriteria benefit} \\ \frac{\min_i(x_{ij})}{x_{ij}} & \text{jika } j \text{ adalah kriteria cost} \end{cases}$$

Keterangan:

- 1) r_{ij} = nilai rating ternormalisasi alternatif ke-i pada kriteria ke-j
- 2) x_{ij} = nilai rating kinerja alternatif ke-i pada kriteria ke-j
- 3) $\max(x_{ij})$ = nilai terbesar dari setiap kolom kriteria ke-j
- 4) $\min(x_{ij})$ = nilai terkecil dari setiap kolom kriteria ke-j

Hasil normalisasi tersebut membentuk matriks ternormalisasi R:

$$R = \begin{pmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ r_{21} & r_{22} & \cdots & r_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ r_{m1} & r_{m2} & \cdots & r_{mn} \end{pmatrix}$$

d. Menghitung Nilai Preferensi

Tahap akhir adalah menghitung nilai preferensi (V) untuk setiap alternatif dengan cara menjumlahkan hasil perkalian antara nilai normalisasi (r_{ij}) dengan bobot kriteria (W_j). Rumusnya adalah sebagai berikut (Kusumadewi et al., 2006):

$$V_i = \sum_{j=1}^n w_j \cdot r_{ij}$$

Keterangan:

- a) V_i = nilai preferensi alternatif ke-i
- b) w_j = bobot kriteria ke-j
- c) r_{ij} = nilai rating ternormalisasi alternatif ke-i pada kriteria ke-j

Alternatif yang memiliki nilai preferensi V_i tertinggi dinyatakan sebagai alternatif terbaik dan menjadi rekomendasi utama yang dihasilkan oleh sistem.

e. Contoh Penerapan dalam Penelitian

Dalam penelitian ini, alternatif yang digunakan adalah program studi yang tersedia di Perguruan tinggi, sedangkan kriteria yang digunakan meliputi:

- a) C1 = Nilai Akademik (benefit) — semakin tinggi nilai rapor siswa, semakin sesuai dengan program studi tersebut
- b) C2 = Minat Siswa (benefit) — diperoleh dari hasil survei minat
- c) C3 = Bakat Siswa (benefit) — diperoleh dari hasil survei bakat

Karena seluruh kriteria dalam penelitian ini bersifat benefit, maka normalisasi menggunakan rumus: $rij = x_{ij} / \max(x_{ij})$. Hasil normalisasi kemudian dikalikan dengan bobot masing-masing kriteria dan dijumlahkan untuk menghasilkan nilai preferensi V_i setiap program studi. Program studi dengan nilai V_i tertinggi direkomendasikan sebagai pilihan yang paling sesuai bagi siswa.

Berdasarkan penjelasan tersebut, dapat disimpulkan bahwa metode *Simple Additive Weighting* (SAW) merupakan metode pengambilan keputusan multi kriteria yang bekerja melalui proses normalisasi nilai dan penjumlahan terbobot sehingga menghasilkan perankingan alternatif secara objektif dan terukur.

4. Program Studi

Program studi merupakan kesatuan kegiatan pendidikan dan

pembelajaran yang memiliki kurikulum serta metode pembelajaran tertentu dalam satu jenis pendidikan akademik, pendidikan profesi, maupun pendidikan vokasi. Program studi diselenggarakan oleh perguruan tinggi sebagai wadah bagi mahasiswa untuk memperoleh pengetahuan, keterampilan, dan kompetensi sesuai dengan bidang ilmu yang dipilih. Setiap program studi memiliki karakteristik dan kompetensi yang berbeda sehingga pemilihan program studi menjadi salah satu keputusan penting bagi calon mahasiswa dalam melanjutkan pendidikan ke jenjang perguruan tinggi.

Menurut Undang-Undang Republik Indonesia Nomor 12 Tahun 2012 tentang Pendidikan Tinggi, program studi adalah kesatuan kegiatan pendidikan dan pembelajaran yang memiliki kurikulum dan metode pembelajaran tertentu dalam satu jenis pendidikan akademik, pendidikan profesi, dan/atau pendidikan vokasi. Program studi menjadi salah satu unsur dalam penyelenggaraan pendidikan tinggi yang digunakan sebagai wadah pelaksanaan proses pendidikan dan pembelajaran sesuai dengan bidang keilmuan tertentu.

Pemilihan program studi merupakan salah satu tahapan penting bagi siswa yang akan melanjutkan pendidikan ke perguruan tinggi. Banyaknya program studi yang tersedia dapat menyebabkan siswa mengalami kesulitan dalam menentukan pilihan yang sesuai dengan dirinya. (Pakarti, Akbar and Rais, 2024) menjelaskan bahwa banyaknya pilihan program studi dapat menimbulkan kebingungan bagi siswa dalam menentukan program studi yang tepat. Dalam penelitian tersebut, pemilihan program studi dilakukan

berdasarkan minat dan bakat siswa yang direpresentasikan melalui lima kriteria, yaitu kemampuan logika, spasial, praktikum, sosial, dan bahasa.

Penggunaan Sistem Pendukung Keputusan dalam pemilihan program studi dapat membantu calon mahasiswa dalam menentukan pilihan berdasarkan sejumlah kriteria secara lebih terstruktur. Menurut (Halim *et al.*, 2022), pemilihan jurusan pada perguruan tinggi perlu mempertimbangkan minat, bakat, dan kemampuan yang dimiliki oleh calon mahasiswa. Banyaknya pilihan jurusan dapat menyebabkan calon mahasiswa mengalami kesulitan dalam menentukan jurusan yang sesuai. Oleh karena itu, Sistem Pendukung Keputusan dengan metode Simple Additive Weighting (SAW) dapat digunakan untuk membantu proses pemilihan dengan melakukan perhitungan nilai bobot terhadap setiap alternatif berdasarkan kriteria yang telah ditentukan, sehingga menghasilkan rekomendasi jurusan kuliah yang sesuai dengan karakteristik calon mahasiswa.

Selain itu, kemampuan serta karakteristik yang dimiliki calon mahasiswa dapat menjadi pertimbangan dalam menentukan program studi. (Marif, Azmi and Alanda, 2022) mengembangkan Sistem Pendukung Keputusan untuk membantu calon mahasiswa dalam menentukan program studi yang sesuai dengan kemampuannya. Penelitian tersebut menggunakan metode Analytic Hierarchy Process (AHP) untuk melakukan pembobotan kriteria dan Simple Additive Weighting (SAW) untuk menentukan rekomendasi program studi. Kriteria diperoleh melalui pertanyaan dalam bentuk kuesioner, sedangkan aspek yang digunakan dalam proses penentuan

program studi mencakup minat, bakat, dan hobi. Sistem yang dikembangkan bertujuan memberikan saran kepada calon mahasiswa sehingga proses pemilihan program studi dapat dilakukan berdasarkan kriteria yang telah ditentukan.

Penggunaan Sistem Pendukung Keputusan dalam pemilihan program studi dapat membantu siswa mempertimbangkan beberapa kriteria secara lebih terstruktur. (Halim *et al.*, 2022)menerapkan metode Simple Additive Weighting (SAW) dengan menggunakan nilai, minat, ketersediaan fasilitas, peluang karier, dan biaya kuliah sebagai kriteria penilaian. Hasil penelitian menunjukkan bahwa sistem pendukung keputusan dapat membantu proses pengolahan kriteria dan pembobotan dalam memberikan rekomendasi program studi kepada calon mahasiswa.

Penelitian lain yang dilakukan oleh (Pakarti, Akbar and Rais, 2024) menerapkan Sistem Pendukung Keputusan berbasis Analytical Hierarchy Process (AHP) untuk membantu siswa SMA dalam menentukan program studi pada perguruan tinggi. Sistem tersebut menggunakan minat dan bakat siswa sebagai dasar pemilihan program studi dengan mempertimbangkan kemampuan logika, spasial, praktikum, sosial, dan bahasa. Sistem dikembangkan berbasis website agar dapat digunakan secara fleksibel melalui perangkat komputer maupun perangkat lainnya. Penerapan sistem tersebut diharapkan dapat membantu siswa menghasilkan keputusan yang lebih baik dan lebih terinformasi dalam memilih program studi di perguruan tinggi.

Dalam penelitian ini, program studi digunakan sebagai alternatif keputusan yang akan dinilai oleh Sistem Pendukung Keputusan. Data program studi dapat dikonfigurasi secara fleksibel oleh Admin sesuai dengan kebutuhan sekolah. Dengan pendekatan tersebut, sistem tidak terbatas pada program studi dari satu perguruan tinggi tertentu, tetapi dapat digunakan untuk mengelola alternatif program studi dari berbagai perguruan tinggi. Setiap alternatif selanjutnya diproses berdasarkan kriteria dan bobot yang telah ditentukan sehingga sistem dapat menghasilkan rekomendasi program studi yang sesuai dengan hasil penilaian siswa.

Berdasarkan uraian tersebut, dapat disimpulkan bahwa program studi merupakan satuan kegiatan pendidikan dan pembelajaran pada perguruan tinggi yang memiliki kurikulum, metode pembelajaran, serta bidang keilmuan tertentu. Pemilihan program studi merupakan keputusan penting bagi calon mahasiswa karena ketepatan pemilihan program studi dapat berkaitan dengan keberhasilan dalam menjalani proses pendidikan. Oleh karena itu, Sistem Pendukung Keputusan dapat digunakan untuk membantu siswa mempertimbangkan berbagai kriteria seperti kemampuan akademik, minat, bakat, dan faktor pendukung lainnya sehingga proses pemilihan program studi dapat dilakukan secara lebih terstruktur..

5. Website

Website merupakan kumpulan halaman yang saling terhubung dan dapat diakses melalui jaringan internet menggunakan *web browser*. Website digunakan sebagai media untuk menyampaikan informasi, mengelola data,

serta menyediakan layanan yang dapat diakses oleh pengguna secara online. Seiring berkembangnya teknologi informasi, *website* menjadi salah satu platform yang banyak digunakan dalam pengembangan sistem informasi karena mampu memberikan kemudahan akses dan pengelolaan data secara terpusat.

Menurut (Maulidda and Jaya, 2022), *website* merupakan media berbasis internet yang digunakan untuk menyampaikan informasi dan menyediakan layanan digital kepada pengguna melalui halaman-halaman yang saling terhubung. *Website* dapat digunakan untuk berbagai kebutuhan, seperti sistem informasi, media pembelajaran, layanan administrasi, maupun sarana komunikasi.

(Carol, Lirungan and Hakim, 2025) menjelaskan bahwa *website* merupakan suatu sistem yang dibangun menggunakan teknologi web sehingga pengguna dapat mengakses informasi dan layanan melalui perangkat yang terhubung dengan internet. Penggunaan *website* dalam sistem informasi mampu meningkatkan efektivitas pelayanan karena proses pengelolaan data dapat dilakukan secara lebih cepat dan terintegrasi.

Dalam pengembangan sistem informasi modern, *website* tidak hanya berfungsi sebagai media penyajian informasi, tetapi juga sebagai sarana pengolahan data dan interaksi pengguna dengan sistem. Melalui *website*, pengguna dapat melakukan input data, melihat informasi, serta memperoleh hasil pengolahan data secara langsung tanpa harus melakukan instalasi perangkat lunak tambahan.

Pada penelitian ini, *website* digunakan sebagai platform utama dalam pembangunan Sistem Pendukung Keputusan Pemilihan Program Studi. Sistem berbasis *website* memungkinkan Admin, Guru BK, dan Siswa mengakses sistem sesuai hak akses masing-masing melalui browser sehingga memudahkan proses pengelolaan data dan penyampaian hasil rekomendasi program studi.

Berdasarkan uraian tersebut, dapat disimpulkan bahwa *website* merupakan media berbasis internet yang digunakan untuk menyampaikan informasi, mengelola data, dan menyediakan layanan digital yang dapat diakses oleh pengguna secara online.

6. Flowchart

Menurut (Mukhlis *et al.*, 2023), *flowchart* merupakan diagram berbentuk visual yang berfungsi untuk melukiskan rangkaian langkah atau proses yang berlangsung dalam suatu sistem maupun program, di mana alur kerja beserta pergerakan informasi dari satu tahap menuju tahap berikutnya direpresentasikan melalui simbol-simbol grafis yang saling dihubungkan oleh tanda panah penunjuk arah. Senada dengan hal tersebut, (Mukhlis *et al.*, 2023) menjelaskan bahwa *flowchart* merupakan pendekatan diagramatik dalam merepresentasikan algoritma guna menghasilkan solusi atas suatu permasalahan, di mana setiap simbol dengan bentuk berbeda mewakili jenis proses yang berbeda pula, sementara panah-panah penghubung menunjukkan jalur yang harus ditempuh dalam menelusuri alur tersebut.

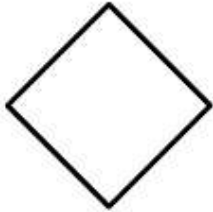
Lebih lanjut, *flowchart* tidak sekadar berfungsi sebagai alat

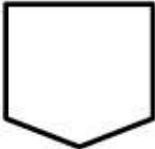
perancangan teknis, melainkan juga memiliki peran komunikatif yang signifikan.(Cahyaningtyas *et al.*, 2025) menegaskan bahwa diagram alir berperan sebagai alat bantu komunikasi antar-*programmer* dalam sebuah tim proyek, sehingga rangkaian logika yang panjang dan rumit dapat disajikan secara lebih mudah dipahami. Pernyataan ini diperkuat (Zimmermann, Zerbato and Weber, 2024) yang menyatakan bahwa *flowchart* dirancang menggunakan simbol-simbol, teks baku, dan/atau elemen grafis yang saling terhubung melalui panah yang merepresentasikan arah informasi maupun proses.

Berdasarkan pengertian *flowchart* di atas, dapat disimpulkan bahwa *flowchart* adalah representasi visual berupa diagram yang menggunakan simbol-simbol grafis dan panah untuk menggambarkan urutan langkah, aliran informasi, serta logika suatu proses atau algoritma secara sistematis. Selain berfungsi sebagai alat bantu perancangan *program*, *flowchart* juga berperan penting dalam memfasilitasi komunikasi dan pemahaman bersama terhadap proses yang kompleks, baik dalam konteks pemrograman maupun penyelesaian masalah secara umum.

Tabel 2.1 Simbol - Simbol Flowchart

Simbol	Nama	Fungsi
	Aliran Data / <i>Flowline</i>	Simbol berbentuk anak panah yang digunakan untuk menunjukkan arah perpindahan data atau informasi dari satu tahapan ke tahapan berikutnya dalam alur proses.

	Terminal / <i>Terminator</i>	Simbol berbentuk oval atau persegi panjang bersudut membulat yang berfungsi sebagai penanda titik awal maupun titik akhir dari suatu alur proses atau program.
	Proses / <i>Process</i>	Simbol berbentuk persegi panjang yang digunakan untuk merepresentasikan suatu tindakan atau serangkaian instruksi yang dijalankan dalam proses atau program.
	Keputusan / <i>Decision</i>	Simbol berbentuk belah ketupat yang digunakan untuk menandai percabangan dalam alur proses berdasarkan suatu kondisi tertentu. Setiap panah yang keluar dari simbol ini mewakili kemungkinan hasil dari keputusan yang diambil.
	Konektor	Simbol berbentuk lingkaran kecil yang berfungsi sebagai penghubung antara bagian-bagian alur proses yang secara fisik terpisah namun memiliki keterkaitan secara logis dalam satu rangkaian proses.
	<i>Input/Output</i>	Simbol berbentuk jajaran genjang yang digunakan untuk merepresentasikan proses pemasukan data ke dalam sistem (<i>input</i>) maupun keluaran hasil

		yang dihasilkan dari proses (<i>output</i>).
	<i>Predefined Process</i>	Simbol berbentuk persegi panjang dengan garis vertikal di kedua sisinya, digunakan untuk mengacu pada suatu sub-proses atau rangkaian langkah yang telah didefinisikan dan dirancang sebelumnya secara terpisah.
	Dokumen	Simbol berbentuk persegi panjang dengan bagian bawah bergelombang yang digunakan untuk merepresentasikan dokumen atau berkas informasi yang terlibat dan diproses dalam suatu alur kerja.
	Penggabungan (<i>Merge</i>) dan Pemisahan (<i>Split</i>)	Simbol yang digunakan untuk merepresentasikan proses penggabungan beberapa aliran proses menjadi satu (<i>merge</i>) atau pemecahan satu aliran menjadi beberapa jalur proses yang berbeda (<i>split</i>).
	<i>Off-page Connector</i>	Simbol berbentuk segi lima menyerupai rumah yang berisi huruf atau angka pengenalan, digunakan untuk menghubungkan langkah-langkah atau bagian-bagian dari <i>flowchart</i> yang berlanjut ke halaman lain yang berbeda.

7. Unified Modeling Language (UML)

Menurut (Yusup and Anwar, 2024) *Unified Modeling Language* (UML) adalah sebuah bahasa pemodelan berbasis grafis yang berperan dalam proses perancangan dan dokumentasi sistem perangkat lunak. Melalui bahasa ini, para pengembang dapat memvisualisasikan serta memahami struktur dan rancangan sistem yang hendak dibangun. Senada dengan hal tersebut, (Wayahdi and Ruziq, 2023) UML adalah bahasa berbasis grafis yang tidak hanya dimanfaatkan untuk memvisualisasikan dan menspesifikasikan suatu sistem, tetapi juga untuk membangun dan mendokumentasikan sistem perangkat lunak yang berorientasi objek. Lebih dari itu, UML memiliki kemampuan untuk diintegrasikan dengan berbagai bahasa pemrograman seperti Java, C++, dan Visual Basic, serta basis data berorientasi objek, sehingga cakupannya tidak terbatas hanya sebagai alat pemrograman visual semata.

Berdasarkan pengertian *Unified Modeling Language* (UML) di atas, dapat disimpulkan bahwa *Unified Modeling Language* (UML) adalah bahasa pemodelan grafis yang mencakup fungsi perancangan, visualisasi, spesifikasi, pembangunan, sekaligus dokumentasi sistem perangkat lunak berbasis objek. Keberadaan UML sangat membantu pengembang dalam memahami dan menggambarkan sistem secara lebih terstruktur dan komprehensif. Selain itu, kemampuannya untuk berintegrasi dengan berbagai bahasa pemrograman maupun basis data berorientasi objek menjadikan UML sebagai salah satu instrumen penting dalam pengembangan perangkat lunak.

Adapun jenis-jenis UML yang digunakan oleh penulis dalam penelitian ini meliputi:

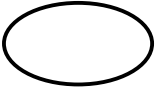
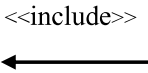
a. *Use Case Diagram*

Menurut (Yusup and Anwar, 2024) *Use Case Diagram* adalah salah satu jenis diagram dalam UML yang digunakan untuk memodelkan bagaimana pengguna berinteraksi dengan sistem, dengan tujuan menampilkan kebutuhan fungsional utama yang perlu disediakan oleh sistem berdasarkan sudut pandang pengguna. Sementara itu, (Wayahdi and Ruziq, 2023) UML menambahkan bahwa *Use Case Diagram* merupakan representasi visual yang menggambarkan hubungan antara aktor eksternal dan sistem guna merepresentasikan fungsi serta alur proses bisnis. Diagram ini juga memiliki peran penting sebagai alat verifikasi kebutuhan, identifikasi aktor, spesifikasi antarmuka, serta sarana komunikasi resmi antara analis sistem dan para pemangku kepentingan.

Berdasarkan pengertian *Use Case Diagram* di atas, dapat disimpulkan bahwa *Use Case Diagram* merupakan salah satu diagram UML yang berfungsi untuk memodelkan interaksi antara pengguna dan sistem secara visual. Diagram ini tidak hanya menggambarkan kebutuhan fungsional utama yang harus dipenuhi oleh sistem, tetapi juga merepresentasikan alur proses bisnis yang berjalan. Di samping itu, *Use Case Diagram* turut berperan dalam proses verifikasi kebutuhan, identifikasi aktor yang terlibat, spesifikasi antarmuka sistem, serta menjadi media komunikasi yang efektif antar entitas yang berkepentingan dalam

pengembangan sistem.

Tabel 2.2 Simbol Use Case Diagram

Simbol	Elemen	Keterangan
	Aktor	Merepresentasikan peran yang dimainkan oleh seseorang, sistem eksternal, atau perangkat tertentu ketika berinteraksi dan berkomunikasi dengan <i>use case</i> yang ada dalam sistem.
	<i>Use Case</i>	Menggambarkan abstraksi dari suatu fungsi atau layanan yang disediakan oleh sistem sebagai hasil dari interaksi yang terjadi antara sistem dengan aktor.
	<i>Association</i>	Merepresentasikan hubungan atau jalur komunikasi yang terbentuk antara aktor dengan <i>use case</i> , menunjukkan bahwa aktor tersebut terlibat dalam pelaksanaan <i>use case</i> yang bersangkutan.
	<i>Generalisasi</i>	Menggambarkan hubungan pewarisan antara aktor atau <i>use case</i> , di mana elemen yang lebih khusus (anak) mewarisi karakteristik dan kemampuan dari elemen yang lebih umum (induk).
	<i>Include</i>	Menandakan bahwa suatu <i>use case</i> secara wajib menyertakan dan menjalankan fungsionalitas dari <i>use case</i> lain sebagai bagian yang tidak terpisahkan dari prosesnya.
	<i>Extend</i>	Menandakan bahwa suatu <i>use case</i> dapat memperluas atau menambahkan fungsionalitas tambahan pada <i>use case</i> lain, namun hanya akan dijalankan apabila kondisi atau syarat tertentu terpenuhi.

Sumber: (Yusup and Anwar, 2024)[Click or tap here to enter text.](#)

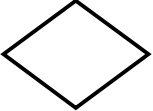
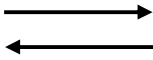
b. *Activity Diagram*

Menurut mukhlis dalam ,(Mukhlis *et al.*, 2023) *activity diagram* merupakan salah satu jenis diagram yang dimanfaatkan dalam pemodelan sistem atau proses bisnis guna mendeskripsikan rangkaian tindakan yang berlangsung secara terstruktur. Sejalan dengan hal tersebut, (Yusup and Anwar, 2024) mengemukakan bahwa *activity diagram* merupakan bagian dari diagram UML yang menggambarkan alur kerja atau serangkaian aktivitas dalam suatu sistem maupun proses bisnis, dengan tujuan untuk memperlihatkan jalannya suatu proses sekaligus mengidentifikasi bagian-bagian yang berpotensi untuk dikembangkan demi tercapainya efisiensi yang lebih optimal.

Berdasarkan pengertian *Activity Diagram* di atas, dapat disimpulkan bahwa *activity diagram* adalah diagram berbasis UML yang difungsikan untuk memodelkan rangkaian aktivitas atau alur kerja dalam sebuah sistem maupun proses bisnis. Diagram ini berperan penting dalam memperlihatkan jalannya suatu proses secara visual, menguraikan tahapan-tahapan yang berlangsung secara berurutan, serta membantu dalam menemukan bagian yang perlu diperbaiki agar sistem dapat beroperasi dengan lebih efisien.

Tabel 2.3 Simbol Activity Diagram

Simbol	Elemen	Keterangan
	<i>Activity</i>	Menggambarkan sebuah aktivitas atau pekerjaan yang dilakukan dalam alur proses, memperlihatkan bagaimana setiap elemen antarmuka saling

		berinteraksi dan berkoordinasi satu dengan yang lainnya.
	<i>Action</i>	Menrepresentasikan kondisi atau keadaan sistem pada saat tertentu yang mencerminkan pelaksanaan sebuah aksi atau tindakan spesifik yang tidak dapat dibagi lebih lanjut.
	<i>Initial Node</i>	Berfungsi sebagai penanda titik permulaan dari suatu diagram aktivitas, menunjukkan dari mana alur proses mulai dijalankan.
	<i>Activity Final Noda</i>	Berfungsi sebagai penanda titik berakhirnya seluruh alur dalam diagram aktivitas, menandakan bahwa keseluruhan proses telah selesai dijalankan.
	<i>Decision</i>	Digunakan untuk merepresentasikan suatu titik percabangan dalam alur proses, di mana terdapat dua atau lebih kemungkinan jalur yang akan ditempuh bergantung pada kondisi atau kriteria tertentu yang dievaluasi.
	<i>Line Connector</i>	Berupa garis penghubung yang digunakan untuk mengaitkan satu simbol dengan simbol lainnya, menunjukkan arah aliran atau urutan dari suatu aktivitas ke aktivitas berikutnya.

Sumber: (Yusup and Anwar, 2024)

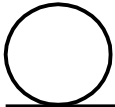
c. *Sequence Diagram*

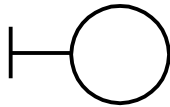
Menurut (Yusup and Anwar,2024), *sequence* diagram merupakan salah satu jenis diagram UML yang menggambarkan interaksi antar objek

dalam suatu sistem secara terurut guna memperlihatkan alur logika dari sebuah fitur atau fungsi tertentu. Diagram ini menyajikan urutan pesan maupun pemanggilan antar objek melalui tanda panah yang disusun berdasarkan dimensi waktu, sehingga dinamika interaksi antar objek dalam sistem dapat terlihat dengan jelas. Senada dengan hal tersebut, (Wayahdi and Ruziq, 2023) UML menjelaskan bahwa *sequence* diagram adalah diagram yang merepresentasikan interaksi secara rinci antar objek dalam sistem, mencakup pesan atau instruksi yang dikirimkan beserta waktu pelaksanaannya, di mana objek-objek yang terlibat pada umumnya disusun dari kiri ke kanan mengikuti jalannya proses.

Berdasarkan pengertian *Sequence* Diagram di atas, dapat disimpulkan bahwa *sequence* diagram merupakan pemodelan interaksi secara rinci antar objek dalam suatu sistem yang disusun secara berurutan mengacu pada aliran waktu. Diagram ini menampilkan rangkaian pesan atau pemanggilan antar objek melalui garis panah, sehingga alur logika suatu fitur atau fungsi dapat tergambarkan dengan jelas, sekaligus memperlihatkan waktu eksekusi setiap pesan yang terjadi di dalam sistem.

Tabel 2.4 Simbol Sequence Diagram

Simbol	Elemen	Keterangan
	Aktor	Merepresentasikan pengguna atau pihak eksternal yang memulai dan terlibat langsung dalam interaksi dengan sistem yang sedang dimodelkan.
	<i>Entity Class</i>	Merepresentasikan objek atau entitas dalam sistem yang menyimpan data dan memiliki

		keterlibatan dalam hubungan atau relasi yang terbentuk selama jalannya suatu skenario interaksi.
	<i>Boundary Class</i>	Merepresentasikan antarmuka yang berfungsi sebagai perantara dalam mengelola komunikasi dan pertukaran informasi antara aktor eksternal dengan lingkungan internal sistem.
	<i>Control Class</i>	Merepresentasikan objek pengendali yang berfungsi sebagai jembatan koordinasi antara kelas <i>boundary</i> yang berinteraksi dengan pengguna dan kelas <i>entity</i> yang mengelola data dalam sistem.
	<i>A Focus of Control & A Life Line</i>	<i>Lifeline</i> merepresentasikan keberadaan suatu objek sepanjang dimensi waktu dalam diagram, sedangkan <i>focus of control</i> menggambarkan periode aktif di mana objek tersebut sedang menjalankan atau mengendalikan suatu proses hingga selesai.
	<i>A Message</i>	Merepresentasikan komunikasi atau instruksi yang dikirimkan dari satu objek kepada objek lainnya dalam rangkaian interaksi, digambarkan melalui garis panah yang menunjukkan arah pengiriman pesan tersebut.

Sumber: (Yusup and Anwar, 2024)

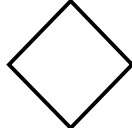
d. *Class Diagram*

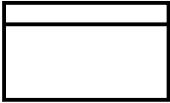
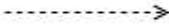
Menurut (Wayahdi and Ruziq, 2023) UML, *class diagram* merupakan salah satu jenis diagram struktural dalam UML yang secara eksplisit menggambarkan kelas, atribut, metode, serta relasi antar objek. Diagram ini memiliki sifat statis karena lebih menitikberatkan pada

representasi hubungan antar kelas daripada alur interaksi yang terjadi di dalamnya. Sejalan dengan hal tersebut, (Yusup and Anwar, 2024) mengemukakan bahwa *class* diagram adalah diagram dalam UML yang digunakan untuk merepresentasikan struktur kelas beserta keterkaitannya dalam suatu sistem, dengan tujuan memberikan gambaran menyeluruh mengenai objek-objek yang ada beserta relasi di antara objek-objek tersebut.

Berdasarkan pengertian *Class* Diagram di atas, dapat disimpulkan bahwa *class* diagram adalah diagram UML yang merepresentasikan struktur kelas dalam suatu sistem, meliputi atribut, metode, serta hubungan antar kelas atau objek. Diagram ini bersifat statis karena menekankan pada struktur dan keterkaitan antar kelas, bukan alur interaksi, sehingga memberikan gambaran menyeluruh mengenai komponen sistem dan bagaimana objek-objek tersebut saling berhubungan.

Tabel 2.5 Simbol Class Diagram

Simbol	Elemen	Keterangan
	Garis lurus (<i>Generalization</i>)	Merepresentasikan hubungan pewarisan antara kelas turunan (<i>subclass</i>) dan kelas induk (<i>superclass</i>), di mana kelas turunan mengadopsi dan mewarisi perilaku serta struktur data yang dimiliki oleh kelas induknya.
	<i>Nary Association</i>	Merepresentasikan hubungan asosiasi yang melibatkan lebih dari dua kelas sekaligus dalam satu relasi, digunakan ketika sebuah asosiasi tidak dapat cukup

		digambarkan hanya dengan menghubungkan dua objek saja.
	<i>Class</i>	Merepresentasikan cetak biru atau <i>template</i> dari sekumpulan objek dalam sistem yang memiliki kesamaan dalam hal <i>atribut</i> (sifat) dan operasi (perilaku) yang dimilikinya.
	<i>Collaboration</i>	Menggambarkan kerja sama antar objek atau kelas melalui serangkaian aksi yang terkoordinasi dalam sistem, dengan tujuan menghasilkan suatu keluaran atau hasil yang dapat diukur dan diverifikasi.
	<i>Realization</i>	Merepresentasikan hubungan antara sebuah antarmuka (<i>interface</i>) dengan kelas yang mengimplementasikannya, menunjukkan bahwa kelas tersebut benar-benar mewujudkan dan menjalankan operasi-operasi yang telah ditetapkan oleh antarmuka.
	<i>Dependency</i>	Menggambarkan hubungan ketergantungan antar elemen, di mana perubahan yang terjadi pada elemen <i>independen</i> (yang tidak bergantung) berpotensi memberikan dampak atau pengaruh terhadap elemen lain yang bersifat <i>dependen</i> (bergantung) padanya.
	<i>Association</i>	Merepresentasikan hubungan struktural yang menghubungkan satu kelas dengan kelas lainnya, menandakan bahwa objek dari kelas yang satu memiliki

keterkaitan atau dapat berkomunikasi dengan objek dari kelas yang lain.

Sumber: (Yusup and Anwar, 2024)

8. *Framework Laravel*

Menurut (Febriyanti *et al.*, 2026), Laravel merupakan framework berbasis PHP yang banyak dimanfaatkan dalam pengembangan aplikasi web karena menyediakan berbagai komponen yang membantu proses pembangunan sistem menjadi lebih terstruktur. Penggunaan framework ini memungkinkan pengembang mengelola kode program dengan lebih mudah sehingga proses pengembangan dan pemeliharaan aplikasi dapat dilakukan secara lebih efektif. Selain itu, Laravel menerapkan konsep Model View Controller (MVC) yang memisahkan logika aplikasi, tampilan antarmuka, dan pengelolaan data secara lebih baik.

Selain menyediakan struktur pengembangan yang sistematis, Laravel dilengkapi dengan berbagai fitur pendukung seperti autentikasi pengguna, pengelolaan routing, middleware, migrasi basis data, serta mekanisme keamanan. Fitur-fitur tersebut sangat membantu pengembang dalam membangun aplikasi yang efisien, aman, dan mudah dipelihara dalam jangka panjang. Dengan demikian, Laravel dapat dipahami sebagai framework PHP terstruktur berarsitektur MVC yang menyediakan infrastruktur lengkap untuk membangun aplikasi web yang aman, efisien, serta mudah dikembangkan.

9. *React.js*

React.js merupakan pustaka JavaScript yang digunakan untuk

membangun antarmuka pengguna (*User Interface*) pada aplikasi web. React.js dikembangkan oleh Meta (*Facebook*) dan pertama kali dirilis pada tahun 2013. React.js menerapkan konsep *component-based architecture* yang memungkinkan antarmuka aplikasi dibangun dari komponen-komponen yang dapat digunakan kembali (*reusable components*).

Menurut (Pamungkas, Parwati and Putranto, 2025), React.js merupakan pustaka JavaScript berbasis komponen yang banyak digunakan dalam pengembangan aplikasi web modern karena mampu menghasilkan antarmuka yang responsif dan interaktif. React.js menggunakan pendekatan Virtual DOM yang memungkinkan proses pembaruan tampilan dilakukan secara selektif dan lebih cepat dibandingkan manipulasi DOM secara langsung sehingga meningkatkan performa aplikasi secara keseluruhan.

(Pramadipta, 2024) menjelaskan bahwa React.js menjadi salah satu teknologi *frontend* yang populer karena mendukung pengembangan *Single Page Application* (SPA), mudah diintegrasikan dengan REST API, serta memiliki ekosistem yang luas dengan dukungan komunitas pengembang yang besar. Penggunaan React.js dalam pengembangan sistem informasi berbasis web mampu meningkatkan pengalaman pengguna karena perubahan data pada antarmuka dapat ditampilkan secara dinamis tanpa perlu memuat ulang halaman secara keseluruhan.

Pada penelitian ini, React.js digunakan sebagai frontend sistem yang berfungsi untuk menampilkan antarmuka pengguna kepada Admin, Guru BK, dan Siswa. React.js digunakan untuk mengelola tampilan dashboard, formulir

input data, pengisian survei minat dan bakat, serta penyajian hasil rekomendasi program studi yang dihasilkan oleh sistem.

Berdasarkan uraian tersebut, dapat disimpulkan bahwa React.js merupakan pustaka JavaScript yang digunakan untuk membangun antarmuka pengguna secara interaktif, responsif, dan efisien melalui pendekatan berbasis komponen serta teknologi Virtual DOM yang mendukung performa tinggi pada aplikasi web modern.

10. MySQL

Pengertian MySQL merupakan perangkat lunak yang digunakan untuk menampung serta mengelola data yang tersusun secara terstruktur dan saling berhubungan di dalam sebuah basis data. Sistem ini termasuk dalam kategori *Database Management System* (DBMS) yang banyak digunakan karena kemampuannya dalam mengelola data secara efisien serta dapat diunduh dan digunakan secara bebas melalui situs resmi MySQL. Dengan kemampuannya tersebut, MySQL sering dimanfaatkan dalam pengembangan aplikasi untuk menyimpan, mengatur, dan memproses data secara sistematis.

MySQL merupakan sistem manajemen basis data yang menggunakan model relasional, di mana data disimpan dalam bentuk tabel yang saling berhubungan sehingga mampu menjaga konsistensi dan integritas data pada suatu sistem informasi (Afif Muhamad *et al.*, 2025). Struktur relasional tersebut juga memungkinkan adanya hubungan antar tabel yang mempermudah pengelolaan serta pengolahan data sehingga informasi

dapat tersimpan secara terorganisir dan sistematis dalam sebuah basis data (Dachi *et al.*, 2025).

Berdasarkan penjelasan tersebut, MySQL dapat dimaknai sebagai sistem basis data relasional yang mendukung penyimpanan dan pengelolaan data terstruktur untuk kebutuhan sistem informasi berbasis web.

11. Black Box Testing

Menurut (Julianifa Putri *et al.*, 2026), *Black Box Testing* merupakan salah satu pendekatan pengujian perangkat lunak yang berfokus pada analisis fungsi sistem dan pemetaan alur kerja pengguna tanpa perlu memperhatikan detail dari internal *software* tersebut. Metode ini dilakukan untuk menentukan area fungsionalitas yang akan diuji guna memastikan bahwa proses pengujian berjalan secara akurat dan mampu mendukung keandalan kinerja sistem secara menyeluruh.

Dalam pengembangan sistem informasi, proses pengujian perangkat lunak dilakukan untuk memastikan sistem dapat berfungsi dengan baik serta meminimalkan kemungkinan terjadinya kesalahan yang dapat menghambat penggunaan sistem oleh pengguna (Sahyudi and Voutama, 2025). Salah satu metode yang sering digunakan adalah *Black Box Testing* yang berfokus pada pengujian fungsionalitas sistem dengan cara memberikan berbagai input dan mengamati output yang dihasilkan tanpa melakukan analisis terhadap struktur atau kode internal program. Melalui pendekatan tersebut, pengujian dapat digunakan untuk memvalidasi fitur sistem, proses pengolahan data, serta kesesuaian hasil keluaran dengan kebutuhan

pengguna (Dwi Andika *et al.*, 2025)

Berdasarkan uraian tersebut, pengujian *Black Box* dapat dimaknai sebagai metode pengujian fungsional yang memastikan sistem berjalan sesuai dengan spesifikasi yang telah ditetapkan.

12. Metode Pengembangan Sistem Waterfall

Metode Waterfall merupakan salah satu pendekatan dalam *Software Development Life Cycle* (SDLC) yang digunakan dalam proses pengembangan perangkat lunak secara sistematis dan bertahap. Menurut (Sofian and Ghozali, 2026), proses pengembangan menggunakan metode Waterfall dilakukan secara berurutan melalui tahapan analisis kebutuhan, desain, implementasi, pengujian, hingga pemeliharaan. Setiap tahapan pengembangan dilaksanakan secara terstruktur sehingga proses pembangunan perangkat lunak dapat dilakukan sesuai dengan kebutuhan sistem yang telah ditentukan.

Menurut (Lesmana and Thomas, 2026), metode Waterfall merupakan model pengembangan perangkat lunak yang dilakukan secara bertahap melalui proses analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan pemeliharaan. Setiap tahapan diselesaikan secara sistematis sebelum melanjutkan ke tahap berikutnya. Dengan tahapan yang terstruktur tersebut, proses pengembangan sistem dapat dilakukan secara lebih terarah mulai dari identifikasi kebutuhan hingga pemeliharaan sistem setelah implementasi.

Metode Waterfall dipilih dalam penelitian ini karena kebutuhan

sistem sudah dapat dirumuskan secara jelas sejak awal. Hal ini sesuai dengan (Sofian and Ghozali, 2026) yang menjelaskan bahwa metode Waterfall cocok digunakan dalam pengembangan perangkat lunak ketika kebutuhan sistem telah jelas dan diperkirakan tidak mengalami banyak perubahan selama proses pengembangan. Dengan demikian, penggunaan metode Waterfall dalam penelitian ini memungkinkan proses pengembangan sistem dilakukan secara sistematis, terstruktur, dan sesuai dengan tahapan yang telah direncanakan.

Berikut integrasi rancangan teknologi sistem ke dalam masing-masing tahapan Waterfall:

a. Tahap Analisis Kebutuhan (*Requirements Analysis*)

Pada tahap ini dilakukan analisis mendalam terhadap kebutuhan fungsional dan non-fungsional sistem yang akan dibangun. Kebutuhan fungsional utamanya adalah kemampuan mengolah data kriteria dan bobot, serta melakukan kalkulasi rekomendasi program studi secara akurat. Untuk memenuhi kebutuhan tersebut, sistem mengintegrasikan metode Simple Additive Weighting (SAW) sebagai algoritma utama dalam menyelesaikan pembobotan dan perangkingan multi-kriteria guna merekomendasikan program studi.

b. Tahap Perancangan Sistem (*Design*)

Tahap ini bertujuan menerjemahkan hasil analisis kebutuhan ke dalam bentuk arsitektur, struktur data, antarmuka, dan pemodelan sistem sebelum masuk ke tahap pengodean. Perancangan dibagi menjadi dua fokus utama:

- 1) Pemodelan Sistem (UML) blueprint jalannya sistem dirancang menggunakan *Unified Modeling Language* (UML) untuk memetakan bagaimana sistem merespons masukan pengguna, mencakup *Use Case Diagram* (interaksi aktor dengan fungsionalitas sistem), *Activity Diagram* (aliran aktivitas proses bisnis), *Sequence Diagram* (interaksi antar-objek berdasarkan urutan waktu), dan *Class Diagram* (struktur dan relasi antar-kelas/tabel).
 - 2) Perancangan Basis Data digunakan MySQL untuk menyimpan data relasional terstruktur (kriteria, nilai, dan pengguna) dengan performa tinggi serta kompatibilitas penuh terhadap framework yang digunakan.
- c. Tahap Pembuatan Kode Program (*Implementation/Coding*)
- Setelah rancangan selesai, seluruh pemodelan diterjemahkan ke dalam bahasa pemrograman dengan mempertimbangkan efisiensi, keamanan, performa, dan skalabilitas. Struktur kode dibagi menjadi dua arsitektur terpisah (*decoupled architecture*):
- 1) Backend: PHP 8.4 sebagai bahasa pemrograman sisi server, framework Laravel 12 yang menyediakan struktur MVC terorganisir, dan Laravel Sanctum untuk autentikasi token API yang aman.
 - 2) Frontend: React JS sebagai pustaka JavaScript berbasis komponen, Vite sebagai build tool untuk mempercepat pengembangan, Tailwind CSS untuk styling yang efisien, dan Shadcn UI sebagai pustaka komponen antarmuka pra-desain yang responsif.
- d. Tahap Pengujian (*Verification/Testing*)

Setelah seluruh fungsi selesai dikodekan, dilakukan pengujian akhir menggunakan Black Box Testing untuk memastikan setiap masukan menghasilkan keluaran sesuai harapan dan sistem terbebas dari kesalahan logika fungsional, dengan fokus pada pemenuhan kebutuhan fungsional tanpa perlu menelaah kode internal secara mendalam.

e. Penjelasan Rancangan Teknologi Sistem

Pemilihan teknologi pada setiap tahap di atas didasarkan pada kebutuhan fungsional dan non-fungsional sistem. *Backend* menggunakan PHP 8.4 dan Laravel 12 karena menawarkan struktur MVC terorganisir, fitur autentikasi bawaan melalui *Laravel Sanctum*, serta kemudahan mengelola logika bisnis dan kalkulasi SAW. *Frontend* menggunakan *React JS* yang dipadukan dengan Vite, Tailwind CSS, dan Shadcn UI untuk menghasilkan antarmuka yang cepat, efisien, dan responsif. Basis data menggunakan MySQL karena mendukung data relasional terstruktur dengan performa tinggi dan kompatibilitas penuh terhadap Laravel. Pemodelan sistem menggunakan UML (*Use Case*, *Activity*, *Sequence*, dan *Class Diagram*), sedangkan pengujian dilakukan dengan Black Box Testing untuk memastikan seluruh fungsi berjalan sesuai kebutuhan fungsional.

B. Kajian Empiris

Kajian empiris merupakan tinjauan terhadap penelitian-penelitian terdahulu yang relevan dan dapat dijadikan sebagai acuan dalam pengembangan penelitian ini. Beberapa penelitian terdahulu yang berkaitan dengan Sistem

Pendukung Keputusan (SPK), metode *Simple Additive Weighting* (SAW), dan rancang bangun sistem informasi berbasis web.

Penelitian pertama yang dilakukan oleh (Dela Novita and Lidya Wati, 2025) memiliki kesamaan dengan penelitian ini dalam hal penggunaan metode *Simple Additive Weighting* (SAW) untuk mendukung pemilihan program studi di perguruan tinggi berdasarkan kriteria akademik, minat, dan bakat siswa. Perbedaannya terletak pada platform yang digunakan, di mana penelitian ini mengembangkan sistem berbasis web dengan framework Laravel dan React.js yang diterapkan secara spesifik untuk pemilihan program studi perguruan tinggi bagi siswa SMA Negeri 4 Kota Madiun.

Penelitian kedua oleh (Of, Assistance and Banyuurip, 2024) menunjukkan persamaan dalam penggunaan metode SAW pada sistem pendukung keputusan berbasis web serta penerapan *Black Box Testing* sebagai metode pengujian. Adapun perbedaannya adalah penelitian ini secara khusus mengintegrasikan fitur survei minat dan bakat siswa ke dalam sistem, serta dikembangkan menggunakan framework Laravel sebagai backend dan React.js sebagai frontend.

Penelitian ketiga oleh (Imami *et al.*, 2024) memiliki kemiripan dengan penelitian ini karena sama-sama mengembangkan sistem berbasis web menggunakan framework Laravel dan metode *Simple Additive Weighting* (SAW) serta menggunakan *Black Box Testing* untuk menguji fungsionalitas sistem. Penelitian tersebut menerapkan metode SAW dalam sistem identifikasi minat dan bakat berdasarkan konsep *Multiple Intelligence*. Hasil pengujian

menunjukkan bahwa seluruh fitur yang dikembangkan dapat berjalan dengan baik berdasarkan skenario pengujian yang dilakukan. Perbedaannya terletak pada tujuan sistem, di mana penelitian (Imami *et al.*, 2024), berfokus pada identifikasi jenis kecerdasan dominan serta rekomendasi kegiatan berdasarkan minat dan bakat, sedangkan penelitian ini mengembangkan Sistem Pendukung Keputusan pemilihan program studi perguruan tinggi dengan mengintegrasikan nilai akademik, survei minat, dan survei bakat siswa menggunakan Laravel sebagai *backend* dan React.js sebagai *frontend*.

Penelitian keempat oleh (Ramdhani and Thantawi, 2024) menunjukkan kesamaan dalam aspek metodologi pengembangan sistem, yaitu penggunaan metode Waterfall dan pengujian *Black Box Testing* dalam membangun sistem berbasis web untuk keperluan institusi pendidikan. Perbedaannya adalah penelitian tersebut membangun sistem informasi perpustakaan, sementara penelitian ini membangun Sistem Pendukung Keputusan yang dilengkapi dengan metode SAW dan dikembangkan menggunakan framework Laravel serta React.js.

Penelitian kelima oleh (Martin and Nilawati, 2024) memiliki kesamaan dalam penggunaan framework Laravel dan metode Waterfall untuk membangun sistem berbasis web yang mengubah proses manual menjadi terkomputerisasi. Perbedaannya terletak pada domain permasalahan, di mana penelitian tersebut mengembangkan sistem peminjaman arsip, sedangkan penelitian ini mengembangkan Sistem Pendukung Keputusan pemilihan program studi dengan metode SAW disertai fitur React.js sebagai antarmuka pengguna.

Berdasarkan lima penelitian terdahulu yang telah dikaji, dapat disimpulkan beberapa hal penting. Pertama, metode SAW terbukti efektif dalam sistem pengambilan keputusan multikriteria dan menghasilkan rekomendasi yang objektif dan terukur. Kedua, framework Laravel terbukti mampu mendukung pengembangan sistem berbasis web secara terstruktur dan efisien. Ketiga, metode Waterfall sesuai digunakan pada proyek dengan kebutuhan yang telah terdefinisi sejak awal. Keempat, *Black Box Testing* secara konsisten mampu memvalidasi fungsionalitas sistem secara menyeluruh.

Namun dari kelima penelitian tersebut, belum ditemukan penelitian yang membangun SPK pemilihan program studi yang bersifat universal dan tidak terbatas pada satu perguruan tinggi, sekaligus mengintegrasikan tiga sumber data nilai akademik, survei minat, dan survei bakat dalam satu sistem dengan arsitektur Laravel dan React.js. Oleh karena itu, penelitian ini hadir untuk mengisi celah tersebut dengan mengembangkan SPK pemilihan program studi yang fleksibel, dapat dikonfigurasi untuk berbagai perguruan tinggi, dan mampu memberikan rekomendasi secara otomatis dan objektif kepada siswa SMA kelas XII.

C. Kerangka Berpikir

Kerangka berpikir merupakan gambaran alur logis penelitian yang menjelaskan keterkaitan antara permasalahan yang ditemukan, pendekatan yang digunakan, model pengembangan yang diterapkan, subjek penelitian, serta luaran yang diharapkan dari penelitian ini.

1. Rumusan dan Identifikasi Masalah

Berdasarkan hasil observasi dan wawancara di SMA Negeri 4 Kota Madiun, diperoleh data bahwa terdapat 312 siswa kelas XII, di mana sekitar 50% dari mereka berminat untuk lanjut kuliah. Dari kondisi tersebut, ditemukan permasalahan bahwa proses pemberian rekomendasi program studi kepada siswa masih dilakukan secara manual oleh Guru BK. Proses tersebut bersifat subjektif, memakan waktu, dan tidak terstandarisasi. Data nilai rapor, minat, dan bakat siswa juga masih tersimpan secara terpisah sehingga menyulitkan proses analisis secara terintegrasi.

2. Metode dan Pendekatan

Untuk menyelesaikan permasalahan tersebut, penelitian ini menggunakan pendekatan solusi berupa pengembangan Sistem Pendukung Keputusan (SPK) berbasis web. SPK dipilih karena mampu membantu proses pengambilan keputusan yang melibatkan banyak kriteria secara objektif dan terstruktur. Berdasarkan kajian empiris, metode yang digunakan adalah Simple Additive Weighting (SAW). Metode ini dipilih karena efektif sebagai sistem rekomendasi multikriteria. Adapun kriteria yang digunakan meliputi nilai akademik (bobot 40%), minat siswa (bobot 35%), dan bakat siswa

(bobot 25%).

3. Model Pengembangan dan Teknologi Sistem

Sistem dikembangkan menggunakan model pengembangan Waterfall yang berjalan secara sekuensial, meliputi tahapan: analisis, desain, implementasi, pengujian, *deployment, maintenance*. Untuk mendukung implementasi, teknologi sistem dibangun menggunakan *framework* Laravel 12 dan PHP 8.4 sebagai *backend*, React JS dan Tailwind sebagai *frontend*, serta MySQL sebagai basis data. Pengujian sistem nantinya akan dilakukan menggunakan metode *Black Box Testing*.

4. Subjek dan Objek Penelitian

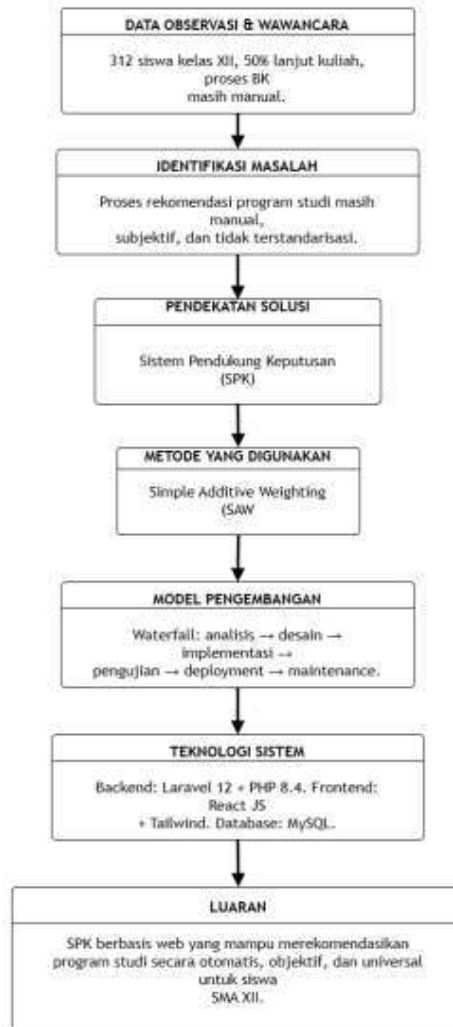
Subjek penelitian adalah siswa kelas XII di SMA Negeri 4 Kota Madiun beserta Guru Bimbingan dan Konseling (BK) yang terlibat dalam proses pemberian rekomendasi program studi. Objek penelitian adalah Sistem Pendukung Keputusan Pemilihan Program Studi perguruan tinggi yang dikembangkan menggunakan metode SAW dengan tiga kriteria penilaian, yaitu nilai akademik (bobot 40%), minat siswa (bobot 35%), dan bakat siswa (bobot 25%).

5. Luaran Penelitian

Luaran yang diharapkan dari penelitian ini adalah sebuah SPK berbasis web yang mampu merekomendasikan program studi secara otomatis, objektif, dan universal untuk siswa SMA kelas XII. Sistem ini diharapkan dapat membantu siswa dalam menentukan pilihan program studi yang sesuai dengan kemampuan akademik, minat, dan bakat yang dimiliki, serta meringankan

beban Guru BK dalam proses pemberian rekomendasi.

Berikut adalah alur kerangka berpikir penelitian ini secara ringkas:



Gambar 2. 1 Kerangka Berpikir Penelitian

BAB III

METODOLOGI PENELITIAN

A. Rencana Jadwal Penelitian

1. Tempat Penelitian

Penelitian dan pengembangan sistem ini dilaksanakan di SMA Negeri 4 Kota Madiun yang beralamat lengkap di Jl. Serayu No.80, Pandean, Kec. Taman, Kota Madiun, Jawa Timur 63133. Objek penelitian secara spesifik difokuskan pada ruang lingkup layanan Bimbingan dan Konseling (BK) bagi siswa kelas XII.

2. Waktu Penelitian

Pelaksanaan penelitian direncanakan berlangsung selama 6 (enam) bulan, terhitung sejak tahap pengajuan proposal awal hingga penyelesaian laporan akhir penelitian. Rincian jadwal kegiatan diuraikan pada Tabel 1.1 berikut:

Tabel 3. 1 Rencana Jadwal Penelitian

No	Keterangan	Maret				April				Mei				Juni				Juli			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Studi Literatur																				
2	Analisa Kebutuhan sistem																				
3	Perancangan Sistem																				