

## **BAB II**

### **KAJIAN PUSTAKA**

#### **A. Kajian Teoritis**

##### **1. Sistem Pendukung Keputusan**

###### **a. Sistem**

Sistem merupakan suatu struktur yang terdiri dari prosedur-prosedur yang saling terhubung, disusun secara menyeluruh dan terorganisir berdasarkan suatu pola tertentu (Endaryati, 2021). Sistem, menurut (Jumroni et al., 2022) merupakan sekumpulan elemen atau komponen yang saling terhubung dan berinteraksi untuk mencapai suatu tujuan tertentu. Komponen-komponen tersebut dihubungkan dan disusun sesuai dengan tujuannya. Guna mencapai tujuan yang diinginkan, sistem dibentuk dari berbagai komponen yang saling terintegrasi satu sama lain (Syafariani et al., 2022).

Berdasarkan penjelasan di atas, sistem dapat diartikan sebagai suatu himpunan elemen, proses, dan subsistem yang saling terhubung secara terstruktur untuk mencapai tujuan tertentu. Setiap bagian tidak dapat dipisahkan karena masing-masing memiliki peran dan fungsi yang khas serta saling melengkapi.

###### **b. Sistem Pendukung Keputusan**

Sistem Pendukung Keputusan (SPK) merupakan sistem komputer yang dibuat untuk mendukung proses pengambilan keputusan dalam menyelesaikan masalah. SPK menyajikan informasi, model, serta alat analisis data secara interaktif guna meningkatkan efektivitas dan efisiensi dalam pengambilan keputusan (Ina et al., 2024). Menurut (I Gede Iwan Sudipa, Suyono, Jefri Junifer Pangaribuan et al., 2023), Sistem Pendukung Keputusan (SPK) berperan dalam membantu pengguna mengambil keputusan dengan menyajikan informasi yang relevan serta melakukan analisis sesuai dengan kebutuhan pengguna. Sistem ini sangat bermanfaat ketika keputusan harus diambil dalam situasi yang kompleks dan tidak sepenuhnya dapat diprediksi, karena mampu mengelola berbagai informasi yang tersedia dan mengolahnya menjadi dasar pertimbangan yang objektif.

Berdasarkan penjelasan di atas, dapat disimpulkan bahwa sistem pendukung keputusan merupakan alat bantu berbasis komputer yang efektif dalam mendukung pengambilan keputusan, terutama pada proses yang melibatkan banyak kriteria seperti seleksi beasiswa. SPK membantu menyajikan data dan analisis secara objektif untuk menghasilkan keputusan yang lebih tepat dan efisien.

## **2. Beasiswa**

Beasiswa merupakan bentuk bantuan keuangan yang diberikan kepada individu untuk mendukung pendidikan mereka (Khaliq et al., 2023). Bantuan ini dapat berasal dari institusi pendidikan, organisasi atau

pemerintah, dan tujuannya adalah untuk mendukung keterjangkauan pendidikan, meningkatkan capaian akademik, serta membantu individu dalam mengembangkan potensinya guna meraih tujuan pendidikan dan karier.

Menurut (Dalla & Kewuel, 2023), beasiswa tidak hanya berfungsi sebagai bantuan finansial, tetapi juga sebagai alat untuk mendorong keberlangsungan studi mahasiswa. Namun, mereka menyoroti adanya ketimpangan akses beasiswa, di mana mahasiswa dengan kemampuan akademik tinggi lebih berpeluang mendapatkan beasiswa, sementara mahasiswa dengan kemampuan akademik rendah yang mungkin lebih membutuhkan bantuan finansial justru kesulitan mengaksesnya. Beasiswa juga berperan dalam meningkatkan motivasi belajar mahasiswa. Studi oleh (Mastur et al., 2023) menunjukkan bahwa pemberian beasiswa memiliki dampak positif dan signifikan terhadap motivasi belajar mahasiswa, serta berkontribusi dalam meningkatkan semangat belajar mereka.

Berdasarkan penjelasan di atas, dapat disimpulkan bahwa beasiswa adalah bantuan keuangan yang diberikan kepada pelajar atau mahasiswa guna mendukung kelangsungan pendidikan mereka. Bantuan dapat bersifat prestasi maupun kebutuhan, dan bertujuan untuk meringankan beban biaya pendidikan serta meningkatkan motivasi dan prestasi belajar penerima.

### 3. Metode *Simple Additive Weighting*

Metode *Simple Additive Weighting* (SAW) merupakan salah satu teknik pengambilan keputusan yang banyak digunakan dalam sistem pendukung keputusan karena kesederhanaan dan efektivitasnya. Metode ini bekerja dengan cara menjumlahkan nilai dari setiap alternatif berdasarkan sejumlah kriteria yang telah diberikan bobot tertentu (Bagoes et al., 2024). Nilai alternatif terlebih dahulu dinormalisasi agar berada dalam skala yang sama, kemudian dikalikan dengan bobot masing-masing kriteria. Hasil perkalian ini selanjutnya dijumlahkan untuk mendapatkan nilai akhir yang menjadi dasar dalam menentukan pemilihan alternatif terbaik.

(Vikki, 2022) menjelaskan bahwa metode SAW digunakan untuk mencari alternatif optimal dari sejumlah alternatif dengan mempertimbangkan beberapa kriteria yang relevan. Ia menyebut metode ini sebagai metode penjumlahan terbobot karena mengandalkan proses pemberian bobot pada setiap kriteria, lalu menjumlahkan nilai-nilai yang telah dinormalisasi. Sementara itu, (Firmasyah et al., 2024) menambahkan bahwa metode SAW dilakukan dengan menjumlahkan hasil dari nilai normalisasi yang dikalikan dengan bobot, dan sangat cocok diterapkan dalam pengambilan keputusan berbasis banyak kriteria.

Berikut adalah rumus metode SAW secara umum dalam konteks Sistem Pendukung Keputusan (SPK):

**Indikator/Atribut**

1. Nilai
2. Gaji Ayah dan Gaji Ibu
3. Status Orang Tua
4. Bobot Indikator/Atribut
5. Perhitungan

Berikut langkah-langkah perhitungan yang digunakan dalam proses penerimaan kelayakan beasiswa adalah sebagai berikut:

**Langkah 1:** Menentukan Matriks Keputusan

Langkah awal adalah menyusun data dari seluruh calon penerima beasiswa ke dalam matriks keputusan, di mana setiap baris mewakili alternatif (calon penerima), dan setiap kolom mewakili indikator/atribut penilaian.

**Langkah 2:** Normalisasi Matriks Keputusan

Setiap indikator/atribut memiliki satuan dan skala yang berbeda, maka perlu dilakukan normalisasi agar semua nilai berada pada skala yang seragam. Normalisasi dilakukan berdasarkan jenis indikator/atribut, yaitu:

- 1) *Benefit* (semakin tinggi semakin baik)

Contoh: Nilai

$$R_{ij} = \frac{X_{ij}}{\max(X_j)}$$

Keterangan:

$R_{ij}$  : Nilai hasil normalisasi untuk alternatif ke-i pada indicator/atribut ke-j

$X_{ij}$  : Nilai asli alternatif ke-i pada indicator/atribut ke-j

$Max(X_j)$  : Nilai maksimum dari seluruh alternatif pada indicator/atribut ke-j

2) *Cost* (semakin rendah semakin baik)

$$R_{ij} = \frac{\min(x_j)}{x_{ij}}$$

Keterangan:

$\min(X_j)$  : Nilai minimum dari seluruh alternatif pada indikator/atribut ke-j

**Langkah 3:** Menentukan bobot tiap indicator/atribut

Bobot diberikan berdasarkan tingkat kepentingan masing-masing indicator/atribut.

**Langkah 4:** Menghitung skor akhir setiap alternatif

Setelah semua nilai dinormalisasi, kemudian dikalikan dengan bobot masing-masing indicator/atribut. Nilai akhir dihitung menggunakan rumus berikut:

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

Keterangan:

$V_i$  : Nilai akhir/skor total alternatif ke-i

$W_j$  : Bobot dari indicator/atribut ke-j

$R_{ij}$  : Nilai hasil normalisasi alternatif ke-i pada indicator/atribut ke-j

$n$  : Jumlah indicator/atribut

#### **Langkah 5:** Menentukan Peringkat

Semua skor alternatif dibandingkan dan diurutkan dari skor tertinggi ke terendah. Alternatif dengan nilai tertinggi dianggap paling layak untuk diterima sebagai penerima beasiswa.

Dari uraian di atas dapat disimpulkan bahwa metode *Simple Additive Weighting* (SAW) adalah salah satu metode yang paling sederhana untuk diterapkan, karena memiliki algoritma yang mudah dipahami. Selain itu, metode SAW juga efektif dalam membantu proses pengambilan keputusan dengan berbagai kriteria.

## **4. Website**

### **a. Website**

*Website* adalah kumpulan halaman yang berisi berbagai jenis konten yang dapat mengakses menggunakan mesin pencari atau *search engine*. Sedangkan menurut (Costa, 2022), situs *web* mengklaim bahwa informasi tersedia di setiap halaman di dalam *domain*. Informasi yang dapat disertakan di situs *web* biasanya terdiri dari teks, gambar, animasi, audio, video dan konten lainnya untuk berbagai tujuan (Waruwu et al., 2024).

Pada *website* terdapat dokumen yang disebut *web page* dan *link* dalam *website* yang memungkinkan pengguna dapat berpindah dari satu *page* ke *page* lain, menurut (Haan, 2020). *Website* ialah aplikasi yang menggunakan protokol HTTP (*Hyper Text Transfer Protocol*) yang berisi materi multimedia, menurut (Indayanti et al., 2024). Untuk menciptakan situs *web* yang baik, diperlukan pemenuhan kriteria-kriteria tertentu dalam aspek fungsionalitas, desain, konten, orisinalitas, profesionalisme dan keefektifan (Bekti, 2015, hlm 35).

Dari uraian di atas dapat disimpulkan bahwa *website* adalah sebuah media digital yang terdiri dari sekumpulan halaman yang saling terhubung satu sama lain, menyediakan informasi dan layanan kepada pengguna melalui internet. Dengan evolusi teknologi web, fungsi dan peran *website* terus berkembang, menjadi lebih interaktif, personal, dan integral dalam kehidupan digital masyarakat.

#### **b. CSS (*Cascading Style Sheet*)**

CSS (*Cascading Style Sheet*) adalah salah satu komponen dalam pemrograman *web* yang berfungsi untuk memperindah tampilan serta mengatur struktur *layout* halaman *web* agar lebih menarik dan rapi. CSS dapat mengubah warna, ukuran *font*, dan tata letak menu yang dikontrol oleh CSS. CSS digunakan untuk mengatur *style* atau gaya elemen halaman, mulai dari mengatur teks hingga mengatur tata letak (batas layar). Tujuan penggunaan CSS ini adalah untuk mendapatkan *style* yang konsisten dan menarik untuk elemen tertentu (Permana, 2013).

Menurut (Ardiyansa et al., 2022) *Cascading Style Sheets*, yang juga dikenal sebagai CSS, adalah kumpulan dokumen *web* yang digunakan untuk mendesain tampilan situs *web*. CSS (*Cascading Style Sheets*) merupakan aturan untuk mengatur komponen halaman *web* agar terstruktur dan situs *web* menjadi mudah disesuaikan.

### c. *Javascript*

*JavaScript* merupakan bahasa pemrograman sisi klien, menurut (Sari et al., 2022), dan digunakan untuk membuat situs *web*. Bahasa untuk pemrograman sisi klien *JavaScript* adalah bentuk bahasa pemrograman yang diproses oleh klien. Untuk mengintegrasikan manipulasi *script* HTML dan CSS pada sisi *client/browser*, salah satu bahasa *scripting website* yang paling banyak digunakan adalah *JavaScript*. Penjelasan serupa diberikan oleh (Setiawan, 2015, hlm 2) *JavaScript* adalah bahasa *script* sisi klien, artinya dijalankan disisi klien. *Browser*, seperti Google Chrome, Mozilla Firefox, atau Opera, berfungsi sebagai klien dalam skenario ini. Penjelasan di atas mengarah pada kesimpulan bahwa *JavaScript* adalah bahasa pemrograman yang dijalankan oleh klien seperti Google Chrome dan yang tujuannya meliputi pengembangan situs *web* dan aplikasi seluler, halaman *web* interaktif, *server web* dan aplikasi *server*.

### d. MySQL

MySQL merupakan sistem manajemen basis data yang banyak digunakan dan dikenal luas oleh para pengembang. MySQL dikenal

karena fleksibilitasnya dan kemampuannya dalam menyimpan data, baik dalam jumlah besar maupun kecil. Sistem ini digunakan untuk mengelola basis data dengan memanfaatkan bahasa SQL (*Structured Query Language*). SQL (*Structured Query Language*) adalah bahasa yang dimanfaatkan untuk melakukan akses serta pengelolaan data dalam basis data relasional (Elgamar, 2020). Bahasa ini menjadi bahasa standart yang digunakan dalam manajemen basis data relasional. Dalam MySQL tersedia dalam dua lisensi berbeda, yang pertama adalah perangkat lunak bebas yang dapat digunakan oleh siapa saja, sedangkan yang kedua adalah *shareware*, dimana penggunaan program berpemilik ini dibatasi. (Hamidi et al., 2022) mengemukakan hal serupa, menyatakan bahwa MySQL memiliki hak istimewa yang memastikan semua pengguna hanya dapat menjalankan aktivitas yang diotorisasi untuk mereka.

MySQL adalah perangkat lunak *server* data yang menggunakan SQL, atau bahasa *query* terstruktur. Pengolahan basis data dilakukan dengan menggunakan bahasa terstruktur SQL. Suatu *Database Management System* (DBMS) yang bersifat *open source* disebut MySQL (Prahasti et al., 2022). Menurut definisi yang di atas, MySQL adalah DBMS (*Database Management System*) *open source* yang digunakan untuk pemrograman.

**e. PHP (*Hypertext Preprocessor*)**

PHP adalah bahasa pemrograman yang bersifat dinamis dan umum digunakan dalam pengembangan situs *web*. PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman sisi *server* yang tertanam dalam struktur HTML dan memungkinkan konversi basis kode program menjadi kode mesin yang dapat dipahami komputer (Prahasti et al., 2022). Berdasarkan definisi di atas, dapat disimpulkan bahwa PHP (*Hypertext Preprocessor*), bahasa pemrograman yang digunakan untuk membuat *website*, melakukan tugas mengubah kode sumber program menjadi kode mesin, yang kemudian dimasukkan ke dalam kerangka HTML dan dikenali oleh komputer.

## **5. Perancangan Sistem**


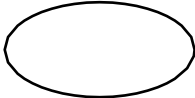


### **a. UML (*Unified Modeling Language*)**

UML (*Unified Modeling Language*) merupakan bahasa yang saat ini secara luas dimanfaatkan untuk memvisualisasikan, merancang, dan mendokumentasikan sistem perangkat lunak. (Syarif & Nugraha, 2020) mengungkapkan hal yang sama. UML adalah bahasa visual yang menggunakan diagram untuk mewakili dan mengkomunikasikan informasi terkait sistem. Diagram adalah blok bangunan utama UML, beberapa spesifik (seperti beberapa jenis diagram waktu), sementara yang lain lebih umum. Bahasa pemodelan yang digunakan untuk perangkat lunak atau sistem berorientasi objek disebut UML atau *Unified Modeling Language* (Suriya & S., 2023).

### **b. *Usecase Diagram***

*Usecase diagram* adalah model yang mendefinisikan perilaku dari sistem yang telah dibangun. Menurut (Tohari, 2014, hlm 47), *usecase diagram* adalah rangkaian atau deskripsi dari sebuah grup yang terhubung satu sama lain dan membentuk sistem yang teratur di bawah kendali seorang aktor. *Usecase diagram* yang dapat dipahami secara lebih ringkas digunakan untuk menentukan fungsionalitas mana yang akan ada pada sistem yang akan dibangun dan siapa yang memiliki kewenangan untuk menggunakannya. Seperti terlihat pada tabel 2.1, terdapat beberapa simbol dalam *usecase diagram* yang digunakan untuk mendeskripsikan diagram.

Tabel 2. 1 Simbol pada *Usecase Diagram*

Simbol	Keterangan
	Aktor: merupakan peran <i>user</i> , sistem atau alat berkomunikasi menggunakan <i>usecase</i>
	<i>Usecase</i> : abstraksi dan interaksi antara sistem dan aktor
	<i>Association</i> : akbstraksi dari penghubung antara aktor <i>usecase</i>
	<i>Generalisasi</i> : spesialisasi aktor guna berpartisipasi dengan <i>usecase</i>
<<Include>>	Menunjukkan <i>usecase</i> secara

Simbol	Keterangan
	keseluruhan yang fungsionalitas yang berasal dari <i>usecase</i> lainnya
<<Extend>>	Menunjukkan <i>usecase</i> tambahan ambahan yang berfungsi melengkapi use case utama apabila kondisi tertentu telah dipenuhi

### c. *Class Diagram*

*Class diagram* adalah suatu spesifikasi yang, ketika diinstansiasi, akan membentuk sebuah objek dan menjadi elemen utama dalam pengembangan serta perancangan berbasis objek. Sementara fungsi kelas yang berbeda diwakili oleh operasi atau metodenya, atributnya adalah variabel kelas. *Class diagram* menunjukkan komposisi dan hubungan antara kelas, paket dan objek, seperti pewarisan, asosiasi.

Menurut (Isnain et al., 2022), *class diagram* menggambarkan sistem dalam bentuk *class* yang masing-masing memiliki deskripsi relasi antar *class* tersebut. Jadi *class diagram* merupakan diagram UML yang menggambarkan *class* dalam sebuah sistem dan berhubungan antara satu dengan yang lain.

### d. *Sequence Diagram*

Menurut (Siska Narulita et al., 2024) *sequence diagram*, yang menggambarkan pesan yang dikirim dan diterima oleh objek, digunakan untuk memvisualisasikan aktivitas atau perilaku objek dalam *usecase diagram*. Akibatnya, untuk melihat diagram urutan, maka harus mengetahui objek yang ada dalam *usecase diagram* serta metode kelas yang berlaku untuk objek tersebut.



Pendapat lain menerangkan bahwa *sequence diagram* menampilkan pesan untuk mewakili interaksi antar komponen sistem termasuk pengguna, antarmuka, dan hal-hal lain di dalam dan sekitar sistem. Dari definisi yang telah dijelaskan sebelumnya, dapat

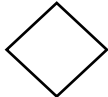


disimpulkan bahwa *sequence diagram* merupakan diagram yang menunjukkan pola interaksi antara objek-objek dalam sistem serta menunjukkan alur pesan atau sinyal yang terjadi di antara objek tersebut.

**e. Activity Diagram**

*Activity Diagram* atau diagram aktivitas “menggambarkan alur kerja atau aktivitas dari suatu sistem atau proses bisnis”, menurut (Yuni Sugiarti, 2013, hlm 75). Penting untuk dicatat bahwa *activity diagram* hanya menunjukkan aktivitas yang dapat dilakukan oleh sistem, bukan tindakan yang mungkin diambil oleh aktor. Selain itu, proses paralel yang dapat berlangsung selama beberapa eksekusi dapat ditampilkan dalam *activity diagram*. Hampir semua status dalam *activity diagram* adalah tindakan.

Tabel 2. 2 Simbol *Activity Diagram*

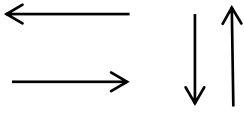
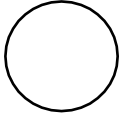
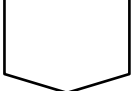
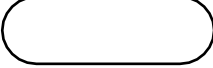

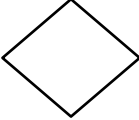
Simbol	Keterangan
 awal	Aktor: merupakan peran orang, ketika berkomunikasi dengan <i>usecase</i>
 aktivitas	<i>Usecase</i> : representasi abstrak dari interaksi antara sistem dan aktor


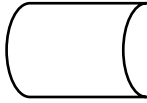


Simbol	Keterangan
 Percabangan	<i>Association</i> : bentuk abstraksi yang menggambarkan hubungan atau koneksi antara aktor dan <i>use case</i>
 Penggabungan	<i>Generalisasi</i> : menunjukkan peran spesifik dari seorang aktor agar dapat terlibat dalam <i>use case</i> tertentu
 akhir	Status akhir sistem menggambarkan kondisi sistem setelah seluruh proses selesai dijalankan

#### f. *Flowchart*

*Flowchart* adalah kumpulan notasi visual simbolik yang menggambarkan alur kerja, aliran data dan urutan proses sistem. *Flowchart* juga dapat dilihat sebagai alat analisis untuk menggambarkan secara rasional, ringkas dan jelas berbagai elemen dari suatu sistem informasi. Pemanfaatan *flowchart* bertujuan untuk membantu *programmer* memahami langkah-langkah dalam menyelesaikan masalah dengan ringkas dan jelas (Khesya, 2020).

Tabel 2. 3 Simbol – simbol pada *Flowchart*

Simbol	Keterangan
 <i>Flow</i>	<p>Simbol untuk menghubungkan satu simbol dengan simbol yang lainnya. Simbol ini dikenal sebagai konektor <i>line</i></p>
 <i>On-page reference</i>	<p>Simbol yang digunakan untuk menandai titik masuk atau keluar suatu proses dalam satu halaman <i>flowchart</i> yang sama</p>
 <i>Off-page reference</i>	<p>Simbol yang digunakan untuk menunjukkan koneksi proses antara halaman yang berbeda dalam <i>flowchart</i></p>
 <i>Terminator</i>	<p>Simbol yang titik awal atau akhir dari suatu alur program</p>
 <i>Process</i>	<p>Simbol yang menunjukkan pelaksanaan suatu proses oleh komputer</p>
 <i>Decision</i>	<p>Simbol yang mencerminkan kondisi tertentu yang mengarah pada kemungkinan jawaban, "YA" atau "TIDAK"</p>

Simbol	Keterangan
 <i>Input/output</i>	Simbol yang menggambarkan proses <i>input</i> dan <i>output</i> secara langsung
 <i>Data Storage</i>	Simbol yang digunakan untuk menunjukkan bahwa data diambil dari atau disimpan ke dalam media penyimpanan seperti disk
 <i>Manual operation</i>	Simbol yang merepresentasikan data yang dibaca dari atau disimpan ke dalam media penyimpanan seperti disk
 <i>Document</i>	Simbol yang digunakan untuk menunjukkan bahwa data <i>input</i> berasal dari dokumen fisik atau bahwa output harus dicetak dalam bentuk fisik

**g. Black Box Testing**

Pengujian *Black Box* menurut (Abdillah et al., 2025) merupakan jenis pengujian kualitas perangkat lunak yang berfokus pada fungsionalitas sistem. Tujuan utama dari pengujian *Black Box* adalah untuk mengidentifikasi fungsi yang tidak berjalan sebagaimana mestinya, antarmuka yang kurang optimal, kesalahan dalam struktur data, kinerja yang tidak sesuai, serta kesalahan saat inialisasi dan

terminasi. Pengujian ini dilakukan setelah program dibuat untuk mengetahui apakah program dapat beroperasi dengan baik dan sesuai dengan spesifikasi, serta untuk mengetahui apakah fungsi, *input* dan *output*nya memenuhi persyaratan tersebut.



Gambar 2. 1 Representasi Metode *Black Box Testing*

Sumber : Olah gambar peneliti

Peneliti menggunakan pengujian *Black Box* sebagai metode pengujiannya. *Black Box Testing* direpresentasikan bagai kotak hitam seperti pada gambar 2.1, sehingga yang bisa dilihat adalah bagian luarnya saja. Dengan kata lain, pengujian perangkat lunak dengan pendekatan seperti ini hanya melihat apa yang terjadi pada bagian luarnya saja tanpa melihat lebih dalam bagaimana proses itu terjadi pada bagian dalam.

## B. Kajian Empiris

Sebelum adanya penelitian ini, terdapat beberapa penelitian terdahulu yang membahas terkait sistem pendukung keputusan penerimaan beasiswa. Berikut ini ialah beberapa penelitian terdahulu yang sesuai dengan topik penelitian ini:

1. Penelitian yang dilakukan oleh (Syifa et al., 2022) dengan judul Sistem Pendukung Keputusan Pemilihan Calon Penerima Beasiswa Tahfidz Menggunakan Metode SAW. Dalam penelitian ini, menggunakan metode SAW dengan empat kategori penilaian meliputi jumlah hafalan, nilai tajwid, nilai makhorijul huruf dan status keluarga santri. Sistem ini dirancang untuk memudahkan dewan penguji Pondok Pesantren ATH dalam melakukan proses seleksi pemilihan beasiswa tahfidz di pondok pesantren ATH.
2. Sistem Pendukung Keputusan Penentuan Penerima Beasiswa BAZNAS Kabupaten Asahan dengan Metode SAW adalah penelitian yang dilakukan oleh (Syahfitri et al., 2024). Dalam penelitian ini, menggunakan metode SAW dengan lima kategori meliputi transkrip nilai, penghasilan orang tua, jumlah tanggungan orang tua, wilayah tempat tinggal dan prestasi. Sistem yang dibangun ini berbasis *web* yang menggunakan bahasa pemrograman PHP dengan MariaDB sebagai *databasenya*. Sistem ini dirancang untuk membantu pihak BAZNAS Kabupaten Asahan dalam proses perhitungan sehingga hasil perhitungan untuk seleksi penerima beasiswa lebih akurat.
3. Pada penelitian dengan judul Perancangan Aplikasi Sistem Pendukung Keputusan Penerimaan Beasiswa Menggunakan Metode SAW, yang

dilakukan (Saputra et al., 2023). Dalam penelitian ini, menggunakan metode SAW. Sistem yang dibangun ini berbasis *web* yang menggunakan bahasa pemrograman PHP dengan MySQL sebagai *databasenya*. Sistem ini dirancang untuk membantu pihak MA Ma'arif NU 5 Sekampung dalam proses seleksi calon penerima beasiswa dengan lebih efisien, akurat, lebih terstruktur dan obyektif.

4. Sistem Informasi Pendukung Keputusan Seleksi Beasiswa Menggunakan Metode SAW merupakan penelitian yang dilakukan oleh (Khaliq et al., 2023). Dalam penelitian ini, metode SAW dengan tiga kategori meliputi IPK, penghasilan dan prestasi. Metode pengembangan sistem yang digunakan adalah *waterfall* dan dibangun menggunakan *framework CodeIgniter*. Sistem ini dirancang untuk memudahkan BAAKPK dalam menyeleksi mahasiswa dan memudahkan mahasiswa dalam memasukkan informasi data diri serta berkas yang diperlukan.
5. Dalam penelitian (Mu'min et al., 2023), yaitu Sistem Pendukung Keputusan Penerimaan Beasiswa Menggunakan Metode *simple Additive Weighting* (SAW). Dalam penelitian ini, menggunakan metode SAW dengan empat kategori penilaian meliputi IPK, penghasilan orang tua, jumlah tanggungan orang tua, keikutsertaan organisasi dan jarak rumah ke kampus. Sistem ini dibangun menggunakan bahasa pemrograman PHP, *Framework Bootstrap* dan *database SQL*. Sistem ini dirancang untuk membantu pihak Universitas Nahdlatul Ulama Sidoarjo (UNUSIDA) untuk menentukan mahasiswa yang

berhak menerima beasiswa KIP dan PPA sehingga hasil perhitungan untuk seleksi penerima beasiswa lebih akurat.

Dari kelima penelitian terdahulu, sistem yang akan dibangun memiliki keunggulan yaitu, dibangun menggunakan metode SAW (*Simple Additive Weighting*). Sistem Pendukung Keputusan Penerimaan Beasiswa dibuat dengan menggunakan *framework Laravel*, sedangkan *databasenya* menggunakan MySQL. *Tools* yang digunakan dalam proses pembuatan sistem ini adalah XAMPP untuk *database* server dan Visual Studio Code untuk *text editor*. Pada pengujian sistem menggunakan *Black Box Testing* yang menguji fungsionalitas perangkat lunak. Sistem yang dibangun diharapkan nantinya dapat memberikan kemudahan pada Pondok Pesantren Kanzul Ulum dalam mengelola konfirmasi pendaftaran, daftar santri, konfirmasi beasiswa dan list penerima.

### C. Kerangka Berfikir

