

BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

Berikut ini beberapa penelitian terdahulu terkait dengan penelitian saat ini. Bangun (2024) melakukan penelitian mengenai penentuan penerima bantuan sosial. Penelitian ini menggunakan kombinasi dua metode, yaitu ELECTRE dan ROC. Metode ELECTRE membandingkan alternatif berdasarkan kriteria, sedangkan Metode ROC menentukan bobot kriteria berdasarkan peringkat yang diberikan oleh pengambil keputusan. Hasil perhitungan mendapatkan alternatif dengan jumlah nilai skor tertinggi yaitu A7.

Siringoringo (2023), melakukan penelitian mengenai pemilihan penerima bantuan yang masih menggunakan penilaian secara subjektif. Penelitian ini membangun sistem pendukung keputusan menggunakan kombinasi dua metode, yaitu ELECTRE dan ROC. ELECTRE digunakan untuk pengambilan keputusan multikriteria berdasarkan perbandingan alternatif berdasarkan kriteria, dan ROC digunakan sebagai metode pembobotan.

Maskhur dan Hadikurniawati (2022), melakukan penelitian mengenai pemilihan pengajar terbaik menggunakan metode ELECTRE. Tujuan penelitian tersebut adalah dapat memudahkan pihak Lembaga Pelatihan Kerja (LPK) dalam penilaian kinerja terhadap pengajar karena masih melakukan perbandingan secara konvensional yang menyebabkan kurang tepat dalam menentukan objektivitas dan efisiensi penilaian. Terdapat lima kriteria yang digunakan dalam pemilihan pengajar terbaik yaitu, penguasaan materi, penyampaian materi, performa, kedisiplinan, dan kerapian. Berdasarkan alternatif yang tersedia, terpilih tiga pengajar terbaik yang direkomendasikan yaitu A1, A2, dan A3. Hasil penelitian yang dilakukan memperoleh keputusan untuk menentukan pengajar terbaik yang terpilih yaitu pengajar A2 karena memiliki skor tertinggi berdasarkan kriteria yang telah ditentukan.

Komsiyah dkk (2019), melakukan penelitian menggunakan metode ELECTRE dalam membangun sistem pendukung keputusan pemilihan vendor

semen. Penelitian tersebut bertujuan untuk memberikan rekomendasi terkait permasalahan dalam pemilihan vendor semen pada PT Wijaya Karya. Terdapat lima kriteria yang digunakan dalam pemilihan vendor semen yaitu, harga, kualitas, waktu tunggu, keuangan, dan K3L. Berdasarkan alternatif yang tersedia, terpilih empat vendor terbaik yang direkomendasikan yaitu A1, A2, A3, dan A4.

Ilmi dkk (2019), melakukan penelitian analisis metode ELECTRE dalam pemilihan proposal program kreativitas mahasiswa yang akan diajukan ke Direktorat Jenderal Pembelajaran dan Kemahasiswaan Kemenristekdikti. Terdapat delapan kriteria penilaian antara lain judul, abstrak, pendahuluan, tujuan, metode, hasil dan pembahasan, kesimpulan, dan daftar pustaka. Berdasarkan penelitian tersebut didapatkan 10 proposal terpilih dari 15 proposal yang ada. Hasil penelitian tersebut dapat memberikan rekomendasi di STIKOM Tunas Bangsa dalam pemilihan program kreativitas mahasiswa.

Sinaga dan Marbun (2020), melakukan penelitian mengenai sistem pendukung keputusan calon pegawai pada PT. Indormarco Primatama menggunakan metode ELECTRE. Penelitian ini dilakukan karena terdapat permasalahan terkait lamanya waktu yang dibutuhkan untuk merekrut calon pegawai karena harus membuat keputusan secara adil dan tepat. Sistem tersebut dirancang menggunakan UML dan dibangun menggunakan PHP, HTML, CSS, Javascript dan MySQL. Terdapat lima kriteria penilaian yaitu, edukasi, pengalaman kerja, tes kemampuan dasar, tes *skill*, dan tes *interview*. Hasil penelitian memperoleh keputusan pelamar A5 berhasil lulus karena memiliki skor tertinggi.

Safitri dan Ibrahim (2019), melakukan penelitian mengenai sistem pendukung keputusan dalam promosi pegawai RRI Palembang menggunakan metode ELECTRE. Penelitian ini dilakukan karena terdapat permasalahan terkait sulitnya melihat kinerja para pegawai yang baik dan memiliki kinerja yang tinggi, serta penilaian kinerja pegawai yang masih dilakukan secara konvensional dengan cara mencatat penilaian pada lembaran kertas. Terdapat tujuh kriteria yang digunakan yaitu target pekerjaan, orientasi pelayanan, integritas, komitmen, kedisiplinan, kerjasam, prestasi kerja, dan masa kerja pegawai. Metode

ELECTRE memproses penilaian berdasarkan kriteria tersebut, kemudian menghasilkan rekomendasi kepada pimpinan untuk menentukan promosi pegawai RRI Palembang.

Haekal (2021), melakukan penelitian untuk mengetahui pengaruh dari karakteristik individu, kinerja dan lingkungan kelembagaan terhadap produktivitas publikasi ilmiah serta terhadap reputasi universitas di Universitas Pancasila. Populasi merupakan seluruh fakultas yang ada di universitas Pancasila. Data dikumpulkan menggunakan kuesioner. Hasil kuesioner diolah menggunakan metode *Structural Equation Modelin* (SEM). Didapatkan hasil dimana produktivitas publikasi ilmiah Universitas Pancasila memiliki pengaruh positif dan signifikan terhadap reputasinya. Dapat disimpulkan strategi untuk meningkatkan reputasi universitas salah satunya melalui produktivitas publikasi ilmiah.

Hozairi dkk (2023), melakukan integrasi ROC dengan *profile match* untuk melakukan seleksi kandidat terbaik sebagai relawan laut nusantara. Metode ROC menghitung bobot kriteria dan sub-kriteria. *Profile Matching* menghitung selisih antara nilai target kriteria dengan nilai yang dihasilkan kandidat, serta melakukan pemeringkatan kandidat terbaik. Terdapat empat kriteria yang digunakan yaitu, kecerdasan, perilaku, sikap kerja, serta kesehatan dan domisili. Agar dapat diterima, nilai kriteria keseluruhan yang didapatkan oleh calon relawan penjaga laut nusantara harus lebih besar atau sama dengan 4. Hasil penelitian mendapatkan, sebanyak enam belas dari tiga puluh calon pendaftar relawan penjaga laut nusantara memenuhi kriteria.

Aisya (2023) melakukan penelitian mengenai implementasi algoritma AHP dan ELECTRE dalam Perekomendasi Program Studi. Penelitian tersebut melakukan perekomendasi program studi berdasarkan nilai mata pelajaran. Hasil dari penelitian yang dilakukan adalah sistem memberikan rekomendasi jurusan berdasarkan ranking alternatif dari inputan perbandingan kriteria oleh siswa.

2.2 Dasar Teori

2.2.1 Sistem Pendukung Keputusan

Sistem pendukung keputusan didefinisikan sebagai suatu sistem informasi yang dirancang dan dibangun untuk dapat mendukung proses pengambilan keputusan dalam bisnis, organisasi, dan manajerial (Ghavami, 2018). Sistem Pendukung Keputusan dapat menjadi suatu alat untuk membantu memberikan informasi berupa rekomendasi yang dapat dijalankan dalam aplikasi komputer (Tasrif, Saputra, Kurniadi, Hidayat, & Mubai, 2021).

Aspek yang ditekankan dalam merancang dan membangun suatu sistem pendukung keputusan antara lain fleksibilitas dan kemampuan beradaptasi yang tinggi. Tahap dalam merancang sistem pendukung keputusan dimulai dari identifikasi masalah, mengumpulkan data-data yang relevan, menentukan pendekatan dalam proses pengambilan keputusan, serta melakukan evaluasi. (Hermawan, Habibie, Sutrisno, Putra, & Aisyah, 2021). Adapun tahapan pengambilan keputusan terbaik adalah sebagai berikut:

1. *Intelligence*

Kecerdasan dapat dipahami sebagai kemampuan untuk menerima informasi dan menyimpannya sebagai pengetahuan yang dapat digunakan.

2. *Design*

Desain diartikan sebagai suatu rencana atau pedoman yang dijadikan acuan dalam pelaksanaan suatu kegiatan atau proses, biasanya dalam wujud prototipe, produk, atau metode.

3. *Choice*

Pilihan adalah langkah untuk menentukan berbagai elemen terkait seperti mencari, menilai, dan menyelesaikan yang dilakukan sesuai dengan model yang telah ditetapkan.

4. *Implementation*

Penerapan adalah cara untuk memahami serta mengimplementasikan kode atau sumber daya pemrograman menjadi sebuah program yang telah dibuat sebelumnya.

2.2.2 Metode ELECTRE

Metode ELECTRE adalah suatu teknik yang digunakan untuk menganalisis pengambilan keputusan yang melibatkan banyak kriteria, dengan cara melakukan perbandingan berbagai alternatif berdasarkan kriteria tertentu (Rahayu, Putri, & Widodo, 2018). Menurut Siregar dkk (2021), metode ELECTRE akan meninggalkan alternatif yang sesuai, karena pilihan yang tidak memenuhi kriteria akan dihapus. Salah satu keuntungan dari metode ELECTRE adalah kemampuannya untuk menyederhanakan proses pengambilan keputusan di tengah ketidakjelasan dan ketidakpastian yang mungkin terjadi dalam situasi dengan banyak pilihan. Langkah-langkah dalam metode ELECTRE adalah (Yosi, Martha, & Imro'ah, 2020):

1. Normalisasi Matriks Keputusan

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^m x_{ij}^2}}, \text{ untuk } i = 1, 2, 3, \dots, m \text{ dan } j \quad (2.1)$$

Keterangan:

r_{ij} = normalisasi pengukuran pilihan dari alternatif ke-i dalam hubungannya dengan kriteria ke-j

x_{ij} = nilai *rating* setiap jawaban responden pada baris dan kolom

m = jumlah alternatif

n = jumlah kriteria

Sehingga dapat matrik R hasil normalisasi sebagai berikut:

$$R = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ r_{21} & r_{22} & \cdots & r_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ r_{m1} & r_{m2} & \cdots & r_{mn} \end{bmatrix} \quad (2.2)$$

Keterangan:

R = matrik yang telah dinormalisasikan

2. Pembobotan pada Matriks yang Telah Dinormalisasi

Setelah mendapatkan matriks R, selanjutnya melakukan pembobotan pada matriks R sehingga mendapatkan Matriks V yang merupakan perhitungan

matriks r_{ij} yang dikalikan dengan nilai bobot preferensi (w_j). Nilai bobot merupakan nilai yang diberikan berdasarkan pada tingkat prioritas suatu kriteria. Proses perhitungan dapat dilakukan pada persamaan berikut:

$$V_{ij} = r_{ij} \times w_j \quad (2.3)$$

$$V = \begin{bmatrix} v_{11} & v_{12} & \dots & v_{1n} \\ v_{21} & v_{22} & \dots & v_{2n} \\ \vdots & \vdots & \vdots & \dots \\ v_{m1} & v_{m2} & \dots & v_{mn} \end{bmatrix} \quad (2.4)$$

Keterangan:

w_j = nilai bobot preferensi

r_{ij} = matriks yang telah dinormalisasi

V = matriks hasil perkalian antara nilai pembobotan dan matriks yang telah dinormalisasi

Matriks V berguna untuk menentukan himpunan concordance dan discordance. Selain itu matriks V digunakan untuk menentukan matriks discordance dengan mencari selisih maksimal dari elemen matriks V tersebut.

3. Menentukan Himpunan *Concordance* dan *Discordance*

Untuk setiap pasang dari alternatif k dan l ($k, l = 1, 2, 3, \dots, m$ dan $k \neq l$) kumpulan J kriteria dibagi menjadi dua himpunan bagian yaitu concordance dan discordance. Sebuah kriteria termasuk concordance jika:

$$C_{kl} = \{j, v_{kj} \geq v_{lj}\}, \text{ untuk } j = 1, 2, 3, \dots, n \quad (2.5)$$

Sebaliknya, dikatakan discordance jika:

$$D_{kl} = \{j, v_{kj} < v_{lj}\} \quad (2.6)$$

Keterangan:

C_{kl} = himpunan concordance

D_{kl} = himpunan discordance

v_{kj} = elemen dari matriks V

v_{lj} = elemen dari matriks V

4. Menghitung Matriks *Concordance* dan *Discordance*

Menghitung matriks *concordance* dengan menjumlahkan elemen yang termasuk pada himpunan *concordance* dengan rumus sebagai berikut:

$$c_{kl} = \sum_{j \in c_{k,l}} w_j \quad (2.7)$$

Sehingga menghasilkan matriks *concordance* sebagai berikut:

$$C = \begin{bmatrix} - & c_{1,2} & c_{1,3} & \dots & c_{1,n} \\ c_{2,1} & - & c_{2,3} & \dots & c_{2,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_{m,1} & c_{m,2} & c_{m,3} & \dots & - \end{bmatrix} \quad (2.8)$$

Sedangkan menghitung matriks *discordance* dengan membagi maksimum selisih kriteria yang termasuk himpunan bagian *discordance* dengan maksimum selisih nilai seluruh kriteria yang ada, dengan rumus sebagai berikut:

$$d_{kl} = \frac{\max[v_{kj} - v_{lj}] j d_{kl}}{\max[v_{kj} - v_{lj}] j} \quad (2.9)$$

Sehingga diperoleh matriks *discordance* sebagai berikut:

$$D = \begin{bmatrix} - & d_{1,2} & d_{1,3} & \dots & d_{1,n} \\ d_{2,1} & - & d_{2,3} & \dots & d_{2,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ d_{m,1} & d_{m,2} & d_{m,3} & \dots & - \end{bmatrix} \quad (2.10)$$

5. Menentukan Matriks Dominan *Concordance* dan *Discordance*

Menentukan matriks dominan *concordance* dapat dilakukan dengan cara membandingkan setiap elemen matriks *concordance* dengan nilai *threshold* c .

$$c_{kl} \geq c \quad (2.11)$$

Nilai *threshold* c yaitu:

$$c = \frac{\sum_{k=1}^m \sum_{i=1}^m c_{ki}}{m(m-1)} \quad (2.12)$$

Sehingga didapatkan elemen matriks F dengan ketentuan sebagai berikut:

$$f_{kl} = \begin{cases} 1, & \text{jika } c_{kl} \geq c \\ 0, & \text{jika } c_{kl} < c \end{cases} \quad (2.13)$$

Sedangkan, menentukan matriks dominan *discordance* dengan bantuan nilai *threshold* d yaitu:

$$d = \frac{\sum_{k=1}^m \sum_{i=1}^m d_{kl}}{m(m-1)} \quad (2.14)$$

Sehingga didapatkan elemen matriks G dengan ketentuan sebagai berikut:

$$g_{kl} = \begin{cases} 1, & \text{jika } d_{kl} \geq d \\ 0, & \text{jika } d_{kl} < d \end{cases} \quad (2.15)$$

6. Menentukan Matriks Agregate Dominant

Menentukan agregat matriks E yaitu dengan melakukan perkalian antara elemen matriks F dengan matriks G yang bersesuaian, dapat dinyatakan sebagai:

$$e_{kl} = f_{kl} \times g_{kl} \quad (2.16)$$

7. Eliminasi Alternatif yang *Less Favourable*

Eliminasi pilihan yang *less favourable* merupakan proses penghapusan matriks dominan agregat atau matriks E. Jika jumlah nilai $e_{kl} = 1$ lebih tinggi dibandingkan dengan alternatif lain, maka opsi tersebut akan dipilih untuk penilaian lebih lanjut. Sementara itu, jika jumlah nilai $e_{kl} = 1$ paling rendah, maka alternatif tersebut akan dihapus. Proses penghapusan matriks E dapat menghasilkan urutan dari berbagai alternatif.

2.2.3 Skala Likert

Skala *Likert* ialah alat ukur dalam psikometri yang sering digunakan dalam kuesioner survei. Ketika responden menjawab pertanyaan yang menggunakan Skala Likert, mereka menunjukkan seberapa setuju atau tidak setuju mereka dengan pernyataan yang diberikan dengan memilih dari opsi yang telah disediakan. (Syofian, Setyaningsih, & Syamsiah, 2015). Pilihan yang tersedia dapat berupa seperti tabel dibawah ini (Azzahra & Ramadhani, 2020):

Tabel 2.1 Bobot Jawaban Skala *Likert*

Kode	Jawaban	Bobot
A	Sangat Mudah/ Sangat Bagus/ Sangat Sesuai/ Sangat Jelas	5
B	Mudah/ Bagus/ Sesuai/ Jelas	4

C	Cukup Sulit/Cukup Bagus/ Cukup Sesuai/ Cukup Jelas	3
D	Sulit/ Jelek/ Tidak Sesuai/ Tidak Jelas	2
E	Sangat Sulit/ Sangat Jelek/ Sangat Tidak Sesuai/ Sangat Tidak Jelas	1

2.2.4 ROC

ROC atau *Rank Order Centroid* merupakan teknik untuk memberikan bobot pada kriteria sesuai dengan urutan prioritasnya. Kriteria yang memiliki bobot tertinggi adalah yang paling signifikan di antara kriteria lainnya (Bangun, 2024). Tahapan untuk menghitung nilai bobot (W) dalam ROC menggunakan rumus berikut:

$$W_k = \frac{1}{k} \sum_{i=1}^k \left(\frac{1}{i} \right) \quad (2.17)$$

Keterangan:

W_k : Bobot untuk kriteria ke-k

k: Jumlah total kriteria

i: Peringkat dari 1 hingga k

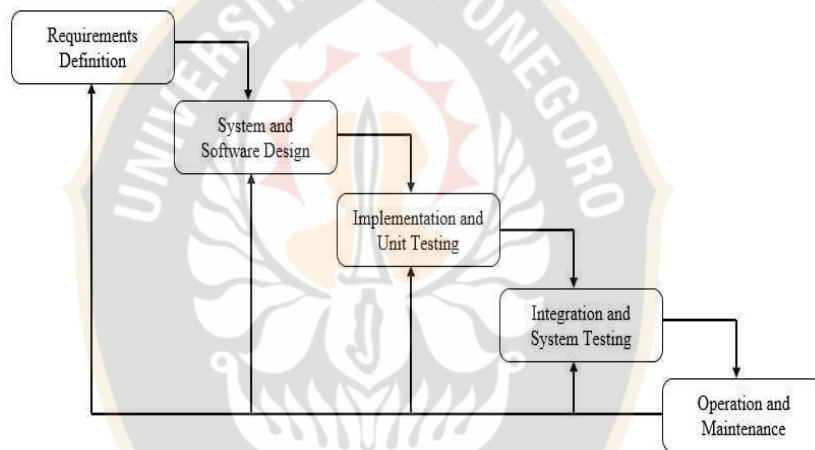
2.2.5 Website

Website atau *World Wide Web* adalah sebuah layanan yang bisa diakses oleh pengguna komputer asalkan terhubung dengan internet. *Website* merupakan halaman informasi yang dapat diakses secara internasional selama terhubung ke jaringan internet (Surentu, 2020).

Website terdiri dari serangkaian halaman *web* yang dioperasikan melalui browser dan diakses melalui internet. Website berlokasi di WWW atau *World Wide Web*, yang merupakan domain atau subdomain. Sebuah *website* dibuat menggunakan bahasa pemrograman HTML (*Hyper Text Markup Language*), atau dapat dikembangkan dengan bahasa pemrograman yang lebih dinamis seperti PHP (*Hypertext Preprocessor*) (Endra, Aprilinda, Dharmawan, & Ramadhan, 2020).

2.2.6 Waterfall

Metode *waterfall* adalah pendekatan sistematis dan bertahap dalam pengembangan perangkat lunak. Pendekatan ini sering dikenal sebagai “Siklus Hidup Klasik”. Metode ini pertama kali diperkenalkan oleh Winston Royce sekitar tahun 1970. Esensi dari metode ini adalah bahwa setiap langkah dilakukan secara berurutan, dan untuk melanjutkan ke langkah berikutnya, tahap sebelumnya harus telah selesai (Suhirman, Hidayat, Saputra, & Saifullah, 2021). Metode *waterfall* menggambarkan pengembangan yang mengalir ke bawah seperti air terjun, seperti yang ditunjukkan pada Gambar 2.1 (Rizaly & Rahman, 2021).



Gambar 2.1 Metode *Waterfall*

Dari Gambar 2.1 terdapat tahapan metode *waterfall* yang dapat dijelaskan sebagai berikut (Firliana, Indriati, Farida, & Niswatin, 2019):

1. *Requirements Definition*

Tahap ini dilakukan pengumpulan spesifikasi yang menjadi kebutuhan *user* pada perangkat lunak yang akan dibangun. Selain itu, dilakukan pula pengumpulan data yang akan menghasilkan dokumen yang berhubungan dalam pembuatan sistem serta menjadi acuan dalam pembuatan desain sistem.

2. *System and Software Design*

Tahap ini melibatkan pembuatan rancangan perangkat lunak berdasarkan hasil analisis kebutuhan dari tahap sebelumnya. Desain yang disusun mencakup

database, struktur perangkat lunak, tampilan antarmuka, dan program yang akan diterapkan.

3. *Implementation and Unit Testing*

Tahap ini dilakukan proses penerjemahan rancangan agar mesin dapat memahami apa yang akan dibuat. Rancangan tersebut harus diubah menjadi bahasa pemrograman. Tahap ini akan menghasilkan program komputer sebagai hasil implementasi dari rancangan yang telah dibuat pada tahap sebelumnya.

4. *Integration and System Testing*

Tahap ini dilakukan pengujian fungsi program untuk mengurangi kesalahan yang mungkin terjadi serta memastikan bahwa perangkat lunak sesuai dengan kebutuhan pengguna.

5. *Pemeliharaan (Maintenance)*

Tahap ini dilakukan perubahan atau pengembangan sistem untuk mengatasi perubahan akibat kebutuhan perkembangan fungsional dari pengguna atau penyesuaian terhadap perubahan lingkungan.

2.2.7 *Unified Modelling Language (UML)*

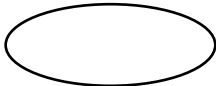
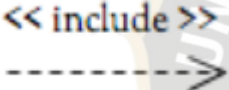
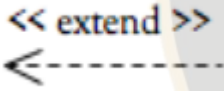
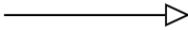
Unified Modelling Language (UML) adalah sebuah standar bahasa yang dimanfaatkan untuk mendokumentasikan desain dalam proses pembangunan perangkat lunak. UML adalah pendekatan dalam pengembangan sistem yang berfokus pada objek (Nistrina & Sahidah, 2022). UML diwakili oleh grafik atau gambar yang berfungsi untuk menggambarkan sistem dalam pengembangan perangkat lunak. UML terdiri dari serangkaian diagram yang mencerminkan aspek-aspek tertentu dari sistem (Aidah & Supriati, 2018).

1. *Use Case Diagram*

Use case diagram adalah sebuah representasi yang menggambarkan bagaimana sistem yang sedang dibuat berfungsi. *Use case* menunjukkan hubungan antara sistem yang dirancang dan pengguna yang akan

memanfaatkan sistem tersebut. Simbol use case yang diterapkan dalam studi ini bisa dilihat pada Tabel 2.2 (S & Shalahuddin, 2018).

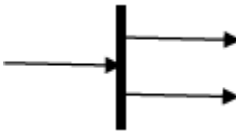
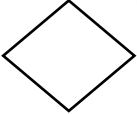
Tabel 2.2 Simbol *Use Case Diagram*

Simbol	Nama	Keterangan
	<i>Use Case</i>	Deskripsi mengenai tindakan yang dilakukan sistem.
	<i>Aktor</i>	Kumpulan peran pengguna yang berinteraksi dengan use case.
	<i>Association</i>	Hubungan antara satu objek dengan objek lainnya.
	<i>Include</i>	Menjelaskan bahwa use case sumber terhubung
	<i>Extend</i>	Menjelaskan bahwa use case target memperluas perilaku dari use case sumber.
	<i>Generalization</i>	Hubungan antara dua use case yang memiliki fungsi lebih umum dibandingkan dengan yang lainnya, dikenal sebagai hubungan generalisasi dan spesialisasi.

2. Activity Diagram

Activity Diagram adalah representasi yang menggambarkan jalur atau urutan dari kegiatan dalam sebuah sistem. Jalur aktivitas yang ditampilkan mencerminkan kegiatan dalam sistem yang sedang dibuat. Simbol-simbol untuk diagram aktivitas bisa ditemukan di Tabel 2.3. (S & Shalahuddin, 2018).

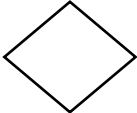
Tabel 2.3 Simbol Activity Diagram

Simbol	Nama	Keterangan
	<i>Activity</i>	Untuk menggambarkan proses sistem.
	<i>Fork</i> (percabangan)	Memisahkan beberapa aliran yang melalui <i>activity</i> menjadi beberapa aliran.
	<i>Join</i> (penggabungan)	Untuk menyinkronkan beberapa aktivitas kembali menjadi satu aliran.
	Status akhir	Status akhir aktivitas sistem.
	Status awal	Status awal aktivitas sistem.
	Percabangan	Percabangan menjadi lebih dari satu aktivitas.
	<i>Swimlane</i>	Memisahkan entitas yang terlibat terhadap aktivitas yang terjadi.

3. Class Diagram

Class diagram merupakan pemodelan yang menjelaskan struktur sistem berdasarkan kelas-kelas. *Class diagram* berfungsi agar pembuat program dapat membuat kelas-kelas sesuai rancangan dan perancangan perangkat lunak yang dibangun dapat sesuai. Adapun simbol *class diagram* dapat dilihat pada Tabel 2.4.

Tabel 2.4 Simbol *Class Diagram*

Simbol	Nama	Keterangan
	<i>Generalization</i>	Hubungan objek anak (<i>descendent</i>) memiliki perilaku dan struktur data dari objek induk (<i>ancestor</i>).
	<i>Nary Association</i>	Digunakan untuk menghindari keterhubungan lebih dari dua objek.
	<i>Class</i>	Kumpulan objek-objek yang memiliki atribut dan operasi yang sama.
	<i>Collaboration</i>	Gambaran mengenai urutan tindakan sistem untuk menghasilkan hasil yang dapat diukur bagi seorang aktor.
	<i>Directed Association</i>	Hubungan antar kelas yang menunjukkan bahwa satu kelas digunakan oleh kelas lainnya.
	<i>Dependency</i>	Hubungan perubahan pada elemen mandiri mempengaruhi elemen yang bergantung padanya.
	<i>Association</i>	Keterhubungan antara satu objek dengan objek lainnya
	Generalisasi	Hubungan antar kelas yang menunjukkan adanya generalisasi dan spesialisasi
	<i>Aggregation</i>	Hubungan antar kelas yang menunjukkan adanya hubungan seluruh-bagian (<i>whole-part</i>)

2.2.8 PHP (*Hypertext Preprocessor*)

PHP (*Hypertext Preprocessor*) merupakan bahasa pemrograman yang bersifat *open-source* dan digunakan untuk pengembangan situs *web*, serta dapat disisipkan langsung ke dalam HTML (Kudratovna, 2022). Keunggulan dari PHP memberikan kesempatan bagi pengguna untuk melakukan perubahan dan pengembangan sesuai kebutuhan mereka (Endra, Aprilinda, Dharmawan, & Ramadhan, 2020).

Agar dapat menjalankan pemrograman PHP, *file* harus diunggah ke *server*, yaitu proses pemindahan data atau *file* dari komputer pengguna ke *web server*. Untuk membuat situs *web* yang interaktif dan mudah diperbarui, diperlukan perangkat lunak yang mampu mengelola data dengan mudah dari komputer pengguna atau *server* di *browser* (Mubarak, 2019).

PHP dibuat untuk dapat bekerja sama dengan *web server* Apache, namun juga dapat digunakan dengan *web server* lainnya seperti IIS (Internet Information Server), PWS (Personal Web Server), dan Xitami. Keunikan PHP terletak pada penggunaan tag khusus yang diawali dengan `<>`, yang membedakannya dari bahasa pemrograman lainnya (Mubarak, 2019).

2.2.9 Laravel

Taylor Otwell menciptakan Laravel dalam suatu proyek yang dimulai pada April 2011. Proyek tersebut diluncurkan karena tidak ada *framework* yang memiliki kelebihan dalam hal pembaruan yang sesuai dengan versi PHP. Keterbatasan sumber daya menjadi kendala dalam pengembangan *framework* yang ada. Oleh karena itu, Otwell mengembangkan *framework* Laravel dengan persyaratan untuk menggunakan PHP versi 5.4 atau yang lebih baru (Utami Sinaga & Samsudin, 2021).

Fungsi-fungsi yang terdapat dalam *framework* Laravel mencakup kode yang dapat diakses melalui perpustakaan dan dapat diinstal dalam Laravel. Salah satu keuntungan menggunakan Laravel adalah ada komunitas besar, yang memungkinkan banyaknya perpustakaan yang dapat diakses. Penelitian menunjukkan bahwa Laravel mampu menangani proyek pengembangan web dari

skala kecil hingga menengah berkat banyaknya perpustakaan yang tersedia (Endra, Aprilinda, Dharmawan, & Ramadhan, 2020).

Laravel merupakan *framework* PHP yang bersifat *open source*. Dengan mengadopsi desain *Model-View-Controller* (MVC), Laravel dirancang untuk pembuatan aplikasi situs *web*. Tujuan dari Laravel adalah untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal, biaya pemeliharaan, dan mengoptimalkan pengalaman dalam bekerja dengan aplikasi melalui sintaks yang ekspresif, bersih, dan efisien (Utami Sinaga & Samsudin, 2021).

2.2.10 MySQL

Basis data adalah kumpulan data yang disusun sesuai dengan aturan tertentu dan saling terkait agar lebih mudah dikelola. Dengan pengelolaan ini, pengguna dapat dengan gampang mencari, menyimpan, dan menghapus informasi yang tersedia. Terdapat lima tipe basis data, yaitu *warehouse database*, *operational database*, *relational database*, *distributed database*, dan *end-user database* (Shidqi, Sulistyo, & Ahda, 2022).

1. MySQL adalah salah satu server basis data yang berfungsi untuk mengelola basis data. MySQL termasuk dalam kategori *relational database*. Dalam *relational database*, data disajikan dalam format dua dimensi yang dikenal sebagai tabel. Tabel ini terdiri dari baris dan kolom (Anggraini, Pasha, Damayanti, & Setiawan, 2020).
2. MySQL memiliki beberapa keuntungan, antara lain dapat digunakan tanpa biaya, handal, selalu mendapatkan pembaruan, dan memiliki banyak forum untuk pengguna yang ingin bertanya dan berdiskusi jika mengalami masalah. Biasanya, MySQL diinstal bersamaan dengan *web server*, sehingga proses instalasinya lebih mudah (Dido, Maman, & Suwita, 2020). Beberapa manfaat tambahan dari MySQL meliputi:
3. Kecepatan MySQL tiga hingga empat kali lipat lebih tinggi dibandingkan server basis data komersial lain yang tersedia saat ini, mudah diatur, dan tidak memerlukan keahlian khusus untuk mengelola penginstalan MySQL.

4. Server Basis Data MySQL dapat mengeluarkan pesan kesalahan dalam berbagai bahasa, seperti Inggris, Jerman, Belanda, Spanyol, Prancis, Portugis, dan Italia.
5. MySQL mampu membuat tabel berukuran sangat besar, dengan ukuran maksimum setiap tabel dapat mencapai 4 GB hingga batas ukuran file yang didukung oleh sistem operasi yang digunakan.
6. MySQL bersifat *open source* dan tersedia secara gratis untuk platform Windows, UNIX, dan OS/2.
7. MySQL terhubung dengan PHP, yang keduanya merupakan perangkat lunak open-source, sehingga koneksi yang dihasilkan lebih cepat dibandingkan dengan menggunakan server basis data lainnya. Modul MySQL dalam PHP sudah terintegrasi secara bawaan, sehingga tidak memerlukan pengaturan tambahan pada berkas konfigurasi PHP.

2.2.11 Blackbox Testing

Tujuan dari pengujian *black box* adalah untuk mengevaluasi fungsionalitas sebuah aplikasi. Proses pengujian ini bertujuan untuk mengetahui masukan dalam sistem serta mengecek hasil yang diharapkan dari aplikasi tersebut. Metode pengujian ini dirancang untuk pihak yang tidak memiliki pengetahuan tentang pemrograman dan dapat diterapkan di semua tahap pengujian perangkat lunak (Utomo D, Kurniawan D, & Astuti Y, 2018). Berikut adalah serangkaian kasus uji yang memenuhi syarat berikut (Fitriyanto, Wulansari, Fahrullah, Wulansari, & Mirwansyah, 2020):

1. Kasus uji yang mengurangi jumlah uji tambahan yang perlu disiapkan agar pengujian bisa dipertanggungjawabkan.
2. Kasus uji yang menentukan ada atau tidaknya jenis kesalahan, bukan hanya berkaitan dengan pengujian tertentu.