

BAB II

TINJAUAN PUSTAKA

2.1 Konsep Sistem

2.1.1 Definisi Sistem

Efendi dkk. (2023) menyatakan bahwa:

“Dewasa ini, istilah sistem menjadi sangat populer. Istilah sistem digunakan untuk banyak hal yang berbeda, terutama untuk pemrosesan data. Sistem menekankan pada prosedur jaringan kerja yang saling berhubungan, berkelompok serta bekerja sama untuk mencapai sasaran yang diinginkan.”

2.1.2 Karakteristik Sistem

Menurut Nestary (2020), suatu sistem memiliki sejumlah karakteristik khusus yang menjadi penanda utama bahwa sesuatu dapat dikategorikan sebagai sebuah sistem, di antaranya:

1. Komponen Sistem (*Component*)

Sistem terdiri dari komponen yang saling berinteraksi membentuk satu kesatuan, bisa berupa subsistem yang menjalankan fungsi tertentu dan memengaruhi proses keseluruhan. Sebuah sistem juga bisa menjadi bagian dari sistem yang lebih besar atau *suprasistem*.

2. Batasan Sistem (*Boundary*)

Batasan sistem adalah pembatas antara sistem dengan sistem lainnya atau dengan lingkungan luar, yang menjadikan sistem dapat dipandang sebagai satu kesatuan yang tidak terpisahkan. Dengan adanya batasan ini, sistem dapat beroperasi secara terorganisir tanpa gangguan dari faktor eksternal.

3. Lingkungan Luar Sistem (*Environment*)

Lingkungan luar merupakan segala sesuatu di luar batas sistem yang dapat memengaruhi jalannya sistem, baik secara positif maupun negatif. Lingkungan luar yang positif perlu dijaga, sedangkan yang merugikan perlu dikendalikan agar tidak mengganggu kelangsungan hidup sistem.

4. Penghubung Sistem (*Interface*)

Penghubung sistem berfungsi sebagai media yang mengalirkan sumber daya antar subsistem dalam sistem. Melalui penghubung ini, keluaran dari suatu subsistem dapat menjadi masukan bagi subsistem lainnya, sehingga tercipta integrasi yang membentuk satu kesatuan sistem.

5. Masukan Sistem (*Input*)

Masukan sistem adalah energi atau data yang dimasukkan ke dalam sistem untuk diolah. Masukan dapat berupa pemeliharaan (*maintenance input*) maupun sinyal (*signal input*) yang akan diproses lebih lanjut menjadi informasi.

6. Keluaran Sistem (*Output*)

Keluaran sistem merupakan hasil pengolahan yang berupa informasi atau hasil lain yang berguna, yang juga dapat menjadi masukan bagi subsistem lain dalam sistem tersebut.

7. Pengolahan Sistem (*Processing*)

Proses dalam sistem berfungsi mengubah masukan menjadi keluaran. Contohnya, sistem akuntansi yang mengolah data transaksi menjadi laporan keuangan yang dibutuhkan manajemen.

8. Tujuan dan Sasaran Sistem (*Goal and Objective*)

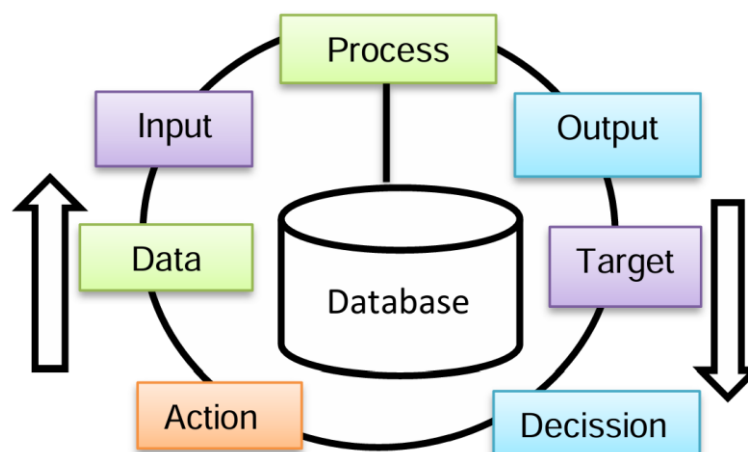
Setiap sistem memiliki tujuan dan sasaran yang jelas. Keberhasilan suatu sistem diukur dari sejauh mana sistem mampu mencapai tujuan yang telah ditentukan. Tanpa sasaran yang jelas, sistem tidak akan memiliki arah dan manfaat yang optimal.

2.2 Konsep Informasi

2.2.1 Definisi Informasi

Menurut Ali & Wangdra (2019) yang dikutip oleh Sitompul & Arnomo (2022) menyatakan bahwa :

“Informasi (*information*) adalah data yang sudah jadi dan suatu bentuk yang berfungsi untuk pengguna atau penerima informasi itu sendiri dan mempunyai nilai yang nyata serta bermanfaat atau dapat digunakan penerima informasi dalam mengambil keputusan yang akan datang atau yang akan digunakan.”



Gambar 2. 1 Siklus Informasi
Sumber: Sitompul & Arnomo (2022)

2.2.2 Kualitas Informasi

Berikut adalah tiga aspek penting yang menentukan kualitas informasi menurut Arifin dkk. (2022):

1. Akurat

Informasi harus benar dan tidak menyesatkan. Saat informasi berpindah dari satu pihak ke pihak lain, ada kemungkinan terjadi gangguan atau perubahan yang bisa memengaruhi keasliannya. Oleh karena itu, keakuratan sangat penting agar informasi dapat dipercaya dan digunakan dengan tepat.

2. Relevan

Informasi harus sesuai dengan kebutuhan pengguna. Informasi akan lebih bernilai jika bermanfaat dalam proses pengambilan keputusan. Meskipun sulit mengukur nilai informasi secara finansial, efektivitasnya dalam membantu pengguna menjadi tolak ukur utama.

3. Tepat Waktu

Informasi harus tersedia saat dibutuhkan. Jika informasi datang terlambat, maka nilainya akan berkurang atau bahkan tidak berguna lagi. Kecepatan dalam memperoleh informasi yang dibutuhkan berperan besar dalam pengambilan keputusan yang efektif.

2.3 Konsep Sistem Informasi

2.3.1 Definisi Sistem Informasi

Sistem informasi merupakan suatu sistem yang dirancang untuk menyediakan informasi yang diperlukan dalam mendukung proses pengambilan keputusan manajerial serta menjalankan operasional perusahaan secara efektif.

Sistem ini terdiri dari kombinasi berbagai elemen, termasuk sumber daya manusia, teknologi informasi, serta prosedur yang terstruktur dan saling terintegrasi. Dengan adanya sistem informasi yang baik, perusahaan dapat meningkatkan efisiensi kerja, mengoptimalkan penggunaan data, serta mempercepat proses dalam mencapai tujuan organisasi (Azwani dkk., 2023).

2.3.2 Komponen Sistem Informasi

Menurut Effendi dkk. (2023), sistem informasi memiliki beberapa komponen utama yang saling terhubung untuk mendukung proses pengolahan data. Setiap komponen berperan dalam mengumpulkan, mengolah, dan menyajikan informasi yang bermanfaat bagi pengambilan keputusan. Berikut adalah komponen-komponen dalam sistem informasi:

1. Blok Masukan (*Input Block*)

Blok masukan dalam suatu sistem mencakup berbagai metode dan media yang digunakan untuk menangkap data sebelum diproses lebih lanjut. Data yang dimasukkan bisa berasal dari berbagai sumber, termasuk dokumen-dokumen dasar yang menjadi dasar bagi sistem informasi.

2. Blok Proses (*Process Block*)

Blok proses merupakan kombinasi dari mekanisme, logika, serta berbagai model yang digunakan untuk memanipulasi data masukan maupun data yang tersimpan dalam basis data. Proses ini bertujuan untuk menghasilkan keluaran berupa informasi yang telah diolah sesuai dengan kebutuhan pengguna.

3. Blok Keluaran (*Output Block*)

Blok keluaran mencakup berbagai bentuk hasil pengolahan data yang dapat berupa dokumen, laporan, atau informasi yang bernilai guna bagi pengguna sistem informasi. Kemudian informasi yang dihasilkan harus berkualitas dan sesuai dengan tujuan sistem agar dapat dimanfaatkan secara optimal oleh pengguna.

4. Blok Teknologi (*Technology Block*)

Blok teknologi berperan dalam menerima masukan, menjalankan proses, menyimpan, mengakses data, dan menghasilkan keluaran yang diperlukan. Komponen teknologi ini mendukung kelancaran seluruh proses dalam sistem, memastikan integrasi yang baik, dan mempercepat pengolahan data.

5. Blok Basis Data (*Database Block*)

Blok basis data terdiri dari kumpulan data yang saling terhubung dan tersimpan dalam perangkat keras, seperti komputer. Data ini digunakan oleh sistem informasi untuk berbagai keperluan, seperti pencatatan, analisis, dan manipulasi data guna menghasilkan informasi yang relevan dan dibutuhkan pengguna.

6. Blok Kendali (*Control Block*)

Blok kendali bertugas untuk mencegah dan menangani berbagai kendala yang dapat menghambat sistem, termasuk kesalahan operasional dan kegagalan sistem. Komponen ini juga memastikan integrasi yang baik serta mendukung pengembangan sistem agar tetap berjalan secara optimal.

2.4 Sistem Pendukung Keputusan (SPK)

Menurut Sarwandi dkk. (2023), Sistem Pendukung Keputusan (SPK) adalah sistem informasi yang dirancang khusus untuk membantu proses pengambilan keputusan melalui pemanfaatan data, model, dan teknik analisis tertentu. SPK berperan mendukung penyelesaian masalah yang kompleks dengan menyediakan informasi yang akurat dan relevan.

Lebih lanjut, mereka menjelaskan bahwa SPK bertujuan meningkatkan efektivitas dan efisiensi pengambilan keputusan. Dengan SPK, pengguna dapat menganalisis data lebih cepat dan mengambil keputusan yang tepat. Selain itu, SPK membantu mengurangi risiko kesalahan, sehingga keputusan yang dihasilkan lebih terpercaya.

2.5 Pemilihan Mahasiswa Berprestasi (Pilmapres)

Pemilihan Mahasiswa Berprestasi (Pilmapres) merupakan program yang bertujuan mengidentifikasi dan memberi penghargaan kepada mahasiswa berprestasi, baik di bidang akademik maupun non-akademik. Program ini memotivasi mahasiswa untuk berinovasi dan memberi penghargaan kepada mereka yang berkontribusi positif pada masyarakat. Meskipun diselenggarakan oleh Balai Pengembangan Talenta Indonesia (BPTI), pelaksanaannya di tingkat wilayah diwenangkan kepada LLDIKTI Wilayah X (Kemdikbudristek, 2024).

2.6 Status *Eligible* Mahasiswa

2.6.1 Definisi Status *Eligible*

Status *eligible* adalah kondisi di mana seorang mahasiswa telah memenuhi seluruh persyaratan administratif dan akademik yang ditetapkan oleh

lembaga penyelenggara, sehingga berhak mengikuti proses seleksi lebih lanjut. Dalam Pilmapres, status *eligible* mengacu pada mahasiswa yang memenuhi syarat sesuai Pedoman Pilmapres 2024 (Kemdikbudristek, 2024).

2.6.2 Kriteria *Eligible* Mahasiswa

Sebelum membahas lebih lanjut mengenai kriteria *eligible* mahasiswa dalam Pilmapres 2024, penting untuk memahami bahwa status *eligible* peserta ditentukan berdasarkan sejumlah faktor yang telah ditetapkan. Kriteria ini mencakup aspek akademik dan persyaratan administratif yang harus dipenuhi oleh setiap calon peserta.

Tabel 2. 1 Panduan Status *Eligible* Peserta Pilmapres

Kriteria	Tingkat Diploma (D3/D4)	Tingkat Sarjana (S1)
Maksimal Semester	Semester VIII	Semester VIII
Usia Maksimal	22 tahun (D3) dan 23 tahun (D4)	23 tahun
Terdaftar di PD-Dikti	✓	✓
IPK Minimal (Sesuai Pedoman Pilmapres)	$\geq 3,00$	$\geq 3,00$

Sumber: Pedoman Pilmapres (2024)

2.7 Bobot Kriteria Penilaian Pilmapres

Penilaian dalam Pemilihan Mahasiswa Berprestasi didasarkan pada kriteria, bobot, dan subkriteria tertentu yang berbeda antara jenjang sarjana dan diploma. Penjabaran masing-masing dapat dilihat pada subbab berikut.

2.7.1 Tingkat Sarjana

Pada tingkat sarjana, penilaian berfokus pada tiga kriteria utama, yaitu Capaian Unggulan (CU), Gagasan Kreatif (GK), dan Kemampuan Berbahasa Inggris (BI). Setiap kriteria memiliki bobot tersendiri dan terdiri atas beberapa subkriteria sebagai dasar evaluasi peserta.

Tabel 2. 2 Penilaian Pilmapres Tingkat Sarjana

Kriteria	Bobot	Subkriteria
CU	0.45	Kompetisi, Pengakuan, Penghargaan, Karir Organisasi, Hasil Karya, Pemberdayaan/Kemanusiaan, Kewirausahaan
GK	0.35	Penyajian Gagasan Kreatif (0.05), Substansi Gagasan Kreatif (0.35), Kualitas Gagasan Kreatif (0.10), Presentasi Gagasan Kreatif (0.25), Tanya Jawab (0.25)
BI	0.20	<i>Content, Accuracy, Fluency, Pronunciation Response, Overall Performance</i>

Sumber: Pedoman Pilmapres (2024)

2.7.2 Tingkat Diploma

Pada tingkat diploma, kriteria penilaian terdiri dari Capaian Unggulan (CU), Produk Inovatif (PI), dan Kemampuan Berbahasa Inggris (BI). Meskipun bobot dan fokusnya berbeda dengan tingkat sarjana, penilaian tetap mengacu pada aspek-aspek utama tersebut untuk menilai prestasi peserta secara komprehensif.

Tabel 2. 3 Penilaian Pilmapres Tingkat Diploma

Kriteria	Bobot	Subkriteria
CU	0.40	Kompetisi, Pengakuan, Penghargaan, Karir Organisasi, Hasil Karya, Pemberdayaan/Kemanusiaan, Kewirausahaan
PI	0.40	Penyajian Produk Inovatif (0.15), Substansi Produk Inovatif (masalah) (0.20), Substansi Produk Inovatif (solusi) (0.35), Kualitas Produk Inovatif (0.30)
BI	0.20	<i>Content, Accuracy, Fluency, Pronunciation Response, Overall Performance</i>

Sumber: Pedoman Pilmapres (2024)

2.8 Metode *Technique for Order of Preference by Similarity to Ideal Solution* (TOPSIS)

2.8.1 Definisi TOPSIS

Menurut Rifqi & Dona (2020) yang dikutip oleh Suri dkk. (2024) menyatakan bahwa:

“Metode TOPSIS merupakan suatu metode pendukung keputusan yang dapat memberikan rekomendasi sesuai dengan yang diharapkan, karena metode TOPSIS didasarkan pada konsep bahwa alternatif yang terbaik tidak hanya memiliki jarak terpendek dari solusi ideal positif, tetapi juga memiliki jarak terpanjang dari solusi ideal negatif.”

2.8.2 Alur Kerja Metode TOPSIS

- Tentukan data kriteria setiap alternatif.
- Subkriteria dan bobot

- Buat matriks keputusan.
- Buat matriks keputusan ternormalisasi:

$$r_{ij} = x_{ij} / \sqrt{(\sum x_{ij}^2)}$$
- Gunakan bobot **W** dari Pedoman resmi untuk pembobotan dalam matriks keputusan terbobot:

$$y_{ij} = W_j * r_{ij}$$
- Hitung solusi ideal positif dan negatif:

$$A^+ = \{y_1^+, y_2^+, \dots, y_m^+\}$$

$$A^- = \{y_1^-, y_2^-, \dots, y_m^-\}$$
- Hitung jarak alternatif terhadap solusi ideal positif dan negatif:

$$D_i^+ = \sqrt{(\sum (y_{ij} - y_j^+)^2)}$$

$$D_i^- = \sqrt{(\sum (y_{ij} - y_j^-)^2)}$$
- Tentukan nilai preferensi setiap alternatif:

$$V_i = D_i^- / (D_i^+ + D_i^-)$$
- Lakukan perankingan.

2.8.3 Keunggulan TOPSIS

Terdapat beberapa keunggulan yang dimiliki metode TOPSIS dalam mendukung proses pengambilan keputusan, di antaranya pendekatan yang sederhana dan sistematis. Alur perhitungannya yang jelas serta tidak kompleks menjadikan metode ini mudah diterapkan. Di samping itu, efisiensi komputasi yang dimiliki oleh TOPSIS menjadikannya salah satu metode yang sesuai untuk diterapkan dalam sistem pendukung keputusan berbasis teknologi yang membutuhkan proses seleksi yang cepat dan terukur (Katili dkk., 2021).

2.8.4 Kelemahan TOPSIS

Menurut Wulandari dkk. (2023), metode TOPSIS memiliki kelemahan karena bobot preferensi tidak didasarkan pada perbandingan langsung antar kriteria secara matematis. Oleh karena itu, TOPSIS sebaiknya dikombinasikan dengan metode lain yang lebih objektif dalam menentukan bobot. Selain itu, TOPSIS juga tidak memiliki mekanisme penentuan prioritas antar kriteria, sehingga dapat memengaruhi validitas hasil jika bobot tidak mencerminkan kepentingan sebenarnya.

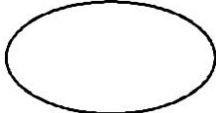
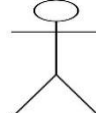
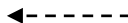
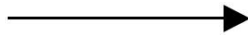
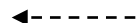
2.9 *Unified Modeling Language (UML)*

Menurut Pakaya dkk. (2020), *Unified Modeling Language (UML)* merupakan suatu bahasa yang digunakan untuk menentukan, memvisualisasikan, membangun, serta mendokumentasikan sistem informasi. Dengan adanya UML, proses pengembangan perangkat lunak menjadi lebih terstruktur karena setiap komponen sistem dapat direpresentasikan dalam bentuk diagram yang mudah dipahami. Adapun, UML membantu mendokumentasikan proses pengembangan, sehingga memudahkan pemeliharaan dan pengembangan di masa depan.

2.9.1 *Use Case Diagram*

Menurut Sularno dkk. (2024), *use case* merupakan pendekatan pemodelan yang digunakan untuk menggambarkan bagaimana suatu sistem informasi berperilaku dalam merespons tindakan dari penggunanya. Melalui pemodelan ini, hubungan interaktif antara aktor dan sistem dapat dijelaskan secara jelas, sehingga alur kerja sistem lebih mudah dipahami dan dikembangkan sesuai kebutuhan.

Tabel 2. 4 Simbol-simbol pada *Use Case Diagram*

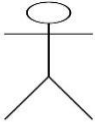
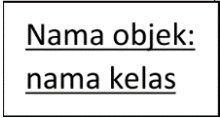
No	Simbol	Keterangan
1.	<i>Use Case</i> 	Perpindahan pesan antar unit atau aktor dalam sistem, diawali kata kerja pada nama <i>use case</i> .
2.	<i>Actor</i> 	Entitas yang berinteraksi dengan sistem, bisa berupa manusia, proses, atau sistem lain.
3.	<i>Assosiasi</i> 	Hubungan antara aktor dan <i>use case</i> yang menunjukkan interaksi dalam sistem.
4.	<i>Extend</i> <<extend>> 	<i>Use case</i> tambahan yang memperluas fungsi utama tanpa mengubahnya.
5.	<i>Generalisasi</i> 	Relasi umum-khusus antar <i>use case</i> , ditunjukkan dengan panah ke <i>use case</i> yang lebih generik.
6.	<i>Include</i> <<include>> 	<i>Use case</i> tambahan yang selalu dijalankan atau dicek saat <i>use case</i> utama berjalan.

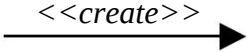
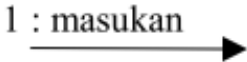
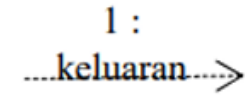
Sumber: Irfan dkk. (2023)

2.9.2 Sequence Diagram

Sequence diagram digunakan untuk membantu dalam memahami persyaratan sistem baru, mendokumentasikan proses, serta memvisualisasikan skenario teknis saat sistem berjalan (*runtime*). Dengan adanya *sequence diagram*, pengguna lebih mudah memahami serta memprediksi bagaimana suatu sistem berperilaku dalam berbagai kondisi operasional (Rohmanto & Setiawan, 2022).

Tabel 2. 5 Simbol-simbol pada *Sequence Diagram*

No	Simbol	Keterangan
1.	<i>Actor</i> 	Entitas eksternal yang berinteraksi dengan sistem, dapat berupa individu, proses, atau sistem lain yang terhubung.
2.	<i>Lifeline</i> 	Mewakili siklus hidup suatu objek dalam sistem, menunjukkan keberadaan objek selama proses berlangsung.
3.	Objek 	Entitas yang berperan dalam interaksi dengan pesan.
4.	Waktu aktif 	Menunjukkan bahwa objek sedang aktif dan berinteraksi.

5.	Pesan tipe <i>Create</i> 	Menunjukkan bahwa suatu objek menciptakan objek lain.
6.	Pesan tipe <i>Send</i> 	Menunjukkan bahwa suatu objek mengirimkan data atau informasi ke objek lain.
7.	Pesan tipe <i>Return</i> 	Menunjukkan bahwa suatu objek mengembalikan hasil ke objek lain setelah menjalankan operasi.

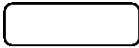
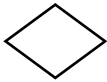
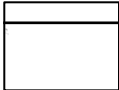
Sumber: Irfan dkk. (2023)

2.9.3 Activity Diagram

Menurut Saputra dkk. (2024), *activity diagram* menunjukkan berbagai alur aktivitas dalam sistem yang sedang dirancang, termasuk bagaimana masing-masing alur bermula, keputusan yang mungkin terjadi, serta bagaimana alur tersebut berlanjut dalam sistem. Diagram ini membantu dalam memahami proses yang terjadi dalam sistem secara lebih terstruktur dan sistematis.

Tabel 2. 6 Simbol-simbol pada Activity Diagram

No	Simbol	Keterangan
1.	Status Awal 	Menunjukkan titik awal proses dalam diagram aktivitas.

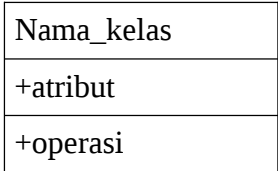
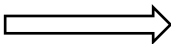
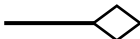
2.	Aktivitas 	Kegiatan yang dilakukan sistem atau aktor dalam suatu proses, biasanya diawali kata kerja.
3.	Percabangan / <i>decision</i> 	Simbol untuk membagi alur proses jika terdapat lebih dari satu pilihan aktivitas.
4.	Penggabungan / <i>join</i> 	Menggabungkan kembali jalur proses yang sebelumnya bercabang menjadi satu alur.
5.	Status akhir 	Menandakan selesainya eksekusi sistem dalam diagram aktivitas.
6.	<i>Swimlane</i> 	Memisahkan peran atau tanggung jawab dalam suatu proses bisnis.

Sumber: Irfan dkk. (2023)

2.9.4 Class Diagram

Class diagram merupakan diagram yang menampilkan berbagai kelas serta paket dalam suatu sistem atau perangkat lunak yang sedang digunakan. Diagram ini juga memberikan gambaran sistem dalam bentuk diagram statis, termasuk hubungan yang terdapat di dalamnya. Dengan *class diagram*, pengembang dapat memahami struktur sistem secara lebih jelas dan terorganisir (Saputra dkk., 2024).

Tabel 2. 7 Simbol-simbol pada *Class Diagram*

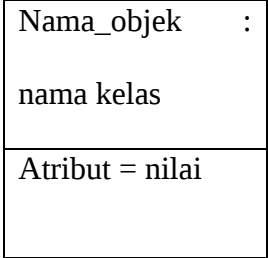
No	Simbol	Keterangan
1.	<i>Class</i> 	Sekumpulan objek dengan atribut dan operasi yang sama.
2.	Antarmuka / <i>Interface</i> 	Kegiatan yang dilakukan sistem atau aktor dalam suatu proses, biasanya diawali kata kerja.
3.	Asosiasi / <i>association</i> 	Hubungan antar <i>class</i> dengan keterkaitan umum, biasanya memiliki <i>multiplicity</i> .
4.	Asosiasi berarah / <i>directed association</i> 	Relasi satu <i>class</i> yang digunakan oleh <i>class</i> lain dengan arah tertentu.
5.	Generalisasi 	Hubungan erat di mana bagian tidak bisa berdiri sendiri tanpa objek utama.
6.	Kebergantungan / <i>dependency</i> 	Ketergantungan satu <i>class</i> terhadap <i>class</i> lain dalam suatu operasi.
7.	Agregasi / <i>aggregation</i> 	Hubungan keseluruhan dan bagian, di mana bagian bisa berdiri sendiri.

Sumber: Suharni dkk. (2023)

2.9.5 Object Diagram

Menurut Destriana dkk. (2022), *Object Diagram* merupakan representasi visual yang menggambarkan objek-objek dalam suatu sistem pada satu titik waktu tertentu. Diagram ini berfungsi untuk memberikan pemahaman yang lebih jelas mengenai struktur dan hubungan antar objek serta bagaimana objek tersebut berinteraksi dalam suatu skenario tertentu. *Object Diagram* lebih menonjolkan perintah-perintah yang dijalankan dibandingkan dengan *class diagram*, sehingga sering disebut sebagai diagram perintah.

Tabel 2. 8 Simbol-simbol pada *Object Diagram*

No	Simbol	Keterangan
1.	Objek 	<i>Instance</i> dari suatu kelas dalam sistem yang sedang berjalan dan memiliki atribut dengan nilai tertentu.
2.	<i>Link</i> 	Menunjukkan hubungan atau keterkaitan antara satu objek dengan objek lainnya dalam sistem.

Sumber: Destriana dkk., (2021)

2.10 Perangkat Lunak Pendukung

Dalam pengembangan sistem, diperlukan berbagai perangkat lunak yang mendukung proses perancangan, pengkodean, serta pengelolaan basis data. Perangkat lunak ini dipilih berdasarkan kompatibilitas, efisiensi, serta kemampuannya dalam mendukung pengembangan sistem berbasis web.

2.10.1 Microsoft Visio

Microsoft Visio merupakan salah satu perangkat lunak dalam paket Microsoft Office yang digunakan untuk membuat berbagai jenis diagram seperti *flowchart*, diagram jaringan, dan ilustrasi prosedur kerja. Keunggulan utama Visio terletak pada kemampuannya menyajikan informasi visual secara rapi dan terstruktur, serta kompatibilitasnya dengan aplikasi Microsoft lainnya seperti Word. Diagram yang dibuat di Visio dapat dengan mudah disisipkan ke dalam dokumen Word untuk memperjelas penjelasan teks, khususnya dalam pembuatan laporan, dokumen SOP, atau dokumentasi sistem informasi (Adawiyah *et al.*, 2021).

2.10.2 Microsoft Excel

Microsoft Excel merupakan perangkat lunak lembar kerja yang digunakan untuk mengolah data, membuat perhitungan otomatis, serta menyajikan informasi dalam bentuk tabel dan grafik. Dengan antarmuka yang mudah digunakan dan fitur yang mendukung analisis data, excel sering dimanfaatkan dalam kegiatan administrasi, keuangan, serta validasi perhitungan pada penelitian berbasis metode numerik seperti pengambilan keputusan (Salsabila & Suhaedi, 2023).

2.10.3 Visual Studio Code

Visual Studio Code merupakan editor kode sumber yang dikembangkan oleh Microsoft untuk sistem operasi Windows, Linux, dan macOS. Editor ini mendukung berbagai bahasa pemrograman serta menyediakan fitur penyorotan *sintaks* dengan variasi warna untuk memudahkan pengembang dalam menulis kode.

Sebagai perangkat lunak *open source*, Visual Studio Code memberikan akses bagi pengembang untuk melihat dan berkontribusi dalam pengembangannya melalui repositori GitHub. Dengan fleksibilitas dan dukungan komunitas yang luas, editor ini menjadi salah satu pilihan utama bagi pengembang dalam membangun berbagai aplikasi di masa depan (Firnando dkk., 2023).

2.10.4 PHP

Menurut Mardiani dkk. (2021), *Hypertext Preprocessor* (PHP) adalah bahasa pemrograman *server-side scripting* yang terintegrasi dengan HTML untuk membangun halaman web yang dinamis. Karena PHP merupakan *server-side scripting*, *sintaks*, dan perintah-perintah yang ditulis dalam PHP akan dieksekusi di server, kemudian hasilnya dikirimkan ke *browser* dalam format HTML. Dengan mekanisme ini, kode program yang ditulis dalam PHP tidak dapat dilihat oleh pengguna, sehingga keamanan halaman web lebih terjamin. Selain itu, PHP memiliki keunggulan dalam kompatibilitas dengan berbagai sistem operasi serta dukungan terhadap banyak jenis *database*, menjadikannya pilihan utama dalam pengembangan aplikasi berbasis web.

2.10.5 Laravel

Menurut Alfarisi dkk. (2023), Laravel merupakan salah satu *framework* PHP yang populer dan banyak digunakan dalam pengembangan aplikasi web. *Framework* ini dirancang untuk menyederhanakan proses pengembangan dengan menyediakan berbagai fitur dan alat yang memungkinkan pengembang menciptakan aplikasi yang efisien, aman, dan mudah dikelola.

Laravel memiliki sintaksis yang ekspresif serta dokumentasi yang lengkap, didukung oleh komunitas yang luas, sehingga menjadi pilihan utama dalam pengembangan aplikasi web. Beberapa fitur utama Laravel meliputi sistem *routing* yang fleksibel, manajemen *otentikasi* yang terintegrasi, serta sistem migrasi basis data yang memudahkan pengelolaan skema *database*.

2.10.6 MySQL

Menurut Noviana (2022), MySQL merupakan salah satu *Database Management System* (DBMS) *open-source* yang mendukung *multiuser* dan *multithreaded*. MySQL dikenal sebagai sistem manajemen basis data yang populer dan gratis. Berdasarkan teori tersebut, dapat disimpulkan bahwa SQL adalah bahasa yang digunakan untuk melakukan permintaan terhadap *database* tertentu. Subbahasa dalam SQL memungkinkan pembuatan serta manipulasi data di dalam *database*. Selain itu, SQL digunakan untuk menjalankan berbagai tugas seperti memperbarui data dalam *database*, yang mengacu pada konsep *Relational Database Management System* (RDBMS).

2.10.7 HTML

Menurut Sari dkk. (2022), HTML (*HyperText Markup Language*) merupakan salah satu bahasa *markup* yang digunakan untuk membangun struktur dasar sebuah *website*. HTML bekerja dengan menggunakan *tag* `< >` yang berfungsi sebagai instruksi bagi *browser* dalam menampilkan elemen-elemen halaman sesuai dengan format yang telah ditentukan. Dalam pengembangan *website*, HTML berperan sebagai fondasi utama sebelum memasuki tahap desain dan pengembangan fitur interaktif.

2.10.8 CSS

CSS (*Cascading Style Sheet*) adalah bahasa yang digunakan untuk mengatur tampilan visual dari elemen-elemen dalam halaman web. CSS berperan penting dalam memisahkan antara isi konten dan desain tampilan, sehingga pengelolaan warna, ukuran huruf, tata letak, serta elemen estetika lainnya menjadi lebih mudah dan fleksibel saat membangun sebuah situs. Dengan menggunakan CSS, pengembang web dapat menciptakan tampilan yang konsisten, menarik secara visual, dan sesuai dengan identitas situs, tanpa harus mencampurkan kode desain ke dalam struktur konten HTML (Rahmatuloh & Revanda, 2022).

2.10.9 JavaScript

Menurut Saepuloh & Seliwati (2022), JavaScript merupakan salah satu bahasa pemrograman *scripting* yang umum digunakan dalam pengembangan aplikasi web karena dijalankan pada sisi klien (*browser*). Bahasa ini bersifat dinamis dan fleksibel, serta mendukung paradigma pemrograman berorientasi objek melalui konsep *object*, *property*, *method*, dan *inheritance (prototype)*.