BAB I
TINJAUAN PUSTAKA

1.1 Landasan Teori

1.1.1 Sistem Informasi

Menurut Sitorus dan Sakban (2021), Sistem Informasi merupakan serangkaian prosedur formal di mana data dikumpulkan, diproses menjadi informasi, serta didistribusikan kepada pengguna untuk mendukung kegiatan operasional, penyampaian informasi, dan pengambilan keputusan di dalam suatu organisasi.

1.1.2 Pengelolaan Obat Esensial

Obat esensial adalah obat yang terpilih dan sangat dibutuhkan dalam pelayanan kesehatan masyarakat. Pengelolaan obat di fasilitas kesehatan tingkat pertama seperti klinik harus mengikuti standar ketat guna menjamin ketersediaan, pemerataan, dan keterjangkauan. Berdasarkan Peraturan Menteri Kesehatan Republik Indonesia Nomor 34 Tahun 2021, pengelolaan sediaan farmasi harus dilakukan secara efektif dan efisien untuk mencegah terjadinya kekosongan atau penumpukan stok yang dapat merugikan pelayanan pasien.

1.1.3 *Early Warning System* (EWS)

Early Warning System (EWS) atau sistem peringatan dini dalam manajemen rantai pasok farmasi didefinisikan sebagai sistem informasi yang mampu mendeteksi potensi ketidakseimbangan stok secara proaktif sebelum mencapai titik

kritis. Menurut Mu'min dkk. (2024), integrasi teknik peramalan ke dalam sistem inventori berfungsi sebagai mekanisme peringatan dini yang memungkinkan pihak manajemen rumah sakit atau klinik untuk mengambil tindakan korektif tepat waktu guna menghindari kerugian finansial akibat stok mati (*dead stock*) atau kekosongan obat. Sistem ini bekerja dengan memvisualisasikan hasil prediksi ke dalam indikator warna atau notifikasi yang menunjukkan tingkat kerawanan stok berdasarkan sisa waktu ketersediaan.

1.1.4 Sistem Pendukung Keputusan (*Decision Support System*)

Menurut Sinaga et al. (2024), Sistem Pendukung Keputusan (SPK) merupakan perangkat lunak berbasis komputer yang dirancang untuk membantu pengambilan keputusan dalam situasi yang kompleks dan tidak terstruktur. SPK sangat bermanfaat pada kondisi saat keputusan yang tepat tidak mudah ditentukan secara langsung.

1.2 Peramalan (*Time Series Forecasting*)

1.2.1 Definisi *Time Series*

Peramalan deret waktu atau *Time Series forecasting* adalah proses penggunaan model statistik untuk memprediksi nilai masa depan berdasarkan data historis yang dikumpulkan secara berkala dalam interval waktu tertentu. Menurut Ankita M. dkk (2024), dalam industri farmasi, pemahaman terhadap data historis sangat krusial untuk memfasilitasi rekomendasi penjualan dan produksi obat melalui

pemahaman pola yang ada. Data deret waktu dalam penelitian ini berupa angka penggunaan obat bulanan di Klinik Pratama Fitria.

1.2.2 Komponen Data (*Trend, Seasonality, Noise*)

Data deret waktu umumnya terdiri dari beberapa komponen yang mempengaruhi akurasi prediksi. Berdasarkan teori yang dipaparkan dalam penelitian Aziz Mu'min dkk (2024), terdapat tiga komponen utama yang harus diidentifikasi dalam pola data inventori farmasi:

1) *Trend*: Menunjukkan kecenderungan data untuk meningkat atau menurun dalam jangka waktu yang panjang.
2) *Seasonality* (Musiman): Menunjukkan fluktuasi periodik yang terjadi pada interval waktu tertentu, seperti peningkatan kebutuhan obat flu di musim hujan.
3) *Noise* (Residu): Variasi data yang tidak teratur atau acak yang tidak dapat dijelaskan oleh tren maupun pola musiman.

1.3 Algoritma Prediksi

1.3.1 ARIMA

Model ARIMA merupakan teknik peramalan yang mengasumsikan adanya hubungan linier antara nilai saat ini dengan nilai-nilai sebelumnya pada data deret waktu. Menurut penelitian Siregar dkk. (2025), ARIMA sangat efektif digunakan di

tingkat fasilitas kesehatan seperti Puskesmas karena mampu menangani data stok obat dengan pola yang cenderung stabil.

Model ini terdiri dari tiga parameter utama:

1. *Autoregressive* (p): Menunjukkan pengaruh nilai masa lalu terhadap nilai saat ini.
2. *Integrated* (d): Tingkat pembedaan (*differencing*) yang dilakukan agar data menjadi stasioner (stabil secara rata-rata dan varians).
3. *Moving Average* (q): Penggunaan kesalahan peramalan masa lalu untuk memperbaiki prediksi masa depan.

Dalam penelitian ini, model ARIMA akan diimplementasikan secara otomatis melalui pustaka pemrograman guna menentukan parameter (p, d, q) yang paling optimal untuk setiap jenis obat esensial.

1.3.2 Facebook Prophet

Prophet adalah algoritma peramalan *open-source* yang dikembangkan oleh Facebook dengan pendekatan model aditif. Berdasarkan penelitian Hidayat dan Sulistiyono (2025), Prophet memiliki keunggulan dibandingkan metode statistik tradisional karena kemampuannya dalam menangani data yang memiliki pola musiman (*seasonality*), tren non-linier, serta adanya data yang hilang (*missing values*).

Persamaan dasar model Prophet adalah:

$$y(t) = g(t) + s(t) + h(t) + \epsilon_t$$

Keterangan:

$g(t)$: Tren yang mewakili perubahan non-periodik dalam nilai deret waktu.

$s(t)$: Perubahan periodik (seperti pola mingguan dan tahunan).

$\boldsymbol{h(t)}$: Efek hari libur (*holidays*) yang mungkin mempengaruhi lonjakan permintaan obat

ϵ_t : Sisaan kesalahan (*error*) yang diasumsikan berdistribusi normal.

1.3.3 Metrik Evaluasi (MAPE & RMSE)

Untuk menentukan tingkat akurasi dan memilih algoritma terbaik antara ARIMA dan Prophet di Klinik Pratama Fitria, penelitian ini menggunakan dua metrik evaluasi standar, yaitu *Mean Absolute Percentage Error* (MAPE) dan *Root Mean Squared Error* (RMSE).

1. *Mean Absolute Percentage Error* (MAPE)

Menurut Hidayat dan Sulistiyono (2025), MAPE memberikan representasi akurasi dalam bentuk persentase sehingga lebih mudah diinterpretasikan oleh manajemen klinik dalam menentukan tingkat risiko kesalahan peramalan. Persamaan matematis untuk MAPE didefinisikan sebagai berikut:

$$\textbf{MAPE} = \frac{\mathbf{1}}{\boldsymbol{n}}\sum_{\boldsymbol{t=1}}^{\boldsymbol{n}}\left|\frac{\boldsymbol{y_t - \hat{y}_t}}{\boldsymbol{y_t}}\right| \times \mathbf{100\%}$$

Keterangan:

- y_t: Nilai data aktual pada periode ke-t.
- $\hat{y}_t$: Nilai hasil peramalan pada periode ke-t.
- n: Jumlah periode atau total data yang dievaluasi.

2. Root Mean Squared Error (RMSE)

RMSE digunakan untuk mengukur seberapa besar penyimpangan hasil prediksi dari nilai aktual dengan memberikan bobot lebih tinggi pada kesalahan (*error*) yang besar melalui proses pengkuadratan. Formula RMSE dinyatakan sebagai berikut:

$$\textbf{RMSE} = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_t - \hat{y}_t)^2}$$

Angka persentase yang dihasilkan oleh kalkulasi *Mean Absolute Percentage Error* (MAPE) memerlukan standarisasi nilai guna mengukur tingkat keandalan model peramalan secara objektif. Dalam memosisikan performa akurasi tersebut, penelitian ini bersandar pada klasifikasi kemampuan peramalan yang dirumuskan oleh Lewis (1982). Batasan teoretis ini membagi tingkat akurasi model ke dalam empat kategori evaluasi yang disajikan pada Tabel 2.1.

Tabel 2.1 Klaster Akurasi Peramalan Berdasarkan Nilai MAPE

Rentang Nilai MAPE	**Kategori Kemampuan Peramalan**
$< 10\%$	***Highly Accurate Forecasting*** **(Sangat Tinggi)**
$10\% - 20\%$	***Good Forecasting*** **(Baik)**
$21\% - 50\%$	***Reasonable Forecasting*** **(Layak / Cukup)**
$> 50\%$	***Inaccurate Forecasting*** **(Tidak Akurat)**

Pembagian klaster penilai dari Lewis (1982) ini memegang peranan krusial sebagai instrumen *guardrail* analitis dalam pangkalan data pendukung keputusan. Melalui batas evaluasi ini, sistem informasi tidak hanya menyajikan deviasi angka mati, melainkan mampu menginterpretasikan derajat validitas hasil prediksi dari algoritma ARIMA maupun Facebook Prophet secara real-time sebelum dikonversi menjadi indikator kedaruratan *Early Warning System* (EWS).

1.4 Teknologi Pendukung & Visualisasi

1.4.1 Dashboard & Visualisasi Data

Dashboard merupakan antarmuka grafis yang menyatukan berbagai metrik dan indikator kinerja utama (*Key Performance Indicators*) ke dalam satu tampilan yang mudah dipahami. Menurut Ankita M. dkk (2024), penggunaan *forecasting dashboard* sangat krusial dalam industri farmasi karena memungkinkan visualisasi tren penjualan dan performa obat secara *real-time*. Dalam penelitian ini, *dashboard* dirancang sebagai bagian dari *Early Warning System* untuk memetakan tingkat kerawanan stok berdasarkan hasil prediksi, sehingga manajemen klinik dapat menentukan prioritas pengadaan obat dengan lebih cepat.

1.4.2 *Tools* Pengembangan Python

Python dipilih sebagai bahasa pemrograman utama karena memiliki ekosistem *library* yang sangat kuat untuk pengolahan data dan statistika. Penelitian ini memanfaatkan *library* Facebook Prophet yang dirancang untuk menangani data deret waktu dengan prosedur otomatis. Menurut Maulana dkk. (2025), penggunaan Python-based Prophet sangat membantu dalam instalasi farmasi karena kemudahannya dalam menangani data yang tidak beraturan serta kemampuan visualisasi komponen tren dan musiman yang mendalam.

1.4.3 Google Collaboratory (Google Colab)

Menurut Bisong (2019), Google Colaboratory atau Google Colab adalah lingkungan *notebook* Jupyter berbasis *cloud* yang dieksekusi secara langsung melalui peramban web. Google Colab memungkinkan peneliti untuk menulis, menjalankan, dan membagikan kode program Python tanpa memerlukan konfigurasi lokal pada perangkat keras komputer. Infrastruktur berbasis *cloud* yang disediakan oleh Google Colab memberikan akses komputasi yang stabil, sehingga sangat efektif digunakan dalam pemodelan algoritma peramalan *Time Series* dengan skala dataset yang kompleks.

1.5 Penelitian Relevan

Tabel 2.2 Penelitian Terdahulu

Judul	**Isi Pokok**	**Kesamaan dengan proyek Penelitian**	**Kekurangan dibandingkan penelitian**
The Utilization of *Time Series* Data *Forecasting* Techniques on Hospitals Pharmaceutical Inventory Jurnal Teknik Informatika & Sistem Informasi (8/2024) (https://journal.maranatha.edu/inde	Penelitian memanfaatkan *Time Series forecasting*, khususnya ARIMA dan Prophet, untuk meramalkan permintaan item farmasi di rumah sakit. Hasil menunjukkan ARIMA menghasilkan prediksi dengan error	Menggunakan ARIMA dan Prophet dalam konteks inventory farmasi berbasis data historis. Fokus pada prediksi stok obat/produk farmasi untuk mendukung perencanaan.	Belum mengaitkan prediksi dengan tingkat kerawanan stok, Tidak mengembangkan dashboard atau sistem peringatan dini (*Early Warning System*) sebagai

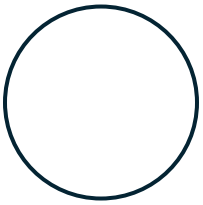
x.php/jutisi/article/view/9352)	terendah dibanding Prophet untuk beberapa item obat.		bagian dari decision support
Analisis Performa Algoritma XGBoost, GRU, dan Prophet dalam Peramalan Penjualan Obat untuk Optimasi Rantai Pasok Farmasi Jurnal Pendidikan & Teknologi Indonesia (2025) (https://jpti.journals.id/index.php/jpti/article/view/562)	Perbandingan performa tiga metode *forecasting* — XGBoost, GRU, dan Prophet — untuk memprediksi penjualan obat, dengan evaluasi metrik error. Hasilnya Prophet tidak unggul dibanding XGBoost dalam dataset ini, tetapi tetap relevan untuk prediksi stok farmasi	Menggunakan Prophet sebagai salah satu pendekatan *Time Series forecasting* bagi data farmasi. Berkaitan langsung dengan manajemen stok/penjualan obat untuk mendukung keputusan operasional	Fokus studi adalah perbandingan model, bukan pengembangan framework prediksi risiko stok atau dashboard EWS. Tidak membahas integrasi prediksi ke sistem pendukung keputusan.
PREDIKSI STOK OBAT TB DENGAN ARIMA DAN ANALISIS VOLATILITAS RESIDUAL DI PUSKESMAS BANDA SAKTI Jurnal Rabit: Jurnal Teknologi & Sistem Informasi (2025) (https://jurnal.univrab.ac.id/index.php/rabit/article/view/6398)	Penelitian ini menerapkan ARIMA untuk memprediksi stok obat TB di Puskesmas, sekaligus mengevaluasi volatilitas residual untuk menilai stabilitas prediksi dalam konteks layanan kesehatan primer.	Metode ARIMA yang sama-sama digunakan untuk meramalkan kebutuhan stok obat berbasis data historis. Dimensi kesehatan primer mirip dengan fokus studi klinik tingkat pertama	Fokus hanya pada ARIMA dan analisis residual, tanpa membandingkan dengan metode lain seperti Prophet. Tidak memformulasikan model prediksi dalam bentuk risk level / dashboard.
Forecasting Medicine Sales Using *Time Series Forecasting* International Journal of	Artikel ini mengeksplorasi model *Time Series forecasting* termasuk ARIMA, LSTM, dan model lain, dan juga	Membedah praktik *forecasting* machines, termasuk ARIMA dan dashboard	Tidak secara khusus ditujukan untuk analisis kerawanan stok obat esensial,

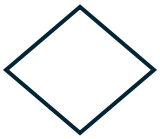
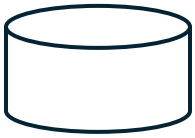
Intelligent Systems and Applications in Engineering (2024) (https://ijisae.org/index.php/IJISAE/article/view/4649)	menyinggung penggunaan dashboard visualisasi prediksi sebagai alat bantu keputusan di dunia farmasi	visualisasi, pada data farmasi. Fokus pada *forecasting* untuk manajemen inventory dan optimalisasi keputusan.	Topiknya berskala industri yang lebih luas, bukan pada unit farmasi klinik
Predicting Tablet Drug Expenditures Using Python-Based Facebook Prophet in Pharmaceutical Installations Jurnal Teknologi dan Open Source (Vol. 8 No. 2) (https://ejournal.uniks.ac.id/index.php/JTOS/article/view/4907)	Studi ini menerapkan model Prophet untuk memprediksi besaran pengeluaran obat (tablet) di instalasi farmasi, mengevaluasi hasilnya menggunakan MAPE & MAE untuk menilai akurasi prediksi medis.	Memanfaatkan Prophet pada aplikasi data farmasi untuk prediksi berdasarkan data historis. Menekankan kemampuan Prophet menangkap tren/seasonal dalam konteks farmasi.	Fokus pada pengeluaran obat, bukan prediksi kerawanan stok/esensial, Belum mengintegrasikan output ke dalam sistem EWS berbasis dashboard

1.6 Aliran Sistem Informasi (ASI) dan Bagan Alir (Flowchart)

Menurut Suherman (2018), Aliran Sistem Informasi (ASI) digambarkan dalam bentuk bagan yang menunjukkan arus pekerjaan sistem secara keseluruhan, serta menjelaskan urutan prosedur yang ada di dalam sistem. Sementara itu, menurut Fatimah dan Samsudin (2019), bagan alir program (*program flowchart*) sifatnya lebih terperinci mengenai langkah-langkah proses di dalam program dari awal sampai akhir, sehingga instrumen ini digunakan untuk menggambarkan struktur logika komputasi dan algoritma secara rinci. Pemodelan ini menggunakan himpunan simbol standar ANSI (*American National Standards Institute)* yang disajikan pada Tabel 2.3

Tabel 2.3 Simbol Aliran Sistem Informasi dan Bagan Alir

No	Simbol	Keterangan
1	Flow Line	Menunjukkan arah aliran proses atau dokumen dari satu tahap ke tahap berikutnya.
2.	Terminator 	Menandai titik awal (*Start*) dan titik akhir (*End*) dari suatu aliran proses atau algoritma.
3.	Manual Operation 	Menyatakan proses fisik yang dilakukan oleh manusia tanpa bantuan komputer (misal: penyerahan resep fisik).
4.	Input/Output(I/O) 	Menyatakan entri data atau pencetakan keluaran tanpa mengikat pada jenis media tertentu.
5.	Pemasukan Manual (*Manual Input*) 	Menyatakan proses entri data ke dalam sistem secara langsung melalui keyboard/antarmuka aplikasi.

6.	Proses (*Processing*) 	Menyatakan pemrosesan data oleh komputer, seperti eksekusi rumus algoritma ARIMA atau Facebook Prophet.
7.	Keputusan (*Decision*) 	Menyatakan percabangan kondisi logika (Ya/Tidak) untuk menentukan jalur proses selanjutnya.
8.	Basis Data (*Database / Data Store*) 	Menyatakan tempat penyimpanan data terstruktur secara digital pada disk (misal: database MySQL/SQLite).
9.	Dokumen (*Document*) 	Menyatakan masukan atau keluaran dalam bentuk cetakan fisik (misal: laporan rekapitulasi stok obat).
10.	Tampilan (*Display*) 	Menyatakan keluaran yang ditampilkan langsung pada layar monitor atau dashboard aplikasi.
11.	Konektor Dalam Halaman (*On-Page Connector*) 	Menyambungkan aliran proses pada halaman dokumen yang sama.
12.	Konektor Antar Halaman (*Off-Page Connector*) 	Menyambungkan aliran proses yang terputus karena berpindah ke halaman lain.

13.	Proses Terdefinisi (*Predefined Process*) 	Menyatakan pelaksanaan suatu bagian prosedur atau sub-program (*sub-routine*) yang modul kodenya didefinisikan secara terpisah (misal: panggilan fungsi pelatihan model ARIMA/Prophet atau *grid search*).

1.7 Unified Modeling Language (UML)

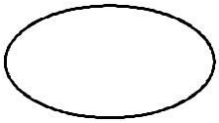
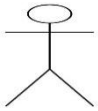
Menurut Nashrullah dan Suprianto (2024), *Unified Modeling Language* (UML) merupakan sebuah alat atau model yang digunakan dalam merancang pengembangan perangkat lunak berbasis berorientasi objek (*object-oriented*). Selain itu, Nashrullah dan Suprianto (2024) juga menjelaskan bahwa UML berfungsi memberikan standar penulisan *blueprint* suatu sistem yang mencakup skema basis data, konsep proses bisnis, penulisan kelas-kelas dalam bahasa pemrograman spesifik, hingga komponen yang dibutuhkan dalam sistem *software*.

1.7.1 *Use Case Diagram*

Menurut Ayu dan Permatasari (2018), *Use Case Diagram* merupakan diagram yang digunakan untuk memvisualisasikan hubungan dan interaksi antara pengguna (*actor*) dengan fungsi-fungsi utama yang disediakan oleh sistem informasi.

Tabel 2.4 Simbol-simbol Pada Use Case Diagram

No	**Simbol**	**Keterangan**

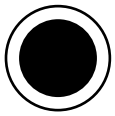
1.	*Use Case* 	Merupakan pekerjaan yang dilakukan oleh aktor. Penamaan *use case* dengan kata kerja
2.	*Actor* 	Merupakan pengguna dari sistem. Penamaan aktor menggunakan kata benda.
3.	*Assosiasi*	Hubungan antara aktor dan *use case*
4.	*Extend* <<*extend*>>	Hubungan antara *use case* dengan *use case*, extends menyatakan bahwa jika pekerjaan yang dilakukan tidak sesuai atau terdapat kondisi khusus, maka lakukan pekerjaan itu .
5.	Generalisasi	Hubungan generalisasi dan spesialisasi antara dua buah *usecase* dimana fungsi yang satu adalah fungsi yang lebih umum dari yang lainnya
6.	*Include* <<*include*>>	Hubungan antara *use case* dengan *use case* lainnya, include menyatakan bahwa sebelum pekerjaan dilakukan

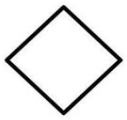
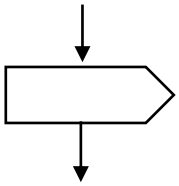
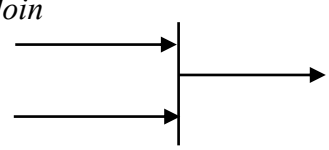
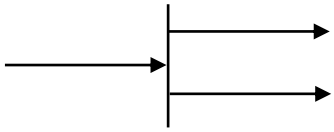
		harus mengerjakan pekerjaan lain terhebih dahulu.

1.7.2 *Activity Diagram*

Menurut Rizka et al. (2023), *activity diagram* mendeskripsikan banyak sekali aliran kegiatan dalam sistem yang sedang didesain, bagaimana masing-masing sirkulasi berawal, *decision* yang mungkin terjadi, serta bagaimana mereka berakhir. Simbol-simbol yang digunakan dalam *Activity Diagram* yaitu:

Tabel 2.5 Simbol-simbol pada Activity Diagram

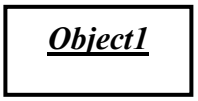
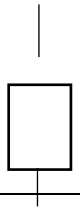
No.	Simbol	Keterangan
1.	*Initial State* 	Menggambarkan titik awal dari setiap aktivitas.
2.	*Final State* 	Menggambarkan titik akhir dari setiap aktivitas.
3.	*Activitiy* 	Menggambarkan aktifitas dari sistem.

4.	*Transition* 	Menunjukan Aliran aktifitas dimana data mengalir dari satu aksi ke aksi berikutnya.
5.	*Decision* 	Menggambarkan titik pengambilan keputusan
6.	*Send* 	Menggambarkan aliran pengiriman
7.	*Join* 	Menggambarkan aliran percabangan
8.	*Fork* 	Menggambarkan aliran penggabungan

1.7.3 *Sequence Diagram*

Menurut Widyatmoko dan Pamungkas (2022), *sequence diagram* digunakan untuk menjelaskan rangkaian langkah-langkah yang dilaksanakan sebagai umpan balik dari sebuah kegiatan guna menghasilkan *output* tertentu yang dibangun oleh sistem aplikasi, serta mendeskripsikan waktu hidup objek dan *message* yang dikirimkan serta diterima antarobjek.

Tabel 2.6 Simbol-simbol pada Sequence Diagram

No	Simbol	Keterangan
1.	*Actor* 	Sebuah ojek yang berasal dari kelas atau dapat dinamai kelasnya saja.
2.	*Lifeline*	*Lifeline* mengindikasikan keberadaan sebuah *object* dalam basis waktu. Notasi untuk *Lifeline* adalah garis putus-putus vertikal yang ditarik oleh objek.
3.	Objek 	Objek merupakan *instance* dari sebuah *class* dan dituliskan tersusun secara *horizontal*. Digambarkan sebagai sebuah *class* (kotak) dengan nama object didalamnya yang diawali dengan titik koma.
4.	*Activation* 	Pada sebuah *lifeline*. Mengindikasikan sebuah obyek yang akan melakukan sebuah aksi.

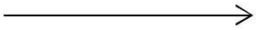
5.	*Message*	*Message* digambarkan dengan anak panah *horizontal* antara *Activation*. *Message* mengindikasikan komunikasi antara objek-objek.

1.7.4 Class Diagram

Diagram kelas atau *class diagram* adalah salah satu jenis diagram struktur pada UML yang menggambarkan struktur serta deskripsi kelas, atribut, metode, dan hubungan dari setiap objek (Wahyuda & Theresiawati, 2022, dalam Himawan & Ikrimach, 2025). Melalui diagram ini, rancangan program dibentuk pada tahapan perancangan sistem sebagai model struktural.

Tabel 2.7 Simbol-simbol pada Class Diagram

No	**Simbol**	**Keterangan**
1.	Kelas Nama Kelas -+Atribut ++Operation()	Kelas pada struktur system

2.	*Interface*  Nama_*Interface*	Sama dengan konsep *interface* dalam pemrograman berorientasi objek
3.	Asosiasi 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain
4	Generalisasi 	Hubungan generalisasi dan spesialisasi antara dua buah *usecase* dimana fungsi yang satu adalah fungsi yang lebih umum dari yang lainnya.
5.	*Agregation* / Agregasi 	Relasi antar kelas dengan makna semua bagian (*whole-part*)

1.7.5 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 Nama_objek : nama kelas Atribut = nilai	Objek dari sistem yang sedang berjalan
2.	*Link* _____	Relasi antar objek