

**IMPLEMENTASI METODE KMEANS DALAM
CLUSTERING PRODUK ECOMMERCE
BERDASARKAN DATA PENJUALAN UNTUK
STRATEGI AWAL BISNIS ONLINE**

SKRIPSI



Oleh:
MUHAMAD AMIR RUDIN YUSUP
211011401487

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS PAMULANG
TANGERANG SELATAN
2025**

**IMPLEMENTASI METODE KMEANS DALAM
CLUSTERING PRODUK ECOMMERCE
BERDASARKAN DATA PENJUALAN UNTUK
STRATEGI AWAL BISNIS ONLINE**

SKRIPSI

Diajukan Untuk Melengkapi Salah Satu Syarat Memperoleh
Gelar Sarjana Komputer



Oleh:
MUHAMAD AMIR RUDIN YUSUP
211011401487

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS PAMULANG
TANGERANG SELATAN
2025**



LEMBAR PERNYATAAN

Yang bertanda tangan dibawah ini:

Nama : MUHAMAD AMIR RUDIN YUSUP
NIM : 211011401487
Program Studi : Teknik Informatika
Fakultas : Ilmu Komputer
Jenjang Pendidikan : Strata 1

Menyatakan bahwa skripsi yang saya buat dengan judul:

IMPLEMENTASI METODE KMEANS DALAM CLUSTERING PRODUK
ECOMMERCE BERDASARKAN DATA PENJUALAN UNTUK STRATEGI
AWAL BISNIS ONLINE

1. Merupakan hasil karya tulis ilmiah sendiri, bukan merupakan karya yang pernah diajukan untuk memperoleh gelar akademik oleh pihak lain, dan bukan merupakan hasil plagiat.
2. Saya ijinkan untuk dikelola oleh Universitas Pamulang sesuai dengan norma hukum yang berlaku.

Pernyataan ini saya buat dengan penuh tanggung jawab dan saya bersedia menerima konsekuensi apapun sesuai aturan yang berlaku apabila di kemudian hari pernyataan ini tidak benar .

Tangerang Selatan, 12 November 2025

MUHAMAD AMIR RUDIN YUSUP
211011401487



LEMBAR PERSETUJUAN

Nama : MUHAMAD AMIR RUDIN YUSUP
NIM : 211011401487
Program Studi : Teknik Informatika
Fakultas : Ilmu Komputer
Jenjang Pendidikan : Strata 1
Judul Skripsi : IMPLEMENTASI METODE KMEANS DALAM CLUSTERING PRODUK ECOMMERCE BERDASARKAN DATA PENJUALAN UNTUK STRATEGI AWAL BISNIS ONLINE

Skripsi ini telah diperiksa dan disetujui oleh pembimbing untuk persyaratan sidang skripsi

Tangerang Selatan, 14 November 2025

Pembimbing

Niki Ratama, S.Kom., M.Kom.
NIDN : 0424119101

Mengetahui,
Ketua Program Studi

Dr Eng Ahmad Musyafa, S.Kom., M.Kom.
NIDN : 0425018609



LEMBAR PENGESAHAN

Nama : MUHAMAD AMIR RUDIN YUSUP
NIM : 211011401487
Program Studi : Teknik Informatika
Fakultas : Ilmu Komputer
Jenjang Pendidikan : Strata 1
Judul Skripsi : IMPLEMENTASI METODE KMEANS DALAM CLUSTERING PRODUK ECOMMERCE BERDASARKAN DATA PENJUALAN UNTUK STRATEGI AWAL BISNIS ONLINE

Skripsi ini telah dipertahankan di hadapan dewan penguji ujian skripsi fakultas Ilmu Komputer, program studi Teknik Informatika dan dinyatakan LULUS.

Tangerang Selatan, 20 November 2025

Penguji I

Penguji II

DOSEN, S.Kom., M.Kom.

NIDN :

DOSEN, S.Kom., M.Kom.

NIDN :

Pembimbing

Niki Ratama, S.Kom., M.Kom.

NIDN : 0424119101

Mengetahui,
Ketua Program Studi

Dr Eng Ahmad Musyafa, S.Kom., M.Kom.

NIDN : 0425018609

ABSTRACT

After bab 5

Keywords:

ABSTRAK

Setelah Bab 5

Kata Kunci:

KATA PENGANTAR

Puji syukur alhamdulillah, penulis panjatkan kehadiran ALLAH SWT yang mana telah melimpahkan rahmat dan karunia-Nya. Sehingga Skripsi yang berjudul “IMPLEMENTASI METODE KMEANS DALAM CLUSTERING PRODUK ECOMMERCE BERDASARKAN DATA PENJUALAN UNTUK STRATEGI AWAL BISNIS ONLINE” dapat terselesaikan, dimana skripsi ini penulis sajikan dengan sebenar-benarnya.

Ucapan terima kasih penulis ucapkan sebesar-besarnya kepada seluruh pihak yang telah membantu penulis dalam penyusunan dan penyelesaian skripsi ini:

1. Bapak Dr. Pranoto, S.E., M.M selaku Ketua Yayasan Sasmita Jaya.
2. Bapak Dr. E. Nurzaman, AM., MM., M.Si. selaku Rektor Universitas Pamulang.
3. Bapak Yan Mitha Djaksana, S.Kom., M.Kom. selaku Dekan Fakultas Ilmu Komputer Universitas Pamulang.
4. Bapak Dr. Eng. Ahmad Musyafa, S.Kom., M.Kom. selaku Ketua Program Studi Teknik Informatika.
5. Bapak Niki Ratama, S.Kom., M.Kom. selaku Dosen Pembimbing yang telah memberikan bimbingan dan semangat untuk menyelesaikan skripsi ini.
6. Kedua Orang tua penulis yang telah memberikan do'a, dukungan, dan kasih sayang yang tiada henti dalam setiap proses penulisan skripsi ini.
7. Rekan-rekan mahasiswa Program Studi Teknik Informatika Universitas Pamulang yang selalu memberikan dorongan dan semangat kepada penulis.
8. Serta semua pihak yang tidak dapat disebutkan satu persatu yang terlibat dalam penyusunan skripsi ini sehingga dapat selesai.

Semoga Allah SWT membalas kebaikan dan selalu mencerahkan hidayah serta berkah-Nya, Aamiin.

Tangerang Selatan, November 2025

MUHAMAD AMIR RUDIN YUSUP

DAFTAR ISI

LEMBAR PERNYATAAN	i
LEMBAR PERSETUJUAN	ii
LEMBAR PENGESAHAN	iii
ABSTRACT	iv
ABSTRAK	v
KATA PENGANTAR.....	vi
DAFTAR ISI.....	vii
DAFTAR GAMBAR.....	ix
DAFTAR TABEL	x
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Identifikasi Masalah.....	3
1.3 Rumusan Masalah.....	3
1.4 Batasan Penelitian.....	4
1.5 Tujuan Penelitian	5
1.6 Manfaat Penelitian	5
1.7 Metodologi Penelitian.....	6
1.7.1 Metodologi Pengumpulan Data	6
1.7.2 Metodologi Pengembangan Sistem.....	7
1.8 Sistematika Penulisan	7
BAB II LANDASAN TEORI	9
2.1 Tinjauan Pustaka	9
2.2 Teori Umum.....	11
2.2.1 Definisi Implementasi	11
2.2.2 Kmeans.....	13
2.2.3 Website.....	15
2.2.4 HTML	17
2.2.5 CSS.....	18
2.2.6 Javascript.....	20
2.2.7 Python	22
2.2.8 Flask	24
2.2.9 Database (Basis Data)	26
2.2.10 MySQL.....	28
2.3 Teori Perancangan Basis Data	29

2.3.1	ERD (<i>Entity Relationship Diagram</i>)	29
2.3.2	Komponen Pada ERD (<i>Entity Relationship Diagram</i>)	30
2.4	Definisi <i>Unified Modeling Language</i> (UML)	31
2.4.1	Jenis-Jenis Diagram <i>Unified Modeling Language</i>	31
2.5	Aplikasi Pendukung	36
2.5.1	Draw.io.....	36
2.5.2	Visual Studio Code	37
2.5.3	XAMPP	38
2.6	Teori Pengujian Sistem	39
2.6.1	Pengujian <i>Black Box</i>	39

DAFTAR GAMBAR

Gambar 2.1 Logo HTML	17
Gambar 2.2 Logo CSS	18
Gambar 2.3 Logo JavaScript.....	20
Gambar 2.4 Logo Python	22
Gambar 2.5 Logo Flask.....	24
Gambar 2.6 Logo MySQL	28

DAFTAR TABEL

Tabel 2.1 Penelitian Terdahulu	9
Tabel 2.2 Komponen Pada ERD	30
Tabel 2.3 Simbol Elemen Activity Diagram.....	31
Tabel 2.4 Simbol Elemen Use case Diagram.....	33
Tabel 2.5 Simbol Elemen Class Diagram.....	34
Tabel 2.6 Simbol Elemen Sequence Diagram.....	35

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam era digitalisasi yang berkembang pesat, kemajuan teknologi internet telah mendorong pertumbuhan e-commerce secara signifikan. E-commerce kini menjadi model bisnis modern yang mempertemukan penjual dan pembeli secara elektronik tanpa dibatasi oleh ruang dan waktu. Di Indonesia, perkembangan ini semakin cepat seiring meningkatnya jumlah pengguna internet dan perangkat seluler, serta adanya dukungan pemerintah dalam menekan biaya transaksi digital (Ratama et al., 2022). Kehadiran e-commerce tidak hanya mengubah pola konsumsi masyarakat, tetapi juga membuka peluang besar bagi pelaku usaha terutama generasi muda untuk memulai bisnis dengan modal yang relatif kecil dan jangkauan pasar yang luas.

Namun, di balik peluang besar tersebut, pertumbuhan pesat e-commerce juga memunculkan tingkat persaingan yang sangat ketat di antara pelaku usaha. Untuk mampu bersaing secara efektif, pelaku bisnis dituntut memahami perilaku konsumen dan pola penjualan secara mendalam. Salah satu tantangan utama yang dihadapi adalah pengelolaan dan pengelompokan produk berdasarkan data penjualan yang bersifat besar, variatif, dan kompleks. Pengelolaan data yang kurang optimal dapat menimbulkan berbagai masalah, seperti kesalahan dalam strategi pemasaran, penumpukan stok pada produk yang kurang diminati, hingga kerugian finansial. Kondisi ini menunjukkan adanya kesenjangan antara besarnya volume data penjualan yang tersedia dengan keterbatasan kemampuan pelaku usaha dalam

mengekstraksi informasi strategis untuk perumusan strategi pemasaran sejak tahap awal (Aryani et al., 2024).

Untuk mengatasi kompleksitas tersebut, diperlukan metode analisis yang mampu mengolah dan mengelompokkan informasi secara sistematis, efektif, dan efisien. Data mining menjadi salah satu solusi yang relevan, khususnya melalui teknik clustering, yang terbukti membantu proses pengambilan keputusan bisnis dengan memberikan gambaran pola penjualan maupun segmentasi produk yang lebih terarah (Singrapati, Dora, & Kurniawan, 2024). Salah satu algoritma clustering yang banyak digunakan adalah K-Means, karena memiliki efisiensi komputasi tinggi, mudah diimplementasikan, serta efektif dalam membagi data ke dalam kelompok homogen berdasarkan kemiripan fitur yang dimilikinya. Dengan demikian, algoritma ini dapat memberikan pemahaman lebih mendalam terhadap pola penjualan dan karakteristik produk (Aryani et al., 2024).

Sejumlah penelitian sebelumnya telah membuktikan efektivitas penerapan K-Means dalam konteks e-commerce, seperti pada segmentasi pelanggan berbasis analisis RFM (Mado & Hendry, 2025) dan pengelompokan reputasi toko (Singrapati et al., 2024). Namun, sebagian besar penelitian tersebut berfokus pada analisis pelanggan atau reputasi toko, sedangkan eksplorasi terhadap pengelompokan produk e-commerce berdasarkan fitur penjualan yang relevan masih relatif terbatas. Padahal, pemahaman terhadap pola pengelompokan produk sangat penting dalam penyusunan strategi inventori, promosi, dan pengambilan keputusan bisnis secara umum.

Berdasarkan uraian tersebut, penelitian ini bertujuan untuk mengembangkan dan mengimplementasikan metode K-Means Clustering dalam

pengelompokan produk e-commerce berdasarkan data penjualan. Hasil yang diharapkan adalah terbentuknya segmentasi produk yang lebih terstruktur dan informatif, sehingga dapat digunakan sebagai dasar dalam perumusan strategi pemasaran awal yang lebih tepat sasaran dan berkontribusi pada peningkatan daya saing pelaku usaha di era bisnis digital yang kompetitif.

1.2 Identifikasi Masalah

Berdasarkan latar belakang di atas, terdapat beberapa permasalahan yang dihadapi yaitu:

1. Kompleksitas Data dan Pengelompokan Produk

Kesulitan pelaku usaha e-commerce mengelompokkan produk secara efektif karena volume data penjualan yang besar dan variatif.

2. Kebutuhan Penerapan Algoritma

Diperlukan penerapan metode Clustering (K-Means) untuk mengolah data penjualan menjadi segmentasi produk yang informatif.

3. Fokus Penelitian K-Means pada Produk

Kurangnya penelitian yang secara spesifik menerapkan algoritma K-Means untuk segmentasi produk (bukan pelanggan atau reputasi toko) berdasarkan data penjualan *e-commerce*.

1.3 Rumusan Masalah

Berdasarkan identifikasi masalah yang ada, maka rumusan masalah untuk penelitian ini adalah sebagai berikut:

1. Bagaimana implementasi algoritma K-Means Clustering dapat diterapkan untuk mengelompokkan produk-produk pada data penjualan *e-commerce*?
2. Bagaimana hasil pengelompokan produk menggunakan K-Means Clustering dapat diinterpretasikan dan dianalisis untuk mendukung perumusan strategi pemasaran awal *e-commerce*?
3. Seberapa efektif kinerja algoritma K-Means dalam menghasilkan segmentasi produk yang homogen dan terstruktur berdasarkan data penjualan *e-commerce*?

1.4 Batasan Penelitian

Agar penelitian ini lebih terarah dan fokus pada tujuan yang ingin dicapai, maka ditetapkan beberapa batasan masalah sebagai berikut:

1. Fokus Metode: Penelitian hanya menggunakan Algoritma K-Means Clustering untuk pengelompokan produk dan tidak membandingkannya dengan algoritma *clustering* lainnya.
2. Lingkup Data: Data yang digunakan dibatasi hanya pada data transaksi penjualan produk dari satu *platform e-commerce* tertentu dalam periode waktu yang telah ditetapkan.
3. Output Analisis: Analisis hasil *clustering* hanya diarahkan untuk perumusan strategi pemasaran awal dan tidak mencakup implementasi atau evaluasi dampak strategi tersebut secara langsung.

1.5 Tujuan Penelitian

Berdasarkan rumusan masalah yang telah ditetapkan, maka tujuan dari penelitian ini adalah sebagai berikut:

1. Mengimplementasikan Algoritma K-Means Clustering untuk mengelompokkan produk-produk berdasarkan data penjualan pada *platform e-commerce* tertentu.
2. Menganalisis dan Menginterpretasikan hasil pengelompokan produk yang dihasilkan oleh K-Means Clustering untuk mendapatkan pola segmentasi produk yang informatif.
3. Memberikan rekomendasi segmentasi produk yang terstruktur sebagai dasar untuk perumusan strategi pemasaran awal yang lebih tepat sasaran bagi pelaku usaha *e-commerce*.

1.6 Manfaat Penelitian

Terdapat manfaat atau keuntungan yang dapat diperoleh dari penelitian ini diantara lain

1. Manfaat Teoritis
 - a) Pengembangan Ilmu Data Mining: Hasil penelitian ini dapat memberikan kontribusi dalam pengembangan dan validasi penerapan algoritma K-Means Clustering pada domain data transaksi *e-commerce*, khususnya untuk segmentasi produk.
 - b) Referensi Akademik: Menjadi referensi dan dasar bagi penelitian-penelitian selanjutnya di bidang Teknik Informatika, khususnya pada sub-bidang *Data Mining* dan sistem pengambilan keputusan bisnis, yang berfokus pada analisis perilaku penjualan *online*.

2. Manfaat Praktis

- a) Bagi Pelaku Usaha *E-commerce*: Memberikan model segmentasi produk yang terstruktur dan informatif, yang dapat digunakan sebagai dasar kuat untuk:
 - i. Mengoptimalkan manajemen inventori (stok).
 - ii. Merumuskan strategi promosi dan pemasaran awal yang lebih terfokus dan efisien.
 - iii. Meningkatkan daya saing di pasar *online*.
- b) Bagi Dunia Industri: Menunjukkan penerapan praktis teknologi *Data Mining* sebagai solusi nyata dalam mengatasi tantangan kompleksitas data di sektor bisnis digital.

1.7 Metodologi Penelitian

Metodologi yang digunakan dalam hal ini ada beberapa metode penelitian yang dipakai antara lain :

1.7.1 Metodologi Pengumpulan Data

Metode pengumpulan data yang digunakan untuk mendapatkan data yang relevan dengan tujuan penelitian adalah:

1. Studi Literatur: Pengumpulan data sekunder melalui kajian pustaka, jurnal ilmiah, dan dokumen resmi terkait teori *Data Mining*, algoritma K-Means, serta konsep strategi pemasaran *e-commerce*.
2. Pengambilan Data Transaksi: Pengumpulan data primer berupa data transaksi penjualan produk (*datasets*) dari *platform e-commerce* yang menjadi objek

penelitian. Data ini akan mencakup atribut-atribut yang diperlukan untuk *clustering* (jumlah terjual, harga, *rating*, dsb.).

3. Wawancara : Melakukan konfirmasi atau validasi interpretasi hasil *clustering* dengan pihak yang berwenang (manajer atau pemilik *e-commerce*) untuk memastikan kesesuaian hasil dengan kebutuhan bisnis.

1.7.2 Metodologi Pengembangan Sistem

Metode yang digunakan untuk mengembangkan dan mengimplementasikan algoritma *clustering* adalah kerangka *Cross-Industry Standard Process for Data Mining* (CRISP-DM). Metode ini dipilih karena sangat sesuai untuk proyek-proyek yang berfokus pada penemuan pengetahuan dari basis data (*knowledge discovery*).

Tahapan utama CRISP-DM yang akan dijalankan meliputi:

1. *Business Understanding* (Pemahaman Bisnis)
2. *Data Understanding* (Pemahaman Data)
3. *Data Preparation* (Persiapan Data)
4. *Modeling* (Pemodelan dengan K-Means)
5. *Evaluation* (Evaluasi dan Interpretasi Hasil)

1.8 Sistematika Penulisan

Sistematika penulisan pada skripsi ini terdiri dari beberapa bab sebagai berikut:

BAB I PENDAHULUAN

Menguraikan konteks dan fondasi penelitian, yang mencakup latar belakang masalah, perumusan masalah yang spesifik, tujuan dan batasan penelitian, hingga metodologi yang akan digunakan

BAB II LANDASAN TEORI

Membahas landasan teoretis yang mendukung penelitian, mencakup konsep utama seperti Bisnis *E-commerce*, Data Mining, penjelasan mendalam mengenai Algoritma K-Means Clustering, metrik validasi (*Silhouette Score*), teori Sistem Pendukung Keputusan (SPK) / Business Intelligence, serta metode pengembangan sistem yang digunakan (CRISP-DM).

BAB III ANALISA DAN PERANCANGAN

Membahas tahap analisis kebutuhan fungsional sistem serta perancangan arsitektur model dan sistem web secara detail, yang divisualisasikan melalui diagram-diagram relevan seperti UML dan desain antarmuka.

BAB IV IMPLEMENTASI DAN PENGUJIAN

Merinci proses implementasi teknis dari hasil perancangan menjadi sebuah sistem yang fungsional, serta menyajikan hasil pengujian performa model prediksi dan pengujian fungsionalitas sistem web.

BAB V PENUTUP

Menyajikan rangkuman kesimpulan dari hasil analisis dan implementasi, sekaligus memberikan saran dan rekomendasi untuk penelitian atau pengembangan sistem di masa depan.

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Berikut ini adalah hasil dari penelitian yang telah dilakukan sebelumnya dengan tema yang sesuai dengan skripsi ini. Penelitian tersebut menjadi referensi dalam menyusun skripsi ini:

Tabel 2.1 Penelitian Terdahulu

Penulis & Tahun	Judul Penelitian	Metode	Masalah Utama	Tujuan	Hasil Kuantitatif Utama
M. Wati & H. Haviluddin (2025)	Penerapan K-Means Clustering untuk Segmentasi Konsumen E-Commerce Berdasarkan Pola Pembelian	K-Means Clustering dengan analisis pola pembelian (e-commerce)	Kurangnya pemahaman mendalam tentang perilaku konsumen dan pola pembelian di e-commerce.	Melakukan segmentasi konsumen e-commerce menggunakan K-Means Clustering untuk menyusun strategi pemasaran yang tepat sasaran.	Menghasilkan segmentasi konsumen ke dalam beberapa klaster berdasarkan pola pembelian, memberikan wawasan untuk menyusun strategi pemasaran yang lebih terpersonalisasi.
F. Khalish, N. M. Piranti, & O. Martadireja (2025)	Implementasi Data Mining Menggunakan Teknik Clustering dengan Metode K-Means	K-Means Clustering (Parameter: Penjualan, Harga, Kategori)	Kesulitan dalam mengidentifikasi produk unggulan dan kurang unggulan berdasarkan data penjualan untuk manajemen stok dan strategi.	Mengelompokkan produk berdasarkan data penjualan untuk membantu manajemen dalam perencanaan stok barang dan strategi promosi yang lebih tepat.	Hasil analisis K-Means menunjukkan pengelompokan produk ke dalam beberapa klaster (misalnya, Cluster 0 (penjualan tinggi/sangat diminati) dan Cluster 1 (penjualan rendah/kurang diminati)).

A. Yani, Z. Azmi, & D. Suherdi (2023)	Implementasi Data Mining Menganalisa Data Penjualan Menggunakan Algoritma K-Means Clustering	K-Means Clustering	Perusahaan/toko ritel kesulitan dalam menentukan pola penjualan produk yang laris dan tidak laris secara manual.	Menerapkan Data Mining dengan algoritma K-Means Clustering untuk menganalisa data penjualan dan menemukan pola penjualan produk.	Pengujian dengan K=3 menghasilkan 3 klaster produk: Cluster 1 (Sangat Diminati), Cluster 2 (Cukup Diminati), dan Cluster 3 (Kurang Diminati), memungkinkan fokus strategi pada produk unggulan.
E. Febrianty, L. Awalina, & W. I. Rahayu (2023)	Optimalisasi Strategi Pemasaran dengan Segmentasi Pelanggan Menggunakan Penerapan K-Means Clustering pada Transaksi Online Retail	Kombinasi Model RFM (Recency, Frequency, Monetary) dan K-Means Clustering	Perlunya strategi pemasaran yang lebih optimal dan terpersonalisasi untuk setiap segmen pelanggan online retail.	Melakukan segmentasi pelanggan online retail menggunakan model RFM yang dikelompokkan dengan K-Means Clustering untuk mengoptimalkan strategi pemasaran.	Menghasilkan 3 klaster pelanggan (misalnya: Cluster 1 High Value, Cluster 2 Medium Value, Cluster 3 Low Value) dengan nilai Silhouette Coefficient rata-rata sebesar 0.575, menunjukkan kualitas clustering yang baik.
C. Hafidz Ardana, A. A. A. A. Khoyum, & M. Faisal (2024)	Segmentasi Pelanggan Penjualan Online Menggunakan Metode K-Means Clustering	K-Means Clustering	Perlunya identifikasi segmen pelanggan berdasarkan penjualan untuk menentukan strategi bisnis yang tepat pada penjualan online.	Mengelompokkan pelanggan berdasarkan data penjualan menggunakan K-Means Clustering untuk membantu manajemen toko online dalam merumuskan strategi pemasaran.	Menghasilkan pengelompokan pelanggan menjadi 3 klaster: Cluster 1 (Pelanggan Loyal/Sangat Potensial), Cluster 2 (Pelanggan Potensial), dan Cluster 3 (Pelanggan

				Kurang Potensial) dengan tingkat penjualan yang berbeda di setiap klaster.
--	--	--	--	--

2.2 Teori Umum

2.2.1 Definisi Implementasi

Secara umum, implementasi merujuk pada proses menerjemahkan suatu rencana, konsep, atau teori ke dalam tindakan nyata atau praktik. Dalam konteks artikel dan jurnal ilmiah, implementasi adalah tahapan krusial di mana temuan, model, metode, atau solusi yang dihasilkan dari penelitian diterapkan, diuji, atau diintegrasikan dalam suatu sistem, lingkungan, atau konteks nyata untuk mencapai tujuan tertentu atau memecahkan masalah praktis (Zhou, 2023)

Implementasi melibatkan penerapan prosedur, alat, atau teknik yang telah diverifikasi secara ilmiah. dalam desain penelitian terapan (*applied research*), implementasi adalah fase di mana hasil penelitian diujicobakan di lapangan (setting nyata) untuk mengukur efektivitas dan kebermanfaatannya. Konsep ini menekankan bahwa keberhasilan suatu penelitian tidak hanya dinilai dari hasil teoritisnya, tetapi juga dari kemampuan hasil tersebut untuk diaplikasikan dan memberikan dampak yang terukur (Sequeira, 2024)

Definisi lain datang dari ranah rekayasa perangkat lunak dan sistem informasi. Pressman dan Maxim (2021, hlm. 209) mendefinisikan implementasi (dalam konteks proyek) sebagai “penerjemahan desain ke dalam kode kerja, pengujian, dan integrasi komponen-komponen yang telah diverifikasi.” Meskipun definisi ini spesifik pada teknologi, intinya tetap sama: implementasi adalah proses

mengubah konsep desain menjadi *artefak* yang berfungsi dan dapat digunakan. Dalam konteks penelitian eksplanatif atau eksperimental, implementasi merujuk pada prosedur yang dilakukan untuk menguji hipotesis di bawah kondisi yang dikendalikan.

Selain itu, studi kasus oleh Pratiwi dan Susanto (2023) mengenai implementasi metode pembelajaran baru menunjukkan bahwa keberhasilan implementasi sangat bergantung pada kesiapan konteks dan ketersediaan sumber daya. Hal ini menggarisbawahi bahwa implementasi bukan sekadar proses teknis, tetapi juga proses manajemen perubahan dan adaptasi.

Tahapan Kunci dalam Implementasi

Implementasi hasil artikel atau jurnal biasanya mengikuti beberapa tahapan sistematis untuk memastikan bahwa solusi yang diusulkan diterapkan secara efektif dan hasilnya dapat dievaluasi secara objektif:

1. Perencanaan dan Adaptasi (*Planning and Adaptation*)

Tahap awal melibatkan peninjauan kembali temuan penelitian dan perumusan strategi implementasi. Strategi ini harus memperhitungkan kebutuhan adaptasi model atau metode agar sesuai dengan kondisi lapangan yang spesifik.

2. Eksekusi Lapangan (*Field Execution*)

Ini adalah fase di mana kegiatan yang direncanakan dilakukan. Sebagai contoh, jika artikel menawarkan *model prediktif* baru, implementasi melibatkan integrasi model tersebut ke dalam sistem produksi atau sistem pengambilan keputusan yang sudah ada. Wiersma dan Jurs (2018) menekankan

pentingnya dokumentasi selama fase eksekusi untuk melacak variabel dan memastikan proses berjalan sesuai protokol.

3. Evaluasi dan Pengujian (*Evaluation and Testing*)

Setelah diterapkan, kinerja implementasi diukur. Ini dapat berupa pengujian hipotesis, evaluasi kinerja sistem, atau studi keberlanjutan. Data yang terkumpul pada fase ini digunakan untuk memvalidasi temuan penelitian dan memberikan masukan untuk perbaikan lebih lanjut (Wiersma & Jurs, 2018).

2.2.2 Kmeans

K-Means clustering adalah algoritma clustering yang didasarkan pada unsupervised learning (pembelajaran tanpa pengawasan) di mana data dan objek dipisahkan menjadi klaster-klaster yang berbeda sedemikian rupa sehingga objek-objek yang memiliki properti serupa ditempatkan dalam satu klaster dan objek-objek yang memiliki properti berbeda ditempatkan dalam klaster yang terpisah. (Mulyadien & Enri, 2024)

Algoritma K-Means seringkali diklasifikasikan sebagai teknik *unsupervised learning* yang digunakan untuk mengelompokkan objek-objek data yang memiliki properti serupa ke dalam satu klaster, tanpa memerlukan label kelas awal. algoritma K-Means Clustering dipergunakan untuk membagi data menjadi beberapa kelompok (*cluster*) dengan sistem partisi dan menerima masukkan berupa data tanpa label kelas.

Kelebihan K-Means

Secara garis besar, keunggulan algoritma K-Means dalam proses *clustering* dapat dikelompokkan menjadi tiga aspek utama, yaitu efisiensi komputasi, kemudahan implementasi, dan kemampuan evaluasi hasil pengelompokan.

a) Efisiensi Komputasi

K-Means dikenal sebagai algoritma *clustering* yang efisien secara komputasi. Dengan prinsip pembaruan centroid secara iteratif, algoritma ini dapat menemukan pola atau kelompok dalam data berukuran besar dengan waktu proses relatif cepat. Dalam konteks penelitian, K-Means menunjukkan performa yang unggul dalam mengelompokkan data ekonomi nelayan karena mampu menghasilkan *cluster* dengan tingkat kesalahan (SSE) rendah dalam waktu singkat (Refialy et al., 2023).

Selain itu, penelitian lain menegaskan bahwa K-Means memiliki kemampuan konvergensi yang cepat dibandingkan dengan algoritma berbasis *fuzzy clustering*, sehingga cocok digunakan untuk dataset berskala besar (Apriyanto, 2024).

b) Kesederhanaan Implementasi

Salah satu keunggulan utama K-Means adalah kemudahannya dalam implementasi. Struktur algoritma yang sederhana—melibatkan proses penentuan centroid awal, perhitungan jarak Euclidean, dan pembaruan anggota cluster—membuatnya mudah diterapkan pada berbagai domain, mulai dari ekonomi, pendidikan, hingga kesehatan.

Penelitian oleh Syelvia dan Widiartha (2024) menunjukkan bahwa algoritma K-Means dapat dengan mudah diadaptasi untuk menganalisis tingkat stunting antarprovinsi di Indonesia tanpa memerlukan penyesuaian parameter yang

kompleks. Hal ini menjadikan K-Means pilihan populer dalam penelitian berbasis *data-driven decision making*.

c) Kemampuan Evaluasi dan Adaptasi

Selain efisien dan sederhana, K-Means juga unggul dalam hal kemampuan evaluasi hasil *clustering*. Melalui penggunaan metrik seperti Silhouette Coefficient dan *Davies-Bouldin Index* (DBI), kualitas pembentukan cluster dapat diukur secara kuantitatif. Studi yang dilakukan oleh Nanda et al. (2022) membuktikan bahwa nilai DBI pada hasil K-Means lebih rendah dibandingkan K-Medoids, yang berarti kualitas pengelompokannya lebih baik.

Lebih lanjut, algoritma ini juga memungkinkan analisis adaptif menggunakan metode *Elbow* untuk menentukan jumlah cluster optimal sesuai karakteristik data (Syelia & Widiartha, 2024). Dengan demikian, K-Means tidak hanya efisien, tetapi juga fleksibel dan mudah dievaluasi.

2.2.3 Website

Secara teknis, sebuah *website* didefinisikan sebagai koleksi *file* atau halaman web yang disimpan dalam sebuah *server* dan dapat diakses di internet menggunakan nama domain tunggal (Azwar et al., 2022). Halaman-halaman ini umumnya disusun menggunakan bahasa markah seperti HTML untuk struktur konten, dan diperindah secara visual menggunakan CSS. Namun, menurut visi penciptanya, konsep dasar *website* adalah untuk menciptakan "ruang informasi universal yang terhubung" secara global, memungkinkan berbagi pengetahuan melalui sistem *hypertext*.

Dalam perkembangannya saat ini, konsep *website* telah bertransformasi dari sekadar dokumen statis menjadi sebuah *platform* yang interaktif. Oleh karena itu,

website yang dianggap berhasil dan efektif wajib memiliki tingkat kegunaan (*usability*) yang tinggi, yang berarti harus mudah dipahami dan dioperasikan tanpa memerlukan panduan yang rumit.

Perkembangan historis *website* dapat diuraikan melalui beberapa fase transformasi yang telah mengubah cara interaksi dan perdagangan global:

- a) Fase 1: Kelahiran dan Era Web 1.0 (1991 - Awal 2000-an)

World Wide Web (WWW) diciptakan oleh Tim Berners-Lee di CERN dengan tujuan utama memfasilitasi pertukaran informasi yang efisien di kalangan ilmuwan. Peluncuran *website* pertama pada tahun 1991 berfungsi sebagai direktori informasi. Masa ini dikenal sebagai Web 1.0 atau "web yang hanya bisa dibaca" (*read-only web*). Karakteristik dominannya adalah konten yang sifatnya statis, hanya diproduksi oleh sedikit penyedia, dan disampaikan kepada audiens yang cenderung pasif. Interaksi pengguna sangat terbatas, paling banyak hanya sebatas formulir kontak atau buku tamu

- b) Fase 2: Revolusi Interaktif Web 2.0 (Sekitar 2004 - 2010)

Istilah Web 2.0 menandai pergeseran fundamental dari *website* sebagai media penyiaran menjadi sebuah *platform* kolaboratif dan partisipatif. Fokus utamanya beralih ke konten yang dihasilkan oleh pengguna (*User-generated content*), kecerdasan kolektif (*collective intelligence*), dan interaktivitas. Manifestasi dari era ini meliputi munculnya blog, wiki (seperti Wikipedia), serta layanan media sosial (seperti YouTube dan Facebook). *Website* tidak hanya lagi menyajikan data, tetapi menjadi sarana bagi pengguna untuk berkreasi, berbagi, dan berkolaborasi.

c) Fase 3: Era Mobile dan Aplikasi Web (2010 - Sekarang).

Adanya peningkatan signifikan dalam penggunaan perangkat bergerak, khususnya *smartphone*, mendorong evolusi lebih lanjut. Standar industri menetapkan prinsip Desain Web Responsif karena *website* harus mampu diakses dan berfungsi optimal di berbagai ukuran layar. Pada fase ini, *website* semakin menyerupai Aplikasi Web (*Web Apps*) yang canggih, menawarkan fungsionalitas yang setara dengan aplikasi desktop. Bagi sektor bisnis, kehadiran di *platform mobile* menjadi faktor krusial untuk menjangkau pelanggan dan menjalankan model *e-commerce* secara efisien.

2.2.4 HTML



Gambar 2.1 Logo HTML

HTML (*HyperText Markup Language*) adalah bahasa markah standar yang menjadi fondasi dari hampir semua konten di World wide Web. Berbeda dengan bahasa pemrograman yang berisi logika dan instruksi, HTML adalah bahasa deskriptif yang menggunakan "tag" untuk memberikan struktur dan makna semantik pada sebuah dokumen digital agar dapat diinterpretasikan dan ditampilkan dengan benar oleh browser web (Manokaran, 2023)

Peran HTML dalam Ekosistem Web *Modern*

HTML tidak bekerja sendirian. Dalam pengembangan web modern, HTML merupakan satu dari tiga pilar teknologi inti yang masing-masing memiliki peran yang jelas, mengikuti prinsip *separation of concerns* (pemisahan urusan) (Felke-Morris, 2021).

a) HTML (Struktur)

Bertanggung jawab atas struktur dan konten dasar dari halaman. Ia diibaratkan sebagai kerangka dari sebuah bangunan atau tubuh manusia.

b) CSS (*Cascading Style Sheets*) (Presentasi)

Mengontrol semua aspek visual, termasuk tata letak, warna, tipografi, dan animasi. CSS diibaratkan sebagai tampilan visual, cat, dan dekorasi bangunan tersebut.

c) JavaScript (Perilaku)

Menambahkan interaktivitas dan fungsionalitas dinamis pada halaman, seperti validasi formulir, konten yang dapat diperbarui tanpa memuat ulang halaman, dan animasi yang kompleks. JavaScript adalah sistem kelistrikan, perpipaan, dan mekanik yang membuat bangunan menjadi fungsional.

2.2.5 CSS



Gambar 2.2 Logo CSS

CSS, singkatan dari *Cascading Style Sheets*, adalah bahasa *stylesheet* yang digunakan untuk mengontrol dan mendeskripsikan presentasi visual dari sebuah dokumen yang ditulis dalam bahasa markah seperti HTML. Jika HTML berfungsi sebagai kerangka (struktur) dari sebuah halaman web, maka CSS adalah segala sesuatu yang berkaitan dengan estetika mulai dari tata letak, warna, tipografi, hingga animasi (Bonsangue, 2023)

Tujuan utama diciptakannya CSS adalah untuk mewujudkan prinsip pemisahan urusan (*separation of concerns*). Sebelum adanya CSS, banyak properti gaya (seperti warna font dan perataan) ditulis langsung di dalam tag HTML. Praktik ini membuat kode menjadi berantakan, sulit dikelola, dan tidak efisien. Dengan CSS, desainer dapat memisahkan instruksi gaya dari struktur konten, sehingga kode menjadi lebih bersih, lebih mudah dipelihara, dan lebih fleksibel (Felke-Morris, 2021).

Bagaimana CSS Bekerja: Konsep Inti

CSS bekerja dengan cara menerapkan serangkaian "aturan" (*rules*) pada elemen-elemen HTML. Setiap aturan terdiri dari dua bagian utama:

1. Selektor (Selector): Bagian ini menargetkan elemen HTML mana yang ingin diberi gaya. Selektor bisa berupa nama tag (misalnya, h1, p), kelas (misalnya, .nama-kelas), ID (misalnya, #id-unik), atau kombinasi yang lebih kompleks.
2. Blok Deklarasi (Declaration Block): Bagian ini berada di dalam kurung kurawal {} dan berisi satu atau lebih deklarasi gaya. Setiap deklarasi terdiri dari properti dan nilai yang dipisahkan oleh titik dua (misalnya, color: blue;).

2.2.6 Javascript



Gambar 2.3 Logo JavaScript

JavaScript (JS) adalah bahasa pemrograman tingkat tinggi, dinamis, dan *just-in-time compiled* yang menjadi teknologi inti ketiga dari *World Wide Web*. Jika HTML menyediakan struktur dan CSS mengatur presentasi, maka JavaScript menambahkan perilaku (*behavior*) dan interaktivitas pada halaman web (Fukuchi, 2022). Ia adalah bahasa yang memungkinkan halaman web untuk merespons aksi pengguna, memanipulasi konten secara dinamis, dan berkomunikasi dengan server tanpa perlu memuat ulang halaman.

Berbeda secara fundamental dengan HTML dan CSS, JavaScript adalah bahasa pemrograman sejati yang dapat menjalankan logika, memproses data, dan mengontrol perangkat. Menggunakan analogi yang sama, jika HTML adalah kerangka dan CSS adalah penampilan fisiknya, maka JavaScript adalah sistem saraf dan otot yang memungkinkan tubuh tersebut untuk bergerak, bereaksi, dan berinteraksi dengan lingkungannya (Felke-Morris, 2021).

Bagaimana JavaScript Bekerja: Konsep Inti

JavaScript, pada dasarnya, dieksekusi di sisi klien (*client-side*), yaitu di dalam browser web pengguna. Kemampuannya yang paling fundamental di dalam browser adalah melalui interaksi dengan DOM (Document Object Model).

a) Manipulasi DOM

Ketika browser memuat sebuah halaman web, ia menciptakan sebuah representasi struktur dari dokumen HTML yang disebut DOM. JavaScript memiliki kemampuan penuh untuk mengakses dan memanipulasi DOM ini. Ia dapat menemukan elemen HTML, mengubah kontennya, memodifikasi gayanya (atribut CSS), serta menambah atau menghapus elemen dari halaman secara langsung. Inilah mekanisme di balik pembaruan konten secara *real-time* (Kadyrkulova & Kyrgyzbaev, 2023)

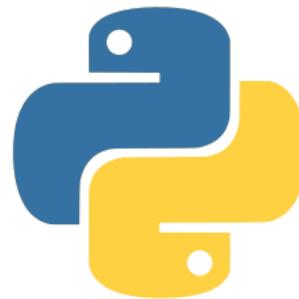
b) Penanganan Peristiwa (Event Handling)

JavaScript dapat "mendengarkan" dan bereaksi terhadap berbagai peristiwa (*event*) yang dipicu oleh pengguna atau *browser*. Peristiwa ini bisa berupa klik mouse (*onclick*), penekanan tombol keyboard (*onkeydown*), pengiriman formulir (*onsubmit*), atau bahkan saat halaman selesai dimuat (*.onload*). Dengan menangani peristiwa ini, developer dapat menjalankan fungsi tertentu sebagai respons, menciptakan pengalaman pengguna yang interaktif (Felke-Morris, 2021).

c) Operasi Asinkron (*Asynchronous*)

Salah satu kekuatan terbesar JavaScript modern adalah kemampuannya untuk melakukan operasi secara asinkron. Menggunakan teknologi seperti AJAX (*Asynchronous JavaScript and XML*) atau *Fetch API* yang lebih modern, JavaScript dapat meminta atau mengirim data ke server di latar belakang tanpa mengganggu atau menghentikan interaksi pengguna dengan halaman.

2.2.7 Python



Gambar 2.4 Logo Python

Python adalah bahasa pemrograman tingkat tinggi (*high-level*), serbaguna (*general-purpose*), dan diinterpretasikan (*interpreted*) yang dikenal luas karena sintaksisnya yang bersih, mudah dibaca, dan elegan. Berbeda dengan bahasa lain yang seringkali memerlukan sintaksis yang rumit, Python dirancang untuk memungkinkan pengembang menulis kode yang ekspresif dengan jumlah baris yang lebih sedikit (Schwarz et al., 2020)

Filosofi inti di balik Python terangkum dalam "*The Zen of Python*", yang menekankan pada keterbacaan (*readability*), kesederhanaan, dan kejelasan. Prinsip utamanya adalah "eksplisit lebih baik daripada implisit" dan "sederhana lebih baik daripada kompleks," yang menjadikan Python tidak hanya kuat tetapi juga sangat mudah dipelajari, bahkan bagi pemula (Ramalho, 2022).

Beberapa karakteristik teknis utama Python meliputi:

- a) Diinterpretasikan: Kode Python dieksekusi baris per baris oleh sebuah interpreter, yang menyederhanakan proses *debugging* dan membuat siklus pengembangan menjadi lebih cepat.

- b) Pengetikan Dinamis (*Dynamic Typing*): Tipe data sebuah variabel ditentukan pada saat program berjalan (*runtime*), bukan pada saat kompilasi, memberikan fleksibilitas yang tinggi.
- c) Manajemen Memori Otomatis: Python secara otomatis menangani alokasi dan dealokasi memori melalui mekanisme *garbage collection*.

Kelebihan dan Penggunaan Populer

Popularitas Python yang meroket didorong oleh fleksibilitasnya yang luar biasa dan ekosistem pustaka (*libraries*) yang sangat matang.

1. Kemudahan Belajar dan Keterbacaan: Sintaksisnya yang bersih dan mirip dengan bahasa Inggris biasa menjadikannya salah satu bahasa terbaik untuk dipelajari oleh pemula. Keterbacaan ini juga membuat pemeliharaan kode dalam proyek-proyek besar menjadi lebih mudah (Schwarz et al., 2020).
2. Ekosistem Pustaka yang Luas ("Batteries Included"): Python menganut filosofi "sudah termasuk baterai", yang berarti pustaka standarnya (*standard library*) sangat kaya dan menyediakan modul untuk berbagai tugas umum. Di luar itu, terdapat Python Package Index (PyPI), sebuah repositori masif berisi lebih dari ratusan ribu paket pihak ketiga yang dapat diinstal dengan mudah untuk memperluas fungsionalitas Python (Ramalho, 2022).
3. Dominasi dalam Sains Data dan Kecerdasan Buatan (AI): Ini adalah bidang di mana Python bersinar paling terang saat ini. Keberadaan pustaka-pustaka seperti:
 - a) NumPy: Untuk komputasi numerik yang efisien.
 - b) Pandas: Untuk manipulasi dan analisis data terstruktur (diciptakan oleh Wes McKinney).

- c) Matplotlib & Seaborn: Untuk visualisasi data.
 - d) Scikit-learn, TensorFlow, & PyTorch: Untuk pembelajaran mesin (*machine learning*) dan *deep learning*. Pustaka-pustaka ini telah menjadikan Python sebagai bahasa *de facto* bagi para ilmuwan data, analis, dan peneliti AI (McKinney, 2022).
4. Pengembangan Web Sisi Server (*Back-end*): Melalui *framework* yang kuat seperti Django (untuk pengembangan cepat dan skala besar) dan Flask (untuk aplikasi yang lebih ringan dan fleksibel), Python menjadi pilihan populer untuk membangun bagian logis dari aplikasi web.
 5. Otomatisasi dan Skrip: Salah satu kegunaan asli Python adalah sebagai bahasa skrip untuk mengotomatiskan tugas-tugas repetitif, mengelola sistem, dan memproses file, menjadikannya alat yang sangat berharga bagi administrator sistem dan insinyur DevOps.

2.2.8 Flask



Gambar 2. 5 Logo Flask

Flask adalah sebuah *microframework* untuk pengembangan aplikasi web yang ditulis dalam bahasa Python. Disebut sebagai "micro" bukan karena fungsionalitasnya yang terbatas, melainkan karena filosofi desainnya yang

minimalis dan ekstensibel. Inti dari Flask dirancang agar tetap sederhana, menyediakan komponen-komponen esensial untuk membangun aplikasi web, namun membiarkan keputusan-keputusan penting lainnya seperti pilihan database atau sistem autentikasi sepenuhnya di tangan pengembang (“Python for Web Development,” 2022).

Filosofi ini sangat selaras dengan “*The Zen of Python*”, yang mengutamakan kesederhanaan dan kejelasan. Flask tidak memaksakan struktur proyek atau dependensi tertentu, memberikan kebebasan penuh kepada pengembang untuk memilih alat dan pustaka yang paling sesuai dengan kebutuhan spesifik mereka. Pendekatan ini sering disebut sebagai *unopinionated*, berbeda dengan *framework* yang lebih besar dan *opinionated* (Ramalho, 2022).

Meskipun intinya kecil, Flask dibangun di atas dua pustaka eksternal yang sangat kuat dan matang, yang merupakan fondasi dari fungsionalitasnya:

1. Werkzeug (Pustaka WSGI): Werkzeug adalah sebuah pustaka utilitas WSGI (*Web Server Gateway Interface*), yang merupakan spesifikasi standar untuk komunikasi antara server web dan aplikasi Python. Werkzeug menangani tugas-tugas tingkat rendah yang kompleks seperti *request routing* (mengarahkan URL ke fungsi Python yang tepat), objek *request* dan *response*, serta sesi (*session*) dan *cookie*. Pada dasarnya, Werkzeug adalah mesin yang menggerakkan aplikasi Flask (“Python for Web Development,” 2022).
2. Jinja2 (Mesin Templat): Jinja2 adalah sebuah mesin templat (*template engine*) yang kaya fitur dan modern. Fungsinya adalah untuk memisahkan logika aplikasi dari presentasi (HTML). Jinja2 memungkinkan pengembang

untuk membuat file templat HTML yang dapat disisipi dengan variabel, ekspresi, dan struktur logika (seperti *loops* dan *conditionals*). Saat sebuah *request* diproses, Flask menggunakan Jinja2 untuk me-*render* templat ini, mengisi data dinamis ke dalamnya, dan menghasilkan halaman HTML akhir yang dikirimkan ke browser pengguna (Ronacher, 2024).

Kelebihan dan Kasus Penggunaan Umum

1. Fleksibilitas Maksimal: Pengembang memiliki kontrol penuh atas komponen yang mereka gunakan, memungkinkan integrasi dengan berbagai macam teknologi dan pustaka.
2. Ringan dan Cepat: Karena intinya yang kecil, aplikasi Flask yang sederhana memiliki *overhead* yang minimal dan dapat berkinerja sangat baik.
3. Sangat Ekstensibel: Terdapat ekosistem ekstensi yang sangat besar (seperti Flask-SQLAlchemy untuk interaksi database, Flask-Login untuk manajemen sesi pengguna) yang memungkinkan fungsionalitas ditambahkan secara modular sesuai kebutuhan.
4. Mudah untuk Memulai: Kurva belajar untuk membuat aplikasi sederhana ("Hello, World!") di Flask sangat landai, menjadikannya pilihan yang bagus untuk pemula yang ingin memahami dasar-dasar pengembangan web.

2.2.9 *Database (Basis Data)*

Basis data atau *database* adalah sekumpulan data yang saling berhubungan secara logis, yang dikelola oleh sebuah sistem untuk memastikan efisiensi dan

kemudahan dalam penyimpanan dan pengambilan informasi. Secara formal, ini adalah koleksi data terstruktur yang merepresentasikan informasi tentang suatu domain tertentu, misalnya data perbankan, universitas, atau perusahaan (Silberschatz et al., 2020).

Kunci untuk memahami basis data adalah dengan membedakannya dari Sistem Manajemen Basis Data atau DBMS (Database Management System). DBMS adalah sebuah sistem perangkat lunak yang memungkinkan pengguna untuk membuat, memelihara, dan mengontrol akses terhadap basis data. Fungsi utama DBMS mencakup:

- a) Manajemen Penyimpanan: Mengelola penyimpanan data secara efisien pada media fisik.
- b) Bahasa Kueri (*Query Language*): Menyediakan bahasa, seperti SQL, untuk mengambil dan memanipulasi data.
- c) Optimalisasi Kueri: Memastikan bahwa permintaan data dieksekusi dengan cara yang paling efisien.
- d) Manajemen Transaksi: Menjamin bahwa operasi pada data (transaksi) memenuhi properti ACID (Atomicity, Consistency, Isolation, Durability) untuk menjaga integritas data (Silberschatz et al., 2020).

2.2.10 MySQL



Gambar 2.6 Logo MySQL

MySQL adalah sebuah Sistem Manajemen Basis Data Relasional atau RDBMS (Relational Database Management System) yang bersifat *open-source*. Sebagai sebuah RDBMS, fungsi utamanya adalah untuk menyimpan, mengelola, dan mengambil data yang terorganisir dalam format tabel (baris dan kolom), sesuai dengan prinsip-prinsip model relasional. Interaksi dengan data di dalam MySQL dilakukan menggunakan SQL (Structured Query Language), yang merupakan bahasa standar untuk basis data relasional (Silberschatz et al., 2020).

MySQL beroperasi menggunakan arsitektur client-server. Ini berarti terdapat sebuah program server utama (MySQL Server) yang bertanggung jawab penuh atas manajemen data, termasuk penyimpanan, keamanan, dan pemrosesan kueri. Program-program lain, yang disebut klien (seperti aplikasi web, perangkat *command-line*, atau alat bantu grafis), terhubung ke server ini untuk mengirimkan permintaan (kueri) dan menerima hasilnya (Grippa & Kuzmichev, 2021).

Sejarah Singkat dan Kepemilikan

MySQL memiliki sejarah yang kaya dan penting dalam dunia *open-source*:

- a) 1995: Versi pertama MySQL dirilis oleh perusahaan Swedia, MySQL AB, yang didirikan oleh David Axmark, Allan Larsson, dan Michael "Monty" Widenius.
- b) 2008: MySQL AB diakuisisi oleh Sun Microsystems, sebuah langkah besar yang membawa MySQL ke dalam ekosistem korporat yang lebih luas.
- c) 2010: Oracle Corporation mengakuisisi Sun Microsystems, sehingga MySQL pun menjadi milik Oracle. Akuisisi ini menimbulkan kekhawatiran di kalangan komunitas *open-source* mengenai masa depan lisensi dan pengembangan MySQL.
- d) MariaDB: Sebagai respons atas kekhawatiran tersebut, pendiri asli MySQL, Monty Widenius, melakukan *fork* (membuat cabang pengembangan baru) dari basis kode asli MySQL dan menciptakan MariaDB. MariaDB bertujuan untuk tetap sepenuhnya gratis dan *open-source* di bawah lisensi GPL, sambil mempertahankan kompatibilitas tinggi dengan MySQL (Grippa & Kuzmichev, 2021).

2.3 Teori Perancangan Basis Data

2.3.1 ERD (*Entity Relationship Diagram*)

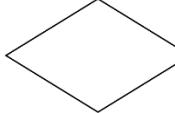
ERD (Entity-Relationship Diagram) adalah sebuah model data konseptual tingkat tinggi yang digunakan untuk merepresentasikan struktur logis dari sebuah basis data secara visual. ERD berfungsi sebagai blueprint atau cetak biru yang menggambarkan entitas-entitas utama dalam suatu sistem, atribut-atribut yang dimiliki oleh setiap entitas, dan bagaimana entitas-entitas tersebut saling berhubungan satu sama lain (Silberschatz et al., 2020).

Tujuan utama dari pembuatan ERD adalah untuk memodelkan dan merancang basis data pada tahap awal siklus pengembangan perangkat lunak. Model ini berfungsi sebagai alat komunikasi yang sangat efektif antara perancang basis data, pengembang aplikasi, dan pemangku kepentingan non-teknis (seperti manajer atau klien) untuk memastikan bahwa semua pihak memiliki pemahaman yang sama mengenai data yang akan dikelola dan aturan bisnis yang berlaku (Hoffer et al., 2020).

2.3.2 Komponen Pada ERD (*Entity Relationship Diagram*)

ERD dibangun dari tiga komponen visual dasar yang disajikan dalam tabel berikut:

Tabel 2.2 Komponen Pada ERD

Simbol	Nama	Keterangan	Fungsi
	Entitas	Objek, orang, tempat, atau konsep di dunia nyata yang datanya perlu disimpan dan dapat diidentifikasi secara unik (Coronel & Morris, 2020).	Berfungsi untuk mewakili objek atau konsep utama yang datanya ingin disimpan
	Atribut	Oval (notasi klasik) atau daftar di dalam kotak entitas (notasi modern)	Berfungsi untuk mendeskripsikan atau memberikan detail informasi tentang sebuah entitas
	Hubungan	Bentuk wajik (notasi klasik) atau garis penghubung dengan simbol di ujungnya (misal: notasi <i>Crow's Foot</i>).	Berfungsi untuk menghubungkan dua entitas atau lebih dan menunjukkan bagaimana mereka berinteraksi atau berhubungan.

2.4 Definisi *Unified Modeling Language* (UML)

2.4.1 Jenis-Jenis Diagram *Unified Modeling Language*

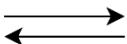
Uml memiliki jenis-jenis diagram yang memiliki banyak model diagram adapun beberapa diagram uml yang dipakai pada penelitian ini :

1. *Activity Diagram*

Activity Diagram adalah menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, decision yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity Diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi (Hasanah & Untari, 2020).

Tabel 2.3 Simbol Elemen *Activity Diagram*

Simbol	Nama	Keterangan
	<i>Initial Node</i>	Menunjukkan titik awal dari suatu alur aktivitas dalam diagram. Ini adalah tempat di mana proses dimulai.
	<i>Activity</i>	Menyatakan suatu aksi atau proses yang dilakukan dalam sistem
	<i>Decision/Per cabangan</i>	Menandakan titik percabangan alur berdasarkan kondisi atau keputusan tertentu. Setiap jalur keluar diberi label kondisi (misal: "Ya" / "Tidak").
	<i>Penggabungan/Join</i>	Menggabungkan beberapa alur menjadi satu jalur untuk dilanjutkan ke proses berikutnya

	<i>Line Connector</i>	Menunjukkan aliran kontrol dari satu aktivitas ke aktivitas berikutnya. Menghubungkan semua simbol dalam diagram.
	<i>Final Node</i>	Menandakan akhir dari seluruh proses dalam <i>Activity Diagram</i> .
	<i>Swimlane</i>	Digunakan untuk mengelompokkan aktivitas berdasarkan aktor atau bagian sistem yang menjalankannya.

2. *Use case Diagram*

Use case diagram adalah teknik untuk merekam persyaratan fungsional sebuah sistem, menggambarkan fungsionalitas yang diharapkan dari sebuah sistem.

Use case diagram menekankan kepada “apa” yang diperbuat oleh sistem, dan bukan “bagaimana”. Sebuah *Use case* merepresentasikan sebuah interaksi antara aktor dengan sistem (Hasanah & Untari, 2020). *Use case* adalah representasi dari suatu aktivitas spesifik yang dilakukan dalam sistem, seperti masuk ke sistem (*Login*), membuat daftar belanja, dan lainnya. Aktor sendiri merupakan entitas baik manusia maupun mesin yang berinteraksi dengan sistem untuk menjalankan aktivitas-aktivitas tersebut.

Menurut (Hasanah dan Untari, 2020), *Use case diagram* sangat berguna dalam menyusun kebutuhan-kebutuhan (*requirement analysis*) suatu sistem. Diagram ini digunakan sebagai alat komunikasi antara pengembang dan klien untuk

menggambarkan rancangan sistem, serta berfungsi sebagai dasar dalam merancang *test case* untuk setiap fitur yang terdapat dalam sistem.

Tabel 2.4 Simbol Elemen *Use case Diagram*

Simbol	Nama	Keterangan
	<i>Actor</i>	Mewakili pengguna atau sistem eksternal yang berinteraksi dengan sistem.
	<i>Use case</i>	Menunjukkan fungsi atau layanan yang disediakan oleh sistem.
	<i>Association</i>	Menggambarkan hubungan interaksi antara aktor dan <i>Use case</i> .
	<i>Generalisasi</i>	Menyatakan hubungan hierarki antar aktor atau antar <i>Use case</i>
	<i>Include</i>	Menunjukkan bahwa suatu <i>Use case</i> selalu menyertakan perilaku dari <i>Use case</i> lain.
	<i>Extend</i>	Menandakan bahwa suatu <i>Use case</i> tambahan hanya dijalankan dalam kondisi tertentu dari <i>Use case</i> utama.

3. *Class Diagram*

Class Diagram merupakan salah satu jenis diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk menggambarkan struktur statis dari sebuah sistem. Diagram ini memodelkan kelas-kelas dalam sistem, lengkap

dengan atribut (data) dan metode (perilaku) yang dimiliki oleh masing-masing kelas. Selain itu, *Class Diagram* juga menunjukkan bagaimana hubungan antar kelas terjalin, baik melalui asosiasi, generalisasi (*inheritance*), agregasi, maupun komposisi. (Arif et al., 2020)

Class Diagram sangat berguna dalam tahap perancangan sistem karena memberikan gambaran menyeluruh tentang bagaimana objek dalam sistem berinteraksi satu sama lain. Ini membantu pengembang memahami struktur internal sistem dan bagaimana bagian-bagian tersebut saling berelasi, sehingga sistem dapat dibangun secara efisien dan sesuai kebutuhan fungsional.

Tabel 2.5 Simbol Elemen *Class Diagram*

Simbol	Nama	Keterangan			
—	<i>Asociation</i>	Menunjukkan hubungan struktural antara dua kelas.			
<table border="1"> <tr><td>Classname</td></tr> <tr><td>+ field: type</td></tr> <tr><td>+ method(type): type</td></tr> </table>	Classname	+ field: type	+ method(type): type	<i>Class</i>	Elemen dasar dalam UML yang merepresentasikan entitas dengan atribut dan operasi (metode) tertentu.
Classname					
+ field: type					
+ method(type): type					
○	<i>Interface</i>	Kelas yang mengimplementasikan antarmuka harus menyediakan implementasi untuk semua metode yang ditentukan dalam antarmuka tersebut.			
—→	<i>Directed Asociation</i>	Bentuk khusus dari <i>association</i> yang menunjukkan hubungan satu arah antara dua kelas.			
→	<i>Generalisasi</i>	Hubungan yang menunjukkan pewarisan atau hubungan <i>is-a</i> (adalah).			

----->	<i>Dependency</i>	Menunjukkan hubungan antar elemen di mana perubahan pada satu elemen dapat mempengaruhi elemen lain, tetapi tidak sebaliknya.
—◇—	<i>Aggregation</i>	Menunjukkan hubungan "bagian-keseluruhan" di mana satu kelas merupakan bagian dari kelas lain, tetapi bagian tersebut dapat berdiri sendiri meskipun keseluruhan dihapus.

4. *Sequence Diagram*

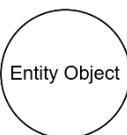
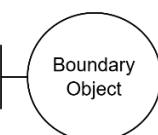
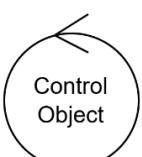
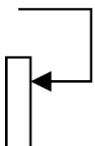
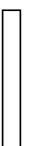
Sequence Diagram adalah salah satu jenis diagram dalam *Unified Modeling Language (UML)* yang digunakan untuk memodelkan interaksi antar objek dalam sistem berdasarkan urutan waktu. Diagram ini menunjukkan bagaimana objek saling berkomunikasi melalui pesan dalam skenario tertentu.

Sequence Diagram berfungsi untuk:

- A. Memodelkan alur interaksi antar objek dalam suatu proses.
- B. Menunjukkan urutan pesan yang dikirim antar objek.
- C. Menggambarkan tahapan aktivitas dalam sebuah skenario sistem.

Tabel 2.6 Simbol Elemen *Sequence Diagram*

Simbol	Nama	Keterangan
 Actor	<i>Actor</i> (Aktor)	Aktor adalah entitas eksternal yang berinteraksi dengan sistem,

	<i>Entity Object</i>	<i>Entity object</i> adalah objek yang mewakili data atau entitas dalam sistem yang menyimpan informasi penting.
	<i>Boundary Object</i>	<i>Boundary object</i> adalah objek yang berfungsi sebagai interface antara sistem dengan dunia luar.
	<i>Control Object</i>	<i>Control object</i> adalah objek yang mengontrol alur eksekusi dalam sistem, termasuk keputusan dan interaksi antar objek.
	<i>Recursive</i>	<i>Recursive</i> dalam konteks <i>Sequence Diagram</i> mengacu pada pemanggilan kembali atau looping suatu metode atau proses.
	<i>Activation</i>	<i>Activation</i> adalah periode waktu ketika suatu objek melakukan aksi atau aktivitas.
	<i>Life Line</i>	<i>life line</i> adalah garis vertikal yang menggambarkan eksistensi suatu objek dalam urutan waktu.

2.5 Aplikasi Pendukung

2.5.1 Draw.io

Draw.io (juga dikenal sebagai diagrams.net) adalah aplikasi berbasis web yang digunakan untuk membuat berbagai jenis diagram, termasuk diagram UML, ERD (*Entity Relationship Diagram*), *Flowchart*, dan lainnya. Aplikasi ini

sangat populer di kalangan pengembang sistem dan analis karena bersifat gratis, ringan, dan mudah diakses tanpa perlu instalasi.

1. Fungsi Utama Draw.io dalam Perancangan Sistem

- A. Membuat Diagram UML: Mendukung pembuatan diagram seperti *Use case*, *class*, *sequence*, dan *Activity Diagram* secara visual.
- B. Merancang ERD: Memungkinkan pengguna membuat relasi antar entitas secara cepat dan terstruktur.
- C. Kolaborasi *Online*: Terintegrasi dengan Google Drive dan OneDrive, memungkinkan kerja kolaboratif dalam tim secara real-time.
- D. Ekspor dan Impor *File*: Mendukung berbagai format eksport seperti PNG, PDF, SVG, dan XML sehingga memudahkan dokumentasi dan presentasi desain sistem.

2. Kelebihan Draw.io

- A. Gratis dan *Open Source*: Tidak memerlukan biaya langganan.
- B. Antarmuka Intuitif: *Drag-and-drop* dengan banyak *template* dan ikon yang tersedia.
- C. Kompatibel di Banyak *Platform*: Dapat diakses melalui browser di berbagai sistem operasi (Windows, macOS, Linux).
- D. *Offline Mode*: Tersedia juga versi desktop yang dapat digunakan tanpa koneksi internet.

2.5.2 Visual Studio Code

Visual Studio Code (VS Code) adalah *Editor* kode sumber terbuka yang dikembangkan oleh Microsoft dan banyak digunakan dalam pengembangan

perangkat lunak. VS Code dirancang untuk mendukung berbagai bahasa pemrograman dan menyediakan berbagai fitur tambahan melalui ekstensi. Aplikasi ini bersifat ringan, dapat dijalankan di berbagai sistem operasi seperti Windows, macOS, dan Linux, serta memiliki antarmuka yang ramah pengguna.

Visual Studio Code menjadi salah satu alat favorit di kalangan pengembang karena kemampuannya dalam menangani proyek pengembangan skala kecil hingga besar. Dalam konteks pengembangan sistem konsultasi dan pemesanan jasa desain grafis berbasis web, VS Code digunakan sebagai lingkungan utama untuk menulis dan mengelola kode program, baik pada sisi frontend maupun backend.

2.5.3 XAMPP

XAMPP berfungsi sebagai lingkungan server lokal yang menyediakan komponen utama untuk menjalankan dan mengelola *Database MySQL*. XAMPP mencakup server Apache yang diperlukan untuk menjalankan aplikasi berbasis web, serta MySQL yang digunakan untuk menyimpan data penting, seperti informasi pengguna, data pemesanan, layanan desain grafis, dan transaksi yang dilakukan melalui sistem.

Dengan menggunakan XAMPP, tim pengembang dapat dengan mudah mengatur *Database MySQL* di lingkungan lokal sebelum melakukan implementasi di server produksi. XAMPP juga memungkinkan pengelolaan MySQL melalui phpMyAdmin, yang menyediakan antarmuka grafis untuk membuat, mengEdit, dan mengelola tabel-tabel *Database* tanpa perlu menulis perintah SQL secara manual. Hal ini sangat mempermudah proses pengembangan, pengujian, dan debugging data terkait dalam sistem yang dikembangkan.

2.6 Teori Pengujian Sistem

Perangkat lunak harus dijaga kualitasnya untuk memastikan kepuasan pelanggan. Dengan menjaga kualitas, perangkat lunak dapat bersaing dengan produk serupa di pasar, mempertahankan pelanggan setia, dan meningkatkan profitabilitas. Pengujian perangkat lunak diperlukan untuk mengurangi kemungkinan kesalahan teknis maupun non-teknis.

2.6.1 Pengujian *Black Box*

Pengujian *Black Box* adalah metode pengujian perangkat lunak yang berfokus pada pengujian fungsionalitas perangkat lunak tanpa melihat struktur internal atau kode sumber perangkat lunak tersebut. Dalam pengujian ini, pengujian hanya berfokus pada input dan output dari sistem, serta bagaimana sistem merespons terhadap berbagai kondisi yang diberikan. Pengujian *Black Box* bertujuan untuk memastikan bahwa perangkat lunak berfungsi sesuai dengan kebutuhan dan spesifikasi yang telah ditetapkan.

Beberapa tujuan utama dari pengujian *Black Box* adalah sebagai berikut:

1. Verifikasi fungsionalitas: Memastikan bahwa perangkat lunak bekerja sesuai dengan spesifikasi dan memenuhi harapan pengguna.
2. Pengujian antarmuka pengguna (UI): Mengidentifikasi apakah antarmuka pengguna berfungsi dengan baik, mudah digunakan, dan sesuai dengan desain yang telah disepakati.
3. Pengujian integrasi antar modul: Memeriksa apakah berbagai komponen atau modul perangkat lunak dapat berinteraksi dengan benar.

4. Menguji kinerja aplikasi: Menilai bagaimana perangkat lunak merespons dalam berbagai kondisi penggunaan dan beban.

Pengujian *Black Box* dilakukan dengan cara menguji perangkat lunak berdasarkan spesifikasi, tanpa memperhatikan struktur kode internalnya.

Ada beberapa jenis teknik dalam pengujian *Black Box*, antara lain:

1. Pengujian fungsional: Fokus pada pengujian apakah perangkat lunak berfungsi sesuai dengan kebutuhan yang telah ditetapkan.
2. Pengujian batasan: Memeriksa apakah perangkat lunak dapat menangani input di luar batas normal yang diizinkan.
3. Pengujian keamanan: Mengidentifikasi potensi celah keamanan dalam aplikasi.