

BAB IV PERANCANGAN

Bagian ini menjelaskan mengenai rencana penelitian yang akan dilakukan oleh peneliti. Dalam pengembangan sistem, diperlukan tahapan yang sistematis untuk memastikan sistem yang dikembangkan mampu beroperasi dengan baik dan optimal. Berikut ini adalah tahapan perancangan yang dilakukan oleh peneliti.

4.1 Deskripsi Sistem

Sistem yang dikembangkan adalah sistem rekomendasi fashion berbasis web yang memanfaatkan algoritma Naive Bayes untuk mengklasifikasikan preferensi pengguna ke dalam dua kategori utama: formal dan kasual. Sistem ini akan mengelola data pengguna, data produk fashion, dan hasil klasifikasi untuk menghasilkan rekomendasi yang relevan.

4.2 Analisis Sistem Terdahulu

Dalam proses pengembangan sistem rekomendasi pada website *E-commerce* menggunakan algoritma *Naive Bayes*, pemahaman terhadap sistem sebelumnya sangatlah penting. Analisis ini bertujuan untuk mengidentifikasi keunggulan dan kekurangan dari sistem terdahulu, serta menentukan bagian-bagian yang perlu ditingkatkan. Adapun beberapa aspek yang perlu dianalisis antara lain:

Tabel 4.1 Analisis Sistem Terdahulu

Pencapaian Ke-1	
Nama Penulis	Dananjoyo, Y., & Muflihah, Y. (2024)
Judul	Sistem Rekomendasi Ukuran Baju pada Aplikasi E-Commerce dengan Metode Naïve Bayes
Hasil	Penelitian ini memanfaatkan 320 catatan survei pengguna yang berisi lima atribut fisiologis—gender, tinggi, berat, lingkar dada, dan lingkar pinggang—untuk memprediksi

	ukuran pakaian (S, M, L). Algoritma yang digunakan adalah Multinomial Naïve Bayes tanpa pembandingan model lain. Uji coba fungsional terhadap tujuh belas skenario black-box menunjukkan seluruh alur sistem berjalan sesuai spesifikasi, sedangkan pengujian lapangan kepada sepuluh responden mencatat presisi rekomendasi ukuran sebesar 85 persen. Keterbatasan yang dilaporkan penulis terletak pada belum dimanfaatkannya atribut perilaku belanja serta belum tersedianya metrik akurasi maupun recall secara numerik.
Pencapaian Ke-2	
Nama Penulis	Strain, N., & Olszewska, J. I. (2020)
Judul	<i>Naïve Bayesian Network for Automated, Fashion Personal Stylist</i>
Hasil	Artikel konferensi ini membangun Naïve Bayesian Network untuk menyarankan kombinasi pakaian uniseks berdasarkan enam atribut umum—warna dominan, jenis item, musim, tingkat formalitas, rentang harga, dan tren. Dataset berisi 1 276 kombinasi outfit yang di anotasi pakar fesyen. Evaluasi lima-fold cross-validation melaporkan Macro-F1 sebesar 0,82 tanpa menggunakan algoritma pembandingan. Penulis menegaskan model ringan dan dapat dijalankan pada perangkat seluler; namun penelitian masih bersifat simulasi karena belum diuji menggunakan data transaksi pengguna nyata.
Pencapaian Ke-3	
Nama Penulis	Sutaryana, Y., & Kharisma, L. P. I. (2024)

Judul	Implementation of Naïve Bayes Algorithm for Children's Clothing Recommendation
Hasil	Studi ini mengumpulkan 450 transaksi pembelian pakaian anak pada sebuah toko daring dan menurunkan delapan atribut, yaitu harga, warna utama, bahan, ukuran, usia pemakai, stok, diskon, dan rating produk. Model Naïve Bayes tunggal dilatih pada 70 persen data dan diuji pada 30 persen sisanya, menghasilkan akurasi 88 persen, precision 0,86, serta recall 0,90. Kelemahan yang diungkap penulis adalah segmentasi khusus anak-anak sehingga generalisasi ke lini dewasa perlu penelitian lanjutan, serta belum dilaporkannya metrik galat probabilistik seperti MAE atau RMSE.

4.3 Spesifikasi Kebutuhan Sistem

Pengembangan *website E-commerce* dengan sistem rekomendasi berbasis algoritma *Naive Bayes* untuk rekomendasi fashion memerlukan perencanaan yang matang dan spesifikasi kebutuhan yang jelas. Berikut adalah spesifikasi kebutuhan sistem yang terperinci:

1. Kebutuhan Fungsional
 - a. Registrasi: Sistem harus mendukung pengguna untuk mendaftar menggunakan email, media sosial, atau metode lainnya agar dapat menggunakan fitur-fitur yang tersedia.
 - b. Profil Pengguna: Pengguna harus dapat mengelola profil mereka, termasuk informasi pribadi dan preferensi fashion.
2. Manajemen Produk:
 - a. Katalog Produk: Sistem harus menyediakan katalog produk fashion yang lengkap dan terstruktur, dengan kategori yang jelas.
 - b. Detail Produk: Setiap produk harus memiliki halaman detail yang mencakup deskripsi, gambar, harga, dan ukuran.

3. Sistem Rekomendasi:

- a. Pengumpulan Data: Sistem harus mampu mengumpulkan data perilaku pengguna, meliputi riwayat pencarian, klik, dan pembelian.
- b. Pemrosesan Data: Data yang dikumpulkan harus diproses dan disimpan dalam format yang dapat digunakan untuk pelatihan model *Naive Bayes*.
- c. Sistem Rekomendasi: Algoritma Naive Bayes diharapkan mampu menyajikan rekomendasi produk yang relevan dengan preferensi pengguna, berdasarkan data yang telah dianalisis dan dikumpulkan sebelumnya.

4. Manajemen Transaksi:

- a. Keranjang Belanja: Pengguna harus dapat menambahkan, menghapus, dan mengelola produk dalam keranjang belanja mereka.
- b. Pembayaran: Sistem harus menyediakan metode pembayaran yang aman dan mudah digunakan.
- c. Riwayat Pembelian: Pengguna harus dapat melihat riwayat pembelian mereka.

5. Antarmuka Pengguna:

Desain Responsif: Antarmuka pengguna harus responsif dan mudah digunakan di berbagai perangkat, termasuk desktop, tablet, dan ponsel. Navigasi Mudah: Sistem harus menyediakan navigasi yang intuitif untuk memudahkan pengguna menemukan produk yang mereka cari.

- a. Kebutuhan Non-Fungsional
 - 1) Keamanan Data: Sistem harus mampu menjaga kerahasiaan data pengguna melalui enkripsi dan langkah-langkah keamanan tambahan.
 - 2) Autentikasi dan Hak Akses: Hanya pengguna yang memiliki izin yang sah yang diperbolehkan mengakses fitur-fitur tertentu pada sistem.
 - 3) Performa Sistem: Waktu Respons: Sistem harus memberikan respons dengan cepat, terutama dalam proses pemberian rekomendasi.
 - 4) Skalabilitas: Sistem perlu dirancang agar mampu menangani pertumbuhan jumlah pengguna dan data tanpa mengalami penurunan performa secara signifikan.
 - 5) Keandalan (Reliabilitas) :

- a) Ketersediaan Sistem (Uptime): Sistem harus didesain untuk selalu tersedia dan meminimalkan indikasi terjadinya downtime.
 - b) Pemulihan Bencana: Perlu disediakan mekanisme pemulihan jika terjadi gangguan, guna melindungi data serta menjamin kontinuitas layanan.
- 6) Kemudahan Penggunaan (Usabilitas)
- a) Antarmuka yang User-Friendly: Desain antarmuka harus intuitif dan mudah digunakan oleh pengguna dari berbagai kalangan.
 - b) Layanan Bantuan: Sistem harus menyediakan layanan dukungan yang memadai meliputi bantuan daring (online help) dan halaman FAQ.
- b. Kebutuhan Perangkat Keras dan Lunak
- 1) Perangkat Keras:
 - a) Server: Server yang kuat dan dapat diandalkan untuk hosting aplikasi dan basis data.
 - b) Penyimpanan: Kapasitas penyimpanan yang cukup untuk menampung data pengguna dan produk.
 - c) Jaringan: Diperlukan koneksi internet yang cepat dan stabil untuk memastikan sistem dapat diakses dengan lancar.
 - 2) Perangkat Lunak:
 - a) Sistem Operasi: Menggunakan sistem operasi server yang kompatibel, meliputi Windows 10.
 - b) Basis Data: Mengandalkan sistem manajemen basis data yang andal, meliputi MySQL.
 - c) Framework Pengembangan: Menggunakan framework web yang mendukung pengembangan aplikasi, meliputi Laravel.
 - d) Library Machine Learning: Memanfaatkan library meliputi scikit-learn untuk mengimplementasikan algoritma Naive Bayes.
 - e) Lingkungan Pengembangan Lokal:

Untuk mendukung proses pengembangan dan pengujian secara lokal, digunakan XAMPP (Cross-platform, Apache, MySQL, PHP,

dan Perl). XAMPP menyediakan paket terintegrasi yang mempermudah pengembang dalam melakukan deploy sistem web tanpa perlu konfigurasi server yang rumit.

3) Kebutuhan Integrasi

a) Integrasi Pembayaran:

Integrasi sistem pembayaran difokuskan pada metode transfer bank.

b) Integrasi Pengiriman:

Integrasi dengan layanan pengiriman untuk pelacakan dan manajemen pengiriman produk.

c) Integrasi Media Sosial:

Integrasi dengan platform media sosial untuk membagikan produk kepada pengguna lain secara mudah melalui berbagai kanal digital.

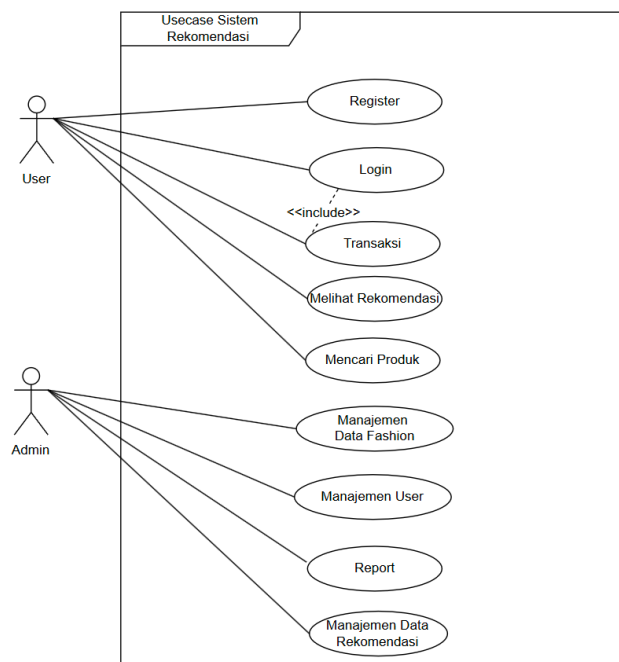
Dengan spesifikasi kebutuhan sistem yang jelas, pengembangan *website E-commerce* dengan algoritma *Naive Bayes* untuk rekomendasi fashion dapat dilakukan secara efektif, memenuhi kebutuhan pengguna, dan memberikan pengalaman belanja yang optimal.

4.4 Perancangan Sistem

Dapat diketahui bahwa baik user maupun admin memiliki akses ke basis data melalui perangkat meliputi komputer atau *smartphone* yang terhubung ke internet. Keduanya akan mengakses antarmuka front-end yang telah disesuaikan dengan peran masing-masing, baik untuk menggunakan fitur-fitur yang tersedia maupun untuk melakukan pengelolaan dalam sistem website *E-commerce*.

4.4.1 Use Case Diagram

Pada *Use Case Diagram* situs *E-Commerce*, terlihat bahwa terdapat dua aktor yang berinteraksi dengan platform UtamaShop. Aktor pelanggan memiliki sejumlah fitur yang dapat diakses, meliputi melakukan pendaftaran akun, membuat transaksi (*Add Order*), melengkapi atau memperbarui informasi pribadi (*Update Account*), mengubah detail pesanan (*Update Order*), serta menghapus pesanan (*Delete Order*).



Gambar 4.1 Use Case Diagram Sistem Rekomendasi Fashion pada Website E-Commerce

Use case diagram pada Gambar 4.1 menggambarkan dua aktor utama dalam sistem, yaitu User dan Admin, yang berinteraksi dengan sistem e-commerce berbasis web untuk fashion. Diagram ini menjelaskan delapan use case atau fungsionalitas utama yang mencerminkan proses-proses penting dalam sistem.

4.4.2 Skenario *Use Case*

Pada sub-subbab ini, skenario pemanfaatan sistem dipetakan ke dalam use-case table (Tabel 4.2). Format tabel mengikuti pedoman tabel use case Pressman yang menekankan kejelasan tujuan, alur operasi, serta batas peran setiap aktor. Delapan skenario inti diidentifikasi—empat mewakili aktivitas User dan empat lainnya hak akses Admin—sehingga seluruh kebutuhan fungsional pada diagram use case tervalidasi secara eksplisit.

Tabel 4.2 Skenario Use-case Login

Nama Use Case	Login
Aktor	Pengguna
Deskripsi Use Case	Pengguna melakukan proses login untuk mengakses sistem rekomendasi fashion
Precondition	Akun pengguna terdaftar & aktif
Tahapan	1. Akses halaman Login 2. Masukkan kredensial 3. Klik Sign in 4. Sistem validasi 5. Dialihkan ke Beranda
Postcondition	Sesi pengguna + token tersimpan; dashboard muncul

Tabel 4.3 Skenario Use-case Registrasi

Nama Use Case	Registrasi
Aktor	Pengguna
Deskripsi Use Case	Pengguna mendaftarkan akun baru pada sistem
Precondition	Semua field wajib terisi valid

Tahapan	1. Akses halaman Registrasi 2. Isi formulir 3. Kirim 4. Sistem simpan 5. Login otomatis
Postcondition	Data user baru di DB; notifikasi "Registrasi sukses"; user login

Tabel 4.4 Skenario Use-case Pencarian Produk

Nama Use Case	Pencarian Produk
Aktor	Pengguna
Deskripsi Use Case	Pengguna mencari produk berdasarkan kata kunci atau filter
Precondition	Pengguna berada di halaman Produk
Tahapan	1. Masukkan kata kunci / filter 2. Klik Cari 3. Sistem query DB 4. Daftar produk tampil
Postcondition	Grid hasil pencarian + pagination

Tabel 4.5 Skenario Use-case Lihat Rekomendasi

Nama Use Case	Lihat Rekomendasi
Aktor	Pengguna
Deskripsi Use Case	Sistem menampilkan produk rekomendasi berdasarkan profil dan histori pengguna

Precondition	Data latih & model NB tersedia
Tahapan	1. Login 2. Sistem ambil profil & histori 3. Hitung probabilitas NB 4. Tampilkan 8 produk rekomendasi
Postcondition	Seksi "Disarankan untuk Anda" dengan badge preferensi

Tabel 4.6 Skenario Use-case Tambah ke Keranjang

Nama Use Case	Tambah ke Keranjang
Aktor	Pengguna
Deskripsi Use Case	Pengguna menambahkan produk ke dalam keranjang belanja
Precondition	Pengguna sudah login
Tahapan	1. Klik "Tambah ke Keranjang" 2. Sistem cek item di cart 3. Insert/Update qty
Postcondition	Item muncul di drawer Cart; ikon keranjang +1

Tabel 4.7 Skenario Use-case Checkout / Transaksi

Nama Use Case	Checkout / Transaksi
Aktor	Pengguna
Deskripsi Use Case	Pengguna melakukan transaksi setelah memilih produk
Precondition	Login berhasil (include use case Login)

Tahapan	1. Buka Keranjang 2. Klik Checkout 3. Isi alamat & bank 4. Konfirmasi 5. Sistem buat invoice & hapus cart
Postcondition	Baris di tabel transaction & transaction_detail + invoice PDF

Tabel 4.8 Skenario Use-case Kelola Produk Fashion

Nama Use Case	Kelola Produk Fashion
Aktor	Admin
Deskripsi Use Case	Admin melakukan manajemen produk fashion
Precondition	Admin terautentikasi
Tahapan	1. Dashboard Admin 2. Menu Produk 3. Tambah/Edit/Hapus 4. Simpan
Postcondition	Tabel product ter update

Tabel 4.9 Skenario Use-case Kelola Dataset Naïve Bayes

Nama Use Case	Kelola Dataset Naïve Bayes
Aktor	Admin
Deskripsi Use Case	Admin melakukan pengelolaan dataset Naïve Bayes
Precondition	File dataset valid

Tahapan	1. Menu Dataset NB 2. Tambah/Edit atribut & label 3. Klik Simpan 4. Sistem re training
Postcondition	Model NB baru (.pkl) tersimpan; log pelatihan sukses

Tabel 4.10 Skenario Use-case Use-case Kelola Pengguna

Nama Use Case	Kelola Pengguna
Aktor	Admin
Deskripsi Use Case	Admin mengelola data pengguna
Precondition	Admin sudah login
Tahapan	1. Menu User 2. Pilih user 3. Edit status / Hapus 4. Simpan
Postcondition	Perubahan status user tercatat

Tabel 4.11 Skenario Use-case Use-case Laporan

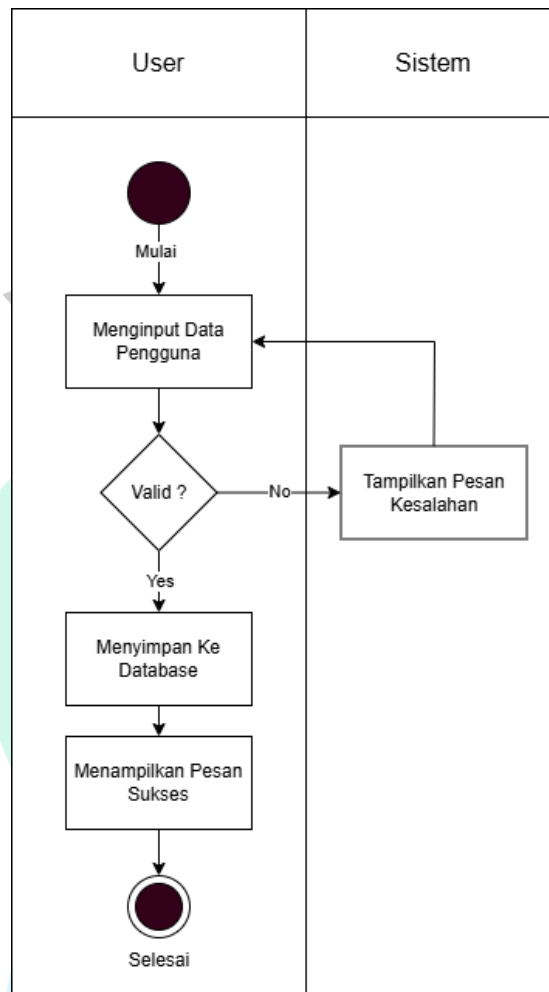
Nama Use Case	Laporan (Report)
Aktor	Admin
Deskripsi Use Case	Admin melihat dan mengunduh laporan sistem
Precondition	Basis data terisi

Tahapan	1. Menu Laporan 2. Pilih jenis (Transaksi/Rekomendasi) 3. Sistem proses 4. Tampil / Unduh PDF/Excel
Postcondition	File laporan (.pdf / .xlsx) siap diunduh

4.4.2 Activity Diagram

Activity Diagram menggambarkan alur aktivitas dari proses bisnis atau sistem, termasuk keputusan dan kondisi yang terjadi di sepanjang proses tersebut. Diagram ini berguna untuk memvisualisasikan bagaimana pengguna atau sistem berinteraksi dari awal hingga akhir dalam suatu aktivitas. Pada penelitian ini, beberapa proses utama dalam sistem digambarkan dalam bentuk activity diagram untuk mempermudah pemahaman alur kerja sistem rekomendasi fashion yang dibangun.

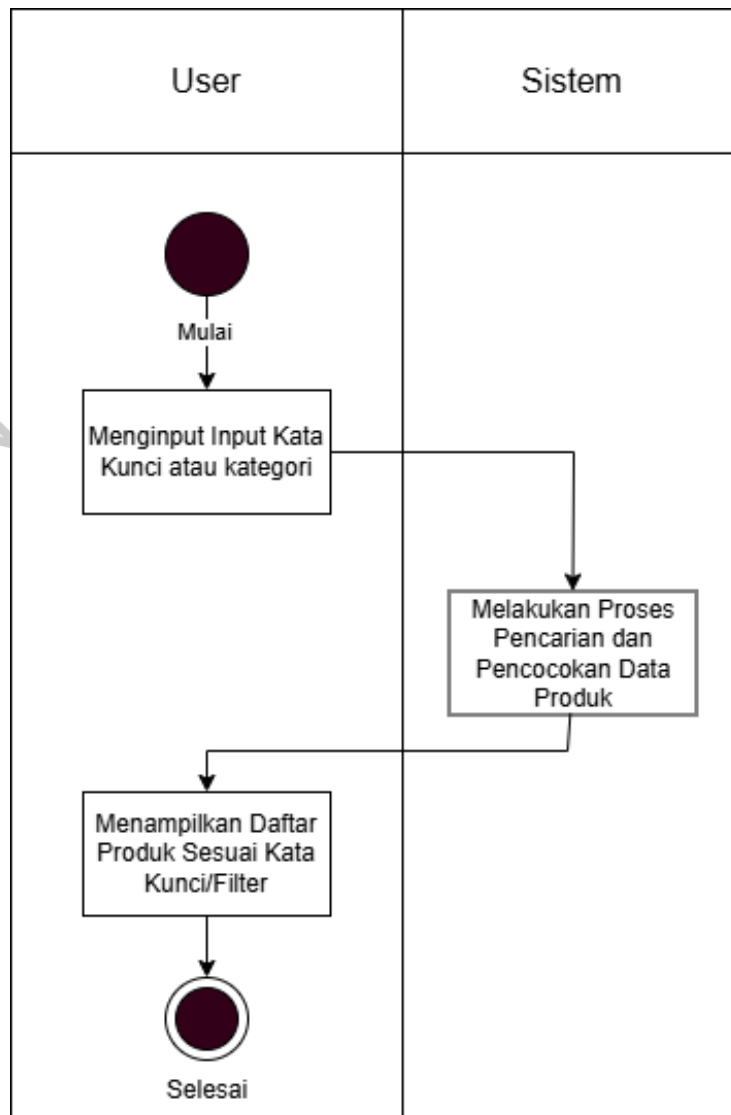
4.4.2.1 Activity Diagram Register



Gambar 4.2 Activity Diagram Proses Registrasi Pengguna

Gambar 4.2 menunjukkan tahapan proses registrasi pengguna, dimulai dari pengisian informasi pribadi, validasi data yang dimasukkan, hingga penyimpanan data oleh sistem dan pemberian notifikasi bahwa proses telah berhasil.

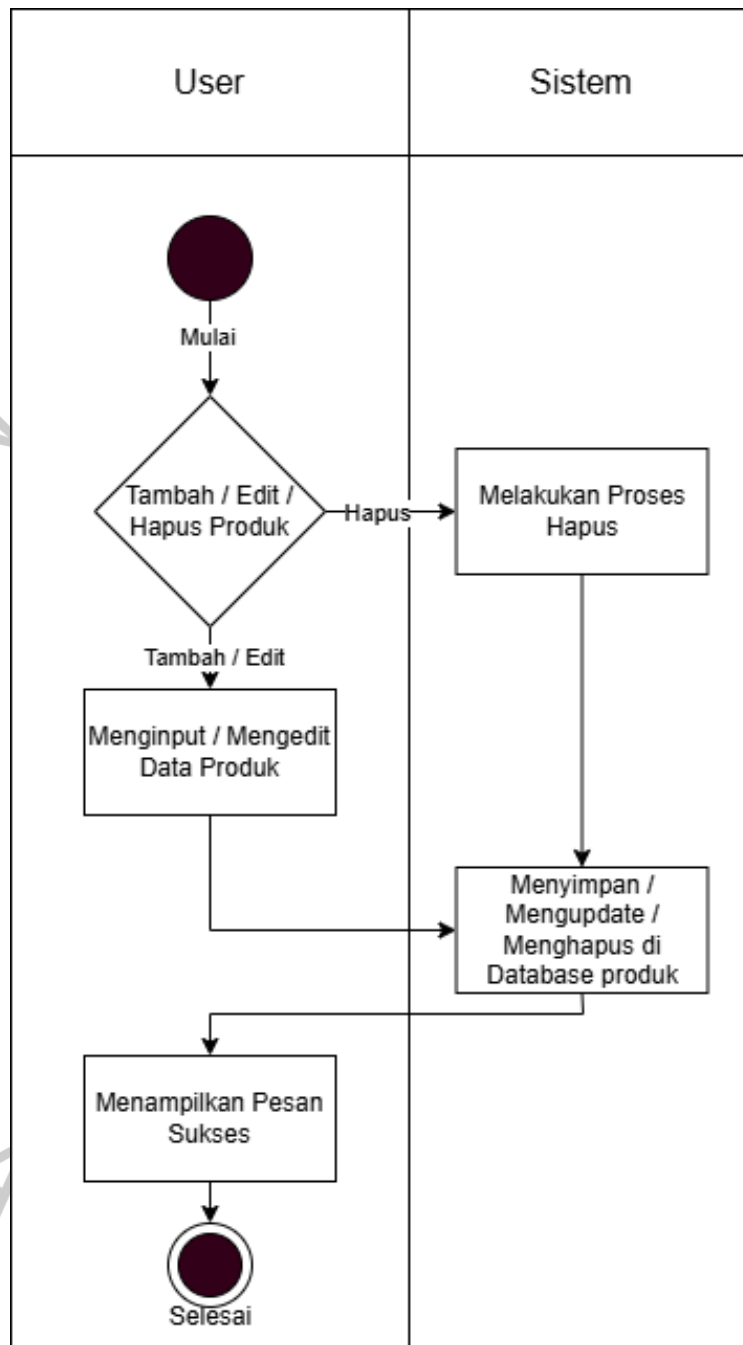
4.4.2.2 Activity Diagram Mencari Produk



Gambar 4.3 Activity Diagram Proses Pencarian Produk oleh Pengguna

Gambar 4.3 memperlihatkan alur aktivitas pengguna saat melakukan pencarian produk, dimulai dari memasukkan kata kunci hingga sistem menampilkan daftar produk yang sesuai.

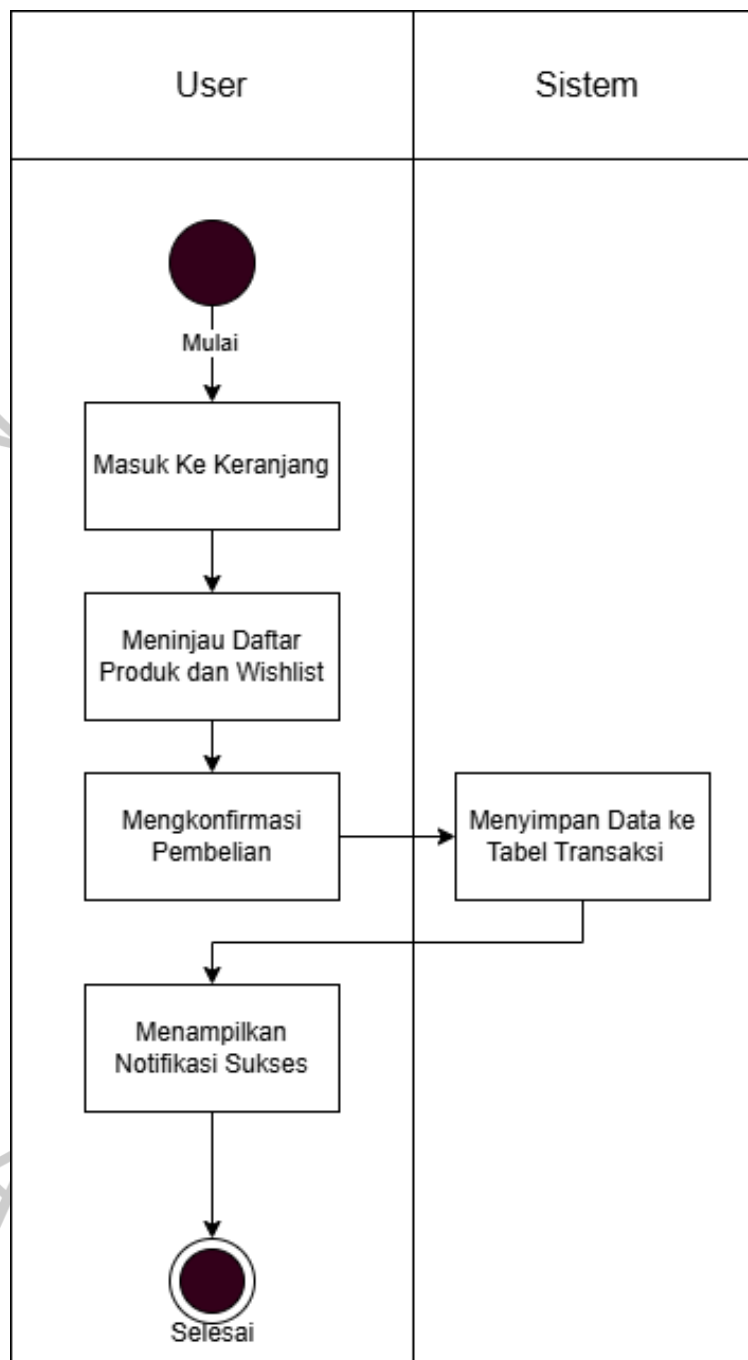
4.4.2.3 Activity Diagram Management Produk



Gambar 4.4 Activity Diagram Manajemen Produk oleh Admin

Gambar 4.4 menampilkan alur kerja admin dalam mengelola data produk, yang mencakup proses penambahan, pengubahan, dan penghapusan informasi produk yang tersimpan di dalam database.

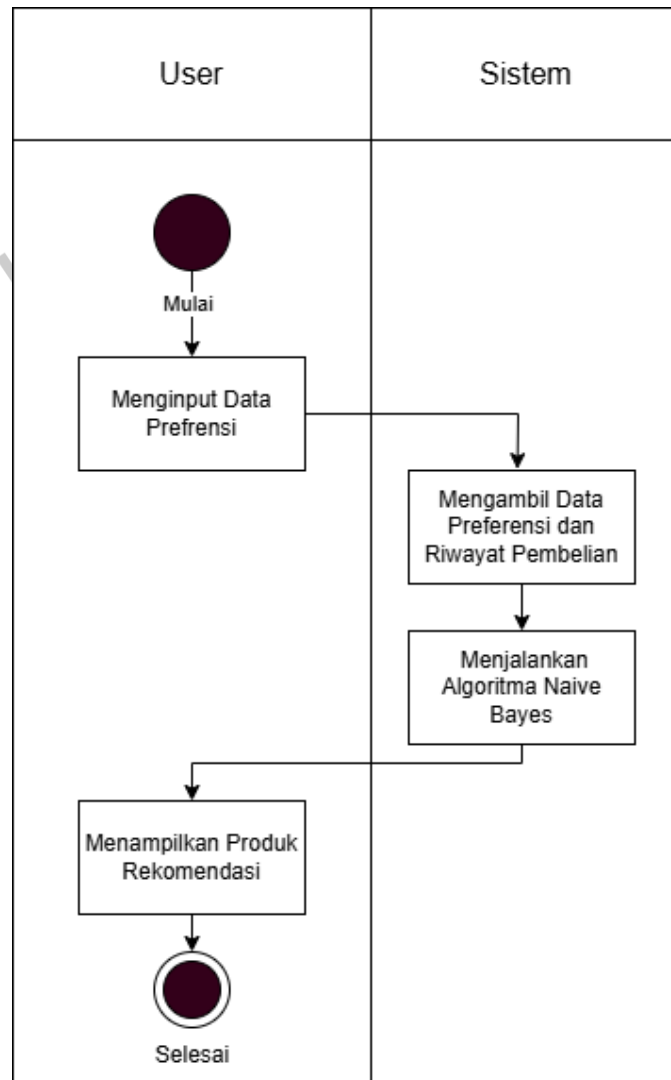
4.4.2.4 Activity Diagram transaksi



Gambar 4.5 Activity Diagram Proses Transaksi Pembelian

Gambar 4.5 memperlihatkan tahapan proses transaksi yang dijalani oleh pengguna, dimulai dari pemilihan produk, proses check-out, hingga tahap konfirmasi pembayaran.

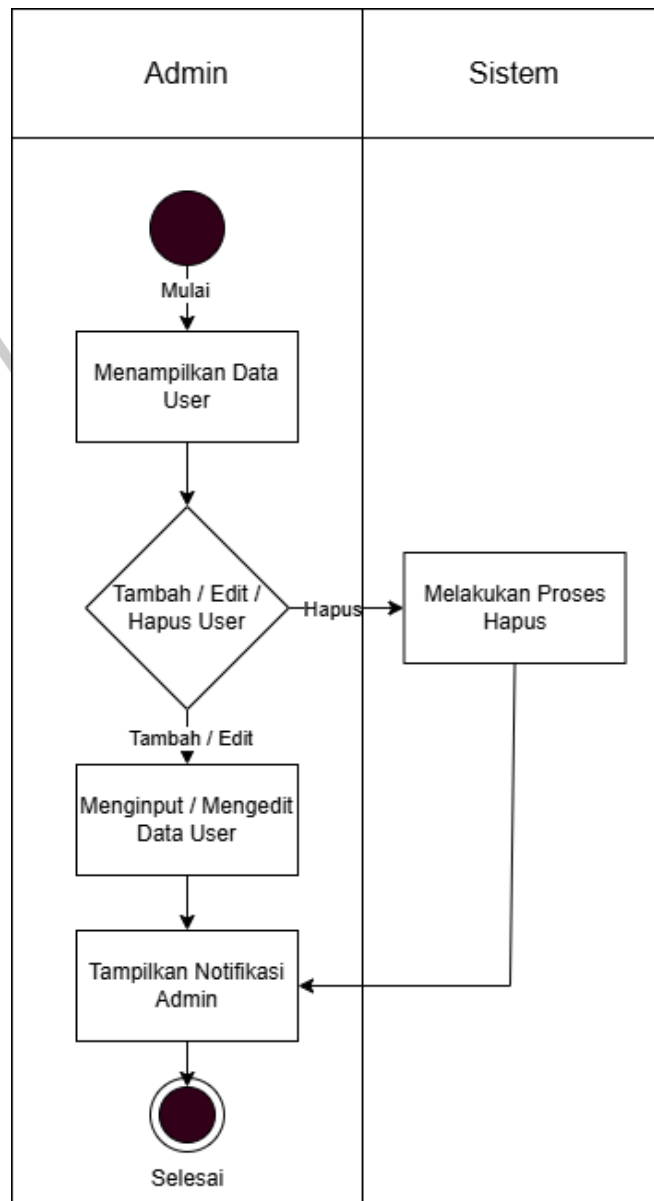
4.4.2.5 Activity Diagram Melihat Rekomendasi



Gambar 4.6 Activity Diagram Melihat Rekomendasi Produk oleh Pengguna

Gambar 4.6 menggambarkan alur proses sistem ketika pengguna mengajukan permintaan rekomendasi produk. Proses diawali dengan pengambilan data histori pembelian atau preferensi pengguna, yang kemudian dianalisis menggunakan algoritma Naive Bayes. Hasil analisis tersebut menghasilkan daftar produk yang ditampilkan kepada pengguna, berdasarkan nilai probabilitas tertinggi dari proses klasifikasi.

4.4.2.6 Activity Diagram Manajemen User

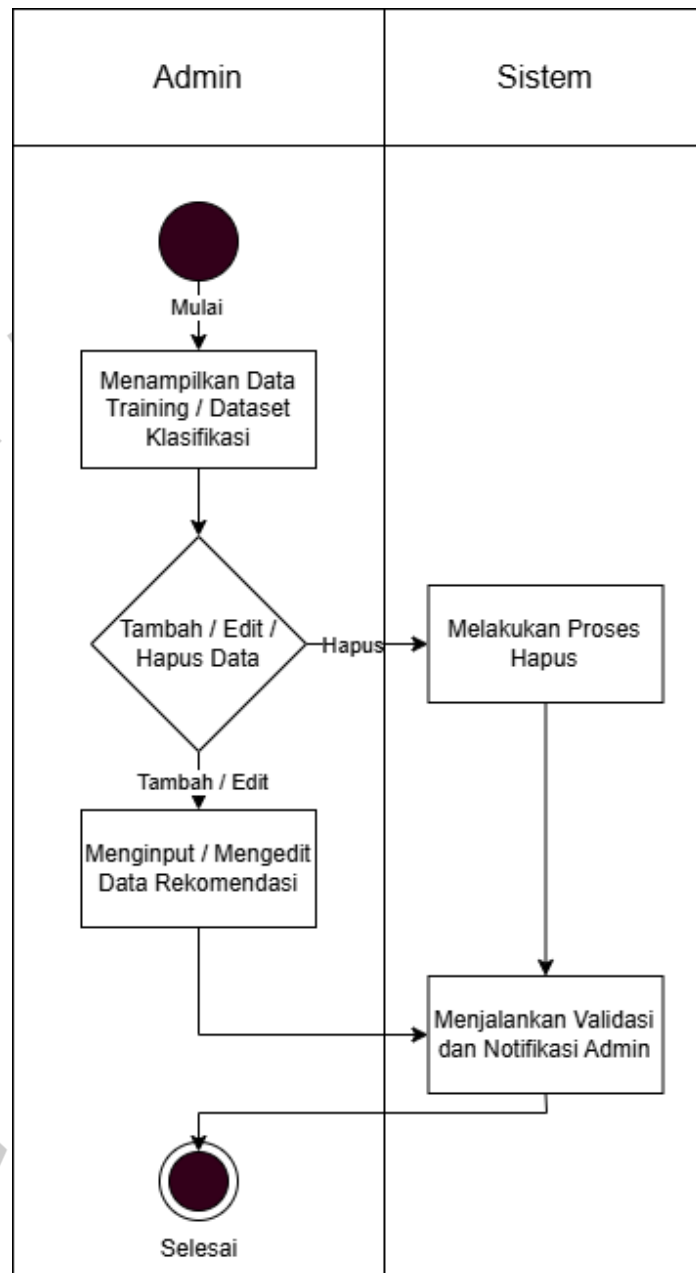


Gambar 4.7 Activity Diagram Manajemen Data Pengguna oleh Admin

Diagram ini menjelaskan bagaimana admin mengelola data pengguna melalui proses CRUD (Create, Read, Update, Delete). Admin dapat menambahkan pengguna baru, mengubah informasi pengguna, atau menghapus data pengguna dari sistem. Proses ini penting untuk memastikan keamanan dan keteraturan data dalam sistem e-commerce.

4.4.2.7

Activity Diagram Manajemen Data Rekomendasi

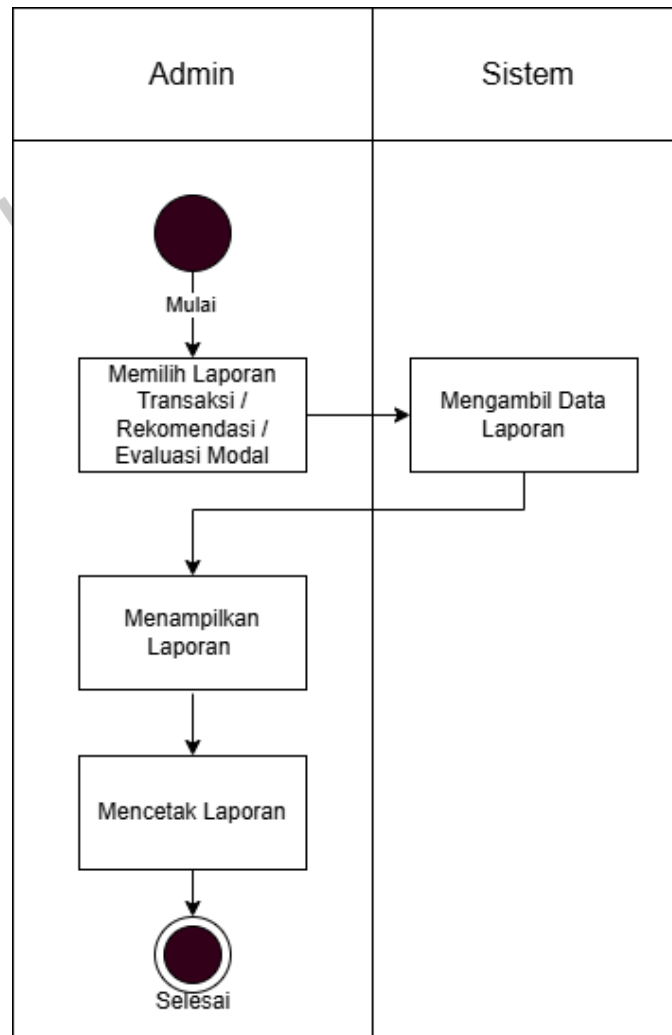


Gambar 4.8 Activity Diagram Pengelolaan Data Rekomendasi oleh Admin

Gambar 4.8 menunjukkan alur kerja admin dalam melakukan input, pembaruan, maupun penghapusan data training yang menjadi dasar bagi sistem rekomendasi. Data tersebut mencakup atribut pengguna meliputi preferensi gaya fashion, warna kesukaan, dan riwayat pembelian. Pengelolaan data training ini memiliki peran

penting dalam menjaga akurasi dan kualitas hasil klasifikasi oleh model Naive Bayes.

4.4.2.8 Activity Diagram Report



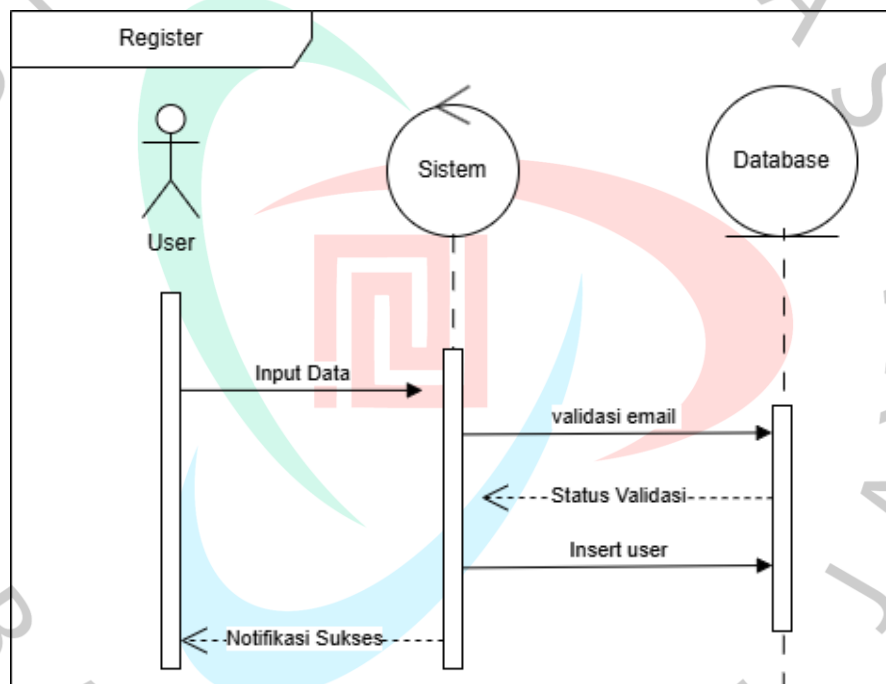
Gambar 4.9 Activity Diagram Proses Pembuatan Laporan oleh Admin

Gambar 4.9 menunjukkan proses kerja admin dalam menyusun laporan sistem. Laporan tersebut mencakup informasi transaksi, data pengguna, serta hasil evaluasi dari kinerja model Naive Bayes. Diagram ini berfungsi sebagai alat bantu untuk memantau serta mengoptimalkan performa sistem secara berkelanjutan.

4.4.3 Sequence Diagram

Sequence diagram adalah diagram interaksi yang menunjukkan bagaimana proses bekerja dengan menampilkan pesan yang dikirim antara objek dalam urutan waktu. Dalam konteks *Naive Bayes* pada *e-commerce* untuk rekomendasi fashion berbasis *website*, sequence diagram akan menjelaskan aliran proses dari saat pengguna berinteraksi dengan sistem hingga sistem memberikan rekomendasi.

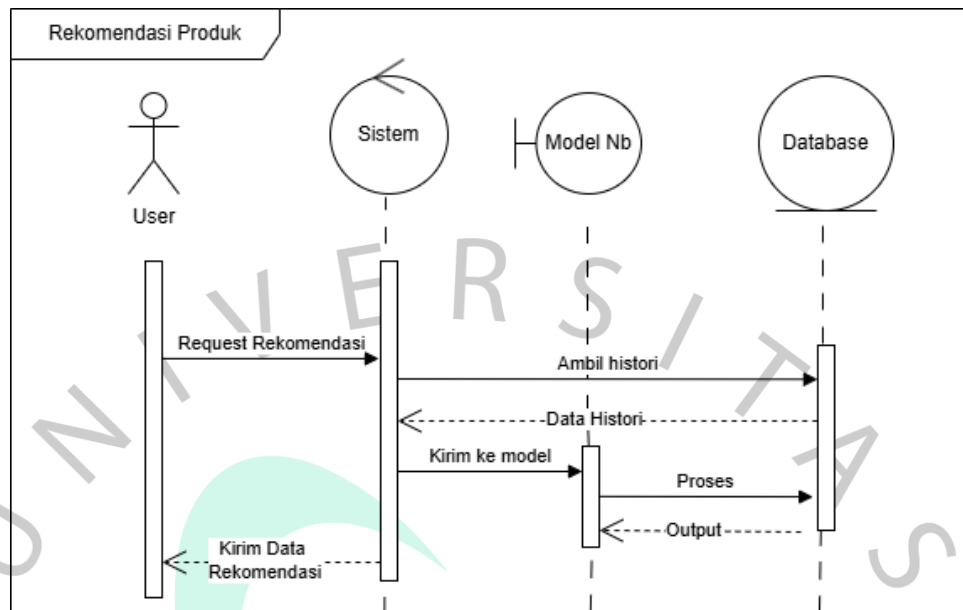
4.4.3.1 Sequence Diagram Register



Gambar 4.10 Sequence Diagram Proses Registrasi Pengguna

Diagram ini menggambarkan alur registrasi pengguna, mulai dari interaksi awal hingga penyimpanan data ke dalam database. Pengguna mengisi formulir pendaftaran, kemudian sistem melakukan validasi dan menyimpan informasi tersebut melalui proses backend. Tahapan ini berperan penting sebagai dasar pembentukan riwayat dan profil pengguna yang akan dimanfaatkan dalam sistem rekomendasi.

4.4.3.2 Sequence Diagram Melihat Rekomendasi

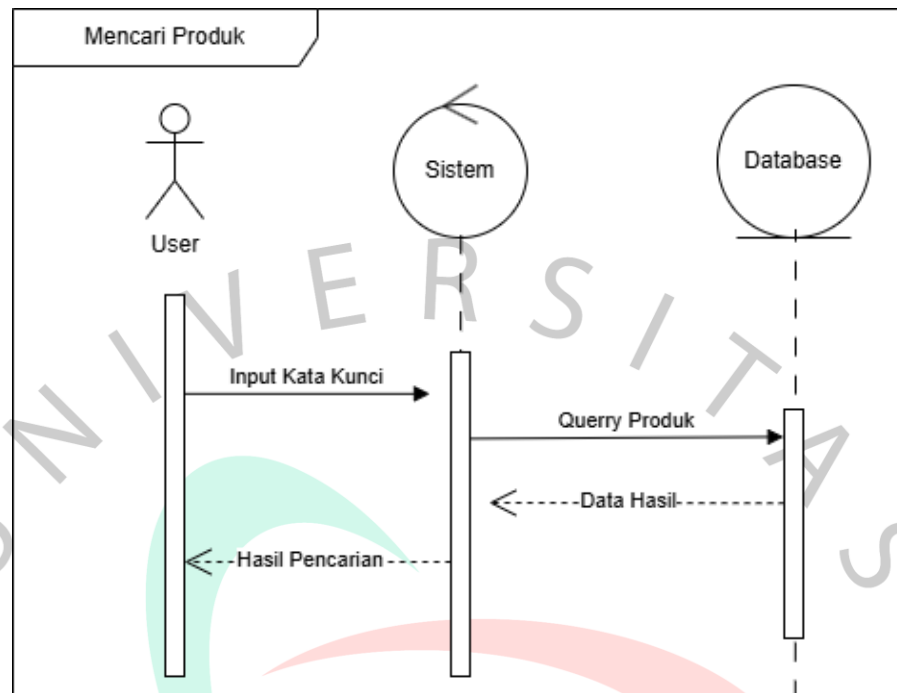


Gambar 4.11 Sequence Diagram Interaksi Melihat Rekomendasi Produk

Diagram ini menunjukkan urutan interaksi ketika pengguna meminta sistem untuk menampilkan rekomendasi. Sistem mengumpulkan data, memprosesnya melalui algoritma Naive Bayes, lalu menampilkan hasil produk yang paling relevan. Proses ini menggambarkan penggunaan model klasifikasi dalam waktu nyata.

4.4.3.3

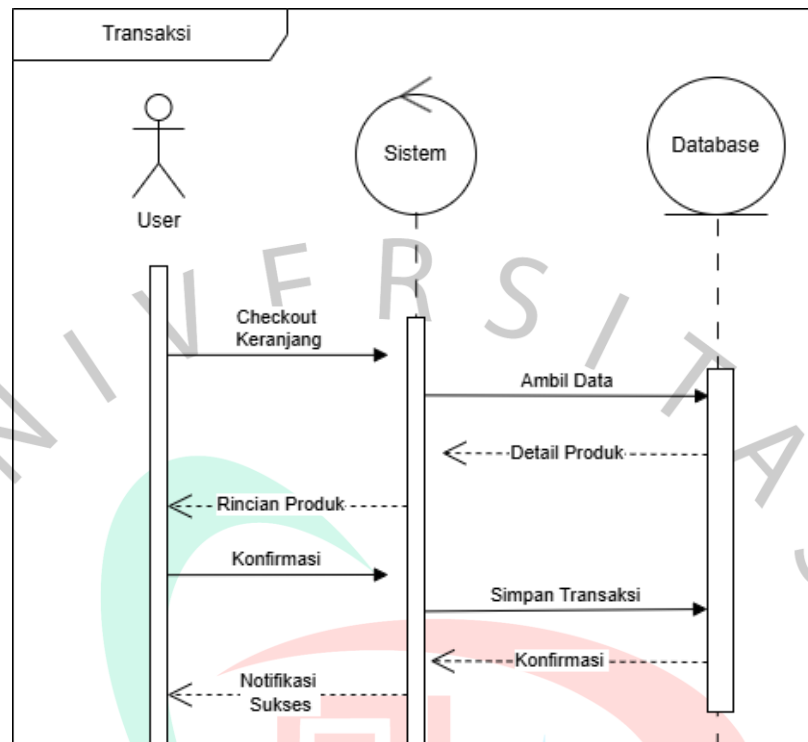
Sequence Diagram Mencari Produk



Gambar 4.12 Sequence Diagram Proses Pencarian Produk

Diagram ini memperlihatkan proses saat pengguna melakukan pencarian produk melalui antarmuka sistem. Sistem akan memproses kueri pencarian tersebut, mencocokkannya dengan data yang ada di database produk, dan kemudian menampilkan hasil yang relevan. Walaupun proses ini tidak secara langsung menghasilkan rekomendasi, data pencarian yang diperoleh dapat dimanfaatkan sebagai bahan pelatihan untuk model prediktif.

4.4.3.4 Sequence Diagram Transaksi



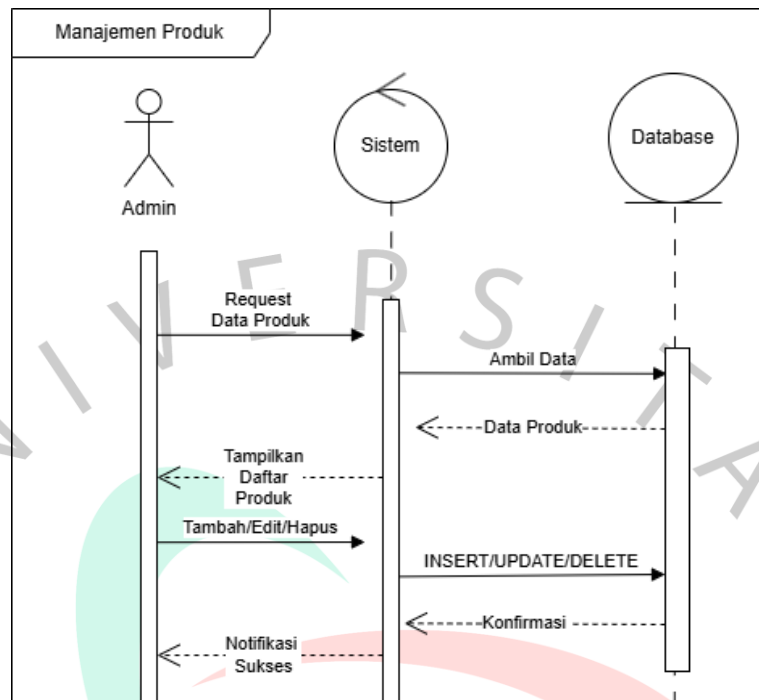
Gambar 4.13 Sequence Diagram Transaksi Pembelian Produk

Diagram ini menggambarkan alur dari saat pengguna memilih produk, mengonfirmasi pembelian, hingga data transaksi tercatat di database. Setiap transaksi yang dilakukan akan menjadi input tambahan untuk model Naive Bayes dalam memahami preferensi pengguna.



4.5.3.5

Sequence Diagram Manajemen Data Fashion

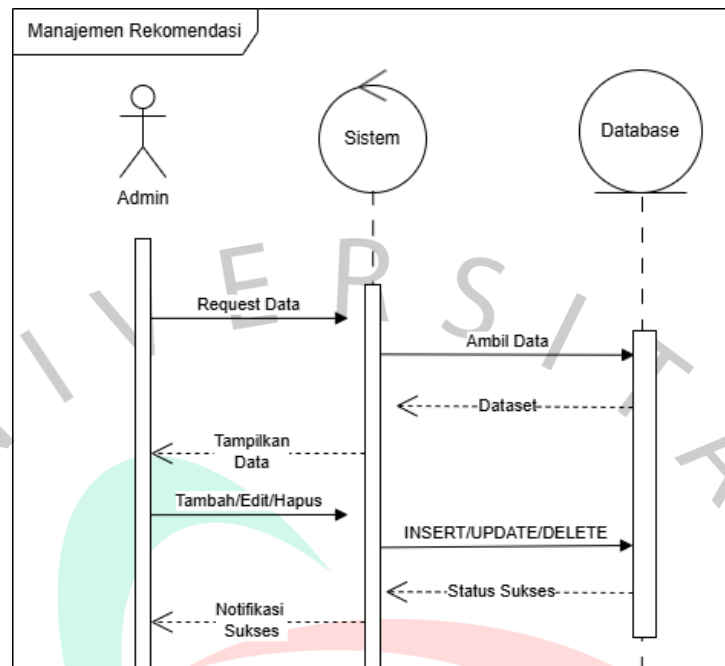


Gambar 4.14 Sequence Diagram Manajemen Data Produk Fashion

Diagram ini menggambarkan bagaimana interaksi antara admin dan sistem berlangsung dalam proses pengelolaan produk fashion. Admin memiliki kemampuan untuk menambahkan, memperbarui, atau menghapus produk dari katalog. Informasi produk yang dikelola ini kemudian dimanfaatkan sebagai salah satu fitur dalam proses klasifikasi untuk sistem rekomendasi.

4.4.3.6

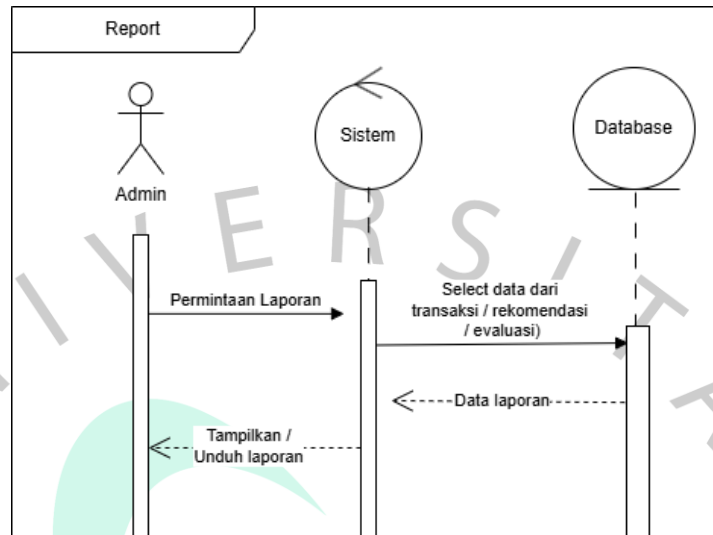
Sequence Diagram Manajemen Data Rekomendasi



Gambar 4.15 Sequence Diagram Pengelolaan Data Rekomendasi

Diagram ini menampilkan urutan proses saat admin mengelola hasil klasifikasi atau input data training untuk sistem rekomendasi. Proses ini penting dalam memastikan bahwa data yang digunakan oleh model Naive Bayes adalah akurat dan mutakhir. Admin dapat mengedit atau menghapus data yang tidak valid.

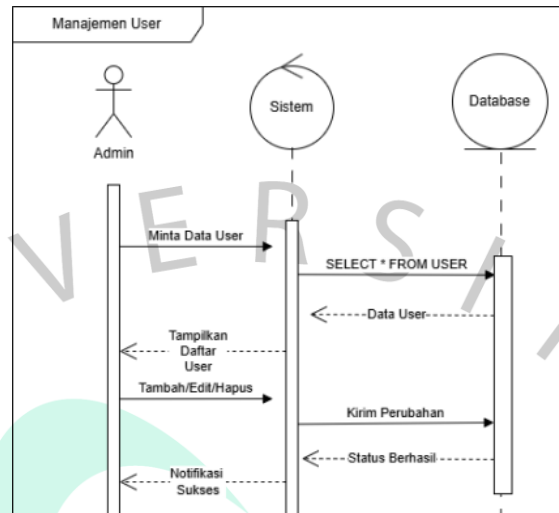
4.4.3.7 Sequence Diagram Report



● Gambar 4.16 Sequence Diagram Penyusunan Laporan oleh Admin ●

Diagram ini menjelaskan bagaimana admin melakukan permintaan laporan kepada sistem. Sistem mengambil data dari database dan menyusunnya menjadi laporan yang mencakup transaksi, pengguna, dan performa model. Proses ini mendukung monitoring dan evaluasi sistem.

4.4.3.8 Sequence Diagram Manajemen User



Gambar 4.17 Sequence Diagram Manajemen User

Diagram ini menunjukkan alur interaksi ketika admin melakukan pengelolaan data pengguna. Proses ini dimulai dari aksi pada UI, dilanjutkan ke backend, hingga data disimpan atau dimodifikasi dalam database. Proses ini penting untuk menjaga struktur data yang mendukung sistem rekomendasi yang akurat.

Komponen Utama dalam Sequence Diagram

1. Actors (Pengguna): Individu yang menggunakan situs *E-commerce*.
2. Frontend (*Website*): Antarmuka pengguna yang berinteraksi dengan pengguna.
3. Backend (Server): Tempat di mana logika aplikasi dan algoritma *Naive Bayes* dijalankan.
4. *Database*: Tempat penyimpanan data pengguna, produk, dan riwayat transaksi.
5. Recommendation Engine: Komponen backend yang mengimplementasikan algoritma *Naive Bayes* untuk memberikan rekomendasi.

Berikut adalah urutan langkah dalam sequence diagram ini:

1. *User Interacts with Website*:
Pengguna membuka situs *E-COMMERCE* dan mulai menjelajahi produk fashion.
2. *Website Requests User Data*:

Frontend mengirimkan permintaan data pengguna ke backend, termasuk riwayat transaksi dan preferensi pengguna.

3. Backend Fetches *User Data*:

- a. Backend mengambil data pengguna dari *database*.
- b. Backend Processes Data:
- c. Backend memproses data pengguna dan menyiapkannya untuk model Naïve Bayes.
- d. *Naive Bayes Model Execution*:
- e. Backend mengirimkan data yang telah diproses ke Recommendation Engine. Recommendation Engine menjalankan model *Naive Bayes* untuk menghitung probabilitas dan menentukan produk yang direkomendasikan.
- f. Recommendation Result: Recommendation Engine mengirimkan hasil rekomendasi kembali ke backend.
- g. Display Recommendations: Backend mengirimkan hasil rekomendasi ke frontend. Frontend menampilkan produk yang direkomendasikan kepada pengguna.

4.5 Pembahasan *Naive Bayes*

4.5.1 Data Training

Data pelatihan dalam penelitian ini berasal dari kombinasi hasil survei pengguna, data transaksi pada tabel *transactions_details*, serta informasi mengenai preferensi fashion dan perilaku pembelian. Dataset tersebut digunakan sebagai input dalam proses klasifikasi dengan algoritma Naive Bayes untuk membangun sistem rekomendasi fashion yang bersifat personalisasi. Jumlah data pelatihan yang digunakan sebanyak 1.500 baris, yang mencakup beragam kombinasi atribut demografis, kebiasaan pengguna, dan preferensi fashion. Seluruh data telah melalui proses pembersihan dan transformasi agar layak digunakan dalam proses machine learning.

Fitur-Fitur Utama pada Dataset:

Tabel 4.12 Daftar Fitur Utama yang Digunakan dalam Dataset

Atribut	Deskripsi
umur	Kelompok umur pengguna, dibagi dalam kategori 18–24, 25–34, dan 35+
gender	Jenis kelamin pengguna (Laki-laki / Perempuan)
kategori	Jenis produk yang dibeli (kaos, jaket, blouse, dress, dll.)
budget_band	Kisaran anggaran belanja pengguna: <100k, 100k–300k, >600k
price_band	Diskretisasi harga aktual produk (murah, sedang, mahal)
brand_pref	Merek yang sering dibeli pengguna (Zara, H&M, Uniqlo, dll.)
buy_freq	Frekuensi pembelian pakaian dalam sebulan (Setiap minggu, 1–2 kali, jarang, dll.)
buy_factor	Faktor utama keputusan pembelian (Harga, Kualitas, Gaya, Brand, Bahan, Diskon, Influencer)
channel_main	Kanal pembelian utama: sosial media, marketplace, atau toko fisik
style_pref	Gaya favorit pengguna: formal, kasual, sporty, minimalis
color_pref	Warna pakaian yang sering dibeli
eco_interest	Minat pengguna terhadap produk ramah lingkungan
eco_paymore	Kesediaan membayar lebih untuk produk eco-friendly
review_consider	Seberapa sering pengguna mempertimbangkan ulasan saat belanja
status	Label target yang diklasifikasikan: kategori rekomendasi (formal / kasual)

Tabel 4.12 menunjukkan fitur umur dikategorikan menjadi tiga kelompok usia, yaitu 18–24 tahun, 25–34 tahun, dan di atas 35 tahun, yang mencerminkan kelompok usia pengguna dalam segmen pasar fashion online. Variabel gender berisi jenis kelamin pengguna yang dikodekan sebagai laki-laki (L) dan perempuan (P), yang juga menjadi indikator penting dalam analisis preferensi fashion. Selanjutnya, atribut kategori berisi jenis produk fashion yang dibeli meliputi dress, kaos, blouse, atau jaket, yang menjadi variabel inti dalam klasifikasi gaya berpakaian. Fitur `budget_band` dan `price_band` masing-masing mewakili kisaran anggaran belanja dan kategori harga produk. Rentang anggaran disusun dalam kelompok <100 ribu, 100–300 ribu, dan >600 ribu, sedangkan harga diklasifikasikan sebagai murah, sedang, atau mahal. Atribut `brand_pref` mengacu pada preferensi merek tertentu yang sering dipilih pengguna, meliputi Zara, Nike, atau Uniqlo, yang menjadi indikasi segmentasi pasar berdasarkan loyalitas terhadap merek.

Selain atribut transaksional, dataset juga mencakup variabel perilaku meliputi `buy_freq`, yang menunjukkan seberapa sering pengguna melakukan pembelian, serta `buy_factor`, yang mencerminkan alasan utama dalam memilih suatu produk—misalnya karena harga, kualitas, atau diskon. Fitur `channel_main` menunjukkan kanal pembelian utama meliputi media sosial atau marketplace, sedangkan `review_consider` mengukur sejauh mana ulasan pelanggan mempengaruhi keputusan pengguna. Data survei juga memperkaya dataset dengan fitur meliputi `eco_interest` dan `eco_paymore`, yang merefleksikan kesadaran dan kepedulian pengguna terhadap produk ramah lingkungan. Dua fitur penting lainnya adalah `style_pref` dan `color_pref`, yang menyimpan informasi tentang gaya dan warna pakaian yang paling disukai pengguna. Akhirnya, fitur status digunakan sebagai label target klasifikasi, yang menunjukkan apakah produk yang dikonsumsi atau disukai pengguna lebih condong pada kategori fashion kasual atau formal. Semua fitur tersebut bersifat kategorikal dan digunakan sebagai input dalam proses pembelajaran model Naive Bayes untuk membentuk distribusi probabilistik

[illegible]

Gambar diatas memperlihatkan 45 baris pertama dari dataset hasil survei pengguna. Data ini diambil dari kuesioner online yang mencakup usia, jenis kelamin, frekuensi belanja, harga, hingga minat terhadap merek dan faktor pembelian. Seluruh data dikompilasi dan dimasukkan ke dalam sistem database sebagai basis pembelajaran algoritma.

Data pada gambar tersebut telah mengalami proses:

- Standarisasi format teks
- Diskretisasi kategori untuk mendukung perhitungan probabilistik dalam Naive Bayes
- Pembersihan dari outlier dan data tidak valid
- Encoding atribut kategorikal menjadi numerik saat pelatihan model

Data pada gambar tersebut telah mengalami proses:

- Dataset ini memainkan peran penting dalam membentuk model klasifikasi yang dapat memprediksi kategori pakaian yang sesuai bagi pengguna baru berdasarkan kemiripan karakteristik.

4.5.2 Menghitung Probabilitas

Semua fitur diperlakukan sebagai kategorikal. Model menerapkan Laplace smoothing ($\alpha = 1$) untuk menghindari nol probabilitas. Untuk setiap fitur f_i dengan nilai v dan untuk tiap kelas label $y \in \{0, 1, 2\}$, probabilitas dihitung:

$$P(f_i = v | y) = \frac{\text{count}(f_i=v, y) + 1}{\text{count}(y) + |V_i|}$$

di mana $f_i = v, y$ dihitung dari trainingSet, $\text{count}(y)$ jumlah instance berlabel y dan $|V_i|$ jumlah nilai unik fitur f_i . Prior juga dismoothing:

$$P(y) = \frac{\text{count}(y) + 1}{N + K},$$

dengan N total instance (~ 1.500) dan K jumlah kelas label :

- Karena label aktual hanya satu nilai (0), $\text{count}(y) = N$, prior $P(0) \approx 1$.
- Jika query diperbaiki menghasilkan beberapa label, rumus di atas berlaku.

4.5.3 Prediksi

Untuk instance baru x (array asosiatif berisi fitur umur, gender, kategori, budget_band, price_band, buy_freq, buy_factor, channel_main, review_consider, eco_interest, eco_paymore, style_pref, color_pref), menghitung skor (posterior tidak ter-normalkan dahulu):

$$\tilde{P}(y | x) = P(y) \prod_i P(f_i(x) = v_i | y).$$

Kemudian di-normalisasi dengan $\sum_y \tilde{P}(y | x)$ untuk mendapatkan $P(y | x)$, dan memilih kelas dengan probabilitas tertinggi.

- Dalam implementasi saat ini, hanya satu kelas $y = 0$ sehingga prediksi selalu kelas 0.
- Jika query diperbaiki menghasilkan beberapa label, rumus di atas berlaku.

4.6 Perhitungan *Naive Bayes* dalam Sistem

Bagian ini menguraikan langkah-langkah verifikasi awal (initial validation) yang ditempuh setelah model klasifikasi *Naive Bayes* selesai dilatih pada 1500 baris data training. Validasi dilakukan dengan data uji yang bersumber langsung dari aktivitas terkini pengguna di sistem e-commerce, sehingga dapat dipastikan bahwa distribusi probabilistik yang dibentuk selama tahap pelatihan benar-benar relevan ketika model diterapkan di lingkungan produksi. Seluruh angka performa (akurasi, precision, recall, F1-score, dll.) tidak dicantumkan di sini; hasil kuantitatif tersebut akan diuraikan secara rinci pada Bab V (Hasil dan Pembahasan).

4.6.1 Penentuan Data Uji

Data uji dipilih menggunakan pendekatan production sampling—yakni, 300 baris rekaman terbaru yang belum pernah diproses pada fase pelatihan. Tiap baris memuat ke-14 atribut kategorikal sebagaimana didefinisikan pada Tabel 4.1 (umur, gender, kategori produk, style pref, dan seterusnya). Pengambilan secara chronological hold-out ini bertujuan:

- Merefleksikan tren fashion terkini (mis. warna musiman, diskon flash-sale).
- Mencegah data leakage sehingga model tidak “mengingat” sampel yang sama ketika diuji.

Tabel 4.13 Ringkasan Dataset Uji (300 Baris, 14 Atribut)

No	Atribut (Kolom)	Tipe Data	Contoh Nilai	Kategori / Rentang	Keterangan Singkat
1	usia	Ordinal	18–24	18–24, 25–34, 35+	Hasil konversi dari users.birth

2	gender	Nominal	m / f	2 kategori: m (male), f (female)	Jenis kelamin dari users.gender
3	kategori	Nominal	sweater	Kategori produk fashion	Diambil dari products.category_id → categories.slug
4	budget_band	Ordinal	<100k	<100k, 100–300k, 300–600k, >600k	Dihitung dari products.price
5	buy_freq	Ordinal	Setiap bulan	Mingguan, Bulanan, Beberapa bulan, Jarang	Kuisisioner frekuensi belanja dari survey_responses
6	buy_factor	Nominal	Harga	Harga, Kualitas, Trend, dll	Faktor utama dari kuisisioner
7	channel_main	Nominal	Marketplace	Marketplace, Sosmed, Web brand, Offline	Kanal belanja dari kuisisioner

8	review_consider	Ordinal	Sering	Tak pernah, Kadang, Sering, Selalu	Seberapa sering mempertimbangkan ulasan
9	eco_interest	Ordinal	Agak tertarik	Tidak tertarik, Agak tertarik, Tertarik	Ketertarikan produk ramah lingkungan
10	eco_paymore	Nominal	Mungkin	Ya, Tidak, Mungkin	Kesediaan bayar lebih untuk eco-friendly
11	color_pref	Nominal	hitam	1–2 warna favorit per user	Dari user_colors.color_slug
12	style_pref (Label)	Nominal	kasual / formal	2 kelas: casual, formal	Label target dari user_styles.style_slug

4.6.2 Pra-Proses Konsisten

Agar probabilitas $P(x_i|C_k)$ yang dihitung tetap valid, **pipeline pra-proses** pada data uji **di-reuse** persis seperti pada data latih:

1. **Pembersihan teks** —penghapusan karakter non-ASCII, *trim* spasi ganda.
2. **Diskretisasi numerik** —mis. harga $\rightarrow$ *price_band* {murah, sedang, mahal}.
3. **Penanganan nilai kosong** —diisi token “Tidak ada”.

4. **Label encoding** —setiap kategori diubah ke indeks bilangan bulat ($0 \dots n-1$).
5. **Laplace smoothing ($\alpha = 1$)** —menjamin tak ada probabilitas nol meskipun muncul nilai kategori baru yang jarang.



4.7 Perancangan Pengujian

Perancangan pengujian menjabarkan metode, skenario, dan tolok ukur yang digunakan peneliti untuk memastikan bahwa (a) fungsi-fungsi sistem e-commerce berjalan sesuai kebutuhan pengguna dan (b) model Naïve Bayes yang di implementasikan mampu memberikan prediksi yang dapat diandalkan. Seluruh pengujian dilakukan secara black-box—yakni berfokus pada keluaran sistem tanpa meninjau kode sumber—serta di lengkapi pengujian performa model menggunakan Confusion Matrix dan turunan metriknya.

4.7.1 Perancangan Pengujian Black Box

Pengujian black box mengevaluasi fungsi sistem dari sudut pandang pengguna tanpa meninjau kode sumber. Rincian skenario tercantum pada tabel 4.14

Tabel 4.14 Perancangan Pengujian Black-Box

No	Fitur yang diuji	Hasil yang diharapkan	Hasil Pengamatan
1	Halaman Home – grid Produk Terbaru	Katalog empat-kolom terpanggil & tampil	–
2	Login	Kredensial valid → session terbentuk & redirect ke / (user) atau /admin/dashboard (admin)	–
3	Registrasi	Input lengkap → data user tersimpan & toast “Pendaftaran Berhasil”	–

4	Pencarian Produk	Ketik kata kunci <i>shirt</i> → daftar hanya berisi produk shirt & URL mengandung ?search=shirt	–
5	Form Kuisioner Rekomendasi	Isi umur, gender, kategori, frekuensi, budget → tombol Dapatkan Rekomendasi memunculkan ≤ 8 produk relevan	–
6	Tambah ke Keranjang – produk baru	Klik Tambah ke Keranjang → ikon cart naik +1	–
7	Update Qty Keranjang	Klik + atau – pada halaman Cart → kolom qty berubah & subtotal ter-update tanpa reload	–
8	Checkout	Isi alamat + bank → tombol Proses Transaksi & mengosongkan keranjang	–
9	Wishlist – Tambah/Hapus	Klik ikon cart pada kartu produk → produk masuk ke halaman Wishlist; klik cart lagi → produk hilang dari Wishlist	–

10	Logout	Klik Logout → session di-flush & redirect ke halaman Login	–
----	---------------	---	---

4.7.2 Perancangan Pengujian White Box

Pengujian white-box difokuskan pada potongan kode yang secara langsung menghasilkan data di tampilan—serupa dengan contoh rekan Anda. Setiap skenario memuat (a) cuplikan kode yang diuji, (b) deskripsi jalur eksekusi, serta (c) hasil yang diharapkan.

Tabel 4.15 Perancangan Pengujian White-Box

N o	Skenario Pengujian	Hasil yang Diharapkan	Hasil
1	<pre> public function index() { // 1) Keranjang & count \$cartModel = \$this->model('cart'); \$carts = \$cartModel->findAll(session('user')); \$count = \$cartModel->total(); // 2) Produk terbaru & kategori untuk header/nav \$products = \$this->model('product')->findLast(8); \$catModel = \$this->model('category');</pre>	Variabel \$naivebayes terisi 8 produk rekomendasi	–

	<pre> \$dropdowns = \$catModel->dropdownMenu(); \$treeOpts = \$catModel->selectTree(); // 3) Siapkan list kategori leaf (slug=>name) untuk form radio \$allCats = \$catModel->getAllForForm(); \$kategoriList = []; foreach (\$allCats as \$c) { if ((int)\$c['parent_id'] !== 0) { \$kategoriList[\$c['slug']] = \$c['name']; } } // 4) List opsi kuisioner lain \$buyFreqList = ['Setiap minggu','Setiap bulan','Beberapa bulan','Jarang']; \$budgetBandList = ['<100k','100- 300k','300-600k','>600k']; // 5) Default rekomendasi = best-seller (lama) \$trxModel = \$this->model('transaction'); </pre>		
--	---	--	--

	<pre> \$naivebayes = []; \$formInput = []; // 6) Kalau form POST lengkap → pakai Bayes if (\$_SERVER['REQUEST_METHOD'] === 'POST' ' && post('umur') && post('kategori') && post('gender') && post('buy_freq') && post('budget_band')) { \$formInput = ['umur' => post('umur'), 'kategori' => post('kategori'), 'gender' => post('gender'), 'buy_freq' => post('buy_freq'), 'budget_band' => post('budget_band'),]; </pre>		
--	---	--	--

	<pre> if (method_exists(\$trxModel,'recommendProducts')) { \$naivebayes = \$trxModel->recommendProducts(\$formInput, 8); } } // 7) Render view \$this->page('home', ['home' => true, 'page' => 'home', 'carts' => \$carts, 'count' => \$count, 'products' => \$products, 'dropdowns' => \$dropdowns, 'categories' => \$treeOpts, 'kategoriList' => \$kategoriList, 'buyFreqList' => \$buyFreqList, 'budgetBandList' => \$budgetBandList, 'formInput' => \$formInput, 'naivebayes' => \$naivebayes, </pre>		
--	---	--	--

	<pre> }); } </pre>		
2	<pre> public function index() { // 1) Keranjang & count \$cartModel = \$this->model('cart'); \$scarts = \$cartModel->findAll(session('user')); \$count = \$cartModel->total(); // 2) Produk terbaru & kategori untuk header/nav \$products = \$this->model('product')->findLast(8); \$catModel = \$this->model('category'); \$dropdowns = \$catModel->dropdownMenu(); \$treeOpts = \$catModel->selectTree(); // 3) Siapkan list kategori leaf (slug=>name) untuk form radio \$allCats = \$catModel->getAllForForm(); \$kategoriList = []; </pre>	\$naivebayes kosong; grid default “best-seller” ditampilkan	–

	<pre> foreach (\$allCats as \$c) { if ((int)\$c['parent_id'] !== 0) { \$kategoriList[\$c['slug']] = \$c['name']; } } // 4) List opsi kuisioner lain \$buyFreqList = ['Setiap minggu','Setiap bulan','Beberapa bulan','Jarang']; \$budgetBandList = ['<100k','100- 300k','300-600k','>600k']; // 5) Default rekomendasi = best-seller (lama) \$trxModel = \$this->model('transaction'); \$naivebayes = []; \$formInput = []; // 7) Render view \$this->page('home', ['home' => true, 'page' => 'home', 'carts' => \$carts, </pre>		
--	--	--	--

	<pre> 'count' => \$count, 'products' => \$products, 'dropdowns' => \$dropdowns, 'categories' => \$treeOpts, 'kategoriList' => \$kategoriList, 'buyFreqList' => \$buyFreqList, 'budgetBandList' => \$budgetBandList, 'formInput' => \$formInput, 'naivebayes' => \$naivebayes,]); } </pre>		
3	<pre> public function addCart() { if (empty(session('isUser'))) { redirect(url('login')); } else { \$model = \$this->model('cart'); \$cart = \$model- >findWhere(session('user'), post('product')); if (empty(\$cart)) { \$qty = post('qty'); </pre>	Baris baru ditulis ke tabel cart dengan qty awal (default 1)	—

	<pre> if (empty(post('qty'))) { \$qty = 1; } \$model->insert(['qty' => \$qty, 'price' => str_replace('.', '', post('price')), 'user_id' => session('user'), 'product_id' => post('product')]); } else { \$qty = post('qty'); if (empty(post('qty'))) { \$qty = intval(\$cart['qty'])+1; } } </pre>		
--	--	--	--

	<pre> \$model->update(\$cart['id'], ['qty' => post('qty')]); } } \$this->toJson(['error' => false]); } </pre>		
4	<pre> public function addCart() { if (empty(session('isUser'))) { redirect(url('login')); } else { \$model = \$this->model('cart'); \$cart = \$model- >findWhere(session('user'), post('product')); { \$qty = post('qty'); if (empty(post('qty'))) </pre>	Kolom qty baris tersebut bertambah 1	–

	<pre> { \$qty = intval(\$cart['qty'])+1; } \$model->update(\$cart['id'], ['qty' => post('qty')]); } } \$this->toJson(['error' => false]); } </pre>		
5	<pre> public function index() { if (empty(session('isUser'))) { redirect(url('login')); } else { \$model = \$this->model('cart'); </pre>	Tabel transaction bertambah 1, baris-baris cart ter-flush	—

	<pre> \$scarts = \$model- >findAll(session('user')); \$count = \$model->total(); \$categories = \$this->model('category'); if (empty(\$scarts)) { redirect(url('product')); } else { \$this->page('checkout', ['home' => false, 'page' => "", 'user' => \$this->model('user')- >findWhere('id', session('user')), 'carts' => \$scarts, 'count' => \$count, 'dropdowns' => \$categories- >dropdownMenu(), 'categories' => \$categories- >selectTree()]); } } } </pre>		
--	--	--	--

6	<pre> public function index() { if (empty(session('isUser'))) { redirect(url('login')); } else { \$model = \$this->model('cart'); \$scarts = \$model- >findAll(session('user')); \$count = \$model->total(); \$categories = \$this->model('category'); redirect(url('product')); } } </pre>	Pengguna dialihkan ke halaman /produk	—
7	<pre> public function index() { \$categories = \$this->model('category'); if (!empty(post('submit'))) { \$cred = trim(post('identity')); \$pw = post('password'); } } </pre>	Session terbuat & redirect ke beranda	—

	<pre> // hash sesuai yang Anda pakai; contoh memakai sha1(md5()) \$hash = sha1(md5(\$pw)); // dapatkan user by email OR phone \$user = \$this->model('user')- >login(\$cred, \$hash); if (!empty(\$user)) { // sukses login if (\$user === 'admin') { redirect(url('admin/dashboard')); } else { redirect(url()); } } // gagal Flasher::danger('Gagal', 'Email/Telepon atau kata sandi salah!'); } </pre>		
--	---	--	--

	<pre> \$this->page('login', ['home' => false, 'page' => 'home', 'count' => 0, 'dropdowns' => \$categories->dropdownMenu(), 'categories' => \$categories->selectTree()]); </pre>		
8	<pre> public function index() { \$categories = \$this->model('category'); if (!empty(post('submit'))) { \$cred = trim(post('identity')); \$pw = post('password'); // hash sesuai yang Anda pakai; contoh memakai sha1(md5()) \$hash = sha1(md5(\$pw)); // dapatkan user by email OR phone </pre>	Flasher menampilkan pesan <i>"Email/Telepon atau kata sandi salah"</i>	

	<pre> \$user = \$this->model('user')->login(\$cred, \$hash); // gagal Flasher::danger('Gagal', 'Email/Telepon atau kata sandi salah!'); } \$this->page('login', ['home' => false, 'page' => 'home', 'count' => 0, 'dropdowns' => \$categories->dropdownMenu(), 'categories' => \$categories->selectTree()]); } </pre>		
9	<pre> public function index() { \$categories = \$this->model('category'); if (! empty(post('submit'))) { \$data['name'] = post('name'); } } </pre>	Baris pengguna baru tersimpan; flash <i>“Pendaftaran Berhasil”</i>	—

	<pre> \$data['role'] = 'user'; \$data['email'] = post('email'); \$data['phone'] = post('phone'); \$data['birth'] = dateToSql(str_replace('/', '-', post('birth'))); \$data['gender'] = post('gender'); \$data['status'] = 1; \$data['address'] = post('address'); \$data['password'] = sha1(md5(post('password'))); if (\$this->model('user')->store(\$data)) { Flasher::success('Sukses', 'Pendaftaran Berhasil'); } else { Flasher::danger('Gagal', 'Pendaftaran gagal ditambahkan'); } } \$this->page('register', ['home' => false, 'page' => 'home', </pre>		
--	---	--	--

	<pre> 'count' => 0, 'dropdowns' => \$categories- >dropdownMenu(), 'categories' => \$categories->selectTree()]); } </pre>		
10	<pre> public function index() { if (empty(session('isUser'))) { redirect(url('login')); } else { \$model = \$this->model('cart'); \$scarts = \$model- >findAll(session('user')); \$scount = \$model->total(); \$wishlists = \$this->model('wishlist')- >findAll(session('user')); \$categories = \$this->model('category'); \$this->page('wishlist', ['home' => false, 'page' => ", </pre>	Redirect otomatis ke halaman /login	

	<pre> 'cars' => \$cars, 'count' => \$count, 'dropdowns' => \$categories- >dropdownMenu(), 'wishlists' => \$wishlists, 'categories' => \$categories- >selectTree()]); } } </pre>		
--	--	--	--

4.7.3 Perancangan Perhitungan Naïve Bayes

Sub-subbab ini mendokumentasikan alur komputasi yang benar-benar diterapkan pada dataset penelitian (1 500 rekaman interaksi pengguna). Atribut yang digunakan:

- umur_band → 20-29, 30-39, ≥ 40 tahun
- gender → Pria / Wanita
- gaya_pref → Kasual / Formal (label tujuan sekaligus kelas)

- Statistik Inti Dataset

Tabel 4.16 Statistik Inti Dataset

Kelas	Jumlah Data	Prior	20-29	30-39	≥ 40	Pria	Wanita
Kasual	894	0,596	410	310	174	520	374
Formal	606	0,404	120	240	246	310	296

Tabel 4.17 Perancangan Perhitungan Naïve Bayes

Usia	Gender	Gaya Favorit	Produk Disukai
23	Pria	Kasual	Ya
25	Pria	Kasual	Ya
30	Wanita	Formal	Tidak

b. Frekuensi Fitur Sesudah Laplace Smoothing

(Tiap sel frekuensi + 1; penyebut bertambah k = 3 untuk usia, 2 untuk gender)

Tabel 4.18 Frekuensi Fitur Sesudah Laplace Smoothing

Fitur	Kategori	P(Kategori Kasual)	P(Kategori Formal)
umur_ band	20-29	$\frac{410 + 1}{894 + 3} = 0,46$	$\frac{120 + 1}{606 + 3} = 0,20$
	30-39	0,35	0,40
	≥ 40	0,19	0,40
gender	Pria	$\frac{520 + 1}{894 + 2} = 0,58$	$\frac{310 + 1}{606 + 2} = 0,51$
	Wanita	0,42	0,49

Setiap nilai frekuensi pada Tabel 4.18 telah ditambah konstanta $\alpha = 1$ (add-one smoothing) untuk menghindari probabilitas nol. Rumus yang digunakan:

$$P(x_i | C_k) = \frac{n_{ik} + \alpha}{N_k + \alpha \cdot m}$$

dengan N_{ik} = frekuensi kategori x_i pada kelas C_k ; N_k = jumlah total data dalam kelas C_k ; m = banyaknya kategori unik atribut tersebut.

Contoh – atribut umur_band pada kelas Kasual:

$$P(30-39 \mid \text{Kasual}) = \frac{310 + 1}{894 + 1 \times 3} = 0,35.$$

c. Kasus Uji

Profil pengguna baru

Umur : 24 tahun (band 20-29) Gender : Pria

d. Perhitungan Posterior

$$P(\text{Kasual} \mid X) \propto 0,596 \times 0,46 \times 0,58 = 0,159$$

$$P(\text{Formal} \mid X) \propto 0,404 \times 0,20 \times 0,51 = 0,041$$

$$P(\text{Kasual} \mid X) > P(\text{Formal} \mid X)$$

e. Keputusan Sistem

Karena posterior tertinggi berada pada kelas Kasual, modul rekomendasi akan:

1. Menandai preferensi pengguna sebagai Kasual.
2. Mengambil 8 item tertinggi dalam daftar produk kasual (hoodie, T-shirt, sneakers, dsb.) dan menampilkannya pada section “Disarankan untuk Anda”.
3. Menyimpan label kasual ke tabel preferensi untuk memperkuat pembelajaran model selanjutnya.

4.7.4 Perancangan Pengujian Performa Model

Pengujian performa bertujuan mengevaluasi akurasi klasifikasi model Naïve Bayes dalam membedakan dua kategori rekomendasi (Kasual vs Formal). Dataset uji berjumlah 300 baris dengan 14 atribut kategorikal.

Tabel 4.19 Perancangan Pengujian Performa Model

No	Metode Pengukuran	Rumus	Kriteria Keberhasilan
1	Confusion Matrix	— (TP, TN, FP, FN)	Digunakan sebagai dasar seluruh metrik
2	Akurasi	$\frac{TP+TN}{TP+FP+FN+TN}$	$\geq 80 \%$
3	Precision	$\frac{TP}{TP+FP}$	$\geq 80 \%$
4	Recall (Sensitivity)	$\frac{TP}{TP+FN}$	$\geq 80 \%$
5	F1-Score	$2 \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$	$\geq 80 \%$
6	MAE	$(\frac{1}{n} \sum_{i=1}^n y_i - \hat{y}_i)$	$y_i - \hat{y}_i$
7	RMSE	$\sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$	$\leq 10 \%$

Jika salah satu metrik tidak memenuhi ambang keberhasilan, peneliti akan melakukan hyper-parameter tuning (mis. penyesuaian α pada Laplace smoothing) atau memperluas dataset pelatihan sebelum pengujian ulang.

4.8 Perancangan Antarmuka Pengguna

Sub-bab ini menyajikan rancangan antarmuka (mock-up) yang akan diimplementasikan pada aplikasi web. Setiap gambar dilengkapi keterangan singkat yang menjelaskan komponen antarmuka, alur interaksi, serta keterkaitannya dengan fungsi sistem yang telah dirancang pada sub-bab sebelumnya. Penomoran gambar disesuaikan secara otomatis setelah dokumen difinalkan.

4.8.1 Halaman Beranda

The mockup shows a web page layout for a home page. At the top, there is a header area with a title 'Rancangan Halaman Home' and a 'No CS' label. To the right of the header are two buttons: 'Daftar' and 'Masuk'. Below the header is a search bar with a 'Nama Toko' label, a 'Dropdown Kategori' menu, and a 'Cari Produk' button. Under the search bar is a navigation bar with three tabs: 'Beranda', 'Produk', and 'Kategori'. The main content area is a large rectangle labeled 'Gambar Keranjang Belanja' (Shopping Cart Image). At the bottom of the page is a footer area.

Gambar 4.18 Rancangan Halaman Beranda

Halaman beranda menampilkan bilah navigasi tetap (navbar) dengan menu Beranda, Produk, dan Masuk/Daftar. Di bawah tajuk utama terdapat bilah pencarian dan carousel promosi yang menyoroti koleksi unggulan. Bagian utama halaman memuat seksi “Produk Unggulan” dalam tata letak kisi, sehingga pengunjung memperoleh gambaran awal mengenai produk yang tersedia.

4.8.2

Halaman Produk Terbaru

Rancangan Halaman Produk Terbaru

Beranda Produk Kategori

Produk Terbaru

Produk

Produk

Produk

Gambar 4.19 Rancangan Halaman Produk Terbaru

Halaman ini menyajikan daftar produk fashion terbaru dalam tampilan kisi responsif. Setiap kartu produk menampilkan gambar, nama, harga, penilaian bintang, serta tombol Tambah ke Keranjang. Di bagian atas halaman tersedia kontrol penyaring (kategori) dan pengurutan (harga dan popularitas) untuk memudahkan pengguna menyesuaikan tampilan produk.

4.8.3

Formulir Pendaftaran Pengguna

Rancangan Halaman Daftar

No CS

Daftar Masuk

Nama Toko

Dropdown Kategori Cari Produk

Beranda Produk Kategori

Nama Lengkap

No Telpn

Tanggal Lahir

Jenis Kelamin

Email

Kata sandi

Alamat

Daftar

Informasi Toko

Gambar 4.20 Rancangan Halaman Formulir Pendaftaran Pengguna

Formulir pendaftaran terdiri atas kolom Nama Lengkap, Email, Kata Sandi, Nomor Telepon, dan Alamat. Validasi masukan dilaksanakan secara waktu nyata guna meminimalkan kesalahan pengetikan. Tombol Daftar

akan menginisiasi proses verifikasi surat elektronik dan menyimpan preferensi awal pengguna ke basis data.

4.8.3 Halaman Check-out

The wireframe illustrates the checkout page layout. At the top, there is a header bar with a 'Rancangan Halaman Checkout' title, a 'No CS' label, and 'Daftar' and 'Masuk' buttons. Below the header, a navigation bar includes 'Nama Toko', a 'Dropdown Kategori' menu, and a 'Cari Produk' search bar. A secondary navigation bar features 'Beranda', 'Produk', and 'Kategori' links. The main content area is divided into two panels. The left panel contains input fields for 'Nama Penerima', 'No Telpn Penerima', and 'Alamat Pengiriman'. The right panel displays a 'Total Keranjang' section with 'Sub total' and a 'Proses Transaksi' button.

Gambar 4.21 Rancangan Halaman Check-out

Halaman check-out terbagi atas dua panel. Panel kiri memuat formulir alamat pengiriman, pilihan kurir, dan catatan pesanan, sedangkan panel kanan menyajikan ringkasan biaya (subtotal, ongkos kirim, diskon, dan total). Tombol Proses Transaksi mengarahkan pengguna ke halaman konfirmasi pembayaran setelah seluruh data tervalidasi.

4.8.3

Halaman Rekomendasi Produk

Rancangan Halaman Produk Terbaru		
Beranda	Produk	Kategori
Dapatkan Rekomendasi Produk Kelompok umur Frekuensi Belanja Gender Budget per Transaksi Gaya / Kategori Favorit Tampilkan		

Gambar 4.22 Rancangan Halaman Rekomendasi Produk

Halaman rekomendasi menampilkan daftar produk hasil analisis algoritma Naïve Bayes berdasarkan profil dan riwayat transaksi pengguna. Setiap kartu produk dilengkapi lencana “Disarankan untuk Anda” dan indikator tingkat kecocokan. Pengguna dapat langsung memilih Beli Sekarang atau Tambah ke Keranjang tanpa meninggalkan halaman ini. Apabila skor kecocokan berada di bawah ambang batas tertentu, sistem menampilkan pesan informatif dan tautan untuk memperbarui preferensi.