

LAPORAN PROYEK AKHIR PRAKTIKUM
MATA KULIAH ALGORITMA DAN PEMROGRAMAN DASAR



MANAJEMEN PRODUK LAPTOP

Oleh:

Kelompok 03

DEWI ASTUTI	2409106007
DWIKI APRILIAN ARYANDA	2409106001
ZEYDAN FAZLE NAWLA	2409106010

PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MULAWARMAN
SAMARINDA
2024

KATA PENGANTAR

Puji syukur kami panjatkan ke hadirat Tuhan Yang Maha Esa atas segala rahmat dan karunia-Nya sehingga kami dapat menyelesaikan laporan penilaian akhir yang berjudul “Manajemen Produk Laptop” mata kuliah Algoritma dan Pemrograman Dasar ini tepat pada waktunya. Laporan ini disusun sebagai untuk menyelesaikan proyek akhir praktikum dan bertujuan untuk mendokumentasikan pemahaman dan keterampilan yang telah kami pelajari dalam menerapkan konsep-konsep dasar algoritma dan pemrograman.

Program yang dibuat adalah sistem manajemen data toko laptop bernama Toko Laptop Wahyu Jaya yang dirancang untuk mengelola penjualan produk laptop. Sistem ini mendukung pengelolaan data produk laptop, memungkinkan pengunjung mendaftar dan membeli produk, serta memberikan akses khusus bagi admin untuk mengelola seluruh data toko. Data disimpan dalam format CSV, sehingga aman untuk digunakan kembali setelah program dijalankan ulang.

Dalam proses pembuatan program kami terkendala Ketika dalam penggunaan csv. kami Menyadari dalam menyelesaikan penulisan laporan penilaian akhir yang berjudul “Manajemen Produk Laptop” mata kuliah Algoritma dan Pemrograman Dasar, kami banyak mendapat bantuan dari berbagai pihak, oleh karena itu kami mengucapkan banyak terima kasih kepada:

1. Bapak Muhammad Bambang Firdaus, S.kom., M.kom selaku dosen pengajar Algoritma pemrograman dasar yang telah memberikan ilmu-ilmu yang bermanfaat.
2. Bapak Awang Harsa Kridalaksana S.kom., M.kom. selaku dosen pengajar Algoritma pemrograman dasar yang telah mengajarkan dan memberikan ilmu yang bermanfaat didalam pemrograman.
3. Bang Farrel selaku Asistem Laboratorium (ASLAB) Pembimbing kami dalam mengerjakan Praktikum Akhir Mata Kuliah Algoritma Pemrograman Dasar.

4. Dan terakhir terimakasih kepada teman-teman dan seluruh pihak terkait yang sudah memberikan dukungan serta membantu kami dalam mengerjakan Praktikum Akhir Mata Kuliah Algoritma Pemrograman Dasar ini.

Demikianlah laporan akhir ini kami buat. Kami menyadari bahwa masih banyak kekurangan dalam program yang telah kami buat, namun kami berharap laporan ini dapat memberikan gambaran tentang kemampuan kami. Kami juga berharap dapat mengembangkan kemampuan kami dalam bidang pemrograman agar dapat memberikan manfaat pada dunia teknologi kedepannya.

Samarinda, 21 November 2024

Kelompok 03

TAKARIR

<i>Input</i>	Memasukkan
Digital	Digital
Efisien	Efisien
Login	Masuk
Register	Daftar
Transaksi	Transaksi
Admin	Pengelola /Administrator
Pengunjung	Pengguna biasa
CSV	Berkas data berbasis teks
File	Berkas
Stok	Persediaan
Total	Jumlah keseluruhan
User	Pengguna
Update	Memperbarui
Interface	Antarmuka
Role	Peran
Create	Tambah
Read	Tampilkan / Baca
Update	Ubah
Delete	Hapus
Rekapitulasi	Ringkasan
Validitas	Keabsahan
Table	Tabel
Password	Kata sandi
Username	Nama pengguna
Local	Lokal

DAFTAR ISI

KATA PENGANTAR	ii
TAKARIR.....	iv
DAFTAR ISI.....	v
DAFTAR GAMBAR	vi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.3 Rumusan Masalah.....	2
1.4 Batasan Masalah	3
1.5 Tujuan	4
BAB II PERANCANGAN	5
2.1 Analisis Program.....	5
2.2 Flowchart	8
2.3 Konsep/Materi Praktikum yang dipakai	15
BAB III HASIL DAN PEMBAHASAN	18
3.1 Tampilan Program	18
3.2 Source Code	25
BAB IV PENUTUP	40
4.1 Kesimpulan	40
4.2 Saran	41
DAFTAR PUSTAKA.....	43
LAMPIRAN.....	44

DAFTAR GAMBAR

Gambar 2.1 Flowchart Manajemen Produk Laptop	8
Gambar 2.2 Flowchart Menu Login	10
Gambar 2.4 Flowchart bagian menu pengunjung	13
Gambar Lampiran Cover Kertas Konsul	43
Gambar Lampiran Konsul Pertama	44
Gambar Lampiran Konsul Kedua.....	45

BAB I

PENDAHULUAN

1.1 Latar Belakang

Menjelaskan Di era digital seperti saat ini, pengelolaan data secara manual menjadi tidak efisien, terutama bagi toko-toko yang memiliki banyak produk dan pelanggan. Toko Laptop Wahyu Jaya, sebagai sebuah usaha yang menjual berbagai produk laptop, membutuhkan sistem yang dapat membantu dalam mengelola informasi secara efektif, cepat, dan akurat. Masalah sering muncul ketika toko harus menangani banyak aktivitas, seperti pencatatan produk, transaksi penjualan, dan manajemen pelanggan, yang memakan waktu jika dilakukan secara manual.

Rincian Masalah dari program diatas secara umum yang memungkinkan adanya program ini, berikut adalah rincian masalahnya;

➤ **Pengelolaan Produk:**

- Kesulitan mencatat dan mengelola informasi produk secara manual, seperti nama, merek, spesifikasi, dan harga.
- Rentan terjadi kesalahan pencatatan, seperti data produk yang hilang atau tidak sesuai.
- Tidak ada mekanisme untuk menghitung total nilai produk dengan mudah.

➤ **Manajemen Pengguna dan Transaksi:**

- Tidak adanya sistem untuk mengelola data pelanggan (pengunjung), sehingga sulit untuk melacak riwayat pembelian.
- Kesulitan mencatat transaksi pembelian dengan benar, terutama jika dilakukan dalam jumlah banyak.

- Kurangnya transparansi dalam memberikan akses yang berbeda untuk admin dan pelanggan.

➤ Efisiensi Waktu:

- Proses manual memakan waktu lebih lama untuk menambah, mengubah, atau menghapus data produk.
- Sulit untuk memberikan layanan pelanggan yang cepat karena data produk atau transaksi tidak terorganisir dengan baik.

➤ Penyimpanan Data:

- Tidak adanya sistem penyimpanan yang terstruktur untuk memastikan data produk dan pelanggan tetap aman dan dapat diakses kembali setelah toko tutup atau ketika diperlukan di kemudian hari.
- Antarmuka Sederhana untuk Pengguna:
- Kebutuhan akan sistem yang mudah digunakan, baik oleh admin maupun pelanggan, agar tidak menyulitkan dalam pengoperasiannya.

Masalah-masalah di atas menunjukkan perlunya solusi digital berupa sistem manajemen data yang sederhana, efisien, dan mudah digunakan. Program ini dirancang untuk membantu Toko Laptop Wahyu Jaya mengelola data produk, pengguna, dan transaksi dengan lebih baik, sekaligus meningkatkan pengalaman pelanggan melalui akses yang terorganisir dan transparan.

1.3 Rumusan Masalah

1. Bagaimana sistem dapat menampilkan, menambahkan, mengubah, dan menghapus produk laptop secara efektif?
2. Bagaimana sistem dapat membedakan antara pengguna dengan peran admin dan pengunjung melalui mekanisme login?
3. Bagaimana admin dapat memantau data pengguna yang terdaftar, termasuk daftar produk yang telah dibeli?

4. Bagaimana pengunjung dapat membeli produk laptop dan secara otomatis menghapus produk tersebut dari daftar?
5. Bagaimana cara menghitung total harga semua produk secara otomatis untuk membantu admin melakukan rekapitulasi?
6. Bagaimana memastikan validitas input pengguna, seperti format harga, nama produk, dan informasi login?
7. Bagaimana menangani input yang tidak valid tanpa menghentikan program?

1.4 Batasan Masalah

1. Sistem hanya menangani fungsi CRUD (Create, Read, Update, Delete) untuk data produk laptop dan pengguna.
2. Proses transaksi pembelian hanya mencakup pembelian produk tanpa fitur pembayaran online.
3. Data hanya disimpan dalam file CSV, sehingga tidak mendukung basis data relasional (RDBMS) untuk pengelolaan data yang lebih kompleks.
4. Sistem tidak mendukung multi-user secara real-time; hanya satu sesi login yang aktif dalam satu waktu.
5. Fitur penambahan, penghapusan, dan pengubahan produk hanya tersedia untuk admin.
6. Pengunjung hanya dapat membeli produk dan melihat daftar produk.
7. Sistem tidak menggunakan enkripsi untuk menyimpan password.
8. Tidak ada mekanisme otentikasi lanjutan, seperti OTP atau verifikasi email.
9. Sistem hanya dirancang untuk penggunaan lokal dan tidak dapat diakses secara online.
10. Tidak mendukung pengelolaan data dalam jumlah besar karena keterbatasan file CSV.

1.5 Tujuan

1. Membangun sistem yang mampu menangani proses CRUD (Create, Read, Update, Delete) produk laptop secara efektif, cepat, dan terorganisir.
2. Memastikan data produk laptop dapat disimpan secara permanen dan mudah diakses melalui file CSV.
3. Memberikan kemudahan kepada pengunjung dalam melihat, memilih, dan membeli produk laptop.
4. Memastikan tampilan antarmuka interaktif yang ramah pengguna dengan animasi dan format tabel yang informatif.
5. Mengimplementasikan sistem login yang dapat membedakan peran admin dan pengunjung, sehingga masing-masing memiliki hak akses yang sesuai.
6. Memberikan fitur pendaftaran bagi pengunjung agar dapat menjadi pengguna resmi sistem.
7. Memberikan admin kemampuan untuk memantau daftar pengguna, termasuk informasi terkait produk yang dibeli.
8. Menghadirkan fungsi perhitungan total harga untuk membantu admin dalam proses rekapitulasi data.
9. Memastikan input data, seperti nama produk, harga, dan informasi login, terverifikasi dengan baik untuk menghindari kesalahan dan menjaga konsistensi data.
10. Memberikan keamanan dasar pada akses sistem dengan autentikasi melalui username dan password.
11. Menyediakan fitur pembelian yang langsung menghapus produk dari daftar tersedia, sehingga stok tetap akurat.
12. Mempermudah pengguna dalam memilih produk dengan tampilan data yang lengkap dan terstruktur.
13. Menyimpan semua data, termasuk produk, pengguna, dan transaksi, dalam file CSV yang dapat diakses dan diperbarui kapan saja.

BAB II

PERANCANGAN

2.1 Analisis Program

Program "Toko Laptop Wahyu Jaya" adalah sebuah program untuk mengelola produk laptop. Program ini memungkinkan pengguna untuk login, mendaftar, mengelola produk (khusus admin), dan membeli produk (pengunjung). Data produk dan pengguna disimpan di file CSV agar tetap konsisten.

Fungsi utama meliputi:

1. Manajemen Produk: Menambah, mengubah, menghapus, dan melihat daftar produk.
2. Sistem Login: Membatasi hak akses sesuai peran (admin atau pengunjung).
3. Transaksi: Pengunjung dapat membeli produk, mengurangi stok produk secara otomatis.
4. Laporan: Menghitung total harga produk dan melihat data pengunjung.

Berikut adalah Penjelasan Alur Program secara singkat ;

1. Tampilan Menu Awal

- Program dimulai dengan menampilkan menu utama berupa tiga pilihan:
 1. Login
 2. Register (untuk pengunjung baru)
 3. Keluar

2. Sistem Login

- Pengguna memasukkan username dan password.
- Jika login berhasil:
 - Admin diarahkan ke menu untuk mengelola produk.
 - Pengunjung diarahkan ke menu pembelian.
- Jika gagal, pengguna diminta mencoba lagi.

3. Menu Admin

Admin memiliki akses ke fitur berikut:

1. Tampilkan Produk: Menampilkan semua produk dalam format tabel interaktif.
2. Tambah Produk: Menambahkan produk baru ke daftar.
3. Ubah Produk: Mengubah informasi produk tertentu.
4. Hapus Produk: Menghapus produk dari daftar.
5. Tampilkan Total Harga: Menghitung dan menampilkan total harga semua produk.
6. Tampilkan Data Pengunjung: Menampilkan daftar pengunjung yang terdaftar dan produk yang telah mereka beli.
7. Keluar: Kembali ke menu awal.

4. Menu Pengunjung

Pengunjung memiliki akses ke fitur berikut:

1. Tampilkan Produk: Menampilkan daftar produk yang tersedia.
2. Beli Produk: Memilih produk untuk dibeli; produk tersebut dihapus dari daftar.
3. Tampilkan Total Harga: Menghitung total harga semua produk (informasi umum).
4. Keluar: Kembali ke menu awal.

5. Pendaftaran Pengunjung

- Pengguna baru dapat membuat akun dengan memasukkan username dan password.
- Setelah berhasil mendaftar, pengguna dapat langsung login menggunakan akun tersebut.

6. Sistem Penyimpanan Data

- Semua data produk disimpan dalam file data_produk.csv.
- Data pengguna juga tersimpan di memori program dan dapat ditampilkan untuk admin.

7. Proses Transaksi

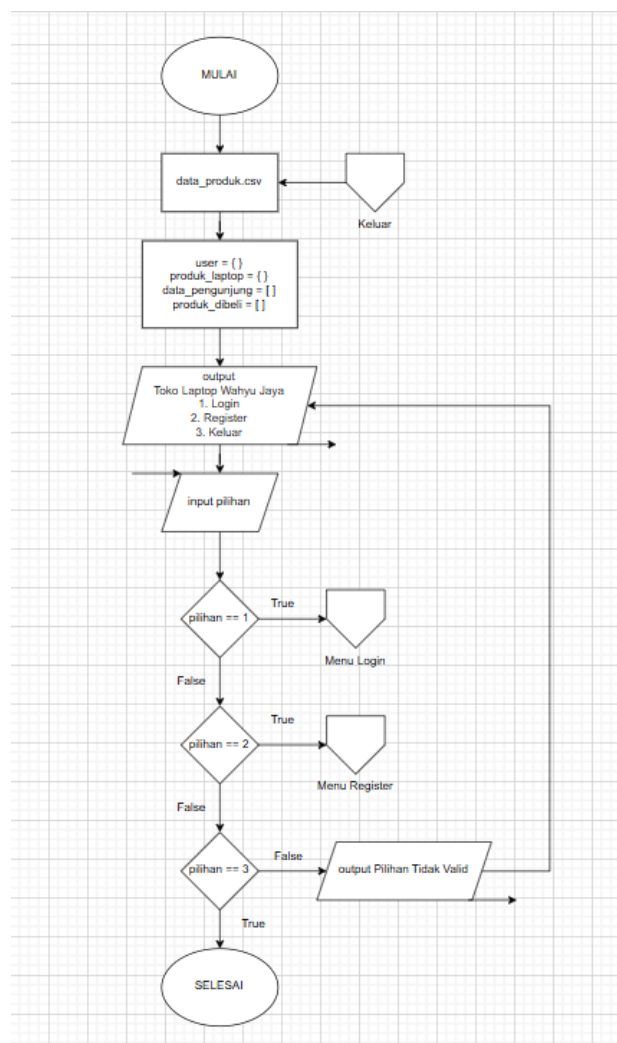
- Pengunjung membeli produk dengan memilih dari daftar.
- Produk yang dibeli langsung dihapus dari daftar produk.

8. Keluar dari Program

- Program menutup sesi dengan pesan terima kasih.

2.2 Flowchart

Flowchart adalah diagram yang digunakan untuk memvisualisasikan alur kerja atau proses secara sistematis dan logis. Flowchart menggunakan simbol-simbol standar, seperti oval, persegi panjang, dan panah, untuk merepresentasikan langkah-langkah dalam suatu proses. Berikut ini merupakan flowchart dari program yang berjudul “Manajemen Produk Laptop”;



Gambar 2.1 Flowchart Manajemen Produk Laptop

Berikut adalah alur dari flowchart yang telah kami buat;

Start

1. Sistem membaca data produk dari file data_produk.csv.
2. Variabel berikut diinisialisasi:
 - user (data pengguna).
 - produk_laptop (data produk laptop).
 - data_pengunjung (data pengunjung toko).
 - produk_dibeli (produk yang telah dibeli pengguna).

3. Tampilan Menu Utama

Sistem menampilkan pilihan menu kepada pengguna:

1. Login
2. Register
3. Keluar

4. Input Pilihan Pengguna

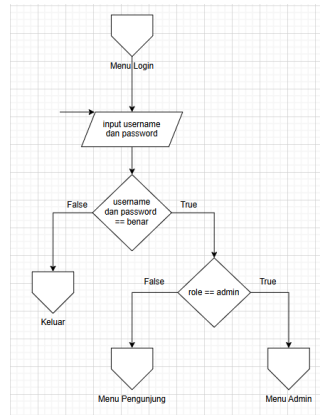
Pengguna diminta memilih salah satu opsi.

5. Logika Pilihan

- Jika Pilihan == 1 (Login):
Proses diarahkan ke Menu Login.
- Jika Pilihan == 2 (Register):
Proses diarahkan ke Menu Register.
- Jika Pilihan == 3 (Keluar):
Sistem akan keluar dan proses berhenti.
- Jika Pilihan Lain:
Sistem memberikan output "Pilihan Tidak Valid" dan kembali ke menu awal.

6. Selesai

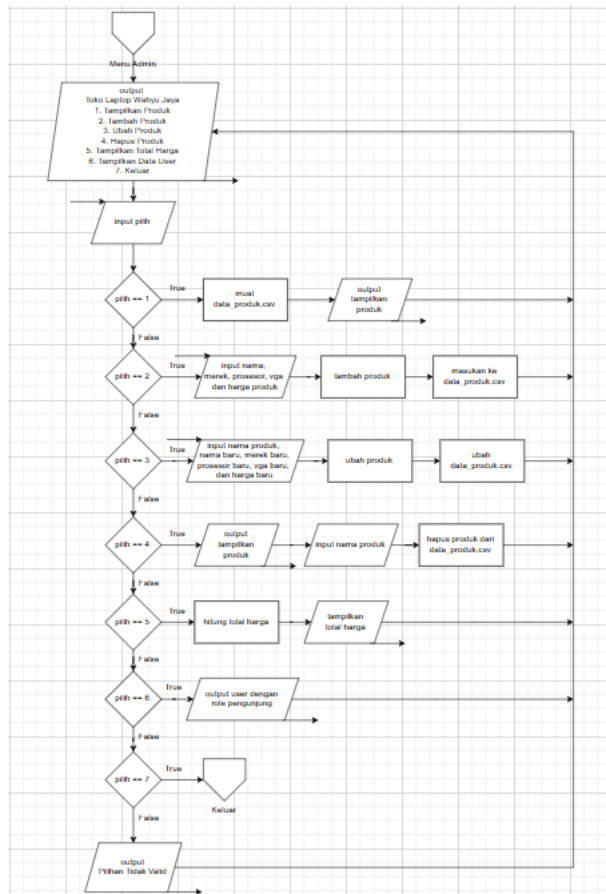
Jika pengguna memilih untuk keluar, sistem berhenti di node "SELESAI."



Gambar 2.2 Flowchart Menu Login

Berikut adalah penjelasan Flowchart pada menu login yaitu;

1. Masuk ke Menu Login
2. Input Username dan Password
3. Validasi Username dan Password
 - Sistem memeriksa apakah kombinasi username dan password yang dimasukkan benar:
 - Jika salah (False):
Sistem keluar dari proses login, ditandai dengan node "Keluar."
 - Jika benar (True):
Sistem melanjutkan proses ke langkah berikutnya.
4. Pengecekan Role Pengguna
 - Sistem memeriksa peran (role) pengguna:
Jika role == admin:
 - Pengguna diarahkan ke Menu Admin.
Jika role != admin:
 - Pengguna diarahkan ke Menu Pengunjung.
5. Akhir Proses



Gambar 2.3 Flowchart bagian Menu Admin

Berikut Adalah Penjelasan flowchart bagian Menu admin berikut;

1. Menu Admin
2. Sistem menampilkan menu admin dengan beberapa opsi:
 1. Lihat Produk
 2. Tambah Produk
 3. Ubah Produk
 4. Hapus Produk
 5. Tampilkan Total Harga
 6. Keluar

3. Input Pilihan
4. Pengguna diminta untuk memilih salah satu opsi yang ditampilkan pada menu.
5. Memilih menu-menu yang ada

Sistem memproses input sesuai dengan pilihan yang diberikan:

1. Pilihan 1: Lihat Produk

Sistem memuat data dari file data_produk.csv.

Sistem menampilkan daftar produk kepada pengguna.

2. Pilihan 2: Tambah Produk

Pengguna diminta untuk memasukkan:

- Nama produk
- Merek produk
- Varian produk
- Harga produk
- Data yang dimasukkan ditambahkan ke file data_produk.csv.

3. Pilihan 3: Ubah Produk

- Pengguna diminta untuk memasukkan:
- Nama produk yang ingin diubah.
- Merek baru.
- Varian baru.
- Harga baru.
- Sistem memperbarui data produk yang sesuai di file data_produk.csv.

4. Pilihan 4: Hapus Produk

Pengguna diminta untuk memasukkan nama produk yang ingin dihapus.

Sistem menghapus produk yang sesuai dari file data_produk.csv.

5. Pilihan 5: Tampilkan Total Harga

Sistem menghitung total harga dari semua produk yang tersedia di file data_produk.csv.

Sistem menampilkan total harga kepada pengguna.

6. Pilihan 6: Keluar

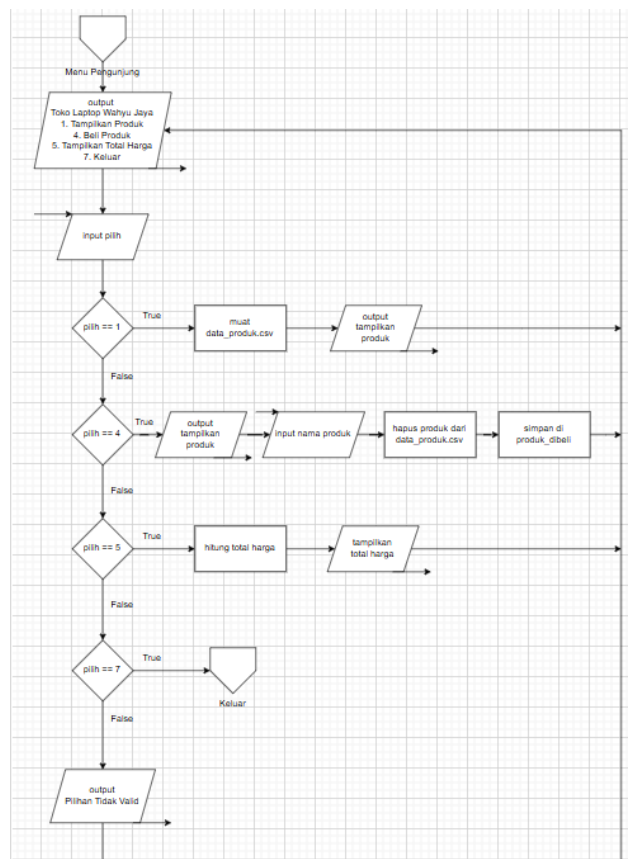
Sistem mengakhiri proses dan keluar dari menu admin.

6. Pilihan Tidak Valid

7. Jika pengguna memasukkan pilihan yang tidak sesuai dengan menu (di luar 1-6), sistem memberikan output "Pilihan Tidak Valid" dan kembali ke menu utama.

8. Selesai

Proses berakhir ketika pengguna memilih opsi keluar atau menyelesaikan seluruh aktivitas.



Gambar 2.4 Flowchart bagian menu pengunjung

1. Menu Utama
2. Program dimulai dengan menampilkan menu utama kepada pengguna. Pilihan yang tersedia adalah:
 - 1: Tampilkan Produk
 - 4: Beli Produk
 - 5: Tampilkan Total Harga
 - 7: Keluar
3. Input Pilihan Pengguna

Pengguna diminta untuk memilih salah satu opsi dari menu utama. Sistem akan mengevaluasi input pengguna berdasarkan pilihan berikut:

 1. Pilihan 1: Tampilkan Produk
 - Sistem memuat data dari file data_produk.csv.
 - Data produk akan ditampilkan kepada pengguna.
 - Setelah itu, sistem kembali ke menu utama.
 2. Pilihan 4: Beli Produk
 - Sistem akan menampilkan daftar produk yang tersedia.
 - Pengguna diminta untuk memasukkan nama produk yang ingin dibeli.
 - Sistem akan menghapus data produk tersebut dari file data_produk.csv.
 - Data produk yang dibeli disimpan di file atau variabel bernama produk_dibeli.
 - Setelah selesai, sistem kembali ke menu utama.
 3. Pilihan 5: Tampilkan Total Harga
 - Sistem menghitung total harga dari produk-produk yang telah dibeli.
 - Total harga akan ditampilkan kepada pengguna.
 - Setelah itu, sistem kembali ke menu utama.

4. Pilihan 7: Keluar

Program akan keluar jika pengguna memilih opsi ini.

5. Pilihan Tidak Valid

Jika pengguna memasukkan input yang tidak sesuai dengan pilihan di menu utama, sistem akan menampilkan pesan "Pilihan Tidak Valid" dan kembali ke menu utama.

6. Selesai flowchart Kembali ke menu utama

2.3 Konsep/Materi Praktikum yang dipakai

Konsep yang kami gunakan dalam pembuatan laporan ini sesuai dengan persyaratan yang terdapat pada modul diantaranya:

1. Fungsi Dasar:

- Fungsi print: Fungsi ini digunakan untuk menampilkan informasi atau keluaran kepada pengguna melalui terminal/console. Program menggunakan print untuk menampilkan menu, informasi produk, kesalahan input, dan status tindakan yang telah dilakukan (seperti "Produk berhasil ditambahkan", "Produk berhasil diubah", dan lainnya). Contohnya, `print(f"Produk {nama_produk} berhasil ditambahkan.")` digunakan untuk memberi tahu pengguna bahwa produk baru telah berhasil ditambahkan.
- Fungsi input: Fungsi ini digunakan untuk menerima input dari pengguna. Input yang diterima digunakan untuk berbagai kebutuhan, seperti login, memilih menu, dan memasukkan data produk. Misalnya, pada pilihan = `input(BOLD+"Masukan Pilihan : "+RESET)`, input digunakan untuk menerima pilihan menu dari pengguna.

2. Struktur Kontrol:

- Percabangan (if-else): Digunakan untuk mengecek kondisi dan mengeksekusi aksi sesuai dengan hasil pengecekan. Salah satu contoh penerapannya adalah di

dalam fungsi login, di mana program mengecek apakah username dan password yang dimasukkan cocok dengan data yang tersimpan. Contoh lain adalah di menu utama, di mana percabangan memeriksa peran pengguna untuk menampilkan pilihan yang sesuai.

- Perulangan (while-for): Struktur perulangan digunakan untuk meminta input dari pengguna berulang kali hingga kondisi tertentu tercapai. Misalnya, pada saat meminta input produk yang ingin dibeli atau produk yang ingin diubah, program akan terus meminta input hingga pengguna memberikan input yang valid.

3. Penyimpanan Data:

- Program ini menggunakan file CSV untuk menyimpan data produk dan data pengguna. Dengan menggunakan modul csv, data produk dapat dimuat dan disimpan dari dan ke dalam file data_produk.csv. Fungsi muat_data_csv membaca file CSV untuk memuat data produk, dan fungsi simpan_data_csv digunakan untuk menyimpan data produk yang telah diubah atau ditambah.

4. Pemrograman:

- Meskipun program ini tidak secara eksplisit menggunakan pemrograman asinkron dengan modul seperti asyncio, terdapat penggunaan fungsi time.sleep() untuk menunda eksekusi agar efek animasi seperti loading atau output teks lebih halus dan memberikan waktu bagi pengguna untuk membaca. Misalnya, fungsi loading() menampilkan animasi loading menggunakan time.sleep():

5. Pemrograman Berbasis Objek dan Modular:

- Program ini terstruktur secara modular dengan fungsi-fungsi yang masing-masing menangani operasi tertentu seperti login, tambah produk, ubah produk, hapus produk, dan beli produk. Fungsi-fungsi ini saling terpisah dan dapat

dipanggil sesuai dengan kebutuhan untuk mengelola produk atau pengguna. Misalnya, ada fungsi `tambah_produk()` yang khusus menangani penambahan produk baru, dan ada pula fungsi `ubah_produk()` yang mengubah data produk yang sudah ada.

6. Penanganan Kesalahan:

- Program menangani kesalahan dengan memeriksa input dari pengguna dan memberikan umpan balik jika input tidak valid. Sebagai contoh, di dalam fungsi `tambah_produk()`, jika pengguna tidak memasukkan harga berupa angka, program akan meminta pengguna untuk memasukkan harga yang valid:

7. Penggunaan Modul dan Library Eksternal:

- `csv`: Modul ini digunakan untuk membaca dan menulis data ke file CSV, yang menyimpan data produk dan informasi pengguna.
- `tabulate`: Digunakan untuk menampilkan data dalam format tabel yang lebih rapi di terminal, memudahkan pembacaan daftar produk dan informasi pengguna.
- `os` dan `sys`: Modul-modul ini digunakan untuk pengelolaan file dan direktori serta untuk mengatur tampilan dan manipulasi terminal. Fungsi `clear()` digunakan untuk membersihkan layar terminal dan memberikan pengalaman interaktif yang lebih baik bagi pengguna.

BAB III

HASIL DAN PEMBAHASAN

3.1 Tampilan Program

1. Tampilan Menu

Pada bagian ini ada 3 menu yang pertama jika anda sudah mempunyai akun dan anda adalah seorang admin anda dapat langsung memilih menu 1, jika anda pengunjung maka anda harus melakukan registrasi terlebih dahulu, dan menu 3 ada keluar yang Dimana disini jika anda sudah tidak memakai programnya anda bisa memilih no 3.

```
=====
|   Toko Laptop Wahyu Jaya   |
=====
|  1. Login                  |
|  2. Register               |
|  3. Keluar                 |
=====
Masukan Pilihan :
```

2. Login Admin

1. Pertama masukkan user admin yang sudah ada sebelumnya

```
Masukkan Username : admin
Masukkan Password : admin
Masuk berhasil! Selamat datang, admin (Role: admin)
Tekan Enter Untuk Melanjutkan...
```

2. Setelah itu akan muncul 7 menu yang dapat anda pilih dan dapat anda ubah sebagai admin

```
=====
|   Toko Laptop Wahyu Jaya   |
=====
|  1. Tampilkan Produk      |
|  2. Tambah Produk         |
|  3. Ubah Produk           |
|  4. Hapus Produk          |
|  5. Tampilkan Total Harga  |
|  6. Tampilkan Data User    |
|  7. Keluar                |
=====
Masukan Pilihan :
```


3. Setelah itu masukkan pilihan 1 pada menu ini akan menampilkan produk-produk yang dijual di dalam toko laptop wahyu jaya

No	Nama	Merek	Prosesor	VGA	Harga
1	ROG STRIX	ASUS	Intel i7	RTX 3060	Rp20000000
2	LEGION	LENOVO	AMD Ryzen 6	RTX 4060	Rp25000000
3	PREDATOR	ACER	Intel i9	RTX 3080	Rp30000000
4	OMEN	HP	Intel i7	RTX 3070	Rp28000000
5	INSPIRON	DELL	Intel i5	GTX 1660	Rp12000000
6	MACBOOK AIR	APPLE	Apple M1	Integrated	Rp16000000
7	MACBOOK PRO	APPLE	Apple M2	Integrated	Rp30000000
8	VIVOBOK	ASUS	Intel i3	Intel UHD	Rp8000000

Tekan Enter Untuk Melanjutkan...

4. Pilihan 2 disini kita dapat menambahkan produk sebagai role admin

```
=====
|   Toko Laptop Wahyu Jaya   |
=====
| 1. Tampilkan Produk        |
| 2. Tambah Produk           |
| 3. Ubah Produk             |
| 4. Hapus Produk            |
| 5. Tampilkan Total Harga    |
| 6. Tampilkan Data User      |
| 7. Keluar                   |
=====
Masukan Pilihan : 2
```

Output program akan muncul seperti dibawah ini

```
Masukan Nama Laptop Yang Ingin Ditambah: GAMING ASUS
Masukan Merek Laptop : ASUS
Masukan Prosesor Laptop : 1
Masukan VGA Laptop : 2
Masukan Harga : 19000000
Produk 'GAMING ASUS' Berhasil Ditambahkan!
Tekan Enter Untuk Melanjutkan...
```

No	Nama	Merek	Prosesor	VGA	Harga
1	ROG STRIX	ASUS	Intel i7	RTX 3060	Rp20000000
2	LEGION	LENOVO	AMD Ryzen 6	RTX 4060	Rp25000000
3	PREDATOR	ACER	Intel i9	RTX 3080	Rp30000000
4	OMEN	HP	Intel i7	RTX 3070	Rp28000000
5	INSPIRON	DELL	Intel i5	GTX 1660	Rp12000000
6	MACBOOK AIR	APPLE	Apple M1	Integrated	Rp16000000
7	MACBOOK PRO	APPLE	Apple M2	Integrated	Rp30000000
8	VIVOBOK	ASUS	Intel i3	Intel UHD	Rp8000000
9	GAMING ASUS	ASUS	1	2	Rp19000000

5. Setelah itu bagian 3 ada ubah produk yang Dimana disini kita dapat memperbaiki atau mengubah harga produk

```
=====
|   Toko Laptop Wahyu Jaya   |
=====
| 1. Tampilkan Produk        |
| 2. Tambah Produk           |
| 3. Ubah Produk             |
| 4. Hapus Produk            |
| 5. Tampilkan Total Harga    |
| 6. Tampilkan Data User      |
| 7. Keluar                  |
=====
Masukan Pilihan : 3
```

Berikut adalah output dari program diatas

```
Masukkan Nomor Produk yang ingin diubah: 9
Masukan Nama Baru : ASUS GAMING
Masukan Merek Baru : ASUS
Masukan Prosesor Baru : 2
Masukan VGA Baru : 2
Masukan Harga Baru : 20000000
Produk Berhasil Diubah!
Tekan Enter Untuk Melanjutkan...
```

No	Nama	Merek	Prosesor	VGA	Harga
1	ROG STRIX	ASUS	Intel i7	RTX 3060	Rp20000000
2	LEGION	LENOVO	AMD Ryzen 6	RTX 4060	Rp25000000
3	PREDATOR	ACER	Intel i9	RTX 3080	Rp30000000
4	OMEN	HP	Intel i7	RTX 3070	Rp28000000
5	INSPIRON	DELL	Intel i5	GTX 1660	Rp12000000
6	MACBOOK AIR	APPLE	Apple M1	Integrated	Rp16000000
7	MACBOOK PRO	APPLE	Apple M2	Integrated	Rp30000000
8	VIVOBOK	ASUS	Intel i3	Intel UHD	Rp8000000
9	ASUS GAMING	ASUS	2	2	Rp20000000

6. Setelah itu ada bagian menu 4 yaitu hapus produk

```
=====
|   Toko Laptop Wahyu Jaya   |
=====
| 1. Tampilkan Produk        |
| 2. Tambah Produk           |
| 3. Ubah Produk             |
| 4. Hapus Produk            |
| 5. Tampilkan Total Harga    |
| 6. Tampilkan Data User     |
| 7. Keluar                  |
=====
Masukan Pilihan : 4
```

```
Masukkan Nomor Produk yang ingin dihapus: 9
Produk dengan nomor 9 berhasil dihapus.
Tekan Enter Untuk Melanjutkan...
```

No	Nama	Merek	Prosesor	VGA	Harga
1	ROG STRIX	ASUS	Intel i7	RTX 3060	Rp20000000
2	LEGION	LENOVO	AMD Ryzen 6	RTX 4060	Rp25000000
3	PREDATOR	ACER	Intel i9	RTX 3080	Rp30000000
4	OMEN	HP	Intel i7	RTX 3070	Rp28000000
5	INSPIRON	DELL	Intel i5	GTX 1660	Rp12000000
6	MACBOOK AIR	APPLE	Apple M1	Integrated	Rp16000000
7	MACBOOK PRO	APPLE	Apple M2	Integrated	Rp30000000
8	VIVOBOK	ASUS	Intel i3	Intel UHD	Rp8000000

7. Setelah itu ada menu bagian 5 yaitu menampilkan total harga

```
Total Harga Semua Produk: Rp169000000
Tekan Enter Untuk Melanjutkan...
```

8. Setelah itu ada bagian data user yang Dimana disini akan terdata siapa saja pengunjung yang telah dating atau telah melakukan registrasi sehingga sebagai admin dapat merekap opengunjung ditiap bulannya.

```
=====
| Toko Laptop Wahyu Jaya |
=====
| 1. Tampilkan Produk |
| 2. Tambah Produk |
| 3. Ubah Produk |
| 4. Hapus Produk |
| 5. Tampilkan Total Harga |
| 6. Tampilkan Data User |
| 7. Keluar |
=====
Masukan Pilihan : 6
```

9. Terakhir ada menu keluar yang Dimana Ketika kita memilih menu keluar ini akan Kembali ke menu utama

```
=====
| Toko Laptop Wahyu Jaya |
=====
| 1. Tampilkan Produk |
| 2. Tambah Produk |
| 3. Ubah Produk |
| 4. Hapus Produk |
| 5. Tampilkan Total Harga |
| 6. Tampilkan Data User |
| 7. Keluar |
=====
Masukan Pilihan : 7
```

```
Terima Kasih Telah Melihat Produk Kami
Loading ...
```

3. Login User
 1. Pertama sebagai user atau pengunjung harus melakukan registrasi terlebih dahulu

```

=====
|   Toko Laptop Wahyu Jaya   |
=====
| 1. Login                   |
| 2. Register                |
| 3. Keluar                  |
=====
Masukan Pilihan : 2

```

- Setelah memilih menu 2 kita akan membuat username dan password yang baru agar kita dapat masuk sebagai pengunjung

```

Masukkan Username Baru : dewi
Masukkan Password Baru : dewi
Pendaftaran Berhasil Untuk dewi Dengan Role pengunjung!
Tekan Enter Untuk Melanjutkan...

```

- Setelah itu kita Kembali ke menu utama dan memilih login menggunakan user yang sudah kita buat tadi

```

Masukkan Username : dewi
Masukkan Password : dewi
Masuk berhasil! Selamat datang, dewi (Role: pengunjung)
Tekan Enter Untuk Melanjutkan...

```

- Pada menu user atau pengunjung hanya terdapat 4 menu yang pertama menu tampilkan produk. Dimana produk milik toko laptop wahyu jaya akan keluar agar pengunjung dapat melihat apa saja laptop yang ada ditoko tersebut.

```

=====
|   Toko Laptop Wahyu Jaya   |
=====
| 1. Tampilkan Produk        |
| 2. Beli Produk             |
| 3. Tampilkan Total Harga   |
| 4. Keluar                  |
=====
Masukan Pilihan : 1

```

Output program yang dihasilkan

No	Nama	Merek	Prosesor	VGA	Harga
1	ROG STRIX	ASUS	Intel i7	RTX 3060	Rp20000000
2	LEGION	LENOVO	AMD Ryzen 6	RTX 4060	Rp25000000
3	PREDATOR	ACER	Intel i9	RTX 3080	Rp30000000
4	OMEN	HP	Intel i7	RTX 3070	Rp28000000
5	INSPIRON	DELL	Intel i5	GTX 1660	Rp12000000
6	MACBOOK AIR	APPLE	Apple M1	Integrated	Rp16000000
7	MACBOOK PRO	APPLE	Apple M2	Integrated	Rp30000000
8	VIVOBOOK	ASUS	Intel i3	Intel UHD	Rp8000000

5. Pada menu kedua disini ada menu beli produk yang Dimana pengunjung dapat memilih produk yang ingin dibeli.

```

=====
|   Toko Laptop Wahyu Jaya   |
=====
| 1. Tampilkan Produk        |
| 2. Beli Produk             |
| 3. Tampilkan Total Harga   |
| 4. Keluar                  |
=====
Masukan Pilihan : 2

```

Ouput tampilan program dari menu ke 2

```

=====
| No | Nama       | Merek | Prosesor | VGA      | Harga      |
=====
| 1  | ROG STRIX  | ASUS  | Intel i7 | RTX 3060 | Rp20000000 |
| 2  | LEGION     | LENOVO | AMD Ryzen 6 | RTX 4060 | Rp25000000 |
| 3  | PREDATOR   | ACER   | Intel i9 | RTX 3080 | Rp30000000 |
| 4  | OMEN       | HP     | Intel i7 | RTX 3070 | Rp28000000 |
| 5  | INSPIRON   | DELL   | Intel i5 | GTX 1660 | Rp12000000 |
| 6  | MACBOOK AIR | APPLE | Apple M1 | Integrated | Rp16000000 |
| 7  | MACBOOK PRO | APPLE | Apple M2 | Integrated | Rp30000000 |
| 8  | VIVOBOOK  | ASUS   | Intel i3 | Intel UHD | Rp8000000  |
=====
Masukkan Nomor Produk yang ingin dibeli: 1
Produk 'ROG STRIX' berhasil dibeli.
Tekan Enter Untuk Melanjutkan...

```

- Setelah itu ada menu tampilkan total harga dari produk yang dibeli oleh pengunjung

```
=====
|   Toko Laptop Wahyu Jaya   |
=====
| 1. Tampilkan Produk         |
| 2. Beli Produk              |
| 3. Tampilkan Total Harga    |
| 4. Keluar                   |
=====
Masukan Pilihan : 3
```

Output tampilan program dari menu ke 3

```
Total Harga Produk yang Anda Beli: Rp20000000
Tekan Enter Untuk Melanjutkan...
```

- Terakhir ada menu keluar Dimana disini user atau pengunjung akan Kembali ke menu utama

```
Terima Kasih Telah Melihat Produk Kami
Loading ...
```

- Setelah itu terakhir ada menu keluar Dimana disini jika kita memilih menu keluar maka kita akan keluar dari program ini.

```
=====
|   Toko Laptop Wahyu Jaya   |
=====
| 1. Login                    |
| 2. Register                 |
| 3. Keluar                   |
=====
Masukan Pilihan :
```

```
Terima Kasih Telah Berkunjung.
Loading .
```

3.2 Source Code

```
import csv
from tabulate import tabulate
import os
```

```

import time
import sys

#Warna
BOLD = '\033[1m'
RED = '\033[31m'
GREEN = '\033[32m'
MAGENTA = '\033[35m'
BRIGHT_YELLOW = '\033[93m'
BRIGHT_BLUE = '\033[94m'
RESET = '\033[0m'

# CRUD Manajemen Produk Laptop

user = {
    "admin": {"password": "admin", "role": "admin"}
}

produk_laptop = {}
data_pengunjung = []
produk_dibeli = []

def clear():
    os.system("cls" if os.name == "nt" else "clear")

def animasi_output(teks, delay=0.0000001):
    for huruf in teks:
        sys.stdout.write(huruf)
        sys.stdout.flush()
        time.sleep(delay)
    print()

def loading(duration=5):

```



```

animation = [
    BRIGHT_BLUE+"Loading . ",
    "Loading .. ",
    "Loading ... ",
    "Loading    "+RESET
]

start_time = time.time()
while time.time() - start_time < duration:
    for frame in animation:
        sys.stdout.write("\r" + frame)
        sys.stdout.flush()
        time.sleep(0.3)

# Fungsi Rekursif Menghitung Total Harga
def hitung_total_harga(index=0, total=0):
    if index >= len(produk_laptop):
        return total
    harga = list(produk_laptop.values())[index]['harga']
    return hitung_total_harga(index + 1, total + int(harga))

# Fungsi login
def login(username, password):
    if username in user and user[username]['password'] == password:
        print(f"Masuk berhasil! Selamat datang, {username} (Role: {user[username]['role']})")
        return user[username]['role']
    else:
        print("Username atau Password Salah!")
        return None

# Fungsi Tampilkan Produk
def tampilkan_produk():
    if len(produk_laptop) == 0:

```

```

        print("Produk Kosong!")
    else:
        tabel_data = [[no, data["nama"], data['merek'], data['prosesor'], data['vga'],
f"Rp{ data['harga']}"]]
        for no, data in produk_laptop.items()]
        print(tabulate(tabel_data, headers=["No", "Nama", "Merek", "Prosesor", "VGA",
"Harga"], tablefmt="grid"))

# Prosedur Untuk Menambah Produk
def tambah_produk():
    nama_produk = input(BOLD + "\nMasukan Nama Laptop Yang Ingin Ditambah: " +
RESET)
    if not nama_produk.strip():
        print(RED + "Input Tidak Boleh Kosong" + RESET)
        return
    nama_produk = nama_produk.upper()

    for produk in produk_laptop.values():
        if produk["nama"] == nama_produk:
            print(RED + f"Produk Dengan Nama '{nama_produk}' Sudah Ada!" + RESET)
            return

    merek = input(BOLD + "Masukan Merek Laptop : " + RESET)
    if not merek.strip():
        print(RED + "Input Tidak Boleh Kosong" + RESET)
        return

    prosesor = input(BOLD + "Masukan Prosesor Laptop : " + RESET)
    if not prosesor.strip():
        print(RED + "Input Tidak Boleh Kosong" + RESET)
        return

```

```

vga = input(BOLD + "Masukan VGA Laptop : " + RESET)
if not vga.strip():
    print(RED + "Input Tidak Boleh Kosong" + RESET)
    return

try:
    harga = int(input(BOLD + "Masukan Harga : " + RESET))
    if harga <= 0:
        print(RED + "Harga Tidak Boleh Minus atau 0" + RESET)
        return
except ValueError:
    print(RED + "Harga Harus Berupa Angka!" + RESET)
    return

# Tambahkan produk ke dictionary
produk_laptop[len(produk_laptop) + 1] = {
    "nama": nama_produk,
    "merek": merek,
    "prosesor": prosesor,
    "vga": vga,
    "harga": harga,
}
simpan_data_csv()
print(GREEN + f"Produk '{nama_produk}' Berhasil Ditambahkan!" + RESET)

# Prosedur Untuk Mengubah Produk
def ubah_produk():
    tampilkan_produk()
    try:
        nomor = int(input(BOLD+"Masukkan Nomor Produk yang ingin diubah: "+RESET))
        if nomor in produk_laptop:
            produk = produk_laptop[nomor]

```

```

nama_baru = input(BOLD+"Masukan Nama Baru : "+RESET) or produk["nama"]

for produk in produk_laptop.values():
    if produk["nama"] == nama_baru.upper():
        print(RED + f"Produk Dengan Nama '{nama_baru}' Sudah Ada!" + RESET)
        return

if not nama_baru.strip():
    print(RED+ 'Input Tidak Boleh Kosong'+RESET)
    return

merek_baru = input(BOLD+"Masukan Merek Baru : "+RESET) or produk["merek"]
if not merek_baru.strip():
    print(RED+ 'Input Tidak Boleh Kosong'+RESET)
    return

prosesor_baru = input(BOLD+"Masukan Prosesor Baru : "+RESET) or
produk["prosesor"]
if not prosesor_baru.strip():
    print(RED+ 'Input Tidak Boleh Kosong'+RESET)
    return

vga_baru = input(BOLD+"Masukan VGA Baru : "+RESET) or produk["vga"]
if not vga_baru.strip():
    print(RED+ 'Input Tidak Boleh Kosong'+RESET)
    return

harga_baru = int(input(BOLD+"Masukan Harga Baru : "+RESET) or produk["harga"])
if harga_baru <= 0:
    print(RED + "Harga Tidak Boleh Minus atau 0" + RESET)
    return

produk_laptop[nomor] = {
    "nama": nama_baru,
    "merek": merek_baru,
    "prosesor": prosesor_baru,

```

```

        "vga": vga_baru,
        "harga": harga_baru
    }
    simpan_data_csv()
    print("Produk Berhasil Diubah!")
else:
    print(RED+"Nomor Tidak Ditemukan!" +RESET)
except ValueError:
    print(RED+"Input Harus Berupa Angka!" +RESET)

# Prosedur Hapus Produk
def hapus_produk():
    tampilkan_produk()
    try:
        nomor = int(input(BOLD+"Masukkan Nomor Produk yang ingin dihapus: " +RESET))
        if nomor in produk_laptop:
            del produk_laptop[nomor]
            rapikan_nomor_produk()
            simpan_data_csv()
            print(f"Produk dengan nomor {nomor} berhasil dihapus.")
        else:
            print(RED+"Nomor Tidak Ditemukan!" +RESET)
    except ValueError:
        print(RED+"Input Harus Berupa Angka!" +RESET)

def beli_produk(username):
    tampilkan_produk()
    try:
        nomor = int(input(BOLD+"Masukkan Nomor Produk yang ingin dibeli: " +RESET))
        if nomor in produk_laptop:
            produk = produk_laptop.pop(nomor)
            produk_dibeli.append(produk)

```

```

        user[username]["produk_dibeli"].append(produk["nama"])
        rapikan_nomor_produk()
        simpan_data_csv()
        print(f"Produk '{produk['nama']}' berhasil dibeli.")
    else:
        print(RED+"Nomor Tidak Ditemukan!" +RESET)
except ValueError:
    print(RED+"Input Harus Berupa Angka!" +RESET)

# Fungsi Total Harga
def tampilkan_total_harga():
    total = hitung_total_harga()
    print(f"\nTotal Harga Semua Produk: Rp{total}")

# Fungsi Total Harga untuk Produk yang Dibeli Pengunjung
def tampilkan_total_harga_pengunjung(username):
    if not user[username]["produk_dibeli"]:
        print("Anda belum membeli produk apa pun.")
        return

    total_harga = 0
    for produk in produk_dibeli:
        if produk["nama"] in user[username]["produk_dibeli"]:
            total_harga += produk["harga"]
    print(f"\nTotal Harga Produk yang Anda Beli: Rp{total_harga}")

def rapikan_nomor_produk():
    produk_laptop_urut = {i+1: data for i, data in enumerate(produk_laptop.values())}
    produk_laptop.clear()
    produk_laptop.update(produk_laptop_urut)

# Fungsi Simpan Data ke CSV

```

```

def simpan_data_csv():
    with open('data_produk.csv', 'w', newline='') as file:
        writer = csv.writer(file)
        writer.writerow(["Nama", "Merek", "Prosesor", "VGA", "Harga"])
        for data in produk_laptop.values():
            writer.writerow([data["nama"], data["merek"], data["prosesor"], data["vga"],
data["harga"]])

# Fungsi Muat Data dari CSV
def muat_data_csv():
    try:
        with open('data_produk.csv', 'r') as file:
            reader = csv.DictReader(file)
            for i, row in enumerate(reader, start=1): # Tambahkan nomor produk
                produk_laptop[i] = {
                    "nama": row["Nama"],
                    "merek": row["Merek"],
                    "prosesor": row["Prosesor"],
                    "vga": row["VGA"],
                    "harga": int(row["Harga"])
                }
    except FileNotFoundError:
        print(RED+"File data_produk.csv tidak ditemukan"+RESET)

def user_register():
    username_baru = input(BOLD+"\nMasukkan Username Baru : "+RESET)
    if not username_baru.strip():
        print(RED+"Input Tidak Boleh Kosong"+RESET)
    else:
        password_baru = input(BOLD+"Masukkan Password Baru : "+RESET)
        if not password_baru.strip():
            print(RED+"Input Tidak Boleh Kosong"+RESET)

```

```

else:
    if username_baru in user:
        print("Nama Pengguna Sudah Terdaftar!")
    else:
        role_baru = "pengunjung"
        user[username_baru] = {"password": password_baru, "role": role_baru,
"produk_dibeli": []}
        data_pengunjung.append({"username": username_baru, "password":
password_baru})
        print(f"Pendaftaran Berhasil Untuk {username_baru} Dengan Role {role_baru}!")

# Fungsi untuk Menampilkan Data User (Pengunjung)
def tampilkan_data_user():
    if len(data_pengunjung) == 0:
        print("Tidak ada pengunjung yang terdaftar.")
    else:
        print("\nBerikut Adalah Data Pengunjung")
        tabel_data = []
        for username, data in user.items():
            if data["role"] == "pengunjung":
                produk_dibeli = ", ".join(data["produk_dibeli"]) if data["produk_dibeli"] else "Tidak
ada"
                tabel_data.append([username, data['password'], produk_dibeli])

        animasi_output(tabulate(tabel_data, headers=["Username", "Password", "Produk Dibeli"],
tablefmt="grid"), delay=0.0000001)

def login_menu():
    animasi_output(GREEN+"\n===== ",
delay=0.0000001)
    animasi_output("|   Toko Laptop Wahyu Jaya   |", delay=0.0000001)
    animasi_output("===== ", delay= 0.0000001)

```



```

animasi_output("| 1. Login          |", delay= 0.0000001)
animasi_output("| 2. Register          |", delay= 0.0000001)
animasi_output("| 3. Keluar            |", delay= 0.0000001)
animasi_output("===== "+RESET,
delay=0.0000001)

def main_menu():
    if role == "admin":
        animasi_output(MAGENTA+"\n===== ",
delay=0.0000001)
        animasi_output("| Toko Laptop Wahyu Jaya |", delay=0.0000001)
        animasi_output("===== ", delay=0.0000001)
        animasi_output("| 1. Tampilkan Produk    |", delay=0.0000001)
        animasi_output("| 2. Tambah Produk       |", delay=0.0000001)
        animasi_output("| 3. Ubah Produk         |", delay=0.0000001)
        animasi_output("| 4. Hapus Produk        |", delay=0.0000001)
        animasi_output("| 5. Tampilkan Total Harga |", delay=0.0000001)
        animasi_output("| 6. Tampilkan Data User  |", delay=0.0000001)
        animasi_output("| 7. Keluar              |", delay=0.0000001)
        animasi_output("===== "+RESET,
delay=0.0000001)
    else:
        animasi_output(BRIGHT_YELLOW+"\n===== ",
delay=0.0000001)
        animasi_output("| Toko Laptop Wahyu Jaya |", delay=0.0000001)
        animasi_output("===== ", delay=0.0000001)
        animasi_output("| 1. Tampilkan Produk    |", delay=0.0000001)
        animasi_output("| 2. Beli Produk         |", delay=0.0000001)
        animasi_output("| 3. Tampilkan Total Harga |", delay=0.0000001)
        animasi_output("| 4. Keluar              |", delay=0.0000001)
        animasi_output("===== "+RESET,
delay=0.0000001)

```

```

def back():
    input("Tekan Enter Untuk Melanjutkan...")

# Sesi Login
muat_data_csv()
while True:
    clear()
    login_menu()
    pilihan = input(BOLD+"Masukan Pilihan : "+RESET)
    clear()

    if pilihan == "1":
        username = input("\nMasukkan Username : ")
        password = input("Masukkan Password : ")
        role = login(username, password)
        back()
        loading(1)
        clear()

    # Program Utama CRUD
    if role:
        while True:
            main_menu()
            pilih = input(BOLD+"Masukan Pilihan : "+RESET)
            loading(1)
            clear()

            if pilih == "1":
                tampilkan_produk()
                back()
                loading(2)

```

```
clear()

elif pilih == "2" and role == "admin":
    tampilkan_produk()
    tambah_produk()
    back()
    loading(2)
    clear()

elif pilih == "2" and role == "pengunjung":
    beli_produk(username)
    back()
    loading(2)
    clear()

elif pilih == "3" and role == "admin":
    ubah_produk()
    back()
    loading(2)
    clear()

elif pilih == "3" and role == "pengunjung":
    tampilkan_total_harga_pengunjung(username)
    back()
    loading(2)
    clear()

elif pilih == "4" and role == "admin":
    hapus_produk()
    back()
    loading(2)
    clear()
```

```
elif pilih == "4" and role == "pengunjung":  
    print("Terima Kasih Telah Melihat Produk Kami")  
    loading(2)  
    clear()  
    break
```

```
elif pilih == "5" and role == "admin":  
    tampilkan_total_harga()  
    back()  
    loading(2)  
    clear()
```

```
elif pilih == "6" and role == "admin":  
    tampilkan_data_user()  
    back()  
    loading(2)  
    clear()
```

```
elif pilih == "7" and role == "admin":  
    print("Terima Kasih Telah Melihat Produk Kami")  
    loading(2)  
    clear()  
    break
```

```
else:  
    print("Pilihan Tidak Valid!")  
    loading(1)  
    clear()
```

```
# Register Hanya Untuk Pengunjung  
elif pilihan == "2":
```

```
    user_register()
    back()
    loading(2)

# Keluar
elif pilihan == "3":
    print("Terima Kasih Telah Berkunjung.")
    loading(2)
    clear()
    break

else:
    print("Pilihan Tidak Valid!")
    loading(1)
    clear()
```

BAB IV

PENUTUP

4.1 Kesimpulan

Program yang telah dibuat merupakan aplikasi manajemen toko laptop dengan menggunakan konsep CRUD (Create, Read, Update, Delete) untuk pengelolaan produk laptop. Program ini juga memungkinkan pengguna untuk login, mendaftar, dan melakukan pembelian produk. Beberapa fitur utama dari program ini adalah:

1. Fungsi Login dan Register: Pengguna dapat login dengan akun yang sudah terdaftar atau mendaftar sebagai pengunjung. Admin memiliki hak akses untuk menambah, mengubah, dan menghapus produk, sementara pengunjung hanya dapat membeli produk dan melihat daftar produk serta total harga.
2. Pengelolaan Produk (CRUD): Admin dapat menambah, mengubah, dan menghapus produk laptop. Produk yang disimpan dalam program disimpan dalam file CSV, yang memungkinkan data tetap tersimpan meski program dihentikan.
3. Pembelian Produk: Pengunjung dapat memilih produk yang ingin dibeli dan melihat total harga produk yang dibeli.
4. Tampilan yang Interaktif: Dengan menggunakan animasi dan warna pada tampilan teks, program memberikan pengalaman pengguna yang lebih menarik.
5. Penanganan Data: Program menggunakan file CSV untuk menyimpan dan memuat data produk dan pengguna, serta memastikan data tetap aman meskipun aplikasi ditutup.
6. Keamanan dan Validasi Input: Program dilengkapi dengan validasi input yang memastikan data yang dimasukkan oleh pengguna sesuai dengan format yang

benar (misalnya, harga harus berupa angka), dan memberikan notifikasi jika terjadi kesalahan.

4.2 Saran

Berikut beberapa saran umum untuk meningkatkan program yang telah dibuat:

1. **Penambahan Fitur Pencarian Produk:** Agar lebih mudah mencari produk, bisa ditambahkan fitur pencarian berdasarkan nama produk, merek, atau harga. Dengan begitu, pengguna bisa langsung menemukan produk yang diinginkan tanpa harus melihat seluruh daftar.
2. **Keamanan Pengguna:** Saat ini, password pengguna disimpan dalam bentuk teks biasa (plain text). Sebaiknya, password pengguna disimpan dalam bentuk yang lebih aman, seperti menggunakan teknik enkripsi atau hashing, agar lebih terlindungi.
3. **Perbaikan Tampilan Antarmuka:** Program ini masih berjalan di terminal dengan teks yang sederhana. Untuk membuatnya lebih menarik dan mudah digunakan, bisa dipertimbangkan untuk menggunakan antarmuka grafis (GUI) yang lebih ramah pengguna, seperti menggunakan Tkinter atau PyQt.
4. **Tambah Fitur Pengelolaan Pengguna:** Selain login dan registrasi, admin bisa diberikan akses untuk mengubah atau menghapus akun pengguna. Ini akan memudahkan pengelolaan pengguna yang terdaftar.
5. **Perbaikan Penanganan Kesalahan:** Program sudah cukup baik dalam menangani kesalahan, namun ada baiknya jika penanganan kesalahan diperluas, misalnya dengan memberikan pesan yang lebih jelas saat terjadi kesalahan membaca file atau saat input tidak valid.
6. **Integrasi Pembayaran:** Program ini sudah memiliki fitur pembelian produk. Agar lebih lengkap, bisa dipertimbangkan untuk menambahkan sistem

pembayaran virtual, sehingga pengunjung bisa langsung melakukan pembayaran setelah membeli produk.

7. Simpan Data Pengguna dan Produk dengan Lebih Efisien: Penyimpanan data produk dan pengguna dalam file CSV sudah cukup baik, namun untuk pengelolaan data yang lebih efisien, bisa dipertimbangkan untuk menggunakan database (seperti SQLite) agar data lebih mudah dikelola.

DAFTAR PUSTAKA

Rahman, S., Sembiring, A., Siregar, D., Prahmana, I. G., Puspadini, R., & Zen, M. (2023). Python: Dasar dan Pemrograman Berorientasi Objek. *Penerbit Tahta Media*.

Hazrat, R. (2024). Lists and Tuples. In *A Course in Python: The Core of the Language* (pp. 23-47). Cham: Springer Nature Switzerland.

Kiusalaas, J. (2010). *Numerical methods in engineering with Python*. Cambridge university press.

Lutz, M. (2001). *Programming python*. " O'Reilly Media, Inc."

Combrisson, E., Vallat, R., Eichenlaub, J. B., O'Reilly, C., Lajnef, T., Guillot, A., ... & Jerbi, K. (2017). Sleep: an open-source python software for visualization, analysis, and staging of sleep data. *Frontiers in neuroinformatics*, 11, 292274.

Muqorobin, M., & Rais, N. A. R. (2022). Comparison of PHP programming language with codeigniter framework in project CRUD. *International Journal of Computer and Information System (IJCIS)*, 3(3), 94-98.

Rodriguez-Tudela, J. L., Arendrup, M. C., Arikan, S., Barchiesi, F., Bille, J., Chrysanthou, E., ... & Verweij, P. (2008). EUCAST DEFINITIVE DOCUMENT E. DEF 9.1: Method for the determination of broth dilution minimum inhibitory concentrations of antifungal agents for conidia forming moulds. *Def*, 9, 1-13.

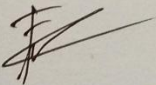
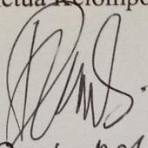
McKinney, W. (2012). *Python for data analysis: Data wrangling with Pandas, NumPy, and IPython*. " O'Reilly Media, Inc."

Shafranovich, Y. (2005). *Common format and MIME type for comma-separated values (CSV) files* (No. rfc4180).

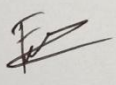
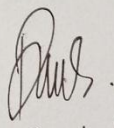
LAMPIRAN



Gambar Lampiran Cover Kertas Konsul

Tanggal Konsultasi : 31/10/2024	
Uraian / Pembahasan : mengenai konsep program yang akan dibuat dan membahas mengenai Flowchart dari program yang akan dibuat	
Asisten Lab  Nama: Christin Farrel	Ketua Kelompok  Nama: Dewi Agustini

Gambar Lampiran Konsul Pertama

Tanggal Konsultasi : Rabu, 13 Nov 24	
Uraian / Pembahasan : 1. Program dicek kembali 2. laporan dikerjakan 3. flowchart dibuat lebih rapih lagi	
Asisten Lab  Nama: Christen Farn	Ketua Kelompok  Nama: Dewi Astuti

Gambar Lampiran Konsul Kedua