APLIKASI MANAJEMEN TOKO GAME “RAYYSTORE GAMES”  
LAPORAN TUGAS STRUKTUR DATA

OLEH:

|  |  |
| --- | --- |
| RAYYA SEVARENO | 24416255201161 |



**IF24D**

**PROGRAM STUDI TEKNIK INFORMATIKA FAKULTAS ILMU KOMPUTER**

**UNIVERSITAS BUANA PERJUANGAN KARAWANG 2025**

# DAFTAR ISI

[DAFTAR ISI 2](#_Toc203253626)

[DAFTAR GAMBAR 3](#_Toc203253627)

[DAFTAR TABEL 5](#_Toc203253628)

[BAB I PENDAHULUAN 6](#_Toc203253629)

[1.1 Latar Belakang 6](#_Toc203253630)

[1.2 Rumusan Masalah 7](#_Toc203253631)

[1.3 Tujuan Proyek 7](#_Toc203253632)

[BAB II METODOLOGI PERANCANGAN SOLUSI 8](#_Toc203253633)

[2.1 Deskripsi Aplikasi 8](#_Toc203253634)

[2.1.1 Gambaran Umum Alur Aplikasi 8](#_Toc203253635)

[2.1.2. Fitur Utama yang Tersedia 9](#_Toc203253636)

[2.1.3. Penjelasan Cara Kerja Aplikasi Secara Keseluruhan 13](#_Toc203253637)

[2.1.4. Struktur File CSV 14](#_Toc203253638)

[BAB III PERANCANGAN SOLUSI 16](#_Toc203253639)

[3.1 Penjelasan Teknis 16](#_Toc203253640)

[3.2 Hasil Uji Coba 26](#_Toc203253641)

[a. Tampilan Menu 26](#_Toc203253642)

[b. Contoh Input–Output 26](#_Toc203253643)

[c. Pengujian Fungsional 32](#_Toc203253644)

[d. Pengujian Integrasi 33](#_Toc203253645)

[e. Catatan Kendala 33](#_Toc203253646)

[BAB IV KESIMPULAN 34](#_Toc203253647)

[DAFTAR PUSTAKA 36](#_Toc203253648)

# DAFTAR GAMBAR

**BAB II – Metodologi Perancangan Solusi**

[Gambar 2.1 Alur Keseluruhan Aplikasi 8](#_Toc203250749)

[Gambar 2.2 Alur Menampilkan Produk Game 9](#_Toc203250750)

[Gambar 2.3 Alur Menambahkan Produk Game 9](#_Toc203250751)

Gambar 2.4 Alur Memperbarui Produk Game 10

Gambar 2.5 Alur Menghapus Produk Game 10

Gambar 2.6 Alur Mencari Produk Game 11

Gambar 2.7 Alur Proses Transaksi Penjualan dan Pembelian 12

Gambar 2.8 Alur Proses Menampilkan Laporan Transaksi 13

**BAB III – Perancangan Solusi**

*Modul fungsi\_util.py*

[Gambar 3.1 Function load\_produk 17](#_Toc203250752)

Gambar 3.2 Function simpan\_produk 17

Gambar 3.3 Function load\_transaksi 17

Gambar 3.4 Function simpan\_transaksi 17

*Modul fungsi\_produk.py*

[Gambar 3.5 Function tampilkan\_produk 18](#_Toc203250753)

Gambar 3.6 Function tambah\_produk (1) 18

Gambar 3.7 Function tambah\_produk (2) 18

Gambar 3.8 Function update\_produk (1) 19

Gambar 3.9 Function update\_produk (2) 19

Gambar 3.10 Function hapus\_produk 19

Gambar 3.11 Function cari\_produk 20

*Modul fungsi\_transaksi.py*

[Gambar 3.12 Function transaksi (1) 20](#_Toc203250754)

Gambar 3.13 Function transaksi (2) 20

Gambar 3.14 Function transaksi (3) 21

Gambar 3.15 Function transaksi (4) 21

*Modul laporan.py*

[Gambar 3.16 Function laporan\_transaksi 21](#_Toc203250755)

*Modul welcome.py*

[Gambar 3.17 Function welcome\_message 22](#_Toc203250756)

*Modul main.py*

[Gambar 3.18 Function main 22](#_Toc203250757)

**Bagian 3.2 Hasil Uji Coba**

[Gambar 3.19 Tampilan Menu 26](#_Toc203250758)

Gambar 3.20 Input-output Tambah Produk 26

Gambar 3.21 Hasil Tambah Produk (CSV) 26

Gambar 3.22 Input-output Update Produk 26

Gambar 3.23 Hasil Update Produk (CSV) 26

Gambar 3.24 Input-output Hapus Produk 26

Gambar 3.25 Hasil Hapus Produk (CSV) 27

Gambar 3.26 Input-output Cari Produk 27

Gambar 3.27 Input-output Transaksi Penjualan 27

Gambar 3.28 Input-output Transaksi Pembelian 27

Gambar 3.29 Hasil Transaksi di transaksi.csv 27

Gambar 3.30 Input-output Laporan Transaksi 27

Gambar 3.31 Output Laporan Transaksi 27

# DAFTAR TABEL

Tabel 2.1 struktur produk.csv 14

Tabel 2.1 struktur transaksi.csv 15

Tabel 3.2 pengujian fungsional proyek 11

# BAB I PENDAHULUAN

## Latar Belakang

Dalam perkembangan era digital saat ini, proses pengelolaan data secara manual mulai digantikan oleh program digital yang jauh lebih mudah dan efisien. Salah satu sector yang juga memerlukan program digital adalah bidang manajemen penjualan toko dan distribusi game. Banyaknya jenis game yang tersedia serta tingginya minat masyarakat terhadap pembelian game menuntut adanya sistem manajemen data yang efisien, terutama bagi toko game yang menjual berbagai produk secara fisik maupun digital. Oleh karena itu, dibutuhkan suatu aplikasi manajemen jual beli game yang mampu membantu proses pengelolaan stok, transaksi penjualan dan pembelian, serta pelaporan penjualan dengan cara yang sistematis dan mudah digunakan.

Untuk mengatasi permasalahan tersebut, dikembangkanlah sebuah aplikasi manajemen penjualan toko game berbasis Python dengan memanfaatkan struktur data algoritma seperti array/list, queue, dan data searching. Aplikasi ini juga menggunakan file CSV sebagai sistem database untuk membuat, membaca, mengubah, menghapus data status manajemen beserta laporan transaksi secara permanen. Aplikasi ini menyediakan fitur pengelolaan produk (stok, harga, dan deskripsi), pencatatan transaksi penjualan dan pembelian, serta laporan penjualan harian, mingguan, dan bulanan.Proyek ini juga menjadi penerapan nyata dari konsep-konsep yang telah dipelajari dalam mata kuliah Struktur Data, seperti pengolahan array, antrian (queue), dan algoritma pencarian data.

Aplikasi ini penting karena dapat membantu pemilik atau pengelola toko game, khususnya skala kecil hingga menengah, dalam mencatat data secara lebih rapi dan terstruktur. Target pengguna dari aplikasi ini adalah pengelola toko game dan pemilik usaha kecil yang belum memiliki sistem manajemen data berbasis digital. Hasil yang ingin dicapai dari penggunaan aplikasi ini adalah terciptanya sistem manajemen data yang praktis, cepat, dan mampu menampilkan laporan penjualan secara otomatis.

## Rumusan Masalah

Berdasarkan latar belakang di atas, maka rumusan masalah dalam proyek ini adalah:

1. Bagaimana merancang konsep aplikasi manajemen toko game dalam pemrograman Python
2. Bagaimana mengimplementasikan struktur data array/list, queue, searching dalam mengelola produk game dan sistem transaksi
3. Bagaimana mengaplikasikan sistem database secara permanen menggunakan file CSV

## Tujuan Proyek

Tujuan dari pembuatan proyek ini adalah:

1. Membangun aplikasi manajemen data jual beli untuk toko game yang dapat mengelola stok, transaksi, dan laporan.
2. Mengimplementasikan konsep struktur data algoritma seperti array/list, queue, dan searching dalam kasus nyata
3. Menggunakan file CSV sebagai penyimpanan data yang efisien dan mudah diakses.

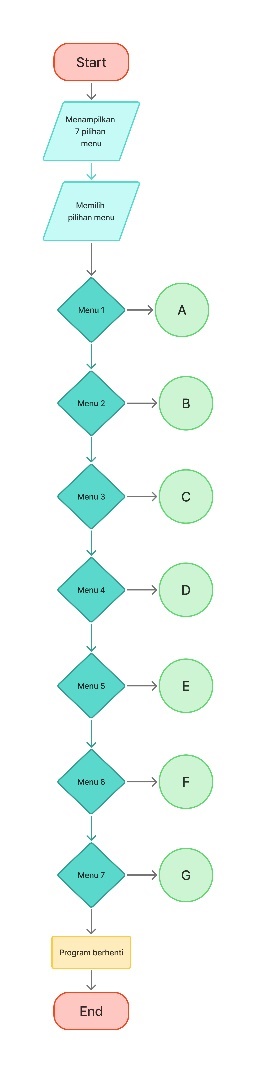
# BAB II METODOLOGI PERANCANGAN SOLUSI

## Deskripsi Aplikasi

Aplikasi yang dikembangkan bernama "RayyStore Games", yaitu sebuah sistem manajemen toko game berbasis terminal yang menggunakan file CSV (Comma-Separated Values) sebagai media penyimpanan data. Aplikasi ini dirancang untuk mengelola data produk, mencatat transaksi penjualan dan pembelian, serta menampilkan laporan penjualan berdasarkan periode waktu tertentu (harian, mingguan, dan bulanan).

### ****2.1.1 Gambaran Umum Alur Aplikasi****

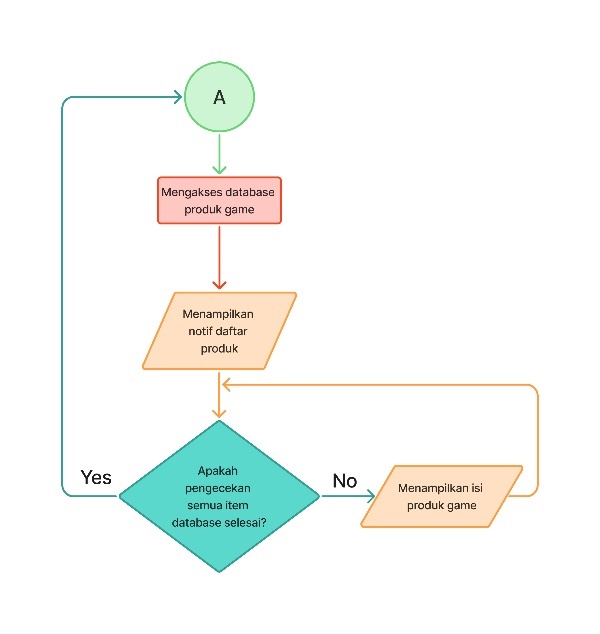
Saat aplikasi dijalankan, pengguna akan disambut oleh tampilan menu utama berisi beberapa pilihan daftar fitur. Setiap menu akan menjalankan fungsi fitur sesuai kebutuhan pengguna. Alur ini dapat digambarkan dalam bentuk flowchart berikut:



#### **Gambar 2.1 Alur Keseluruhan Aplikasi**

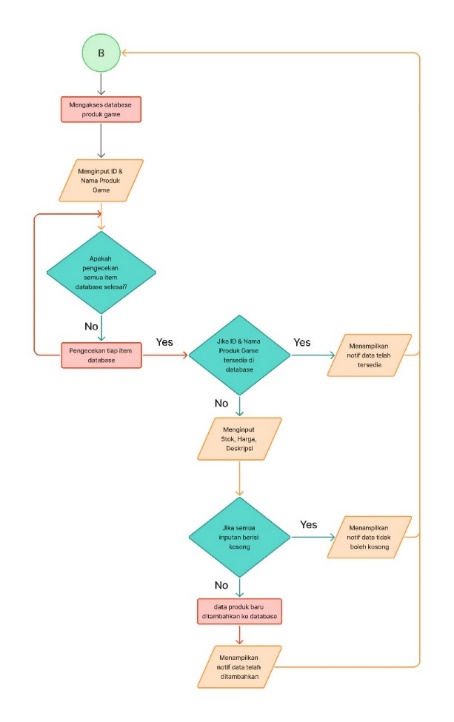
### ****2.1.2. Fitur Utama yang Tersedia****

1. **Manajemen Produk Game**
   * Menampilkan seluruh produk yang tersedia dalam file CSV.



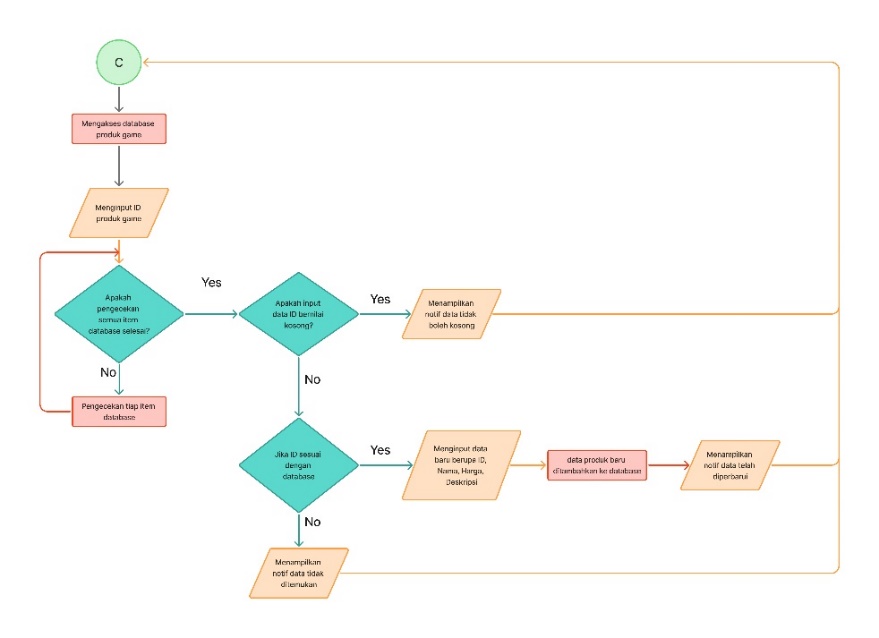
#### **Gambar 2.1 Alur Menampilkan Produk Game**

* + Menambahkan produk baru dengan validasi ID dan nama agar tidak duplikat.



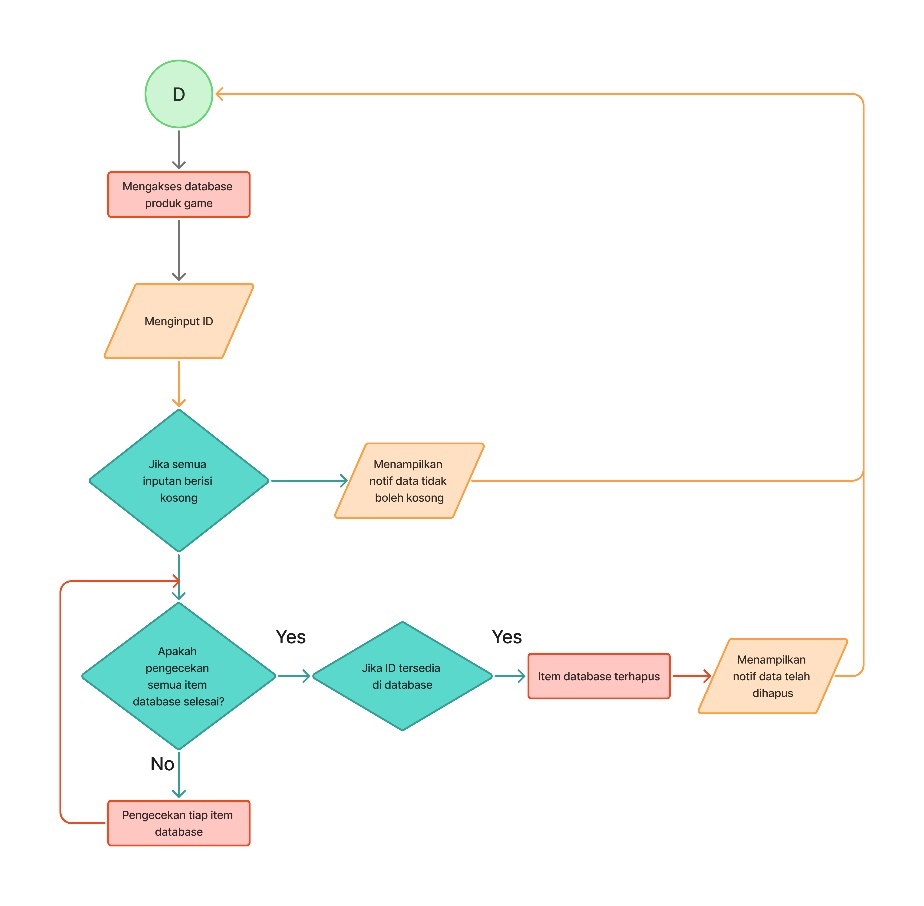
#### **Gambar 2.1 Alur** Menambahkan **Produk Game**

* + Memperbarui data produk berdasarkan ID.



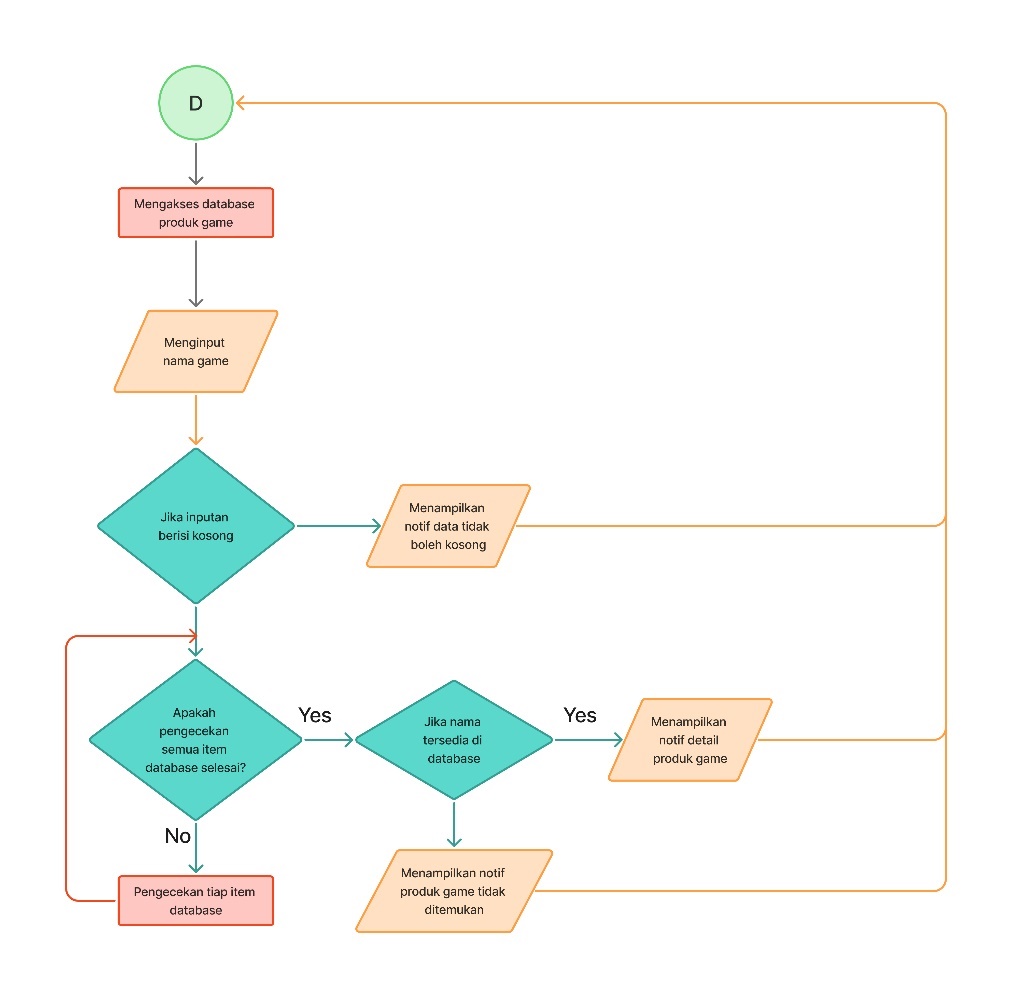
#### **Gambar 2.1 Alur** Memperbarui **Produk Game**

* + Menghapus produk dari data.



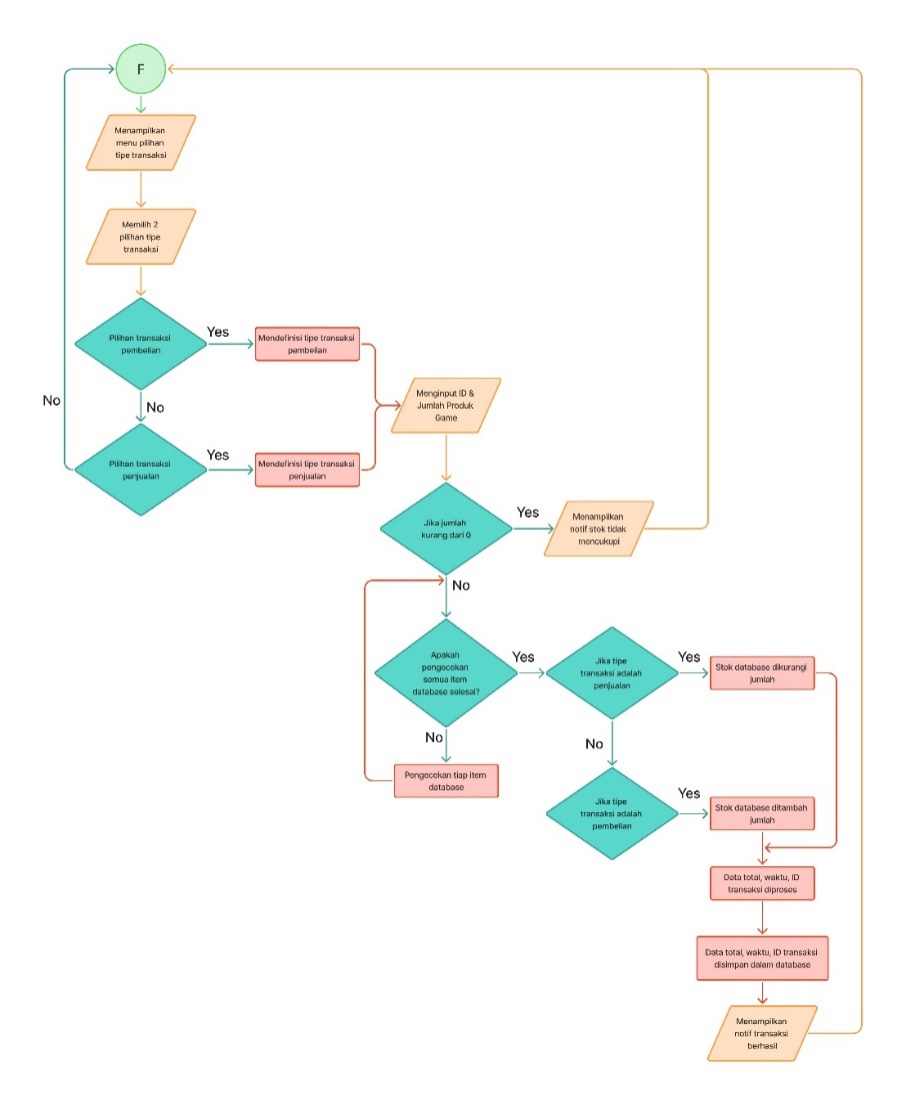
#### **Gambar 2.1 Alur** Menghapus **Produk Game**

* + Mencari produk berdasarkan nama.



#### **Gambar 2.1 Alur** Mencari **Produk Game**

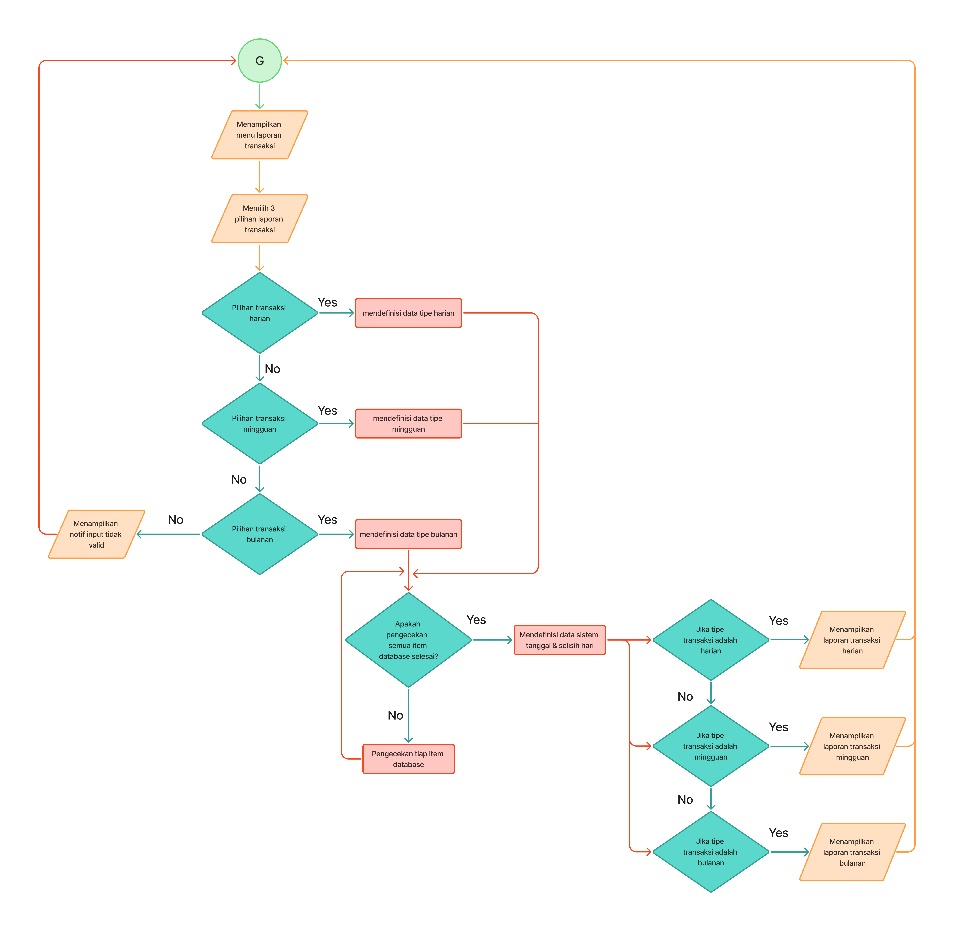
1. **Transaksi**

****

#### **Gambar 2.1 Alur** Proses Transaksi Penjualan dan Pembelian

* + Melayani transaksi **penjualan** (mengurangi stok) dan **pembelian** (menambah stok).
  + Validasi stok tersedia sebelum penjualan dilakukan.
  + Perhitungan total harga berdasarkan jumlah dan harga produk.
  + Mencatat transaksi ke file CSV dan queue.

1. **Laporan Transaksi**



#### **Gambar 2.1** **Alur Proses Menampilkan Laporan Transaksi**

* + Menyediakan laporan berdasarkan rentang waktu:
    - **Harian** (≤1 hari)
    - **Mingguan** (≤7 hari)
    - **Bulanan** (≤30 hari)
  + Laporan ditampilkan dalam format yang memuat ID transaksi, tanggal, jumlah, total, dan tipe transaksi.

### ****2.1.3. Penjelasan Cara Kerja Aplikasi Secara Keseluruhan****

Secara keseluruhan, aplikasi ini bekerja melalui struktur beberapa fungsi yang kemudian dipisah menjadi beberapa file dan semua modular saling keterkaitan dengan file utama:

* **main.py** adalah pusat eksekusi program. Di dalamnya terdapat perulangan menu yang terus berjalan sampai pengguna memilih keluar. File ini menghubungkan semua fitur ke pengguna.
* **welcome.py** hanya menyimpan fungsi sambutan awal dengan daftar menu agar antarmuka lebih terstruktur.
* **fungsi\_produk.py** berisi semua fungsi untuk pengelolaan produk seperti menambah, menampilkan, mengubah, menghapus, dan mencari data game yang disimpan di produk.csv. Semua input divalidasi agar tidak kosong, dan ID serta nama game tidak boleh duplikat saat menambah data.
* **fungsi\_transaksi.py** menangani proses transaksi penjualan dan pembelian. Transaksi akan memvalidasi ketersediaan stok, menghitung total, serta mencatat transaksi tersebut ke dalam file transaksi.csv dan juga ke dalam queue (antrian data sementara).
* **laporan.py** mengelola pembuatan laporan transaksi berdasarkan waktu. File ini memproses data dari transaksi.csv dan memfilter transaksi berdasarkan hari, minggu, atau bulan.
* **fungsi\_util.py** berisi fungsi pembantu seperti membaca (load\_produk, load\_transaksi) dan menyimpan data (simpan\_produk, simpan\_transaksi) dari dan ke file CSV.

Seluruh data produk dan transaksi disimpan secara lokal dalam file CSV, yang kemudian dimuat ulang saat dibutuhkan. Perubahan terhadap data seperti penambahan, penghapusan, dan pembaruan akan langsung ditulis ulang ke file CSV agar data tetap konsisten.

### ****2.1.4. Struktur File CSV****

#### **1. File produk.csv**

File ini menyimpan informasi tentang produk game. Kolom-kolom yang terdapat di dalamnya adalah:

|  |  |
| --- | --- |
| Kolom | Keterangan |
| id | ID unik produk, biasanya berformat huruf+angka (contoh: P001) |
| nama | Nama game yang dijual |
| stok | Jumlah stok game yang tersedia |
| harga | Harga satuan dari produk game |
| deskripsi | Penjelasan singkat mengenai game |

#### **Tabel 2.1 struktur produk.csv**

#### **2. File transaksi.csv**

File ini mencatat seluruh riwayat transaksi, baik pembelian maupun penjualan. Kolom-kolom yang disimpan:

|  |  |
| --- | --- |
| Kolom | Keterangan |
| Tanggal | Tanggal transaksi dilakukan (YYYY-MM-DD) |
| id\_transaksi | ID transaksi (contoh: T0001) |
| id\_produk | ID dari produk yang dibeli/dijual |
| Jumlah | Jumlah unit yang dibeli/dijual |
| Game | Keterangan nama game |
| total | Total harga dari transaksi (harga x jumlah) |
| tipe | Jenis transaksi Penjualan atau Pembelian |

#### **Tabel 2.1 struktur** transaksi**.csv**

Dengan pendekatan modular dan penyimpanan berbasis CSV, aplikasi ini dapat dijalankan secara ringan di berbagai perangkat tanpa database tambahan. Struktur dan alur kerja yang terorganisir juga memudahkan pemeliharaan serta pengembangan fitur di masa depan.

# BAB III PERANCANGAN SOLUSI

## 3.1 Penjelasan Teknis

Aplikasi RayyStore Games dikembangkan menggunakan bahasa Python dengan pendekatan modular. Struktur data utama menggunakan **list of dictionary** yang disimpan dan dimuat melalui file CSV. Setiap modul memiliki fungsi-fungsi khusus yang mendukung fitur aplikasi, seperti manajemen produk, transaksi, laporan, dan tampilan antarmuka terminal.

#### **a. Struktur Data**

Aplikasi menggunakan dua struktur data utama:

* **List of Dictionary** untuk menyimpan data dari file CSV.
* **Queue** (antrian) dari collections.deque untuk menyimpan transaksi sementara.
* **Searching untuk menc**ari item produk game

#### **b. Modul fungsi\_util.py (Folder: fungsi\_manajemen)**

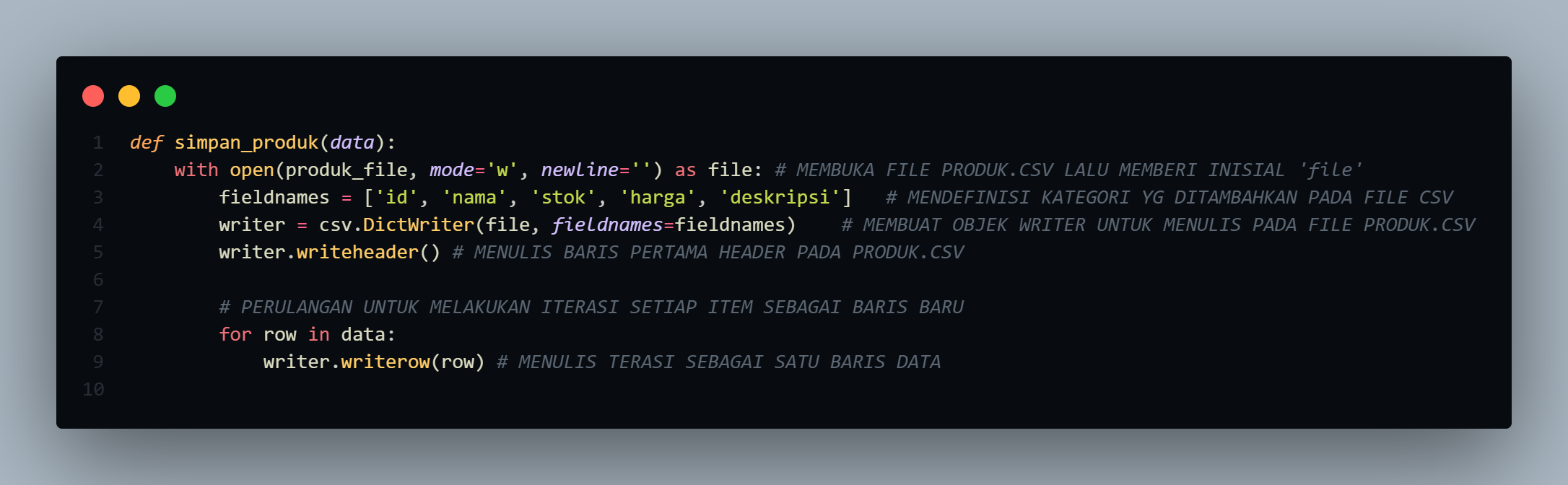
Modul ini bertanggung jawab memuat dan menyimpan data ke dalam file CSV.

* **load\_produk()**  
  Membaca data dari produk.csv dan mengembalikan list of dict.



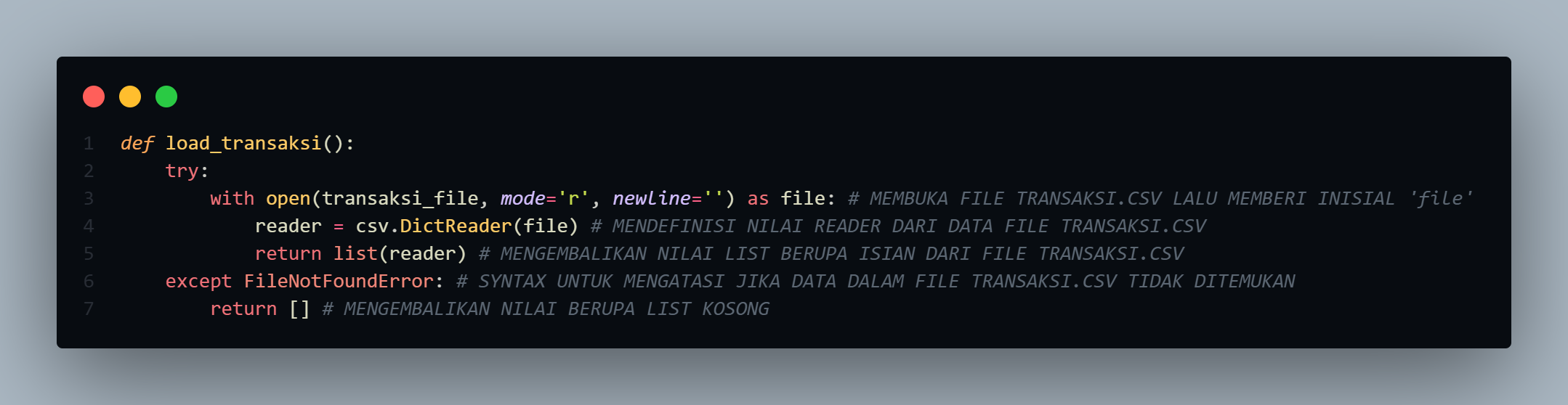
#### **Gambar 3.1 function load\_produk**

* **simpan\_produk(data)**  
  Menulis ulang data produk ke produk.csv.



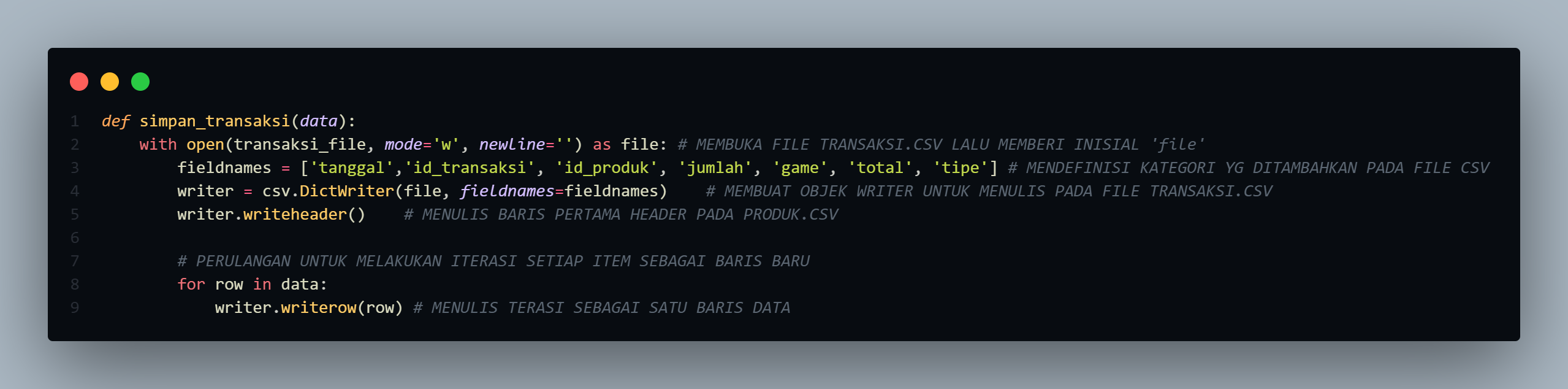
#### **Gambar 3.1 function simpan\_produk**

* **load\_transaksi()**  
  Membaca data dari transaksi.csv.



#### **Gambar 3.1 function load\_transaksi**

* **simpan\_transaksi(data)**  
  Menulis ulang data transaksi ke transaksi.csv.

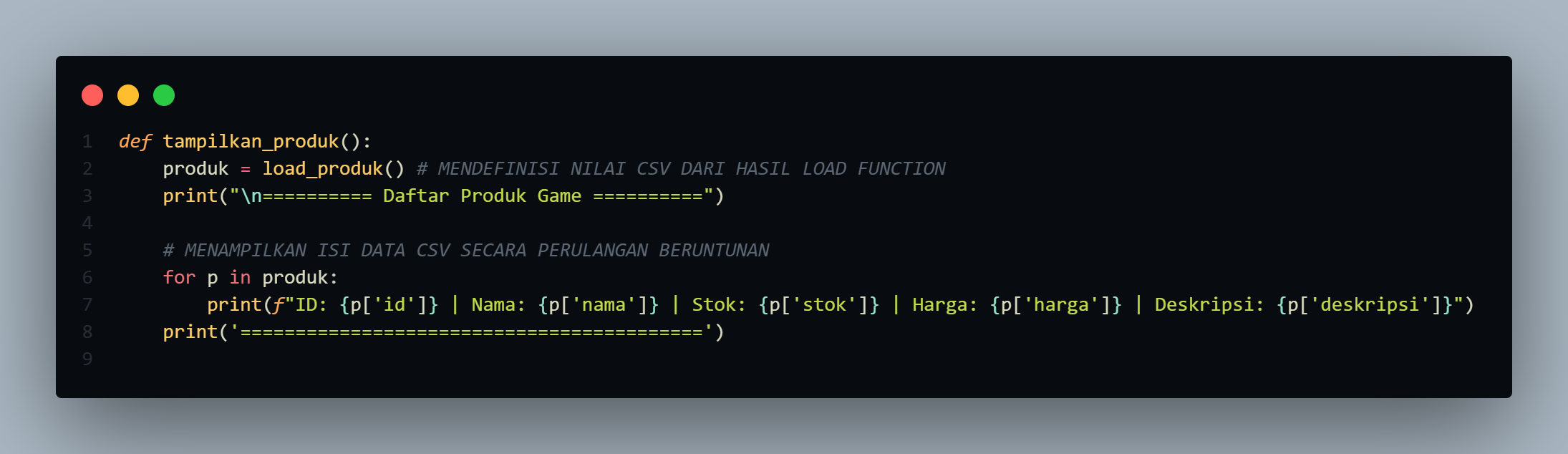


#### **Gambar 3.1 function simpan\_transaksi**

#### **c. Modul fungsi\_produk.py (Folder: fungsi\_manajemen)**

Modul ini menangani seluruh operasi CRUD terhadap data produk.

* **tampilkan\_produk()**  
  Menampilkan daftar semua produk game yang tersimpan.



#### **Gambar 3.1 function tampilkan\_produk**

* **tambah\_produk()**  
  Menambahkan produk baru dengan validasi:
  + ID dan nama tidak boleh duplikat
  + Semua input wajib diisi (tidak boleh kosong)



#### **Gambar 3.1 function tambah\_produk 1**



#### **Gambar 3.1 function tambah\_produk 2**

* **update\_produk()**  
  Memperbarui data produk berdasarkan ID. Namun, implementasinya belum menghapus data lama, hanya menambahkan data baru.



#### **Gambar 3.1 function update\_produk 1**



#### **Gambar 3.1 function update\_produk 2**

* **hapus\_produk()**  
  Menghapus produk dari produk.csv berdasarkan ID.



#### **Gambar 3.1 function hapus\_produk**

* **cari\_produk()**  
  Mencari dan menampilkan data game berdasarkan nama.



#### **Gambar 3.1 function cari\_produk**

#### **d. Modul fungsi\_transaksi.py (Folder: fungsi\_manajemen)**

Modul ini menangani proses transaksi **penjualan** dan **pembelian** produk.

* **transaksi()**  
  Meminta pengguna memilih jenis transaksi:
  + Jika penjualan, stok akan dikurangi
  + Jika pembelian, stok akan ditambah  
    Transaksi disimpan ke transaksi.csv dan queue (deque) untuk antrian data sementara.



#### **Gambar 3.1 function transaksi 1**



#### **Gambar 3.1 function transaksi 2**



#### **Gambar 3.1 function transaksi 3**



#### **Gambar 3.1 function transaksi 4**

#### **e. Modul laporan.py**

Modul ini menyajikan laporan transaksi berdasarkan periode:

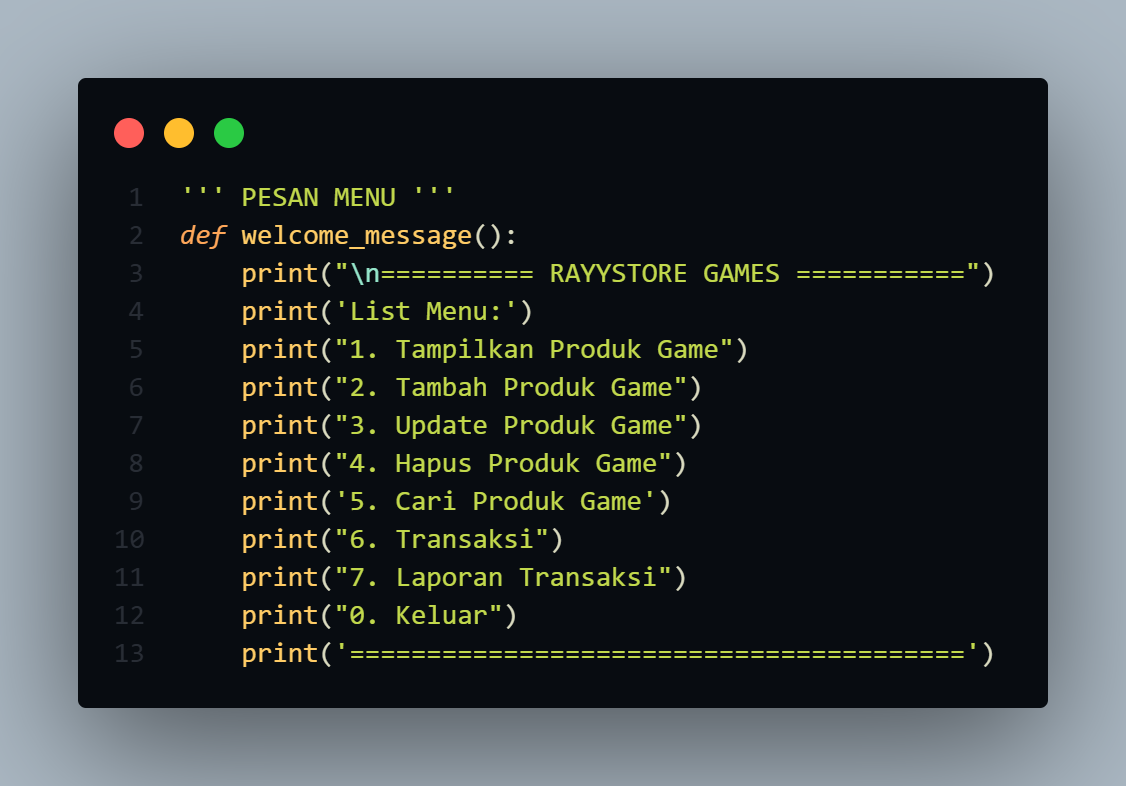
* **laporan\_transaksi()**  
  Memfilter transaksi berdasarkan waktu:
  + Harian (selisih hari <= 1)
  + Mingguan (<= 7 hari)
  + Bulanan (<= 30 hari)



#### **Gambar 3.1 function laporan\_transaksi**

#### **f. Modul welcome.py (Folder: libs)**

* **welcome\_message()**  
  Menampilkan menu utama kepada pengguna. Ini merupakan antarmuka awal dari sistem CLI (command-line interface).



#### **Gambar 3.1 function welcome\_message**

#### **g. Modul main.py**

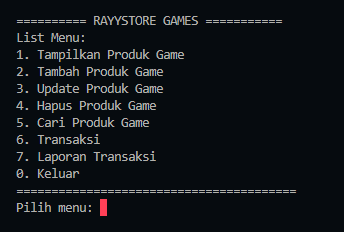
Modul utama ini mengatur alur aplikasi. Fungsi menu() melakukan loop tak hingga untuk menerima input pengguna dan memanggil fungsi sesuai menu.



#### **Gambar 3.1 function main**

## 3.2 Hasil Uji Coba

## a. Tampilan Menu



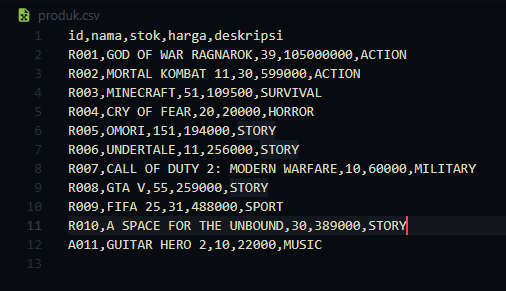
#### **Gambar 3.2 tampilan menu**

## b. Contoh Input–Output

#### **Tambah Produk Game**



#### **Gambar 3.2 tampilan input-output tambah produk game**

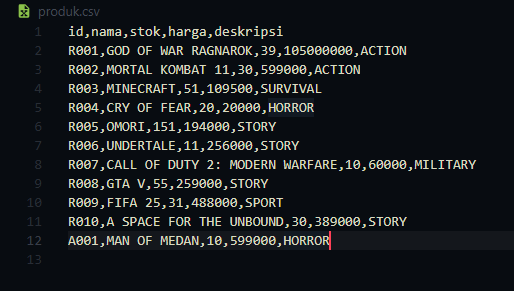


#### **Gambar 3.2 hasil tambahan produk di file produk.csv**

#### **Update** **Produk** **Game**



#### **Gambar 3.2 tampilan input-output update produk game**

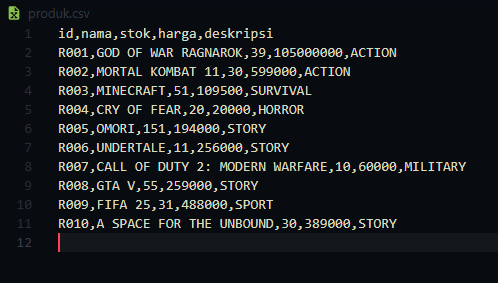


#### **Gambar 3.2 hasil pengubahan produk di file produk.csv**

#### **Hapus Produk Game**

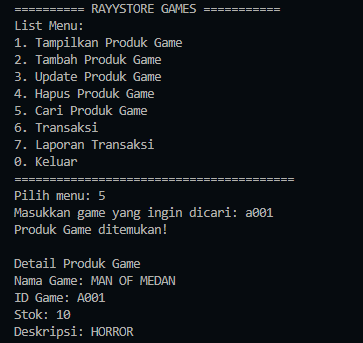


#### **Gambar 3.2 tampilan input-output hapus produk game**



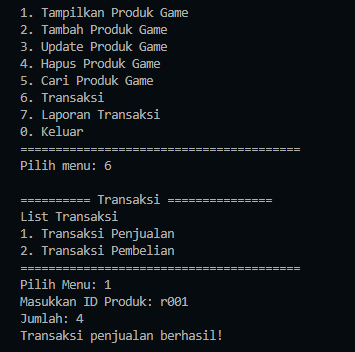
#### **Gambar 3.2 hasil penghapusan produk di file produk.csv**

#### **Cari Produk** **Game**

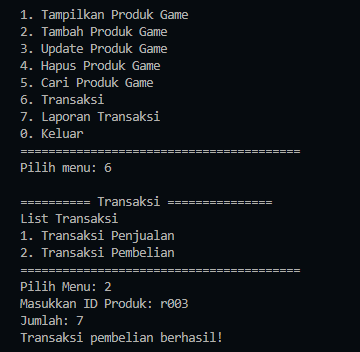


#### **Gambar 3.2 tampilan input-output cari produk game**

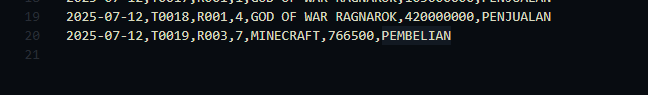
#### **Transaksi**



#### **Gambar 3.2 tampilan input-output transaksi penjualan**



#### **Gambar 3.2 tampilan input-output transaksi pembelian**

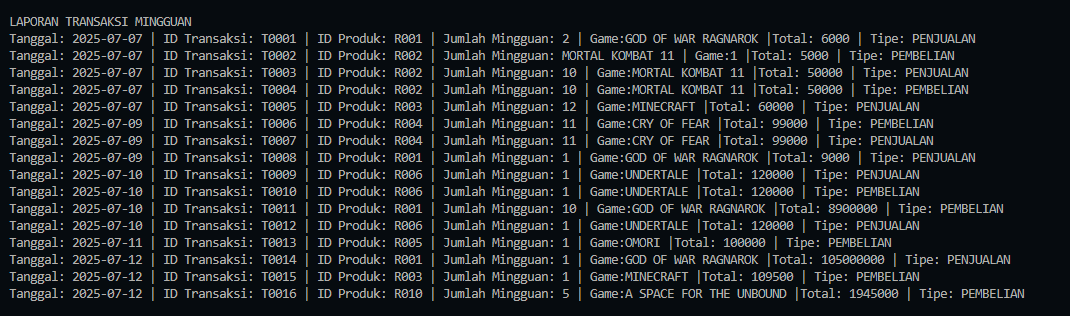


#### **Gambar 3.2 tampilan hasil transaksi pada transaksi.csv**

#### **Laporan Transaksi**



#### **Gambar 3.2 tampilan input-output laporan transaksi**



#### **Gambar 3.2 output laporan transaksi**

## c. Pengujian Fungsional

|  |  |  |
| --- | --- | --- |
| Fitur | Status | Keterangan |
| Tambah Produk | Berhasil | Validasi input dan penulisan ke CSV berjalan baik |
| Update Produk | Berhasil | Produk diubah dari CSV sesuai inputan pengguna |
| Hapus Produk | Berhasil | Produk dihapus dari CSV sesuai ID |
| Cari Produk | Berhasil | Mencari berdasarkan nama game |
| Transaksi Penjualan | Berhasil | Stok dikurangi dan transaksi disimpan |
| Transaksi Pembelian | Berhasil | Stok ditambah dan transaksi dicatat |
| Laporan Harian/Mingguan/Bulanan | Berhasil | Filter berdasarkan tanggal berfungsi |

#### **Tabel 3.2 penguji**an fungsional proyek

## d. Pengujian Integrasi

* Semua modul saling terhubung dengan baik melalui main.py.
* Uji coba skenario lengkap:

1. Tambah produk
2. Lakukan transaksi
3. Lihat laporan  
   → semua berjalan tanpa error dan file CSV terupdate otomatis.

## e. Catatan Kendala

* **Validasi jumlah:** belum ada pengecekan apakah input jumlah bertipe integer sebelum diproses.
* **Rollback:** belum ada penanganan jika proses transaksi gagal di tengah-tengah (misalnya saat file gagal dibuka).

# BAB IV KESIMPULAN

Aplikasi RayyStore Games telah berhasil dikembangkan sebagai sistem manajemen toko game berbasis *command-line interface* (CLI) dengan memanfaatkan file CSV sebagai media penyimpanan data. Aplikasi ini mampu menjalankan berbagai fungsi penting seperti pengelolaan produk (tambah, ubah, hapus, cari), pencatatan transaksi (penjualan dan pembelian), serta penyajian laporan transaksi berdasarkan periode waktu harian, mingguan, dan bulanan.

Berdasarkan hasil uji coba dan analisis fungsionalitas, aplikasi menunjukkan kinerja yang stabil dan dapat dijalankan tanpa error selama semua input diberikan sesuai format. Namun, terdapat beberapa hal yang dapat dijadikan rekomendasi pengembangan untuk versi mendatang, antara lain:

1. **Peningkatan Fitur Update Produk**  
   Perlu dilakukan revisi logika update agar benar-benar mengganti data lama, bukan hanya menambahkan data baru. Mekanisme pencarian dan penghapusan item lama berdasarkan ID sebaiknya ditambahkan.
2. **Validasi Tipe Data Input**  
   Pengguna dapat memasukkan data non-numerik pada kolom seperti stok, harga, dan jumlah transaksi. Disarankan untuk menambahkan pengecekan tipe data (misal isdigit() atau try-except) agar lebih aman dan akurat.
3. **Rollback Otomatis saat Transaksi Gagal**  
   Saat transaksi gagal di tengah jalan, sebaiknya sistem bisa membatalkan proses secara otomatis dan tidak menyimpan perubahan parsial ke file CSV.
4. **Antarmuka yang Lebih Ramah Pengguna**  
   Pengembangan antarmuka grafis (GUI) sederhana menggunakan library seperti tkinter atau PyQt dapat dipertimbangkan untuk meningkatkan pengalaman pengguna.
5. **Penerapan Fitur Pencarian dan Filter Lanjutan**  
   Filter berdasarkan rentang tanggal, nama game, atau jenis transaksi bisa ditambahkan untuk meningkatkan fleksibilitas laporan.
6. **Penggunaan Database**  
   Untuk skala yang lebih besar dan sistematis, disarankan mengganti penyimpanan berbasis file CSV dengan sistem basis data seperti SQLite atau PostgreSQL, agar aplikasi lebih efisien dan aman dalam menyimpan data.
7. **Penambahan Fitur Login/Admin**  
   Agar lebih aman, aplikasi dapat dilengkapi sistem autentikasi sederhana untuk membedakan peran pengguna (admin vs staf biasa), terutama saat menghapus atau mengubah data.

Dengan menerapkan rekomendasi tersebut, diharapkan aplikasi RayyStore Games dapat berkembang menjadi sistem manajemen penjualan yang lebih lengkap, fleksibel, dan andal, serta mampu mendukung kebutuhan operasional toko game digital dalam skala yang lebih besar di masa depan.

# DAFTAR PUSTAKA

[1] Python Software Foundation, “csv — CSV File Reading and Writing,” Python 3.12 Documentation. [Online]. Available: <https://docs.python.org/3/library/csv.html>.

[2] Python Software Foundation, “collections — Container datatypes,” Python 3.12 Documentation. [Online]. Available: <https://docs.python.org/3/library/collections.html#collections.deque>.