

**LAPORAN PROYEK AKHIR PRAKTIKUM  
MATA KULIAH ALGORITMA DAN PEMROGRAMAN DASAR**



**MANAJEMEN TOKO MEBEL**

**Oleh:**

**Kelompok 6 B1 2024**

<b>Muhammad Riva Fachrodhiya</b>	<b>2409106053</b>
<b>Alya Mayasha</b>	<b>2409106054</b>
<b>Muhammad Dzaki Rifa'i</b>	<b>2409106056</b>

**PROGRAM STUDI INFORMATIKA  
FAKULTAS TEKNIK  
UNIVERSITAS MULAWARMAN  
SAMARINDA 2024**

## KATA PENGANTAR

Puji syukur kehadiran Tuhan Yang Maha Esa atas limpahan rahmat dan karunia-Nya sehingga kami dapat menyelesaikan laporan proyek akhir praktikum algoritma pemrograman ini yang berjudul “Manajemen Toko Mebel”. Laporan ini merupakan hasil dari pengaplikasian konsep algoritma pemrograman dalam membangun sistem untuk mengelola data toko mebel. Tujuan dari penulisan laporan proyek akhir ini adalah untuk memenuhi penugasan proyek akhir dari praktikum algoritma pemrograman dasar.

Program yang berjudul “Manajemen Toko Mebel” ini merupakan program dengan Bahasa pemrograman python untuk mengelola inventaris dan keuangan toko mebel. Pada program ini terdapat fitur lengkap untuk melakukan operasi CRUD (*Create, Read, Update, Delete*) dengan dua pilihan role yaitu sebagai *owner* yang dapat melakukan semua operasi CRUD termasuk melihat profit dan sebagai pembeli yang dapat melakukan operasi read dan pembayaran dengan fitur *top-up*.

Dalam proses pembuatan program ini, kami menyadari bahwa terdapat beberapa kesulitan atau kendala yang perlu diperhatikan seperti dalam proses pembuatan saldo dan riwayat pembelian yang mengharuskan program untuk *crosscheck* dari file eksternal excel-json. Oleh karena itu segala masukan dan kritik membangun dari pihak-pihak yang terlibat akan sangat berharga guna meningkatkan kualitas dari proyek akhir yang kami buat.

Selanjutnya, kami ingin mengucapkan terima kasih kepada Bapak Muhammad Bambang Firdaus, S.Kom, M.Kom. dan Bapak Awang Harsa Kridalaksana,

S.Kom, M.Kom. selaku dosen pengampu mata kuliah Algoritma Pemrograman Dasar serta Bang Adi Muhammad Syifai dan Bang Ifnu Umar selaku asisten laboratorium yang telah memberikan dukungan dan bantuan dalam proses pengerjaan proyek akhir praktikum ini. Baik itu berupa saran, maupun dukungan moril yang sangat berarti bagi kami dalam melakukan pengerjaan proyek akhir praktikum yang berjudul “Manajemen Toko Mebel” ini

Samarinda, 20 November 2024

Kelompok 6

## TAKARIR

<i>Database</i>	Basis Data
<i>Managemen</i>	Mengatur
<i>Input</i>	Memasukkan
Owner	Pemilik
Profit	Keuntungan
Transaction	Transaksi
Login	Masuk
Register	Daftar
Riwayat	Sejarah
Excel	Lembar Kerja Elektronik
JSON	Notasi Objek JavaScript
Input	Masukan
Output	Keluaran
User-friendly	Ramah Pengguna
Library	Pustaka
Error Handling	Penanganan Kesalahan
Authentication	Autentikasi
Implementation	Implementasi
Interactive	Interaktif
Terminal	Antarmuka Baris Perintah
Menu	Daftar Pilihan
Top-Up	Isi Ulang
Module	Modul
System	Sistem

## DAFTAR ISI

KATA PENGANTAR .....	ii
TAKARIR.....	iv
DAFTAR ISI.....	v
DAFTAR GAMBAR .....	vi
BAB I PENDAHULUAN .....	1
1.1 Latar Belakang .....	1
1.2 Kebutuhan Fungsional.....	1
1.3 Rumusan Masalah .....	2
1.4 Batasan Masalah .....	2
1.5 Tujuan .....	2
BAB II PERANCANGAN.....	4
2.1 Analisis Program.....	4
2.2 Flowchart .....	6
2.3 Konsep/Materi Praktikum yang dipakai .....	10
BAB III HASIL DAN PEMBAHASAN .....	12
3.1 Tampilan Program.....	12
1. Tampilan <i>Menu Utama</i> .....	12
2. Tampilan <i>Menu Login</i> .....	12
3. Tampilan <i>Menu Owner</i> .....	12
4. Tampilan <i>Menu Register</i> .....	16
5. Tampilan <i>Menu Pembeli</i> .....	17
3.2 Source Code .....	18
BAB IV PENUTUP .....	37
4.1 Kesimpulan .....	37
4.2 Saran .....	37
DAFTAR PUSTAKA .....	39
LAMPIRAN.....	40

## DAFTAR GAMBAR

Gambar 3. 1 Tampilan <i>menu</i> utama .....	12
Gambar 3. 2 Tampilan <i>menu</i> login.....	12
Gambar 3. 3 Tampilan <i>menu</i> owner.....	12
Gambar 3. 4 Menu <i>owner</i> Tambah data.....	13
Gambar 3. 5 Menu <i>owner</i> lihat data.....	14
Gambar 3. 6 Menu <i>owner</i> perbarui data .....	15
Gambar 3. 7 Menu <i>owner</i> hapus produk.....	15
Gambar 3. 8 Menu <i>owner</i> melihat profit.....	16
Gambar 3. 9 Tampilan menu <i>register</i> .....	16
Gambar 3. 10 Tampilkan menu pembeli.....	17
Gambar 3. 11 Tampilan riwayat pembelian.....	17
Gambar 3. 12 fungsi <i>input</i> tengah.....	18
Gambar 3. 13 Fungsi print tengah.....	18
Gambar 3. 14 Fungsi lanjut.....	18
Gambar 3. 15 Fungsi baca user .....	19
Gambar 3. 16 Simpan user data .....	19
Gambar 3. 17 Fungsi simpan data produk .....	19
Gambar 3. 18 fungsi membersihkan terminal .....	20
Gambar 3. 19 Fungsi loading .....	20
Gambar 3. 20 Fungsi load data .....	20
Gambar 3. 21 Fungsi lihat produk.....	21
Gambar 3. 22 Fungsi tambah produk.....	21
Gambar 3. 23 Fungsi perbarui produk .....	22
Gambar 3. 24 Fungsi hapus produk .....	23
Gambar 3. 25 Fungsi hitung profit.....	23
Gambar 3. 26 Fungsi pembelian .....	24
Gambar 3. 27 Fungsi tampilkan riwayat .....	25
Gambar 3. 28 fungsi top up saldo .....	25
Gambar 3. 29 fungsi lihat saldo .....	26
Gambar 3. 30 Fungsi tampilan <i>menu</i> biasa .....	26
Gambar 3. 31 fungsi tampilkan <i>menu</i> utama .....	27
Gambar 3. 32 Fungsi tampilkan pembeli <i>menu</i> .....	28
Gambar 3. 33 Fungsi tampilkan <i>owner menu</i> .....	29
Gambar 3. 34 Fungsi <i>owner menu</i> .....	30
Gambar 3. 35 Fungsi pembeli <i>menu</i> .....	31
Gambar 3. 36 Fungsi main .....	32

Gambar 3. 37 Fungsi daftarUser .....	32
Gambar 3. 38 Fungsi cekUsername .....	33
Gambar 3. 39 fungsi cekPassword .....	33
Gambar 3. 40 Fungsi Login .....	34
Gambar 3. 41 Fungsi tambahUser .....	34
Gambar 3. 42 Fungsi <i>register</i> .....	35

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Manajemen stok dan transaksi penjualan merupakan aspek yang penting dalam berjalannya operasional toko, termasuk toko mebel. Seiring berkembangnya era digitalisasi, manajemen dapat dilakukan secara efektif menggunakan teknologi berbasis komputer. Salah satu cara mengimplementasikannya adalah dengan menggunakan program sederhana python untuk membantu mengelola stok barang, memproses pembelian, menghitung *profit*, dan menyimpan riwayat transaksi.

Dalam proyek akhir ini kami merancang sebuah sistem berbasis Python untuk mempermudah manajemen toko mebel. Program ini memanfaatkan beberapa Pustaka python seperti Pandas, Tabulate, dan modul bawaan seperti os, time, serta json. Kami merancang program ini agar dapat memproses data secara dinamis, menyimpan dan memanipulasi data melalui file, serta menampilkan hasil dalam format yang *user-friendly*.

### **1.2 Kebutuhan Fungsional**

Sistem manajemen toko mebel ini dirancang untuk memenuhi kebutuhan fungsional berikut:

1. Pengelolaan produk seperti menambah, membaca, memperbarui, dan menghapus data produk.
2. Perhitungan *profit* dengan menghitung total keuntungan berdasarkan data produk.
3. Pengelolaan transaksi dengan membantu pencatatan pembelian oleh pengguna.
4. Manajemen akun pengguna dengan menampilkan saldo, riwayat pembelian, dan fitur *top-up* saldo.



5. Pengolahan data dengan menyimpan dan memuat data produk serta pengguna dari file eksternal (Excel/JSON).

### 1.3 Rumusan Masalah

1. Bagaimana cara merancang sistem yang dapat melakukan operasi CRUD (*create, read, update, delete*) data produk secara efisien?
2. Bagaimana cara sistem menghitung *profit* berdasarkan data produk yang tersedia?
3. Bagaimana sistem dapat mengintegrasikan manajemen pengguna, seperti pencatatan saldo dan riwayat pembelian dalam satu program yang terstruktur?

### 1.4 Batasan Masalah

Program Manajemen Toko Mebel ini memiliki beberapa batasan:

1. Program ini hanya dapat diakses pada *platform* berbasis Python tanpa antarmuka grafis.
2. Data produk disimpan dalam file Excel (*produk.xlsx*), sementara data pengguna disimpan dalam file JSON (*datausername.json*).
3. Program tidak dilengkapi dengan autentikasi keamanan tingkat lanjut untuk data pengguna.

### 1.5 Tujuan

Adapun tujuan dari pembuatan program Manajemen Toko Mebel meliputi:

1. Membuat sistem manajemen toko mebel yang mencakup fungsi CRUD (*create, read, update, delete*).

2. Menyediakan fitur tambahan seperti perhitungan *profit*, *top-up* saldo pengguna, dan riwayat pembelian.
3. Mengimplementasikan pustaka python untuk memanipulasi data secara efisien dan interaktif.
4. Membantu pengguna mengelola data produk dan transaksi secara lebih mudah dan terorganisasi.

## BAB II

### PERANCANGAN

#### 2.1 Analisis Program

Program ini merupakan sistem yang dirancang untuk mengelola akses pengguna (*login* dan *register*) serta menyediakan *menu* utama berdasarkan peran pengguna (*owner* atau pembeli). Program menggunakan fungsi dari modul *menu*, *login*, *regis* dan CRUD yang berisi implementasi untuk tampilan *menu*, proses autentikasi, *registrasi*, pengelolaan produk, transaksi pembelian, perhitungan *profit*, dan *top-up* saldo pengguna. Modul yang digunakan pada program ini meliputi:

1. *Menu*: berisi fungsi untuk menampilkan *menu* utama dan *menu* spesifik berdasarkan peran pengguna.
2. *Login*: berisi fungsi untuk proses *login*, termasuk memverifikasi *username* dan *password*.
3. *Regis*: berisi fungsi untuk proses *registrasi* pengguna baru.
4. *CRUD*: berisi fungsi untuk mengelola produk (menambah, membaca, memperbarui, dan menghapus).

Fungsi main sebagai fungsi utama bertanggung jawab untuk menampilkan *menu* utama melalui fungsi *tampilkanMenuutama* dimana pengguna dapat memilih opsi :

- 1 untuk *login*.
- 2 untuk *registrasi*.
- 0 untuk keluar dari program.

Jika *login* berhasil, program akan memeriksa peran pengguna:

- *Owner* akan dialihkan ke *menu* pemilik.
- Pembeli akan dialihkan ke *menu* pembeli.

Jika memilih *register*, maka fungsi *register* akan dipanggil untuk menambah pengguna baru. Setelah selesai, fungsi lanjut digunakan untuk menampilkan prompt melanjutkan ke langkah berikutnya.

Apabila pengguna memilih keluar, program akan menampilkan pesan perpisahan dan keluar menggunakan fungsi *exit*.

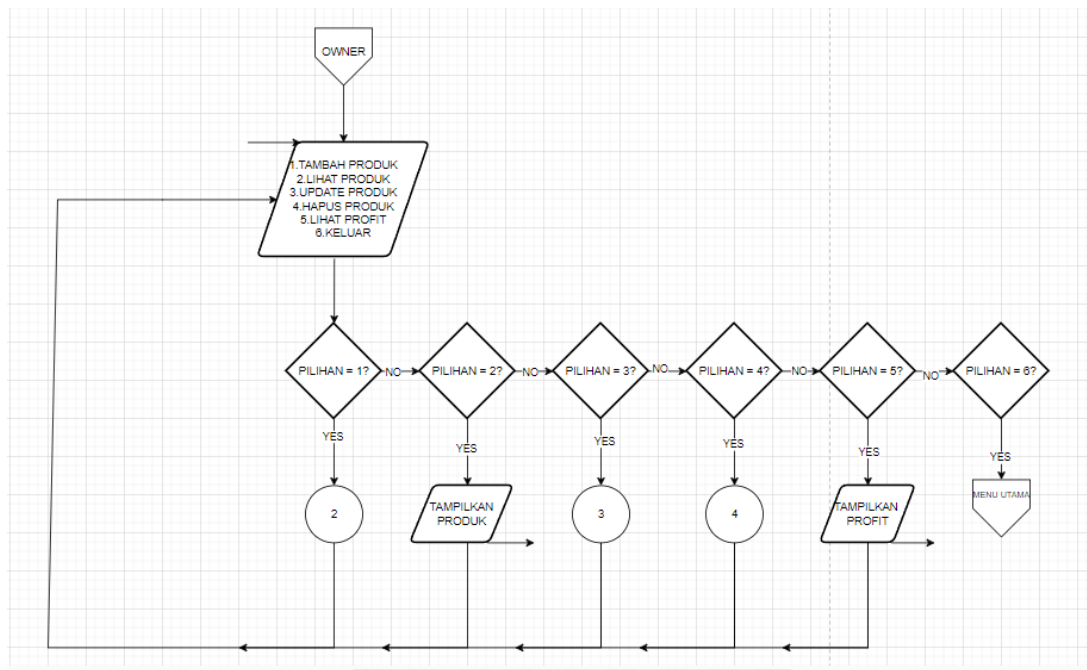
## 2.2 Flowchart

### chart Menu Utama

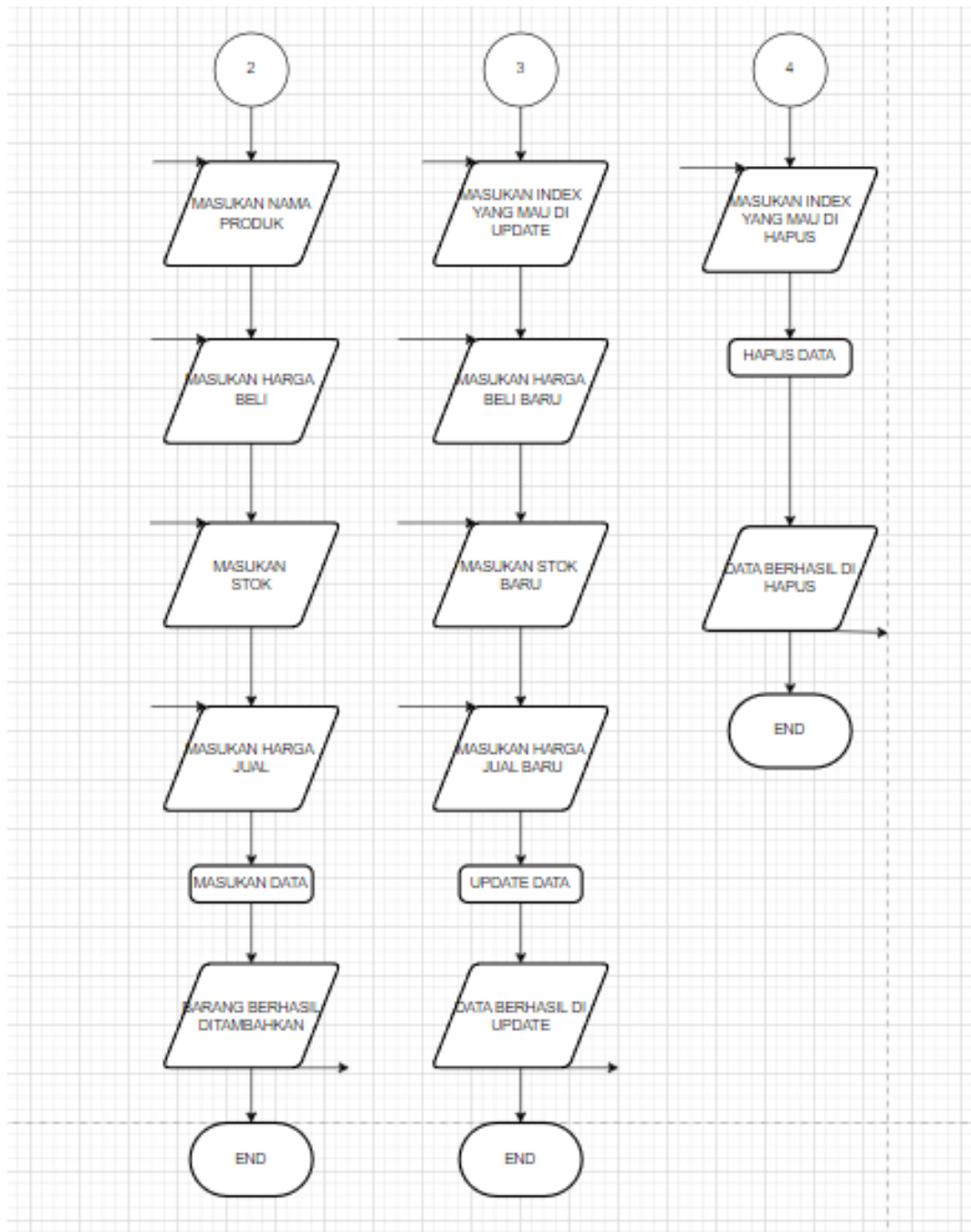
Flowchart dimulai dengan *start* dimana pengguna diminta untuk memilih salah satu opsi yaitu *login*, *register*, atau keluar. Apabila pengguna memilih *login* dengan menginput 1, maka pengguna akan diminta memasukkan *username* dan *password*, lalu akan diarahkan ke *menu owner* apabila login menggunakan akun *owner* dan ke akses akun pembeli apabila akun selain *owner*.

Apabila pengguna memilih *register* dengan menginput 2, maka akan diminta menginput username, dan jika username tersedia pengguna akan melanjutkan dengan menginput password dan user baru berhasil ditambahkan. Namun jika username tidak tersedia, maka akan terjadi perulangan ke *input* username.

Jika pengguna menginput 0, maka program akan berhenti sesuai dengan ketentuan



Gambar 2.2 Flowchart Menu Owner



Gambar 2.3 Flowchart Menu Owner (2)

Apabila pengguna telah berhasil *login* dengan akun *owner*, maka akan diberikan pilihan untuk dapat menambah, melihat, memperbarui, dan menghapus produk serta melihat *profit*. Jika pengguna memilih pilihan 1 atau *menu* menambah produk, maka

pengguna akan diminta menginput nama produk, *stock* produk, dan harga jual, setelah data berhasil ditambahkan maka program akan menampilkan sebuah *output* yang memberikan tanda bahwa data telah berhasil ditambahkan.

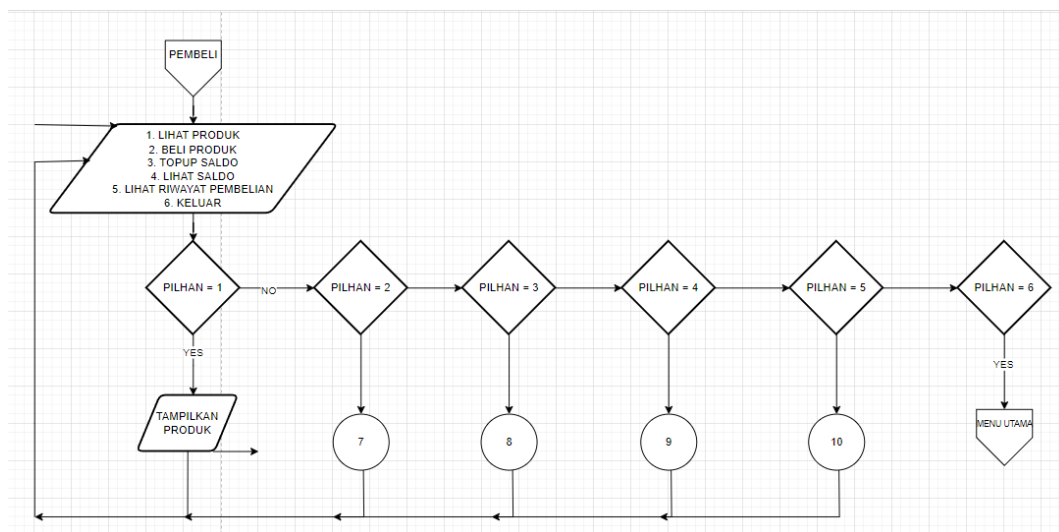
Jika pengguna memilih pilihan 2 atau *menu* lihat produk, maka program akan menampilkan data produk yang tertera.

Jika pengguna memilih pilihan 3 atau *menu update*, pengguna akan diminta memasukkan index dari data yang ingin diupdate lalu pengguna akan memasukkan harga beli baru, *stock* baru, dan harga jual baru. Setelah data berhasil diperbarui maka program akan menampilkan sebuah *output* yang memberikan tanda bahwa data berhasil diperbarui.

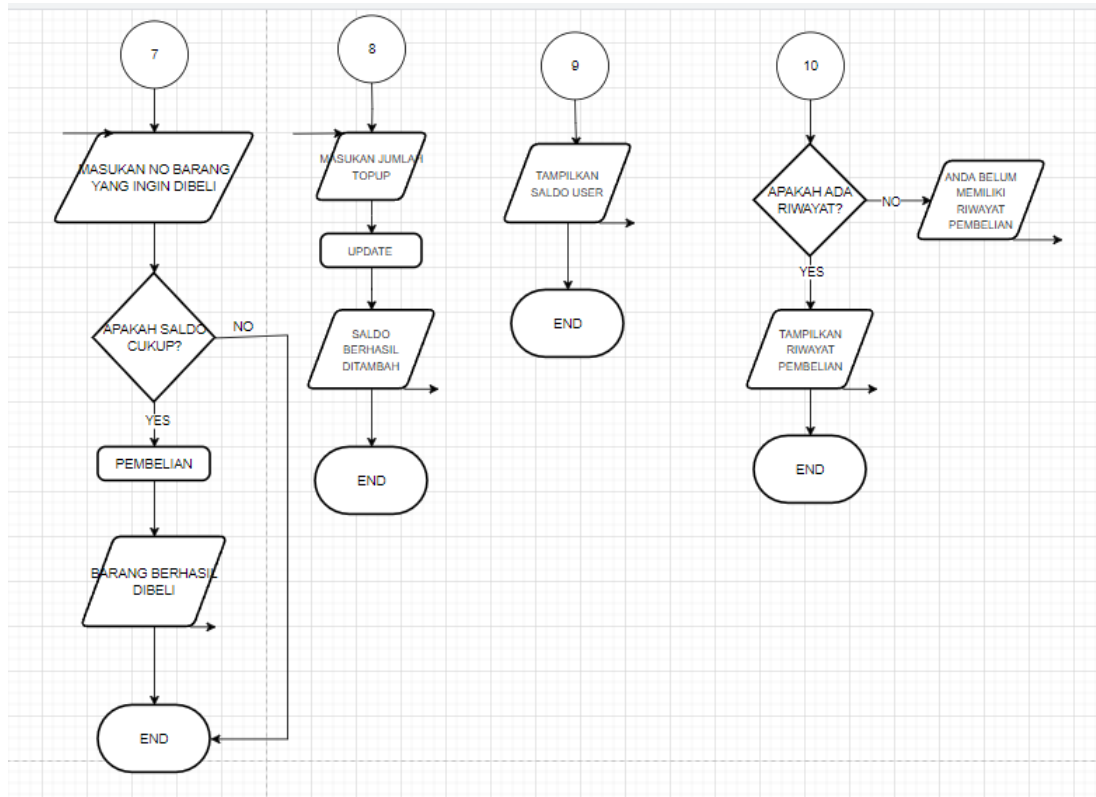
Jika pengguna memilih pilihan 4 atau *menu* menghapus produk, maka pengguna akan diminta index dari data yang ingin dihapus, setelah data berhasil dihapus maka program akan menampilkan sebuah *output* yang memberikan tanda bahwa data berhasil dihapus.

Jika pengguna memilih pilihan 5 atau *menu* melihat profit, maka program akan menampilkan data profit yang tersedia.

Jika pengguna memilih pilihan 6 atau keluar maka akan kembali pada *menu* utama.



Gambar 2.4 Flowchart Menu Pembeli



Gambar 2.5 Flowchart Menu Pembeli (2)

Apabila pengguna telah berhasil *login* dengan akun pembeli, maka akan diberikan pilihan untuk melihat produk, membeli produk, *top-up* saldo, melihat nominal saldo, dan riwayat pembelian. Jika pengguna memilih pilihan 1 atau *menu* lihat produk, maka program akan menampilkan data produk.

Jika pengguna memilih pilihan 2 atau *menu* membeli produk, maka pengguna akan diminta memasukkan no index dari produk yang akan dibeli dan program akan melakukan pengecekan apakah saldo cukup untuk membeli barang tersebut, apabila saldo cukup maka proses transaksi barang akan berhasil.

Jika pengguna memilih pilihan 3 atau *menu top-up* saldo, pengguna akan diminta memasukkan jumlah nominal saldo yang ingin diisi dan program akan memperbarui jumlah saldo pengguna.

Jika pengguna memilih pilihan 4 atau *menu* lihat saldo, maka program akan menampilkan nominal saldo pengguna.



Jika pengguna memilih pilihan 5 atau *menu* riwayat pembelian, maka program akan melakukan pengecekan apakah pengguna pernah melakukan transaksi pembelian sebelumnya, jika iya maka program akan menampilkan data riwayat pembelian.

Jika pengguna memilih pilihan 6 atau keluar maka akan kembali pada *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 yang kami gunakan adalah fungsi print.

Fungsi print pada program merupakan fungsi yang umum dipakai untuk menampilkan suatu keluaran pada layar peraga.

2. Fungsi percabangan: pada program kami menggunakan fungsi percabangan *if/elif/else* yang digunakan untuk menentukan tindakan dan perintah sesuai logika/kondisi yang kita berikan.

3. Fungsi perulangan: pada program kami menggunakan fungsi perulangan *for* dan *while* sebagai instruksi yang digunakan untuk memerintahkan komputer melakukan sesuatu secara berulang-ulang dengan jumlah yang ditentukan atau selama masih memenuhi suatu kondisi yang telah ditentukan.

4. Fungsi list: pada program kami menggunakan fungsi list untuk menyimpan data *user*.

5. Fungsi *dictionary*: pada program kami menggunakan fungsi *dictionary* untuk menampung data *user*.

6. Fungsi *def*: pada program kami menggunakan fungsi *def* dengan tujuan memecah program besar menjadi sub program yang lebih sederhana, agar masing-masing fitur pada program dapat dibuat dalam satu fungsi.

7. *Error handling*: pada program kami menggunakan *try-except* untuk menangani error yang mungkin terjadi selama eksekusi program, salah satunya pada *menu login* dan *register* agar pengguna dapat memasukkan *username* dan *password* sesuai ketentuan.
8. File eksternal: pada program kami menggunakan file excel (*produk.xlsx*) untuk menyimpan data produk, file JSON (*datausername.json*) untuk menyimpan nama pengguna, saldo pengguna, dan riwayat pembelian pengguna..
9. Library: pada program kami menggunakan *library* pandas untuk membaca dan menyimpan data dalam format excel, menggunakan *tabulate* untuk merapikan tampilan, menggunakan *prettytable* untuk merapikan *menu* terminal kami, *os* dan *shutil* untuk memastikan keberadaan file sesuai dengan ukuran terminal.

## BAB III

### HASIL DAN PEMBAHASAN

#### 3.1 Tampilan Program

##### 1. Tampilan Menu Utama

```
+-----+
|               SELAMAT DATANG DI MENU UTAMA               |
+-----+
|               1.Login                                     |
|               2.Register                                   |
|               0.Keluar                                    |
+-----+
|               Pilihan anda ->                |
+-----+
```

Gambar 3. 1 Tampilan menu utama

##### 2. Tampilan Menu Login

```
+-----+
|               >> MENU LOGIN <<               |
+-----+
|               Username -> Admin1                 |
|               Password -> Admin1         |
+-----+
```

Gambar 3. 2 Tampilan menu login

##### 3. Tampilan Menu Owner

No	Menu Toko Mebel
1	Tambah Produk
2	Lihat Produk
3	Update Produk
4	Hapus Produk
5	Lihat Profit
6	Keluar

Pilih menu (1-6):

Gambar 3. 3 Tampilan menu owner

No	Nama	Harga Beli	Stok	Harga Jual
1	IKEA LINNMON - MEJA KERJA	1200000	8	2000000
2	OLYMPIC STUDY DESK - MEJA BELAJAR	500000	6	750000
3	INFORMA FIONA - MEJA TAMU	1500000	11	1750000
4	MODERA MAESTRO - MEJA KOMPUTER	1200000	9	1400000
5	LIVIVEN DINING TABLE - MEJA MAKAN	2000000	5	2300000
6	OLYMPIC - LEMARI PAKAIAN 2 PINTU	1500000	6	1600000
7	INFROMA GRAYSON - LEMARI SLIDING 3 PINTU	3000000	4	3500000
8	FABELIO CLASSIC - LEMARI KAYU JATI	4500000	2	5000000
9	NAPOLLY - LEMARI PLASTIK	300000	11	450000
10	IKEA BILLY - LEMARI BUKU	1000000	10	1250000
11	ERGOTEC - KURSI KERJA	1200000	12	1450000
12	REXUS RGC - KURSI GAMING	1500000	12	1750000
13	OLYMPIC - KURSI KAYU LIPAT	200000	8	300000
14	INFROMA OPAL - KURSI MAKAN	300000	13	450000
15	LIVIEN - KURSI TAMU MINIMALIS	750000	10	1000000

Masukkan nama barang: INFORMA - MEJA KOMPUTER  
 Masukkan harga barang: 1300000  
 Masukkan stok barang: 15  
 Masukkan harga pokok barang: 1500000

Gambar 3. 4 Menu owner Tambah data

No	Nama	Harga Beli	Stok	Harga Jual
1	IKEA LINNMON - MEJA KERJA	1200000	8	2000000
2	OLYMPIC STUDY DESK - MEJA BELAJAR	500000	6	750000
3	INFORMA FIONA - MEJA TAMU	1500000	11	1750000
4	MODERA MAESTRO - MEJA KOMPUTER	1200000	9	1400000
5	LIVIVEN DINING TABLE - MEJA MAKAN	2000000	5	2300000
6	OLYMPIC - LEMARI PAKAIAN 2 PINTU	1500000	6	1600000
7	INFROMA GRAYSON - LEMARI SLIDING 3 PINTU	3000000	4	3500000
8	FABELIO CLASSIC - LEMARI KAYU JATI	4500000	2	5000000
9	NAPOLLY - LEMARI PLASTIK	300000	11	450000
10	IKEA BILLY - LEMARI BUKU	1000000	10	1250000
11	ERGOTEC - KURSI KERJA	1200000	12	1450000
12	REXUS RGC - KURSI GAMING	1500000	12	1750000
13	OLYMPIC - KURSI KAYU LIPAT	200000	8	300000
14	INFROMA OPAL - KURSI MAKAN	300000	13	450000
15	LIVIEN - KURSI TAMU MINIMALIS	750000	10	1000000
16	INFORMA - MEJA KOMPUTER	1300000	15	1500000

Tekan Enter untuk melanjutkan. . . .

Gambar 3. 5 Menu owner lihat data

No	Nama	Harga Beli	Stok	Harga Jual
1	IKEA LINNMON - MEJA KERJA	1200000	8	2000000
2	OLYMPIC STUDY DESK - MEJA BELAJAR	500000	6	750000
3	INFORMA FIONA - MEJA TAMU	1500000	11	1750000
4	MODERA MAESTRO - MEJA KOMPUTER	1200000	9	1400000
5	LIVIVEN DINING TABLE - MEJA MAKAN	2000000	5	2300000
6	OLYMPIC - LEMARI PAKAIAN 2 PINTU	1500000	6	1600000
7	INFROMA GRAYSON - LEMARI SLIDING 3 PINTU	3000000	4	3500000
8	FABELIO CLASSIC - LEMARI KAYU JATI	4500000	2	5000000
9	NAPOLLY - LEMARI PLASTIK	300000	11	450000
10	IKEA BILLY - LEMARI BUKU	1000000	10	1250000
11	ERGOTEC - KURSI KERJA	1200000	12	1450000
12	REXUS RGC - KURSI GAMING	1500000	12	1750000
13	OLYMPIC - KURSI KAYU LIPAT	200000	8	300000
14	INFROMA OPAL - KURSI MAKAN	300000	13	450000
15	LIVIEN - KURSI TAMU MINIMALIS	750000	10	1000000
16	INFORMA - MEJA KOMPUTER	1300000	15	1500000

Masukkan index barang yang ingin diupdate: 16  
Masukkan harga beli baru: 1200000  
Masukkan stok baru: 18  
Masukkan harga jual baru: 1650000

Gambar 3. 6 Menu owner perbarui data

No	Nama	Harga Beli	Stok	Harga Jual
1	IKEA LINNMON - MEJA KERJA	1200000	8	2000000
2	OLYMPIC STUDY DESK - MEJA BELAJAR	500000	6	750000
3	INFORMA FIONA - MEJA TAMU	1500000	11	1750000
4	MODERA MAESTRO - MEJA KOMPUTER	1200000	9	1400000
5	LIVIVEN DINING TABLE - MEJA MAKAN	2000000	5	2300000
6	OLYMPIC - LEMARI PAKAIAN 2 PINTU	1500000	6	1600000
7	INFROMA GRAYSON - LEMARI SLIDING 3 PINTU	3000000	4	3500000
8	FABELIO CLASSIC - LEMARI KAYU JATI	4500000	2	5000000
9	NAPOLLY - LEMARI PLASTIK	300000	11	450000
10	IKEA BILLY - LEMARI BUKU	1000000	10	1250000
11	ERGOTEC - KURSI KERJA	1200000	12	1450000
12	REXUS RGC - KURSI GAMING	1500000	12	1750000
13	OLYMPIC - KURSI KAYU LIPAT	200000	8	300000
14	INFROMA OPAL - KURSI MAKAN	300000	13	450000
15	LIVIEN - KURSI TAMU MINIMALIS	750000	10	1000000
16	INFORMA - MEJA KOMPUTER	1200000	18	1650000

Masukkan indeks barang yang ingin dihapus: 16

Gambar 3. 7 Menu owner hapus produk

No	Nama	Harga Beli	Stok	Harga Jual	Profit
1	IKEA LINNMON - MEJA KERJA	1200000	8	2000000	6400000
2	OLYMPIC STUDY DESK - MEJA BELAJAR	500000	6	750000	1500000
3	INFORMA FIONA - MEJA TAMU	1500000	11	1750000	2750000
4	MODERA MAESTRO - MEJA KOMPUTER	1200000	9	1400000	1800000
5	LIVIVEN DINING TABLE - MEJA MAKAN	2000000	5	2300000	1500000
6	OLYMPIC - LEMARI PAKAIAN 2 PINTU	1500000	6	1600000	600000
7	INFROMA GRAYSON - LEMARI SLIDING 3 PINTU	3000000	4	3500000	2000000
8	FABELIO CLASSIC - LEMARI KAYU JATI	4500000	2	5000000	1000000
9	NAPOLLY - LEMARI PLASTIK	300000	11	450000	1650000
10	IKEA BILLY - LEMARI BUKU	1000000	10	1250000	2500000
11	ERGOTEC - KURSI KERJA	1200000	12	1450000	3000000
12	REXUS RGC - KURSI GAMING	1500000	12	1750000	3000000
13	OLYMPIC - KURSI KAYU LIPAT	200000	8	300000	800000
14	INFROMA OPAL - KURSI MAKAN	300000	13	450000	1950000
15	LIVIEN - KURSI TAMU MINIMALIS	750000	10	1000000	2500000

-----  
Total Profit: Rp 32,950,000  
-----  
Tekan Enter untuk melanjutkan. . . .

Gambar 3. 8 Menu owner melihat profit

#### 4. Tampilan Menu Register

```

+-----+
|               >> MENU REGISTER <<               |
+-----+
+-----+
| Username -> User1                                   |
| Password -> User1[]                               |
+-----+

```

Gambar 3. 9 Tampilan menu register

## 5. Tampilan Menu Pembeli

No	Nama	Harga Beli	Stok	Harga Jual
1	IKEA LINNMON - MEJA KERJA	1200000	8	2000000
2	OLYMPIC STUDY DESK - MEJA BELAJAR	500000	6	750000
3	INFORMA FIONA - MEJA TAMU	1500000	11	1750000
4	MODERA MAESTRO - MEJA KOMPUTER	1200000	9	1400000
5	LIVIVEN DINING TABLE - MEJA MAKAN	2000000	5	2300000
6	OLYMPIC - LEMARI PAKAIAN 2 PINTU	1500000	6	1600000
7	INFROMA GRAYSON - LEMARI SLIDING 3 PINTU	3000000	4	3500000
8	FABELIO CLASSIC - LEMARI KAYU JATI	4500000	2	5000000
9	NAPOLLY - LEMARI PLASTIK	300000	11	450000
10	IKEA BILLY - LEMARI BUKU	1000000	10	1250000
11	ERGOTEC - KURSI KERJA	1200000	12	1450000
12	REXUS RGC - KURSI GAMING	1500000	12	1750000
13	OLYMPIC - KURSI KAYU LIPAT	200000	8	300000
14	INFROMA OPAL - KURSI MAKAN	300000	13	450000
15	LIVIEN - KURSI TAMU MINIMALIS	750000	10	1000000

Masukkan nomor produk yang ingin dibeli: 1  
Saldo tidak mencukupi.  
Tekan Enter untuk melanjutkan. . . .

Gambar 3. 10 Tampilkan menu pembeli


ID	TANGGAL	PRODUK	HARGA
1	2024-11-22 17:51:53	IKEA LINNMON - MEJA KERJA	2000000

Tekan Enter untuk melanjutkan. . . .

Gambar 3. 11 Tampilan riwayat pembelian




### 3.2 Source Code



```
1 def inputTengah(kata):
2     lebarTerminal = shutil.get_terminal_size().columns
3     sampling = (lebarTerminal - len(kata)) // 2
4     print(" " * sampling + kata, end="")
5
6     userInput = input()
7     return userInput
```

Gambar 3. 12 fungsi input tengah



```
1 def print_tengah(text):
2     lebarTerminal = shutil.get_terminal_size().columns
3     teks = text.center(lebarTerminal)
4     print(teks)
```

Gambar 3. 13 Fungsi print tengah



```
1 def lanjut():
2     print_tengah("Tekan Enter untuk melanjutkan. . . .")
3     input("")
```

Gambar 3. 14 Fungsi lanjut

```

1 def bacaUser():
2     if not os.path.exists('datausername.json'):
3         raise FileNotFoundError("Data user tidak ditemukan")
4     with open('datausername.json', 'r') as file:
5         return json.load(file)
6

```

*Gambar 3. 15 Fungsi baca user*

```

1 def simpanUserData(data, file_path='datausername.json'):
2     with open(file_path, 'w') as file:
3         json.dump(data, file, indent=4)

```

*Gambar 3. 16 Simpan user data*

```

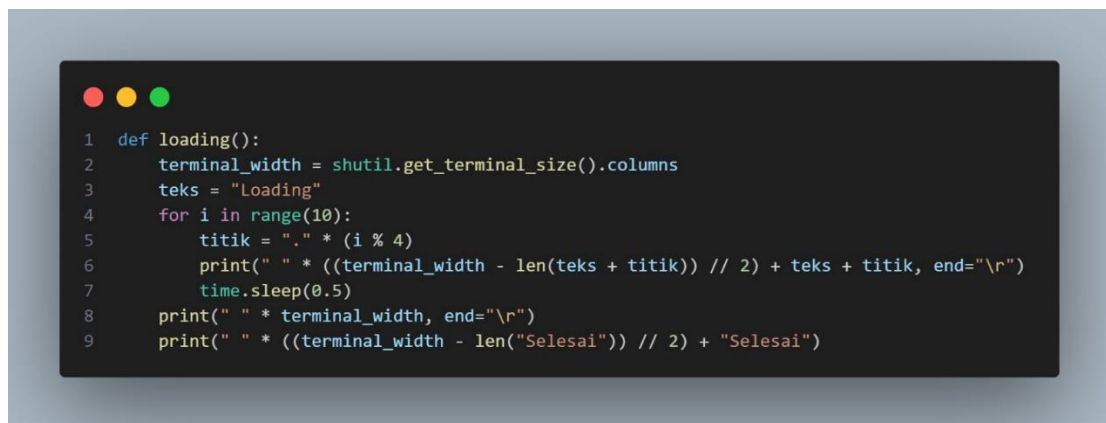
1 def simpanDataProduk(df):
2     df.reset_index(drop=True, inplace=True)
3     df['No'] = df.index + 1
4     df = df[['No', 'Nama', 'Harga Beli', 'Stok', 'Harga Jual']]
5     df.to_excel('produk.xlsx', index=False)
6

```

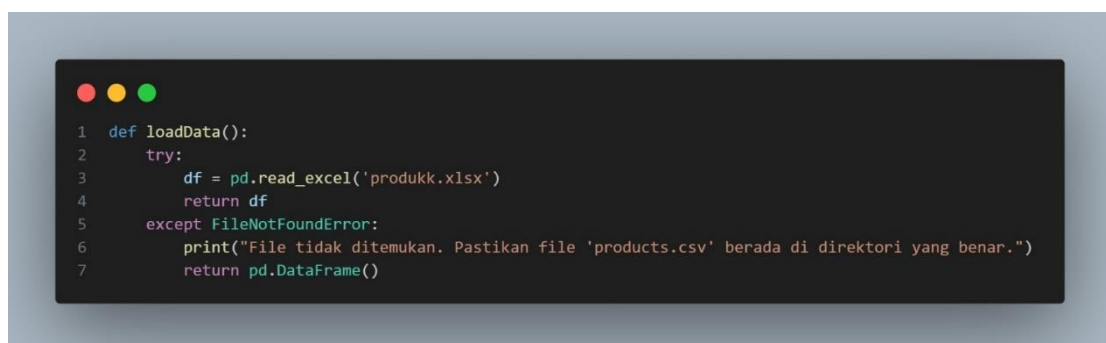
*Gambar 3. 17 Fungsi simpan data produk*



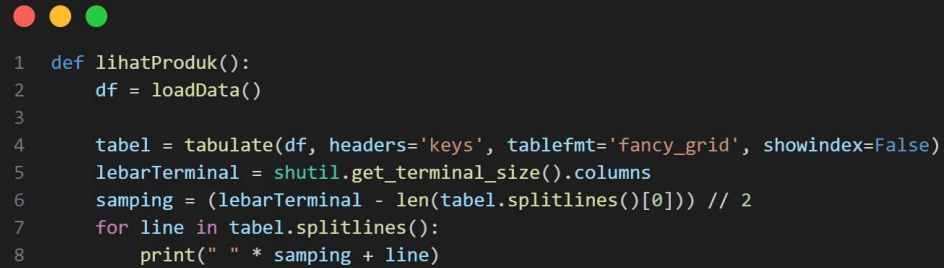
Gambar 3. 18 fungsi membersihkan terminal



Gambar 3. 19 Fungsi loading



Gambar 3. 20 Fungsi load data

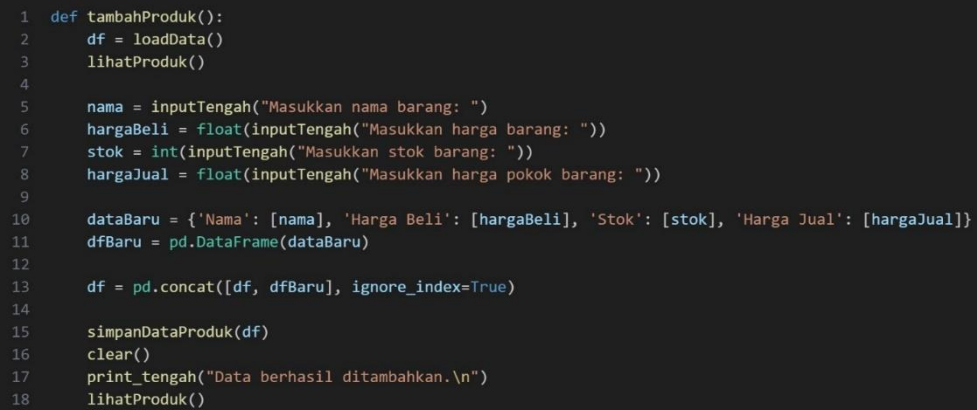


```

1 def lihatProduk():
2     df = loadData()
3
4     tabel = tabulate(df, headers='keys', tablefmt='fancy_grid', showindex=False)
5     lebarTerminal = shutil.get_terminal_size().columns
6     sampling = (lebarTerminal - len(tabel.splitlines()[0])) // 2
7     for line in tabel.splitlines():
8         print(" " * sampling + line)

```

Gambar 3. 21 Fungsi lihat produk



```

1 def tambahProduk():
2     df = loadData()
3     lihatProduk()
4
5     nama = inputTengah("Masukkan nama barang: ")
6     hargaBeli = float(inputTengah("Masukkan harga barang: "))
7     stok = int(inputTengah("Masukkan stok barang: "))
8     hargaJual = float(inputTengah("Masukkan harga pokok barang: "))
9
10    dataBaru = {'Nama': [nama], 'Harga Beli': [hargaBeli], 'Stok': [stok], 'Harga Jual': [hargaJual]}
11    dfBaru = pd.DataFrame(dataBaru)
12
13    df = pd.concat([df, dfBaru], ignore_index=True)
14
15    simpanDataProduk(df)
16    clear()
17    print_tengah("Data berhasil ditambahkan.\n")
18    lihatProduk()

```

Gambar 3. 22 Fungsi tambah produk

```

1  def updateProduk():
2      df = loadData()
3      lihatProduk()
4      try:
5          indexNo = int(inputTengah("Masukkan index barang yang ingin diupdate: "))
6          trueindex = indexNo - 1
7          if 0 <= trueindex < len(df):
8              hargaBeliBaru = float(inputTengah("Masukkan harga beli baru: "))
9              stok_baru = int(inputTengah("Masukkan stok baru: "))
10             hargaJualBaru = float(inputTengah("Masukkan harga jual baru: "))
11
12             df.at[trueindex, 'Harga Beli'] = hargaBeliBaru
13             df.at[trueindex, 'Stok'] = stok_baru
14             df.at[trueindex, 'Harga Jual'] = hargaJualBaru
15
16             df.to_excel('produk.xlsx', index=False)
17
18             inputTengah("Produk berhasil diubah! Tekan Enter untuk melanjutkan...")
19             clear()
20             lihatProduk()
21             print_tengah("Data berhasil diupdate.\n")
22         else:
23             print_tengah("Index tidak valid.\n")
24     except ValueError as e:
25         clear()
26         print_tengah(f"Input tidak valid: {e}\n")
27     except Exception as e:
28         clear()
29         print_tengah(f"Terjadi kesalahan: {e}\n")

```

Gambar 3. 23 Fungsi perbarui produk

```

1 def hapusProduk():
2     df = loadData()
3     lihatProduk()
4     try:
5         indexNo = int(inputTengah("Masukkan indeks barang yang ingin dihapus: "))
6         trueindex = indexNo - 1
7         if 0 <= trueindex < len(df):
8             df.drop(index=trueindex, inplace=True)
9             simpanDataProduk(df)
10            print_tengah("Data berhasil dihapus.\n")
11        else:
12            print_tengah("Indeks tidak valid.\n")
13    except ValueError:
14        clear()
15        print("Input tidak valid. Harus berupa angka.\n")
16    except Exception as e:
17        clear()
18        print(f"Terjadi kesalahan: {e}\n")

```

Gambar 3. 24 Fungsi hapus produk

```

1 def hitungProfit():
2     df = pd.read_excel('produk.xlsx')
3
4     df['Profit'] = (df['Harga Jual'] - df['Harga Beli']) * df['Stok']
5
6     totalProfit = df['Profit'].sum()
7     simpanDataProduk(df)
8
9     lihatProduk()
10
11    print_tengah("-"*101)
12    print_tengah(f"Total Profit: Rp {totalProfit:,}")
13    print_tengah("-"*101)

```

Gambar 3. 25 Fungsi hitung profit

```

1 def pembelian(username):
2     data = bacaUser()
3     akun = next((a for a in data['akun'] if a['username'] == username), None)
4     if akun is None:
5         print("Akun tidak ditemukan.")
6         return
7
8     df_produk = pd.read_excel('produk.xlsx')
9     lihatProduk()
10
11     try:
12         no_produk = int(inputTengah("Masukkan nomor produk yang ingin dibeli: "))
13         produk_index = df_produk[df_produk['No'] == no_produk].index[0]
14         produk = df_produk.loc[produk_index]
15     except (IndexError, ValueError):
16         print_tengah("Nomor produk tidak valid.")
17         return
18
19     if produk['Stok'] <= 0:
20         print_tengah("Stok produk habis.")
21         return
22
23     harga = produk['Harga Jual']
24     if akun['saldo'] < harga:
25         print_tengah("Saldo tidak mencukupi.")
26         return
27
28     akun['saldo'] -= harga
29
30     df_produk.at[produk_index, 'Stok'] -= 1
31
32     transaksi = {
33         "ID": len(akun['riwayatPembelian']) + 1,
34         "TANGGAL": datetime.now().strftime("%Y-%m-%d %H:%M:%S"),
35         "PRODUK": produk['Nama'],
36         "HARGA": f"{harga}"
37     }
38     akun['riwayatPembelian'].append(transaksi)
39
40     simpanUserData(data)
41     df_produk.to_excel('produk.xlsx', index=False)
42     clear()
43     loading()
44     print_tengah(f"Pembelian berhasil! Saldo tersisa: Rp {akun['saldo']}")
45     time.sleep(2)
46     print_tengah(f"Stok {produk['Nama']} tersisa: {df_produk.at[produk_index, 'Stok']}")
47     time.sleep(2)
48     print_tengah("Riwayat pembelian diperbarui.")

```

Gambar 3. 26 Fungsi pembelian

```

1 def tampilkanRiwayat(username):
2     data = bacaUser()
3     akun = next((a for a in data['akun'] if a['username'] == username), None)
4
5     if akun is None or not akun['riwayatPembelian']:
6         print_tengah("Riwayat pembelian tidak ditemukan.")
7         return
8
9     dfRiwayat = pd.DataFrame(akun['riwayatPembelian'])
10    tabel= tabulate(dfRiwayat, headers='keys', tablefmt='fancy_grid', showindex=False)
11    lebarTerminal = shutil.get_terminal_size().columns
12    sampling = (lebarTerminal - len(tabel.splitlines()[0])) // 2
13    for line in tabel.splitlines():
14        print(" " * sampling + line)
15

```

Gambar 3. 27 Fungsi tampilkan riwayat


```

1 def topupSaldo(username, jumlah, file_path='datausername.json'):
2     data = bacaUser()
3
4     for akun in data['akun']:
5         if akun['username'] == username:
6             if 'saldo' not in akun:
7                 akun['saldo'] = 0
8
9             akun['saldo'] += jumlah
10
11         simpanUserData(data, file_path)
12
13     print_tengah(f"Saldo berhasil ditambah sebesar Rp {jumlah}. Saldo saat ini: Rp {akun['saldo']}\n")
14     return

```

Gambar 3. 28 fungsi top up saldo






```

1 def lihatSaldo(username):
2     data = bacaUser()
3     for akun in data['akun']:
4         if akun['username'] == username and 'saldo' in akun:
5             print_tengah(f"Saldo {akun['username']} sebesar Rp {akun['saldo']}.\n")
6             return akun['saldo']
7     print_tengah("Username tidak ditemukan atau saldo belum diatur.\n")
8     return None

```

Gambar 3. 29 fungsi lihat saldo

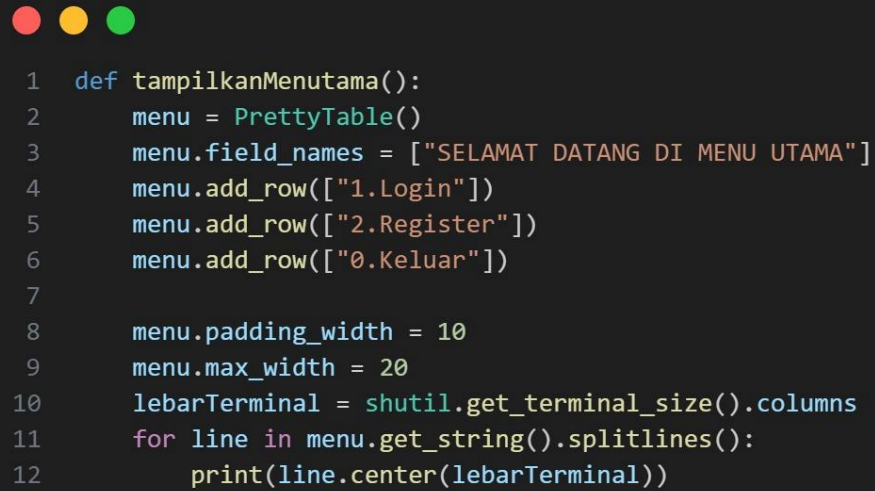


```

1 def tampilkanMenuBiasa(kata):
2     menu = PrettyTable()
3     menu.field_names = [f">> MENU {kata} <<"]
4     menu.padding_width = 19
5     lebarTerminal = shutil.get_terminal_size().columns
6
7     for line in menu.get_string().splitlines():
8         print(line.center(lebarTerminal))

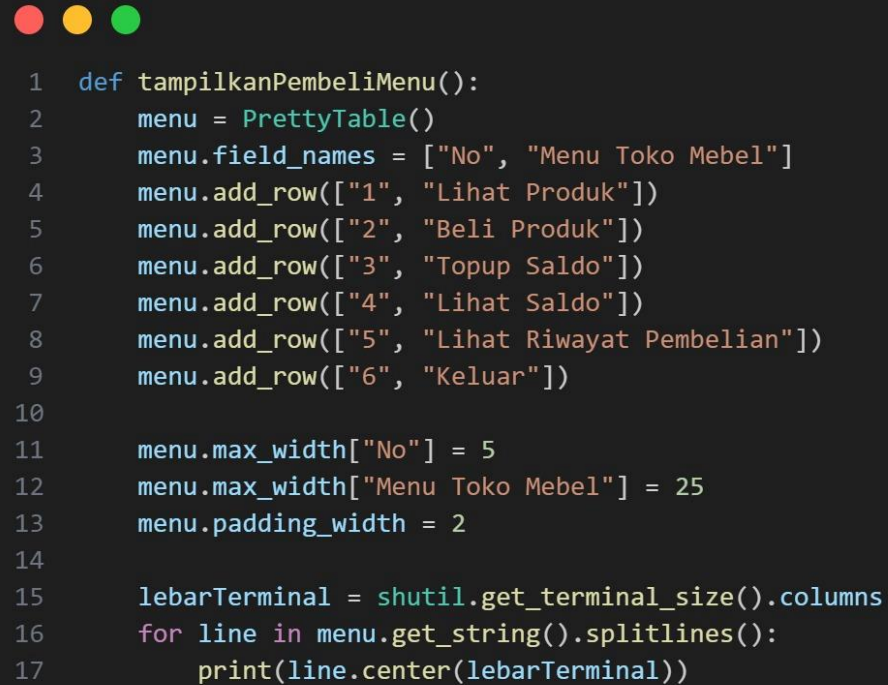
```

Gambar 3. 30 Fungsi tampilan menu biasa



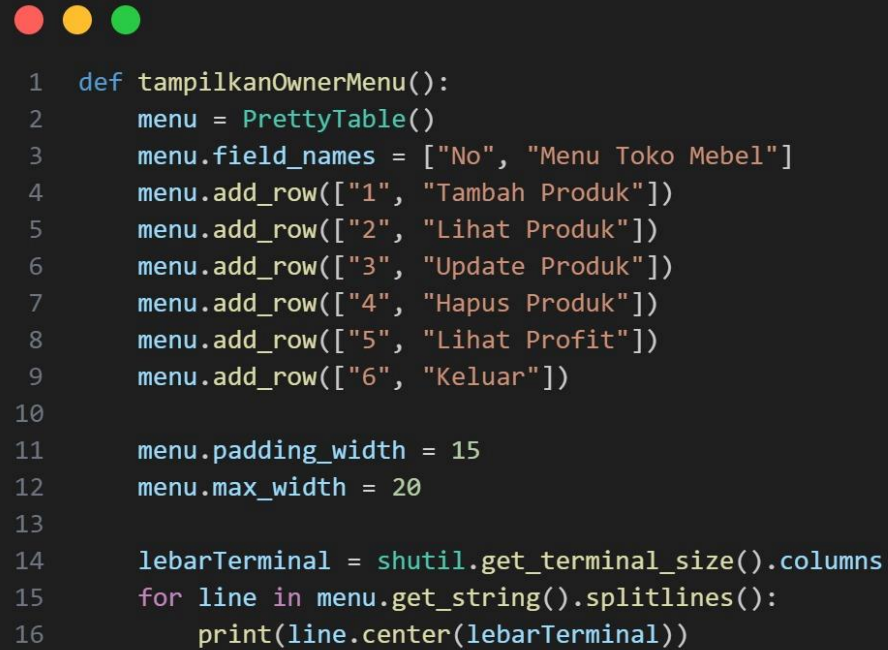
```
1 def tampilkanMenutama():
2     menu = PrettyTable()
3     menu.field_names = ["SELAMAT DATANG DI MENU UTAMA"]
4     menu.add_row(["1.Login"])
5     menu.add_row(["2.Register"])
6     menu.add_row(["0.Keluar"])
7
8     menu.padding_width = 10
9     menu.max_width = 20
10    lebarTerminal = shutil.get_terminal_size().columns
11    for line in menu.get_string().splitlines():
12        print(line.center(lebarTerminal))
```

Gambar 3. 31 fungsi tampilkan menu utama



```
1 def tampilkanPembeliMenu():
2     menu = PrettyTable()
3     menu.field_names = ["No", "Menu Toko Mebel"]
4     menu.add_row(["1", "Lihat Produk"])
5     menu.add_row(["2", "Beli Produk"])
6     menu.add_row(["3", "Topup Saldo"])
7     menu.add_row(["4", "Lihat Saldo"])
8     menu.add_row(["5", "Lihat Riwayat Pembelian"])
9     menu.add_row(["6", "Keluar"])
10
11     menu.max_width["No"] = 5
12     menu.max_width["Menu Toko Mebel"] = 25
13     menu.padding_width = 2
14
15     lebarTerminal = shutil.get_terminal_size().columns
16     for line in menu.get_string().splitlines():
17         print(line.center(lebarTerminal))
```

Gambar 3. 32 Fungsi tampilkanpembeli menu



```
1 def tampilkanOwnerMenu():
2     menu = PrettyTable()
3     menu.field_names = ["No", "Menu Toko Mebel"]
4     menu.add_row(["1", "Tambah Produk"])
5     menu.add_row(["2", "Lihat Produk"])
6     menu.add_row(["3", "Update Produk"])
7     menu.add_row(["4", "Hapus Produk"])
8     menu.add_row(["5", "Lihat Profit"])
9     menu.add_row(["6", "Keluar"])
10
11     menu.padding_width = 15
12     menu.max_width = 20
13
14     lebarTerminal = shutil.get_terminal_size().columns
15     for line in menu.get_string().splitlines():
16         print(line.center(lebarTerminal))
```

Gambar 3. 33 Fungsi tampilkan owner menu

[illegible]

Gambar 3. 34 Fungsi owner menu

[illegible]

Gambar 3. 35 Fungsi pembeli menu



```

1 def cekUsername(username):
2     if not username:
3         raise ValueError("Username tidak boleh kosong.")
4     if not re.match("^[A-Za-z0-9]*$", username):
5         raise ValueError("Username tidak boleh menggunakan karakter spesial.")
6     if not any(char.isdigit() for char in username):
7         raise ValueError("Username harus mengandung minimal 1 angka.")
8     if not any(char.isupper() for char in username):
9         raise ValueError("Username harus mengandung minimal 1 huruf kapital.")
10

```

Gambar 3. 38 Fungsi cekUsername

```

1 def cekPassword(password):
2     if not password:
3         raise ValueError("Password tidak boleh kosong.")
4     if len(password) <= 5:
5         raise ValueError("Password harus lebih dari 5 karakter.")
6     if not re.match("^[A-Za-z0-9]*$", password):
7         raise ValueError("Password tidak boleh menggunakan karakter spesial.")
8     if not any(char.isdigit() for char in password):
9         raise ValueError("Password harus mengandung minimal 1 angka.")
10    if not any(char.isupper() for char in password):
11        raise ValueError("Password harus mengandung minimal 1 huruf kapital.")
12

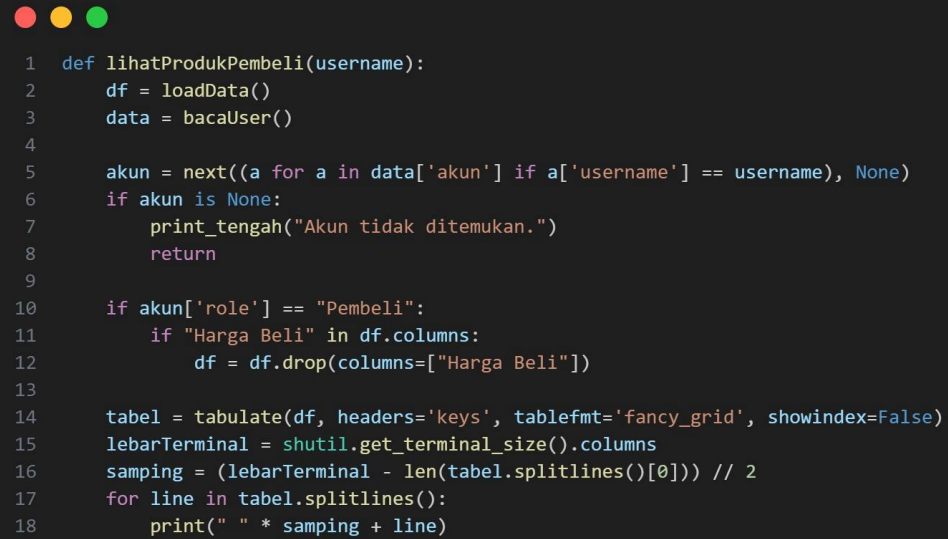
```

Gambar 3. 39 fungsi cekPassword









```
1 def lihatProdukPembeli(username):
2     df = loadData()
3     data = bacaUser()
4
5     akun = next((a for a in data['akun'] if a['username'] == username), None)
6     if akun is None:
7         print_tengah("Akun tidak ditemukan.")
8         return
9
10    if akun['role'] == "Pembeli":
11        if "Harga Beli" in df.columns:
12            df = df.drop(columns=["Harga Beli"])
13
14    tabel = tabulate(df, headers='keys', tablefmt='fancy_grid', showindex=False)
15    lebarTerminal = shutil.get_terminal_size().columns
16    sampling = (lebarTerminal - len(tabel.splitlines()[0])) // 2
17    for line in tabel.splitlines():
18        print(" " * sampling + line)
```

Gambar 3. 43 Fungsi lihatprodukpembeli

## **BAB IV**

### **PENUTUP**

#### **4.1 Kesimpulan**

Program dengan judul "Manajemen Toko Mebel" ini bertujuan untuk membantu mengelola toko mebel secara praktis dan efisien. Program ini menyediakan berbagai fitur seperti autentikasi pengguna (*login* dan *registrasi*), manajemen produk (tambah, baca, perbarui, hapus), serta penghitungan *profit*. Selain itu, terdapat fitur khusus bagi pelanggan, seperti pembelian produk, pengecekan saldo, *top-up*, dan riwayat transaksi. Implementasi ini memanfaatkan beberapa pustaka Python seperti *pandas* untuk pengolahan data, *tabulate* untuk tampilan data yang terformat, serta *json* untuk menyimpan data pengguna.

Program ini juga dirancang untuk memvalidasi *input* dari pengguna agar tidak terjadi kesalahan dengan menggunakan *error handling*. Sistem ini dirancang dengan pertimbangan keamanan data pengguna dan dilengkapi dengan validasi *input* untuk mencegah kesalahan. Penggunaan library *Pandas* dan *Tabulate* meningkatkan efisiensi dalam pengelolaan dan visualisasi data, sedangkan library *JSON* digunakan untuk menyimpan data pengguna. Secara keseluruhan, program ini memberikan solusi sederhana namun efektif untuk mengelola toko mebel.

#### **4.2 Saran**

Program dengan judul "Manajemen Toko Mebel" ini dirancang untuk membantu pengelolaan toko secara digital. Beberapa saran perbaikan untuk meningkatkan kualitas program meliputi keamanan sistem perlu ditingkatkan, misalnya dengan menambahkan enkripsi untuk data sensitif seperti saldo pengguna dan riwayat pembelian, serta pembatasan percobaan *login*. Kemudian, antarmuka pengguna dapat dibuat lebih menarik dan informatif dengan menambahkan elemen visual

seperti warna atau border pada tampilan terminal, serta menyediakan *menu* bantuan untuk pengguna baru. Selanjutnya, penambahan fitur seperti laporan keuangan berkala, pencarian produk, dan pengelolaan pengembalian barang akan semakin mempermudah operasional toko mebel secara digital. Dengan perbaikan ini, program diharapkan akan menjadi lebih andal, aman, dan memberikan pengalaman yang lebih baik bagi pengguna.

## **DAFTAR PUSTAKA**

## LAMPIRAN

### Aturan Konsultasi :

- A. Kartu Konsul wajib dibawa saat dilakukan konsultasi
- B. Ketua Kelompok dan Anggota diwajibkan untuk hadir tiap konsultasi
- C. Konsul dilaksanakan minimal 2 kali dengan ketentuan sebagai berikut :
  - Konsul 1 : Konsep Program
  - Konsul 2 : Penyelesaian Program

N.B : Batas waktu konsultasi ialah H+7 untuk konsul 1 dan H+14 untuk konsul 2 ( Dimulai sejak pengumuman dan pembentuk kelompok PA )

### Manajemen Toko Mebel

### Algoritma dan Pemrograman Dasar

KELAS IF B1 24 :

KELOMPOK 6:



Muhammad Riva Fachrodiya/2409106053 (Ketua)



Alya Mayasha/2409106054

Muhammad Dzaki Rifa'i/2409106056



LABORATORIUM FAKULTAS TEKNIK  
UNIVERSITAS MULAWARMAN 2024

Tanggal Konsultasi : 01 / 11 / 2024	
Uraian / Pembahasan : Pembahasan mengenai Flowchart dan Konsep program : - Menyusun ulang Flowchart dengan rapi. - Menambahkan pembayaran ke dalam flowchart. - Menggantikan ETD di akhir on Page Reference pada Flowchart. - Menambahkan pembayaran ke dalam program. - Membuat fitur error handling ke dalam program agar program tidak berhenti. - Password dan username tidak boleh terlalu gampang. - Menambahkan sistem toque untuk melakukan pembayaran.	
Asisten Lab	Ketua Kelompok
	
Nama: Adi Muhammad S.	Nama: Muhammad Riva F.

Tanggal Konsultasi : 13 / 11 / 2024	
Uraian / Pembahasan : Pembahasan mengenai Flowchart, Program, serta Pengejaan Laporan : - Mengetek dan memastikan program tetap berjalan meskipun terjadi kesalahan input. - Memperbaiki tampilan di terminal. - memperbaiki penulisan laporan.	
Asisten Lab	Ketua Kelompok
	
Nama: Adi Muhammad S.	Nama: Muhammad Riva F.