

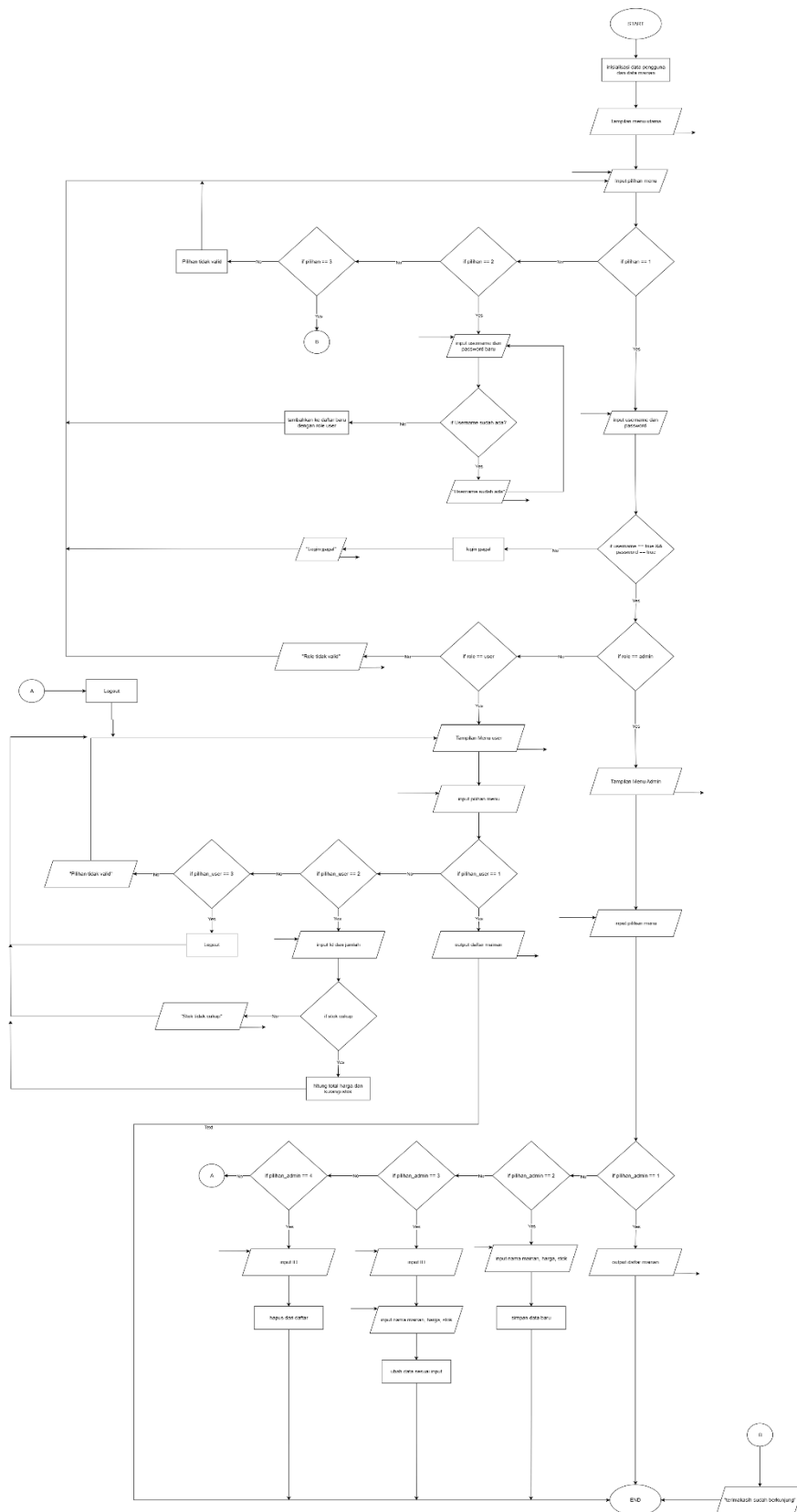
**LAPORAN PRAKTIKUM**  
**POSTTEST 7**  
**ALGORITMA PEMROGRAMAN DASAR**



**Disusun oleh:**  
**Az Zuhri Al Atsary (2509106072)**  
**Kelas (B2 '25)**

**PROGRAM STUDI INFORMATIKA**  
**UNIVERSITAS MULAWARMAN**  
**SAMARINDA**  
**2025**

# 1. Flowchart



Gambar 1.1 Flowchart CRUD toko mainan anak

Penjelasan Singkat Flowchart

1. START
2. Inisialisasi data pengguna dan data mainan
3. Tampilkan menu utama
  1. Login
  2. Register
  3. Keluar
4. Input pilihan menu (1–3)
5. Percabangan menu utama
  - Jika pilih 1 (Login):
    1. Input username dan password.
    2. Cek kecocokan data login.
      - Jika salah → tampilkan pesan “Login gagal” → kembali ke menu utama.
      - Jika benar → lanjut cek peran (role).
    3. Cek role pengguna:
      - Admin → masuk ke menu admin.
      - User → masuk ke menu user.
  - Menu Admin (role = admin):
    1. Lihat mainan → tampilkan daftar mainan.
    2. Tambah mainan → input nama, harga, stok → simpan data baru.
    3. Update mainan → pilih ID → ubah data sesuai input.
    4. Hapus mainan → pilih ID → hapus dari daftar.
    5. Logout → kembali ke menu utama.
  - Menu User (role = user):
    1. Lihat mainan → tampilkan daftar mainan.
    2. Beli mainan → input ID & jumlah beli.
      - Jika stok cukup → hitung total harga & kurangi stok.
      - Jika stok tidak cukup → tampilkan pesan kesalahan.
    3. Logout → kembali ke menu utama.
  - Jika pilih 2 (Register):
    1. Input username dan password baru.
    2. Cek apakah username sudah ada.
      - Jika sudah ada → tampilkan pesan “Username sudah ada”.
      - Jika belum ada → tambahkan user baru dengan role “user”.
    3. Kembali ke menu utama.
  - Jika pilih 3 (Keluar):
    - Tampilkan pesan “Terima kasih sudah berkunjung!”
    - Program berhenti.
  - Jika pilihan tidak valid:
    - Tampilkan pesan kesalahan dan kembali ke menu utama.

6. END

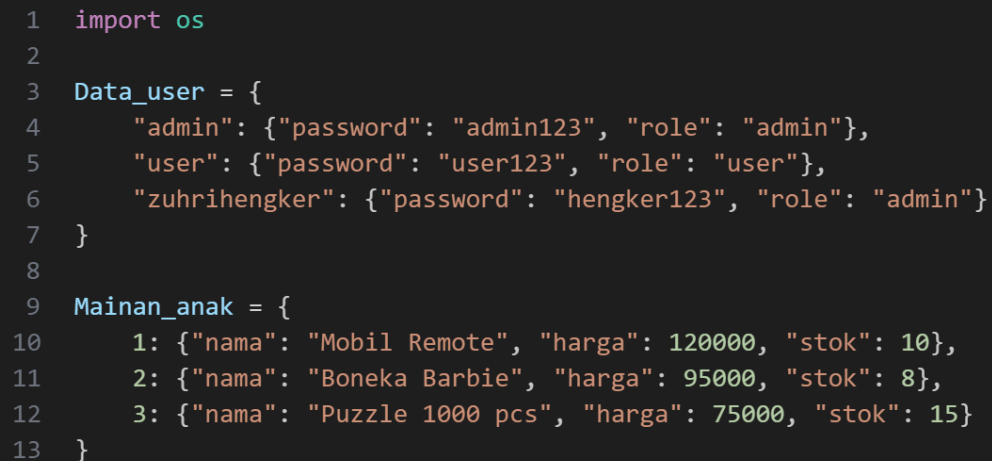
## 2. Deskripsi Singkat Program

Program ini berfungsi sebagai simulasi sistem toko mainan anak yang memungkinkan pengguna untuk login, register, dan mengelola data mainan. Terdapat dua jenis pengguna, yaitu admin dan user, dengan hak akses yang berbeda.

Pengguna admin dapat melihat daftar mainan, menambah, memperbarui, dan menghapus data mainan. Sementara itu, pengguna user dapat melihat daftar mainan yang tersedia dan melakukan pembelian sesuai stok yang ada. Program juga menampilkan total harga pembelian dan mengurangi stok otomatis setelah transaksi. Menu utama akan terus ditampilkan hingga pengguna memilih opsi keluar dari program.

## 3. Source Code

### A. Inisialisasi data pengguna dan data mainan

A screenshot of a code editor with a dark background and light-colored text. The code is written in Python and defines two dictionaries: 'Data\_user' and 'Mainan\_anak'. 'Data\_user' contains three entries for 'admin', 'user', and 'zuhrihengker' with their respective passwords and roles. 'Mainan\_anak' contains three entries for 'Mobil Remote', 'Boneka Barbie', and 'Puzzle 1000 pcs' with their respective prices and stock levels. The code is numbered from 1 to 13.

```
1 import os
2
3 Data_user = {
4     "admin": {"password": "admin123", "role": "admin"},
5     "user": {"password": "user123", "role": "user"},
6     "zuhrihengker": {"password": "hengker123", "role": "admin"}
7 }
8
9 Mainan_anak = {
10     1: {"nama": "Mobil Remote", "harga": 120000, "stok": 10},
11     2: {"nama": "Boneka Barbie", "harga": 95000, "stok": 8},
12     3: {"nama": "Puzzle 1000 pcs", "harga": 75000, "stok": 15}
13 }
```

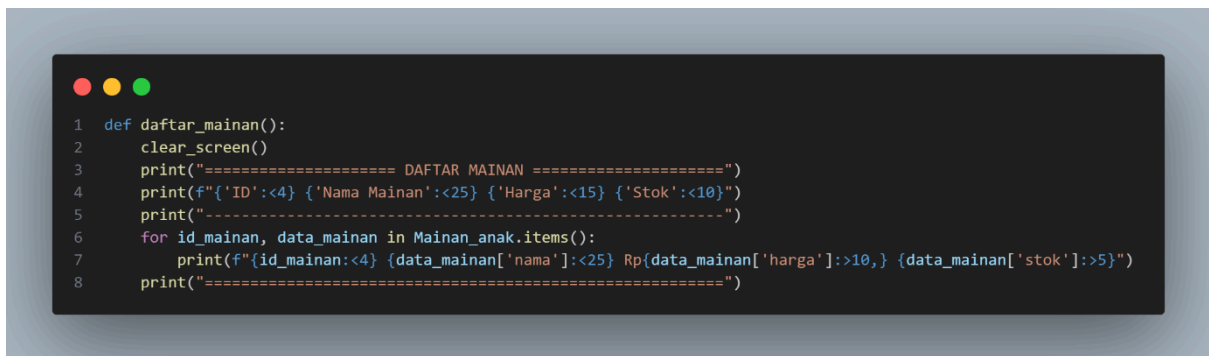
Gambar 3.1 Inisialisasi data pengguna dan data mainan

## B. Prosedur Clear Screen



Gambar 3.2 Prosedur Clear Screen

## C. Prosedur Daftar Mainan



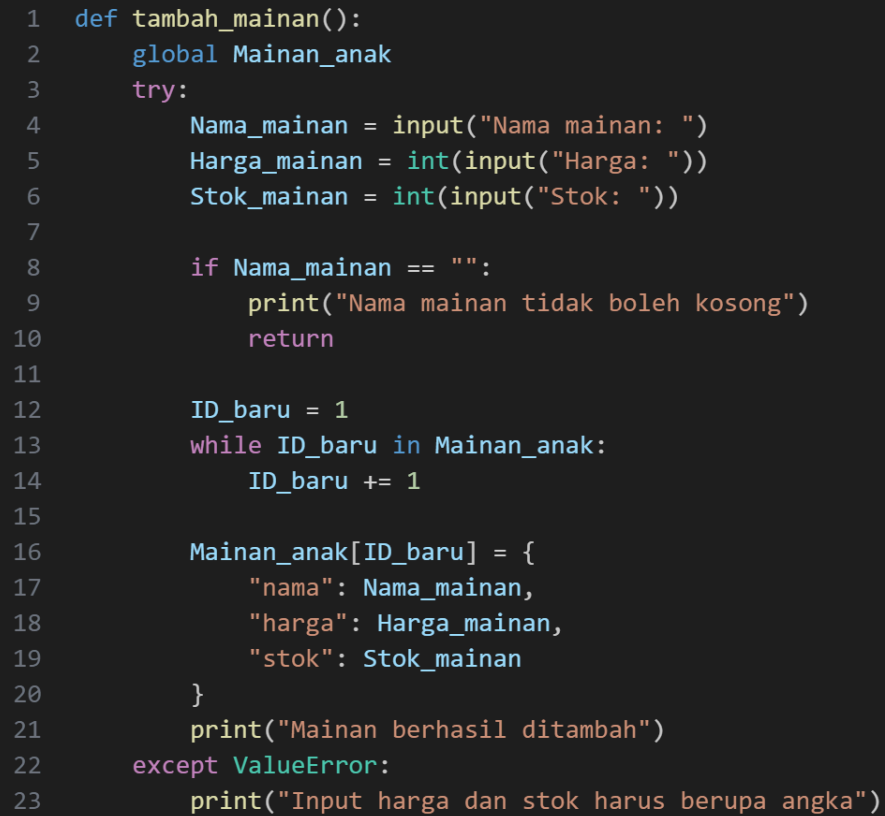
Gambar 3.3 Prosedur Daftar Mainan

## D. Fungsi Mengitung Total Bayar



Gambar 3.4 Fungsi Hitung Total

## E. Fungsi Tambah Mainan

A screenshot of a Python code editor window with a dark background and three colored window control buttons (red, yellow, green) in the top-left corner. The code is a function named 'tambah\_mainan' that interacts with a global list 'Mainan\_anak'. It prompts the user for a toy name, price, and stock, and then adds a new entry to the list. The code includes error handling for non-numeric input for price and stock.

```
1 def tambah_mainan():
2     global Mainan_anak
3     try:
4         Nama_mainan = input("Nama mainan: ")
5         Harga_mainan = int(input("Harga: "))
6         Stok_mainan = int(input("Stok: "))
7
8         if Nama_mainan == "":
9             print("Nama mainan tidak boleh kosong")
10            return
11
12            ID_baru = 1
13            while ID_baru in Mainan_anak:
14                ID_baru += 1
15
16            Mainan_anak[ID_baru] = {
17                "nama": Nama_mainan,
18                "harga": Harga_mainan,
19                "stok": Stok_mainan
20            }
21            print("Mainan berhasil ditambah")
22    except ValueError:
23        print("Input harga dan stok harus berupa angka")
```

Gambar 3.5 Tambah Mainan

## F. Prosedur Update Mainan

```
1 def update_mainan():
2     global Mainan_anak
3     try:
4         daftar_mainan()
5         ID_diperbarui = int(input("ID mainan yang diupdate: "))
6         if ID_diperbarui in Mainan_anak:
7             Data_mainan = Mainan_anak[ID_diperbarui]
8             print("Data lama:", Data_mainan)
9             Nama_baru = input("Nama baru (kosongkan jika tidak diubah): ")
10            Harga_baru = input("Harga baru (kosongkan jika tidak diubah): ")
11            Stok_baru = input("Stok baru (kosongkan jika tidak diubah): ")
12
13            if Nama_baru != "":
14                Data_mainan["nama"] = Nama_baru
15            if Harga_baru != "":
16                Data_mainan["harga"] = int(Harga_baru)
17            if Stok_baru != "":
18                Data_mainan["stok"] = int(Stok_baru)
19
20            print("Data mainan berhasil diperbarui")
21        else:
22            print("ID mainan tidak ditemukan")
23    except ValueError:
24        print("Input ID dan harga/stok harus angka")
```

Gambar 3.6 Prosedur Update Mainan

## G. Prosedur Hapus Mainan

```
1 def hapus_mainan():
2     global Mainan_anak
3     try:
4         daftar_mainan()
5         ID_hapus = int(input("ID mainan yang dihapus: "))
6         if ID_hapus in Mainan_anak:
7             del Mainan_anak[ID_hapus]
8             print("Mainan berhasil dihapus")
9         else:
10            print("ID tidak ditemukan")
11    except ValueError:
12        print("Input harus berupa angka")
```

Gambar 3.7 Prosedur Hapus Mainan

## H. Prosedur Beli Mainan

```
1 def beli_mainan():
2     global Mainan_anak
3     try:
4         daftar_mainan()
5         ID_beli = int(input("Masukkan ID mainan: "))
6         Jumlah_beli = int(input("Jumlah beli: "))
7         if ID_beli in Mainan_anak:
8             Data_mainan = Mainan_anak[ID_beli]
9             if Jumlah_beli <= Data_mainan["stok"]:
10                Total_harga = hitung_total(Data_mainan["harga"], Jumlah_beli)
11                Data_mainan["stok"] -= Jumlah_beli
12                print(f"Total bayar: Rp{Total_harga:,}")
13            else:
14                print("Stok tidak cukup")
15        else:
16            print("Mainan tidak ditemukan")
17    except ValueError:
18        print("Input harus berupa angka")
```

Gambar 3.8 Prosedur Beli Mainan

## I. Menu Utama

```
1 while True:
2     clear_screen()
3     print("===== TOKO MAINAN ANAK =====")
4     print("1. Login")
5     print("2. Register")
6     print("3. Keluar")
7     print("=====")
8     menu = input("Pilih menu (1-3): ")
```

Gambar 3.9 Menu Utama

## J. Fitur Login

```
1 if menu == "1":
2     clear_screen()
3     print("===== LOGIN =====")
4     username = input("Masukkan username: ")
5     password = input("Masukkan password: ")
6     print("=====")
7
8     login = False
9     role = ""
10
11     if username in Data_user and Data_user[username]["password"] == password:
12         login = True
13         role = Data_user[username]["role"]
14
15     if not login:
16         print("Login gagal")
17         input("Enter untuk lanjut...")
```

Gambar 3.10 Fitur Login



## K. Menu Admin

```
1  if role == "admin":
2      while True:
3          clear_screen()
4          print("===== MENU ADMIN =====")
5          print("1. Lihat mainan")
6          print("2. Tambah mainan")
7          print("3. Update mainan")
8          print("4. Hapus mainan")
9          print("5. Logout")
10         print("=====")
11         Pilihan_admin = input("Pilih menu (1-5): ")
12
13         if Pilihan_admin == "1":
14             daftar_mainan()
15         elif Pilihan_admin == "2":
16             tambah_mainan()
17         elif Pilihan_admin == "3":
18             update_mainan()
19         elif Pilihan_admin == "4":
20             hapus_mainan()
21         elif Pilihan_admin == "5":
22             break
23         else:
24             print("Pilihan salah")
25             input("Enter untuk kembali...")
26
27     else:
28         while True:
29             clear_screen()
30             print("===== MENU USER =====")
31             print("1. Lihat mainan")
32             print("2. Beli mainan")
33             print("3. Logout")
34             print("=====")
35
36             Pilihan_user = input("Pilih (1-3): ")
37
38             if Pilihan_user == "1":
39                 daftar_mainan()
40             elif Pilihan_user == "2":
41                 beli_mainan()
42             elif Pilihan_user == "3":
43                 break
44             else:
45                 print("Pilihan tidak valid")
46                 input("Enter untuk kembali...")
47
48 elif menu == "2":
49     clear_screen()
50     print("===== REGISTER =====")
51     Username_baru = input("Masukkan Username Baru: ")
52     Password_baru = input("Masukkan Password: ")
53     print("=====")
54
55     if Username_baru not in Data_user:
56         Data_user[Username_baru] = {"password": Password_baru, "role": "user"}
57         print("Registrasi berhasil")
58     else:
59         print("Username sudah ada!")
60         input("Enter untuk lanjut...")
61
62 elif menu == "3":
63     print("Terima kasih sudah berkunjung!")
64     break
65
66 else:
67     print("Pilihan tidak valid")
68     input("Enter untuk lanjut...")
69
```

Gambar 3.11 Menu Admin

## K. Menu User

```
1  else:
2      while True:
3          clear_screen()
4          print("===== MENU USER =====")
5          print("1. Lihat mainan")
6          print("2. Beli mainan")
7          print("3. Logout")
8          print("=====")
9
10         Pilihan_user = input("Pilih (1-3): ")
11
12         if Pilihan_user == "1":
13             daftar_mainan()
14         elif Pilihan_user == "2":
15             beli_mainan()
16         elif Pilihan_user == "3":
17             break
18         else:
19             print("Pilihan tidak valid")
20             input("Enter untuk kembali...")
```

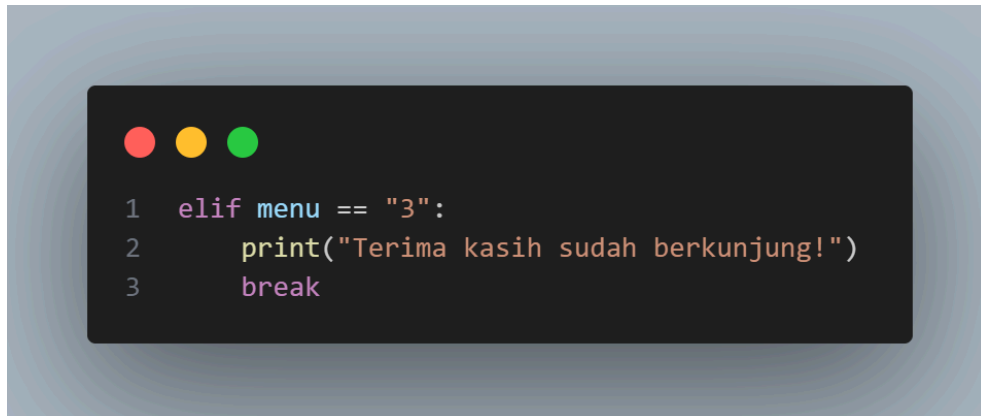
Gambar 3.12 Menu User

## L. Fitur Register

```
1  elif menu == "2":
2      clear_screen()
3      print("===== REGISTER =====")
4      Username_baru = input("Masukkan Username Baru: ")
5      Password_baru = input("Masukkan Password: ")
6      print("=====")
7
8      if Username_baru not in Data_user:
9          Data_user[Username_baru] = {"password": Password_baru, "role": "user"}
10         print("Registrasi berhasil")
11     else:
12         print("Username sudah ada!")
13         input("Enter untuk lanjut...")
```

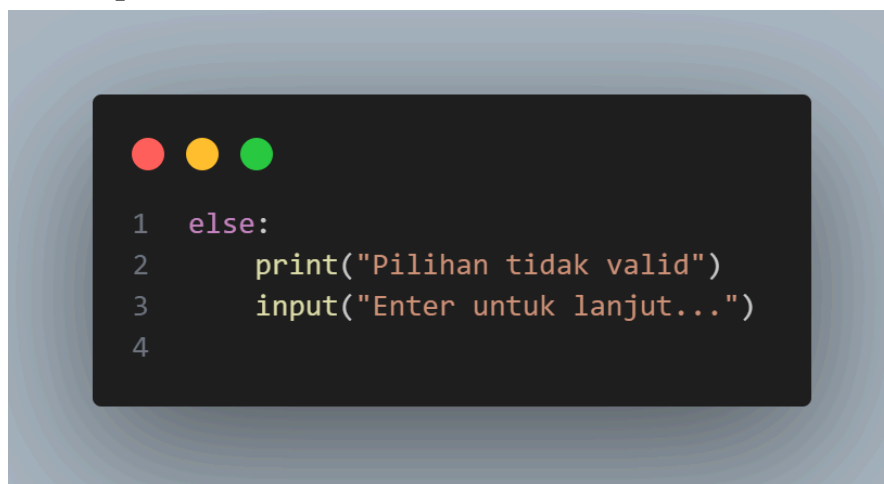
Gambar 3.13 Fitur Register

#### L. Keluar dari Program



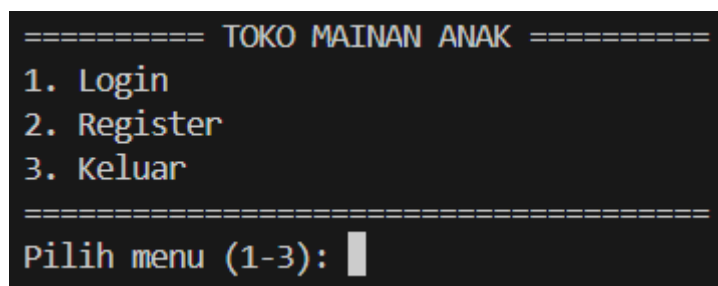
Gambar 3.14 Keluar Program

#### L. Penanganan Input Tidak Valid



Gambar 3.15 Input Tidak Valid

#### 4. Hasil Output



Gambar 4.1 Tampilan awal saat program di jalankan

```

===== TOKO MAINAN ANAK =====
1. Login
2. Register
3. Keluar
=====
Pilih menu (1-3): 11
Pilihan tidak valid
Enter untuk lanjut...

```

*Gambar 4.2 Output pilihan menu tidak valid*

```

===== LOGIN =====
Masukkan username: admin
Masukkan password: admin123

```

*Gambar 4.3 Tampilan menu login*

```

===== LOGIN =====
Masukkan username: aaaaaaa
Masukkan password: aaaaaaa
=====
Login gagal
Enter untuk lanjut...

```

*Gambar 4.4 Output 3 gagal login*

```

===== MENU ADMIN =====
1. Lihat mainan
2. Tambah mainan
3. Update mainan
4. Hapus mainan
5. Logout
=====
Pilih menu (1-5): 

```

*Gambar 4.5 Tampilan menu admin*

```

===== MENU ADMIN =====
1. Lihat mainan
2. Tambah mainan
3. Update mainan
4. Hapus mainan
5. Logout
=====
Pilih menu (1-5): 11
Pilihan salah!
Enter untuk lanjut...

```

Gambar 4.6 Output pilihan menu salah atau tidak sesuai

```

===== DAFTAR MAINAN =====
ID   Nama Mainan           Harga           Stok
-----
1    Mobil Remote          Rp 120,000      10
2    Boneka Barbie          Rp 95,000       8
3    Puzzle 1000 pcs        Rp 75,000      15
=====
Enter untuk kembali...

```

Gambar 4.7 Tampilan daftar mainan sebelum penambahan, penghapusan, atau perubahan data mainan pada menu pilihan 1

```

===== MENU ADMIN =====
1. Lihat mainan
2. Tambah mainan
3. Update mainan
4. Hapus mainan
5. Logout
=====
Pilih menu (1-5): 2
Nama mainan: Lego
Harga: 85000
Stok: 6
Mainan berhasil ditambah!
Enter untuk kembali...

```

Gambar 4.8 Penambahan data mainan pada menu pilihan 2

```
===== DAFTAR MAINAN =====
```

ID	Nama Mainan	Harga	Stok
-----			
1	Mobil Remote	Rp 120,000	10
2	Boneka Barbie	Rp 95,000	8
3	Puzzle 1000 pcs	Rp 75,000	15
4	Lego	Rp 85,000	6

```
=====
Enter untuk kembali...█
```

Gambar 4.9 Tampilan setelah mainan ditambahkan

```
===== DAFTAR MAINAN =====
```

ID	Nama Mainan	Harga	Stok
-----			
1	Mobil Remote	Rp 120,000	10
2	Boneka Barbie	Rp 95,000	8
3	Puzzle 1000 pcs	Rp 75,000	15
4	Lego	Rp 85,000	6

```
=====
ID mainan yang diupdate: 1
Data lama: {'nama': 'Mobil Remote', 'harga': 120000, 'stok': 10}
Nama baru (kosongkan jika tidak diubah): Squiishy
Harga baru (kosongkan jika tidak diubah): 35000
Stok baru (kosongkan jika tidak diubah): 15
Data mainan berhasil diperbarui
Enter untuk kembali...█
```

Gambar 4.10 Update data mainan pada menu pilihan 3

```
===== DAFTAR MAINAN =====
```

ID	Nama Mainan	Harga	Stok
-----			
1	Squishy	Rp 35,000	15
2	Boneka Barbie	Rp 95,000	8
3	Puzzle 1000 pcs	Rp 75,000	15
4	Lego	Rp 85,000	6

```
=====
Enter untuk kembali...█
```

Gambar 4.11 Tampilan data mainan setelah diperbarui

```

===== MENU ADMIN =====
1. Lihat mainan
2. Tambah mainan
3. Update mainan
4. Hapus mainan
5. Logout
=====
Pilih menu (1-5): 4
ID mainan yang dihapus: 2
Mainan dihapus!
Enter untuk kembali...

```

Gambar 4.12 Hapus data mainan pada menu pilihan 4

```

===== DAFTAR MAINAN =====
ID   Nama Mainan           Harga      Stok
-----
1    Squishy                Rp   35,000   15
3    Puzzle 1000 pcs        Rp   75,000   15
4    Lego                   Rp   85,000    6
=====
Enter untuk kembali...

```

Gambar 4.13 Tampilan data mainan setelah dihapus

```

===== REGISTER =====
Masukkan Username Baru: zuhrisigma
Masukkan Password: 123

```

Gambar 4.14 Tampilan menu register

```

===== REGISTER =====
Masukkan Username Baru: zuhrisigma
Masukkan Password: 123
=====
Registrasi berhasil
Enter untuk lanjut...

```

Gambar 4.15 Output ketika registrasi berhasil

```

===== REGISTER =====
Masukkan Username Baru: admin
Masukkan Password: admin123
=====
Username sudah ada
Enter untuk lanjut...

```

Gambar 4.16 Output ketika registrasi username sudah ada

```

===== MENU USER =====
1. Lihat mainan
2. Beli mainan
3. Logout
=====
Pilih (1-3): █

```

*Gambar 4.17 Tampilan menu user*

```

===== DAFTAR MAINAN =====
ID   Nama Mainan           Harga      Stok
-----
1    Mobil Remote           Rp 120,000 10
2    Boneka Barbie           Rp 95,000  8
3    Puzzle 1000 pcs         Rp 75,000 15
=====
Enter untuk kembali...█

```

*Gambar 4.18 Tampilan daftar mainan pada menu user*

```

===== DAFTAR MAINAN =====
ID   Nama Mainan           Harga      Stok
-----
1    Mobil Remote           Rp 120,000 10
2    Boneka Barbie           Rp 95,000  8
3    Puzzle 1000 pcs         Rp 75,000 15
=====
Masukkan ID mainan: 2
Jumlah beli: 6
Total bayar: Rp570,000
Enter untuk kembali...█

```

*Gambar 4.19 Output ketika membeli mainan pada menu user*

```

===== MENU USER =====
1. Lihat mainan
2. Beli mainan
3. Logout
=====
Pilih (1-3): 2
Masukkan ID mainan: 2
Jumlah beli: 12
Stok tidak cukup
Enter untuk kembali...█

```

*Gambar 4.19 Output ketika membeli mainan stok tidak mencukupi pada menu user*



## 5. Langkah-langkah GIT

### 5.1 GIT Add

```
PS C:\Users\azzuh\OneDrive\Documents\Praktikum-APD-B2-25> git add .  
PS C:\Users\azzuh\OneDrive\Documents\Praktikum-APD-B2-25> █
```

Gambar 5.1

*Menandai semua perubahan file di folder agar siap disimpan (commit) ke Git.*

### 5.2 GIT Commit

```
PS C:\Users\azzuh\OneDrive\Documents\Praktikum-APD-B2-25> git commit -m "PostTest7"  
[main 6ea70a1] PostTest7  
2 files changed, 213 insertions(+)  
create mode 100644 Kelas/pertemuan-7.py  
create mode 100644 Post-Test/Post-Test-APD-7/2509106072-AzZuhriAlatsary-PT-7.py  
PS C:\Users\azzuh\OneDrive\Documents\Praktikum-APD-B2-25> █
```

Gambar 5.2

*Menyimpan perubahan yang sudah ditandai ke dalam riwayat Git, dengan pesan sebagai catatan perubahan.*

### 5.3 GIT Push

```
PS C:\Users\azzuh\OneDrive\Documents\Praktikum-APD-B2-25> git push -u origin main  
Enumerating objects: 10, done.  
Counting objects: 100% (10/10), done.  
Delta compression using up to 20 threads  
Compressing objects: 100% (7/7), done.  
Writing objects: 100% (7/7), 2.10 KiB | 716.00 KiB/s, done.  
Total 7 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)  
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.  
To https://github.com/azzuhrialatsary-dev/Praktikum-APD-B2-25.git  
173d107..6ea70a1 main -> main  
branch 'main' set up to track 'origin/main'.  
PS C:\Users\azzuh\OneDrive\Documents\Praktikum-APD-B2-25> █
```

Gambar 5.4

*Mengirim commit (perubahan yang sudah disimpan) dari repositori lokal ke repositori remote seperti GitHub.*