

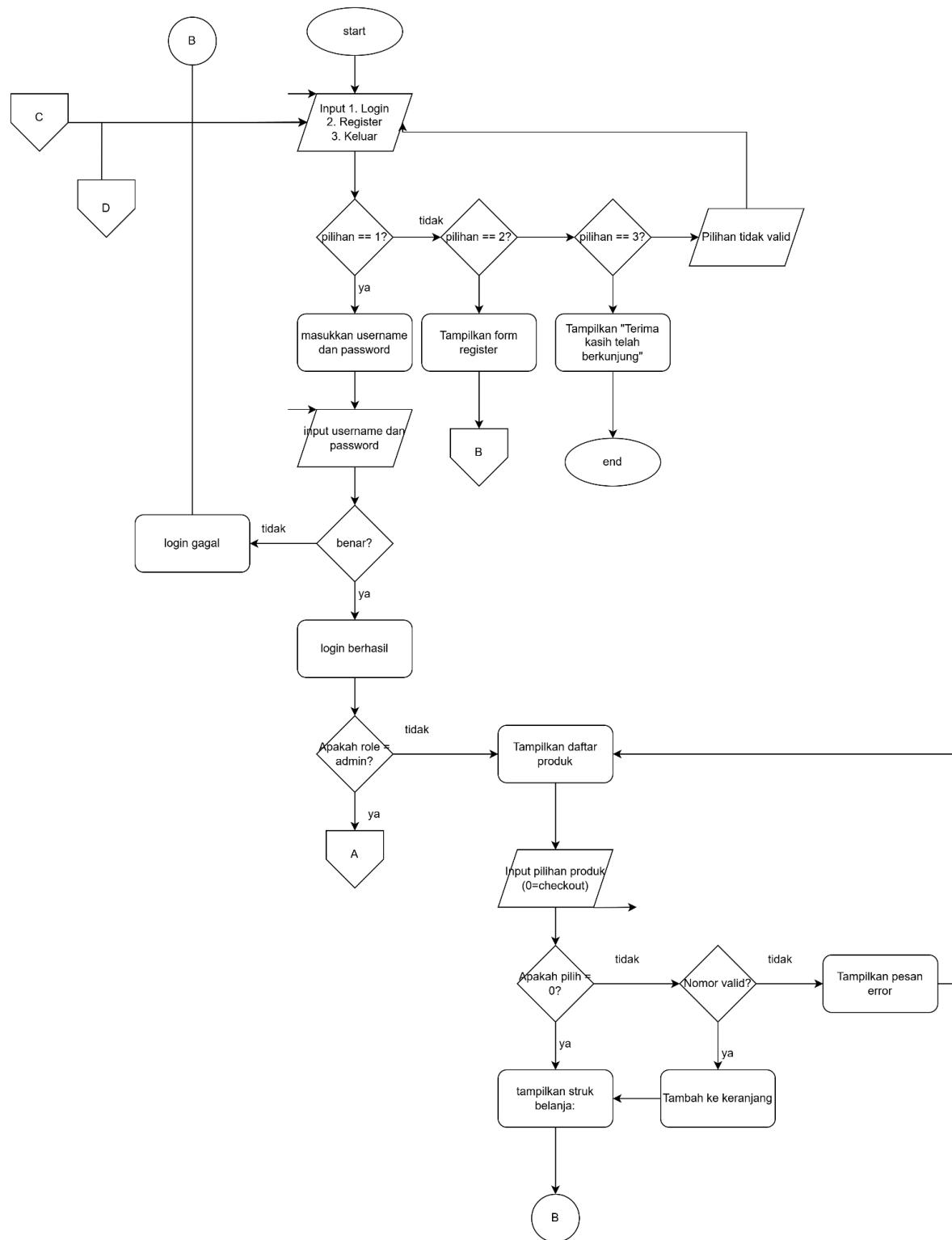
**LAPORAN PRAKTIKUM
POSTTEST (8)
ALGORITMA PEMROGRAMAN DASAR**



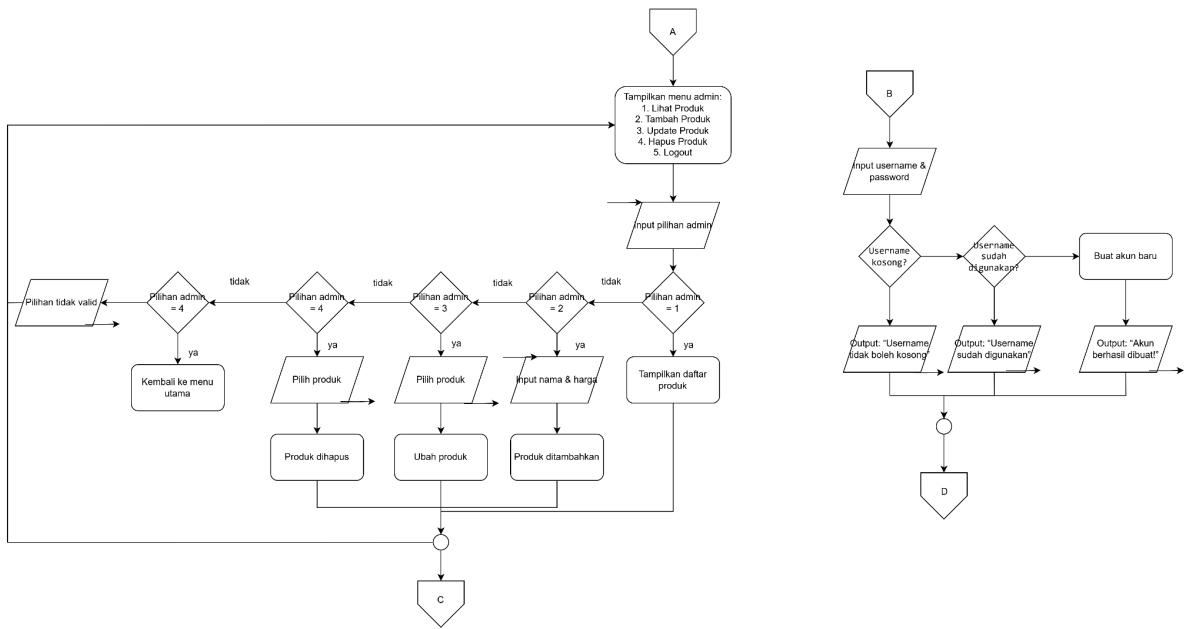
**Disusun oleh:
Bintang Dhana Permadi (2509106014)
Kelas (A1 '25)**

**PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025**

1. Flowchart



Gambar 1.1 Flowchart Program toko baju brand



Gambar 1.2 Flowchart Program toko baju brand

- **Penjelasan singkat flowchart:**

Program “Toko Baju Brand” adalah program Python yang mensimulasikan sistem penjualan pakaian sederhana. Di dalamnya terdapat fitur login, register, serta menu admin dan menu user. Admin dapat menambah, mengubah, dan menghapus produk, sedangkan user dapat melihat produk dan melakukan pembelian.

2. Deskripsi Singkat Program

Program Toko Baju Brand adalah aplikasi toko pakaian sederhana berbasis terminal yang memiliki dua jenis pengguna: admin dan user. Admin dapat mengelola produk (melihat, menambah, mengubah, dan menghapus), sedangkan user dapat melihat daftar produk, menambahkannya ke keranjang, dan melakukan checkout. Program juga mendukung fitur login dan register, serta melakukan validasi input tanpa menggunakan `def()` atau `isdigit()`.

3. Source Code

```
1  from utils import clear_screen, tampilan_produk
2  from data import produk
3
4  def tambah_produk():
5      clear_screen()
6      print("== TAMBAH PRODUK ==")
7      nama = input("Masukkan nama produk: ").strip()
8      harga = input("Masukkan harga produk: ").strip()
9
10     if not nama or not harga.isdigit():
11         print("Input tidak valid!")
12     else:
13         id_baru = max(produk.keys()) + 1 if produk else 1
14         produk[id_baru] = {"nama": nama, "harga": int(harga)}
15         print(f"Produk '{nama}' berhasil ditambahkan!")
16
17 def ubah_produk():
18     clear_screen()
19     print("== UBAH PRODUK ==")
20     tampilan_produk(produk)
21     try:
22         index = int(input("\nMasukkan ID produk yang ingin diubah: "))
23         if index in produk:
24             nama_baru = input("Nama baru (kosongkan jika tidak diubah): ").strip()
25             harga_baru = input("Harga baru (kosongkan jika tidak diubah): ").strip()
26             if nama_baru:
27                 produk[index]["nama"] = nama_baru
28             if harga_baru.isdigit():
29                 produk[index]["harga"] = int(harga_baru)
30                 print("Produk berhasil diperbarui!")
31             else:
32                 print("ID produk tidak ditemukan!")
33         except:
34             print("Input tidak valid!")
35
36 def hapus_produk():
37     clear_screen()
38     print("== HAPUS PRODUK ==")
39     tampilan_produk(produk)
40     try:
41         index = int(input("\nMasukkan ID produk yang ingin dihapus: "))
42         if index in produk:
43             hapus = produk.pop(index)
44             print(f"Produk '{hapus['nama']}' berhasil dihapus.")
45         else:
46             print("Produk tidak ditemukan!")
47     except:
48         print("Input tidak valid!")
49
```

Gambar 2.1 File Sourcs code Admin

```
● ● ●

1 users = {
2     "admin": {"password": "admin123", "role": "admin"},
3     "udin": {"password": "1234", "role": "user"}
4 }
5
6 produk = {
7     1: {"nama": "Kaos Lacoste", "harga": 250000},
8     2: {"nama": "Celana Rucas", "harga": 400000},
9     3: {"nama": "Jaket H&M", "harga": 550000},
10    4: {"nama": "Topi Nike", "harga": 150000},
11    5: {"nama": "Sepatu Adidas", "harga": 750000}
12 }
13
14 keranjang_global = []
15
```

Gambar 2.2 File Sourc code Data

```
1 from utils import clear_screen, tampilan_header
2 from data import users
3 from admin import tambah_produk, ubah_produk, hapus_produk
4 from user import menu_user
5 from utils import tampilan_produk
6 from data import produk
7
8 while True:
9     clear_screen()
10    tampilan_header()
11    print("1. Login")
12    print("2. Register")
13    print("3. Keluar")
14    pilih = input("Pilih menu: ")
15
16 if pilih == "1":
17     clear_screen()
18     print("== LOGIN ==")
19     username = input("Username: ").strip()
20     password = input("Password: ").strip()
21
22 if username in users and users[username]["password"] == password:
23     role = users[username][“role”]
24     print(f"Login berhasil sebagai {role.upper()}!")
25     input("\nTekan Enter untuk melanjutkan...")
26
27 if role == "admin":
28     while True:
29         clear_screen()
30         print("== MENU ADMIN ==")
31         print("1. Lihat Produk")
32         print("2. Tambah Produk")
33         print("3. Ubah Produk")
34         print("4. Hapus Produk")
35         print("5. Logout")
36         menu_admin = input("Pilih menu: ")
37
38 if menu_admin == "1":
39     clear_screen()
40     tampilan_produk(produk)
41     input("\nTekan Enter untuk kembali...")
42 elif menu_admin == "2":
43     tambah_produk()
44     input("\nTekan Enter untuk kembali...")
45 elif menu_admin == "3":
46     ubah_produk()
47     input("\nTekan Enter untuk kembali...")
48 elif menu_admin == "4":
49     hapus_produk()
50     input("\nTekan Enter untuk kembali...")
51 elif menu_admin == "5":
52     break
53 else:
54     print("Pilihan tidak valid!")
55     input("\nTekan Enter untuk kembali...")
56
57 menu_user(username)
58 else:
59     print("Login gagal! Username atau password salah.")
60     input("\nTekan Enter untuk kembali...")
61
62 elif pilih == "2":
63     clear_screen()
64     print("== REGISTER AKUN BARU ==")
65     username = input("Buat username: ").strip()
66     password = input("Buat password: ").strip()
67
68 if username in users:
69     print("Username sudah digunakan!")
70 elif not username or not password:
71     print("Username dan password tidak boleh kosong!")
72 else:
73     users[username] = {"password": password, "role": "user"}
74     print("Akun berhasil dibuat! Silakan login.")
75     input("\nTekan Enter untuk kembali...")
76
77 elif pilih == "3":
78     clear_screen()
79     print("Terima kasih telah mengunjungi Toko Baju Brand!")
80     break
81
82 else:
83     print("Pilihan tidak valid!")
84     input("\nTekan Enter untuk kembali...")
85
```

Gambar 2.3 File Sourc code [main.py](#)

```
● ● ●
1  from utils import clear_screen, tampilan_produk, hitung_total
2  from data import produk, keranjang_global
3
4  def menu_user(username):
5      while True:
6          clear_screen()
7          print(f"== MENU PENGGUNA: {username.upper()} ==")
8          tampilan_produk(produk)
9          print("0. Checkout")
10
11         pilih = input("\nPilih ID produk: ")
12
13         if pilih == "0":
14             checkout()
15             break
16         elif pilih.isdigit() and int(pilih) in produk:
17             item = produk[int(pilih)]
18             keranjang_global.append(item)
19             print(f"{item['nama']} berhasil ditambahkan ke keranjang!")
20         else:
21             print("Input tidak valid!")
22             input("\nTekan Enter untuk lanjut...")
23
24     def checkout():
25         clear_screen()
26         print("== STRUK BELANJA ==")
27         if not keranjang_global:
28             print("Keranjang masih kosong.")
29         else:
30             tampilan_produk({i+1: item for i, item in enumerate(keranjang_global)})
31             total = hitung_total(keranjang_global)
32             print(f"Total Belanja: Rp{total:,}")
33             input("\nTekan Enter untuk keluar...")
34
```

Gambar 2.4 File Sours code [user.py](#)

```
● ● ●

1 import os
2 from prettytable import PrettyTable
3
4 def clear_screen():
5     os.system('cls' if os.name == 'nt' else 'clear')
6
7 def tampilan_header():
8     print("== SELAMAT DATANG DI TOKO BAJU BRAND ==")
9
10 def tampilan_produk(produk):
11     if not produk:
12         print("Belum ada produk.")
13         return
14     table = PrettyTable(["ID", "Nama Produk", "Harga (Rp)"])
15     for id_produk, data in produk.items():
16         table.add_row([id_produk, data['nama'], f"{data['harga']}"])
17     print(table)
18
19 def hitung_total(keranjang):
20     return sum(item["harga"] for item in keranjang)
21
```

Gambar 2.5 File Sours code utils.py

4. Hasil Output

- Menu awal saat datang ke toko

```
==== SELAMAT DATANG DI TOKO BAJU BRAND ====
1. Login
2. Register
3. Keluar
Pilih menu:
```

Gambar 3.1 Output pilihan/opsi awal

- Tampilan login berhasil

```
==== LOGIN TOKO BAJU BRAND ====
Username: udin
Password: 1234

Login berhasil sebagai USER!
Tekan Enter untuk melanjutkan...■
```

Gambar 3.2 Output login berhasil

- Daftar produk

```
==== MENU PENGGUNA: UDIN ====
+-----+
| ID | Nama Produk | Harga (Rp) |
+-----+
| 1 | Kaos Lacoste | 250,000 |
| 2 | Celana Rucas | 400,000 |
| 3 | Jaket H&M | 550,000 |
| 4 | Topi Nike | 150,000 |
| 5 | Sepatu Adidas | 750,000 |
+-----+
0. Checkout

Pilih ID produk: ■
```

Gambar 3.3 Output pilihan daftar produk

- Struk hasil transaksi

```
== STRUK BELANJA ==
+-----+
| ID | Nama Produk | Harga (Rp) |
+-----+
| 1 | Kaos Lacoste | 250,000 |
+-----+
Total Belanja: Rp250,000
```

Gambar 3.4 Output Struk hasil transaksi

- Pembeli membuat akun baru

```
== REGISTER AKUN BARU ==
Buat username: asep
Buat password: 27
Akun berhasil dibuat! Silakan login.

Tekan Enter untuk kembali...■
```

Gambar 3.5 Output membuat akun baru

- Login akun baru

```
== LOGIN TOKO BAJU BRAND ==
Username: asep
Password: 27

Login berhasil sebagai USER!
Tekan Enter untuk melanjutkan...■
```

Gambar 3.6 Output login akun baru

- Struk belanja akun baru

```
== STRUK BELANJA ==
- Topi Nike : Rp150,000
- Sepatu Adidas : Rp750,000
-----
Total Belanja : Rp900,000
Terima kasih sudah berbelanja!

Tekan Enter untuk keluar...■
```

Gambar 3.7 Output struk belanja akun baru

- Tampilan login gagal

```
==== LOGIN TOKO BAJU BRAND ====
Username: asep
Password: 12

Login gagal! Username atau password salah.
Tekan Enter untuk kembali...█
```

Gambar 3.8 Output struk belanja akun baru

- Login admin

```
==== LOGIN ===
Username: admin
Password: admin123
Login berhasil sebagai ADMIN!
Tekan Enter untuk melanjutkan...█
```

Gambar 3.9 Output login akun admin

- Menu Login akun admin

```
==== MENU ADMIN TOKO BAJU BRAND ====
1. Lihat Produk
2. Tambah Produk
3. Update Produk
4. Hapus Produk
5. Logout
Pilih menu: █
```

Gambar 3.10 Output menu login akun admin

5. Langkah-langkah GIT

5.1 GIT Add

```
PS D:\kuliah\SEMESTER 1\APD\praktikum_apd> git add .
```

Gambar 4.1 Git add

Setelah membuat atau mengubah file, Kita perlu memberitahu Git file mana yang akan dimasukkan ke dalam riwayat versi berikutnya. Perintah git add ini biasanya digunakan untuk menambahkan perubahan dari satu atau lebih file ke area staging (area persiapan). Area ini berfungsi sebagai "keranjang" tempat kita mengumpulkan semua perubahan yang ingin disimpan dalam satu commit.

5.2 GIT Commit

```
PS D:\kuliah\SEMESTER 1\APD\praktikum_apd> git commit -m "upload"
[main aede578] upload
 11 files changed, 246 insertions(+)
 create mode 100644 kelas/pertemuan8.py
 create mode 100644 post-test/post-test-apd-8/2509106014-Bintang Dhana Permadi-PT-8/__pycache__/_admin.cpython-313.pyc
 create mode 100644 post-test/post-test-apd-8/2509106014-Bintang Dhana Permadi-PT-8/__pycache__/_data.cpython-313.pyc
 create mode 100644 post-test/post-test-apd-8/2509106014-Bintang Dhana Permadi-PT-8/__pycache__/_user.cpython-313.pyc
 create mode 100644 post-test/post-test-apd-8/2509106014-Bintang Dhana Permadi-PT-8/__pycache__/_utils.cpython-313.pyc
 create mode 100644 post-test/post-test-apd-8/2509106014-Bintang Dhana Permadi-PT-8/_admin.py
 create mode 100644 post-test/post-test-apd-8/2509106014-Bintang Dhana Permadi-PT-8/_data.py
 create mode 100644 post-test/post-test-apd-8/2509106014-Bintang Dhana Permadi-PT-8/_main.py
 create mode 100644 post-test/post-test-apd-8/2509106014-Bintang Dhana Permadi-PT-8/_user.py
 create mode 100644 post-test/post-test-apd-8/2509106014-Bintang Dhana Permadi-PT-8/_utils.py
 create mode 100644 post-test/post-test-apd-8/__pycache__/_utils.cpython-313.pyc
```

Gambar 4.2 Git Commit

Perintah ini menambahkan semua file yang ada di dalam folder ke staging area. Staging area adalah tempat sementara sebelum file dikomit ke dalam repository.

5.3 GIT Push

```
PS D:\kuliah\SEMESTER 1\APD\praktikum_apd> git push origin main
Enumerating objects: 22, done.
Counting objects: 100% (22/22), done.
Delta compression using up to 8 threads
Compressing objects: 100% (18/18), done.
Writing objects: 100% (19/19), 8.00 KiB | 909.00 KiB/s, done.
Total 19 (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/Bintang-2727/praktikum-apd.git
 882d5c8..aede578  main -> main
PS D:\kuliah\SEMESTER 1\APD\praktikum_apd>
```

Gambar 4.3 Git Push

Setelah commit Kita siap, perintah git push digunakan untuk mengunggah commit dari repositori lokal Anda ke repositori jarak jauh (misalnya, GitHub). Ini membuat perubahan yang Anda simpan di komputer pribadi kita bisa dilihat dan diakses oleh orang lain yang berkolaborasi dalam proyek yang sama. Perintah ini biasanya diikuti dengan nama remote dan nama branch, seperti: git push origin main.

