

LAPORAN PRAKTIKUM
POSTTEST 6
ALGORITMA PEMROGRAMAN DASAR



Disusun oleh:
Syarifah Anargya Rizky (2509106007)
Kelas A1'25

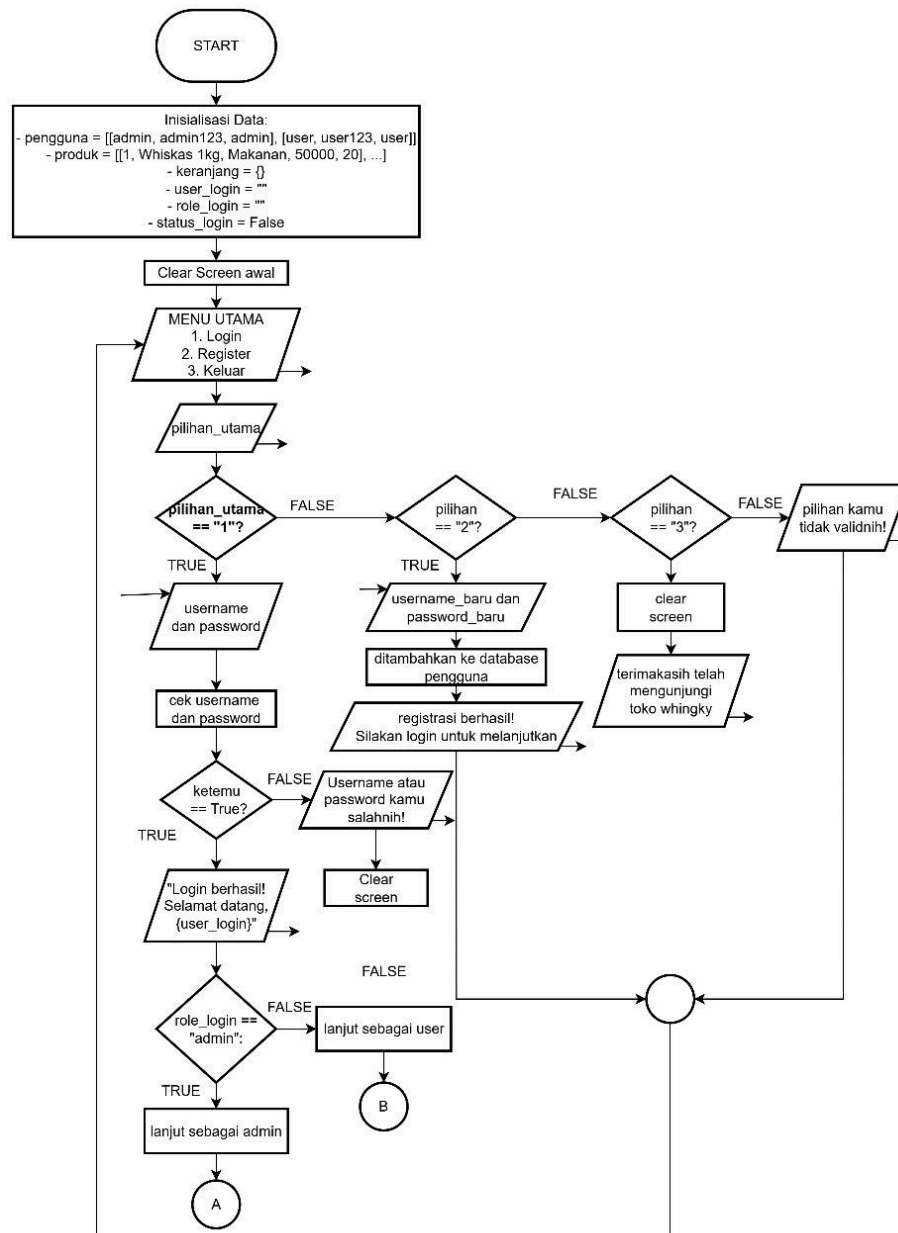
PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

DAFTAR ISI

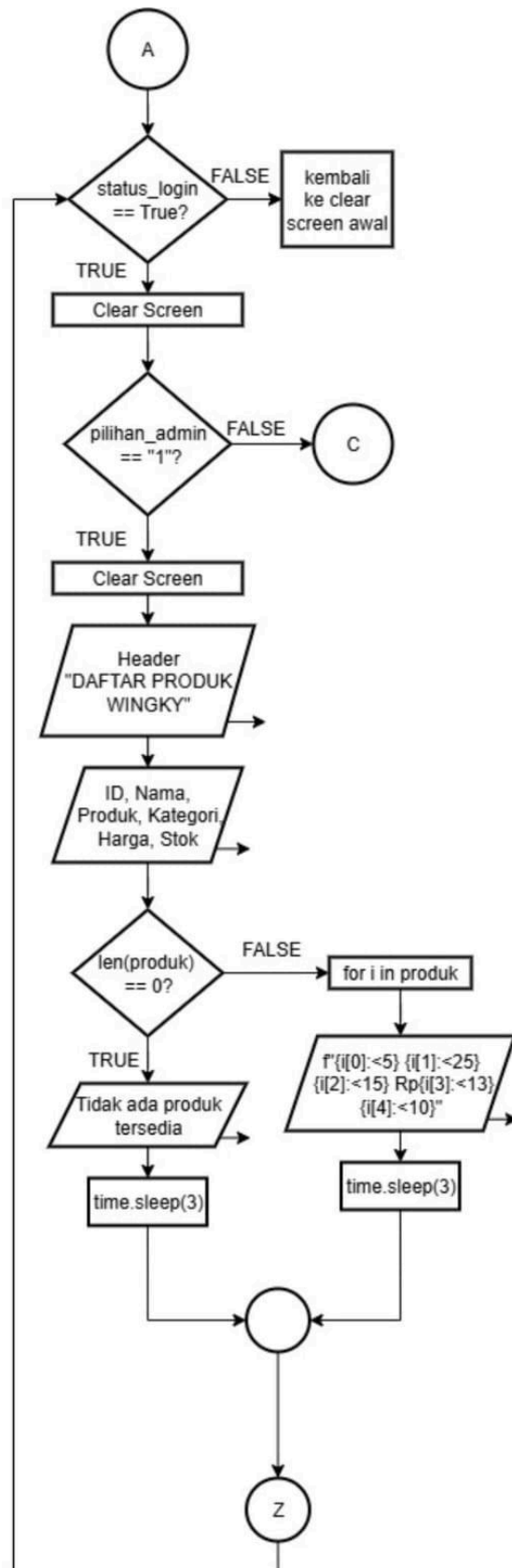
DAFTAR ISI.....	1
1. Flowchart.....	2
Gambar 1.1 Flowchart bagian awal program toko makanan kucing whinky.....	2
Gambar 1.2 Flowchart bagian on page A untuk program toko makanan kucing whinky...	3
Gambar 1.3 Flowchart bagian on page C untuk program toko makanan kucing whinky...	4
Gambar 1.4 Flowchart bagian on page D untuk program toko makanan kucing whinky...	5
Gambar 1.5 Flowchart bagian on page E untuk program toko makanan kucing whinky...	6
Gambar 1.6 Flowchart bagian on page F untuk program toko makanan kucing whinky....	7
Gambar 1.7 Flowchart bagian on page B untuk program toko makanan kucing whinky...	8
Gambar 1.8 Flowchart bagian on page G untuk program toko makanan kucing whinky...	9
Gambar 1.9 Flowchart bagian on page H untuk program toko makanan kucing whinky.	10
Gambar 1.10 Flowchart bagian on page I untuk program toko makanan kucing whinky	11
Gambar 1.11 Flowchart bagian on page J untuk program toko makanan kucing whinky	12
2. Deskripsi Singkat Program.....	13
3. Source Code.....	14
4. Hasil Output.....	26
Gambar 4.1.1 Ketika login sebagai admin di pilihan 1 (login).....	26
Gambar 4.1.2 tampilan layar bila jadi admin.....	27
Gambar 4.1.3 output pilihan 1 (melihat semua produk).....	27
Gambar 4.1.4 output pilihan 2 (menambahkan produk baru pada etalase toko).....	27
Gambar 4.1.5 output pilihan 3 (jadi bisa memperbarui nama/ kategori/ harga/ stok).....	28
Gambar 4.1.6 output pilihan 4 (admin dapat menghapus apapun yang ada di etalase)..	28
Gambar 4.1.7 output pilihan 5 (admin bisa mencari produk lebih cepat dengan fitur ini)	29
Gambar 4.2.1 output pilihan 2 sedari awal dengan tujuan daftar akun baru.....	29
Gambar 4.2.1.a output pilihan 2 sedari awal dengan tujuan daftar akun baru.....	29
Gambar 4.2.2 Ketika login sebagai user (kalau registrasi itu akunnya nanti jadi user) di pilihan 1 (login).....	30
Gambar 4.2.3 output pilihan 1 (melihat semua produk).....	30
Gambar 4.2.4 output pilihan 2 (fitur mencari produk).....	30
Gambar 4.2.5 output pilihan 3 (jadi user bisa nambahin barang kekeranjang sepuasnya *asalkan stoknya cukup yaa).....	31
Gambar 4.2.6 output pilihan 4 (fitur melihat isi keranjang sekaligus udah dihitung kira kira subtotal belanjanya berapa).....	31
Gambar 4.2.7 output pilihan 5 (fitur menghapus produk pilihan yang ada di keranjang).	32
Gambar 4.2.8 output pilihan 6 (fitur checkout).....	32
Gambar 4.2.9 jika pengguna memilih pilihan 3 sejak awal.....	32
5. Langkah-langkah GIT.....	32
5.1 GIT Add.....	32
Gambar 5.1.1 Penggunaan Git Add.....	33
5.2 GIT Commit.....	33
Gambar 5.2.1 Penggunaan Git Commit.....	33
5.3 GIT Push.....	33

1. Flowchart

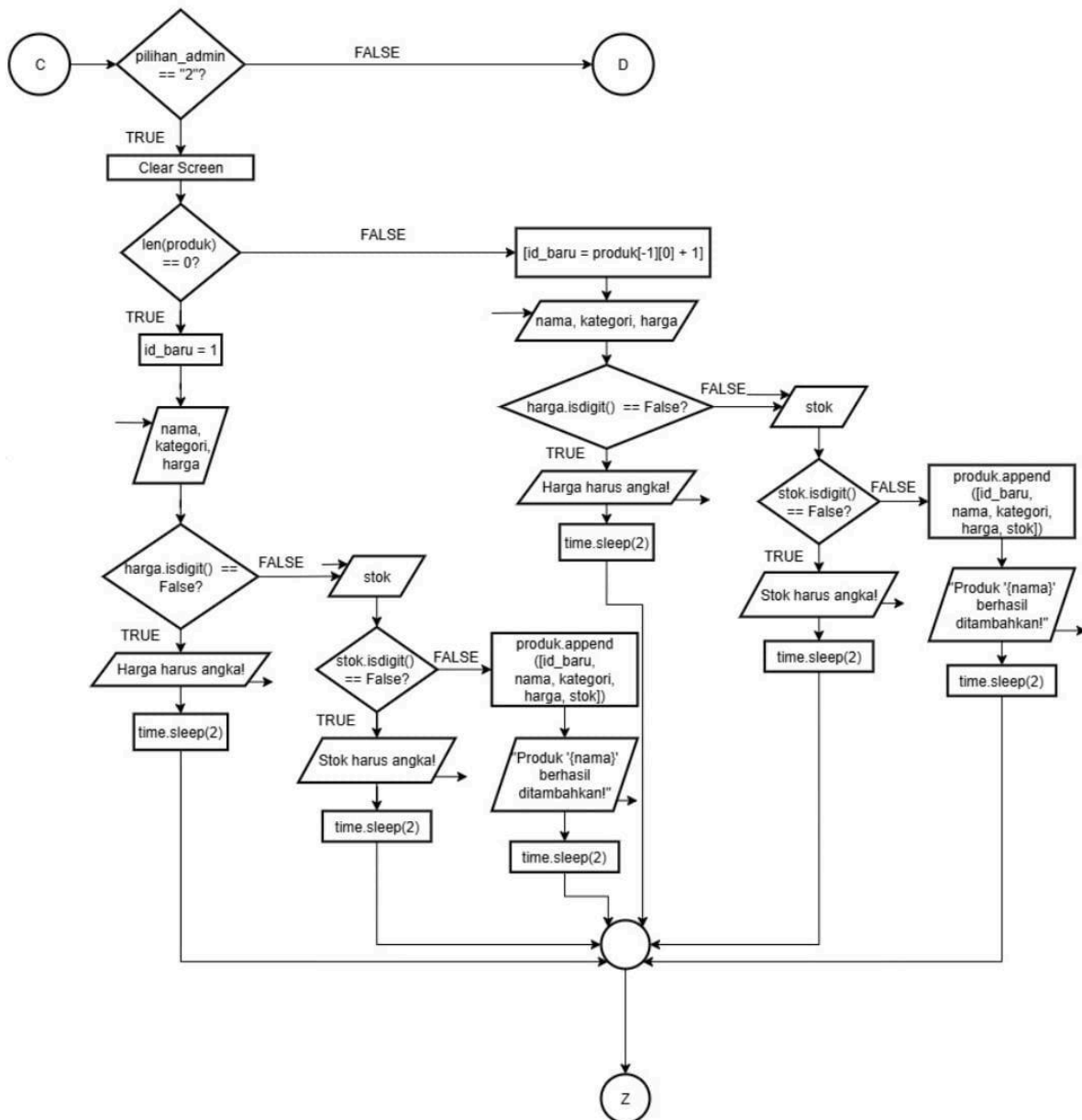
Flowchart merupakan diagram yang biasa digunakan dalam dunia informatika dengan tujuan agar dapat menggambarkan dengan lebih jelas dari suatu langkah-langkah, keputusan, dan alur kerja dari suatu proses dalam sistem. Flowchart terdiri dari berbagai simbol-simbol dengan makna tersendiri yang digunakan untuk menyederhanakan rangkaian prosedur agar memudahkan pemahaman terhadap informasi yang ingin disampaikan.



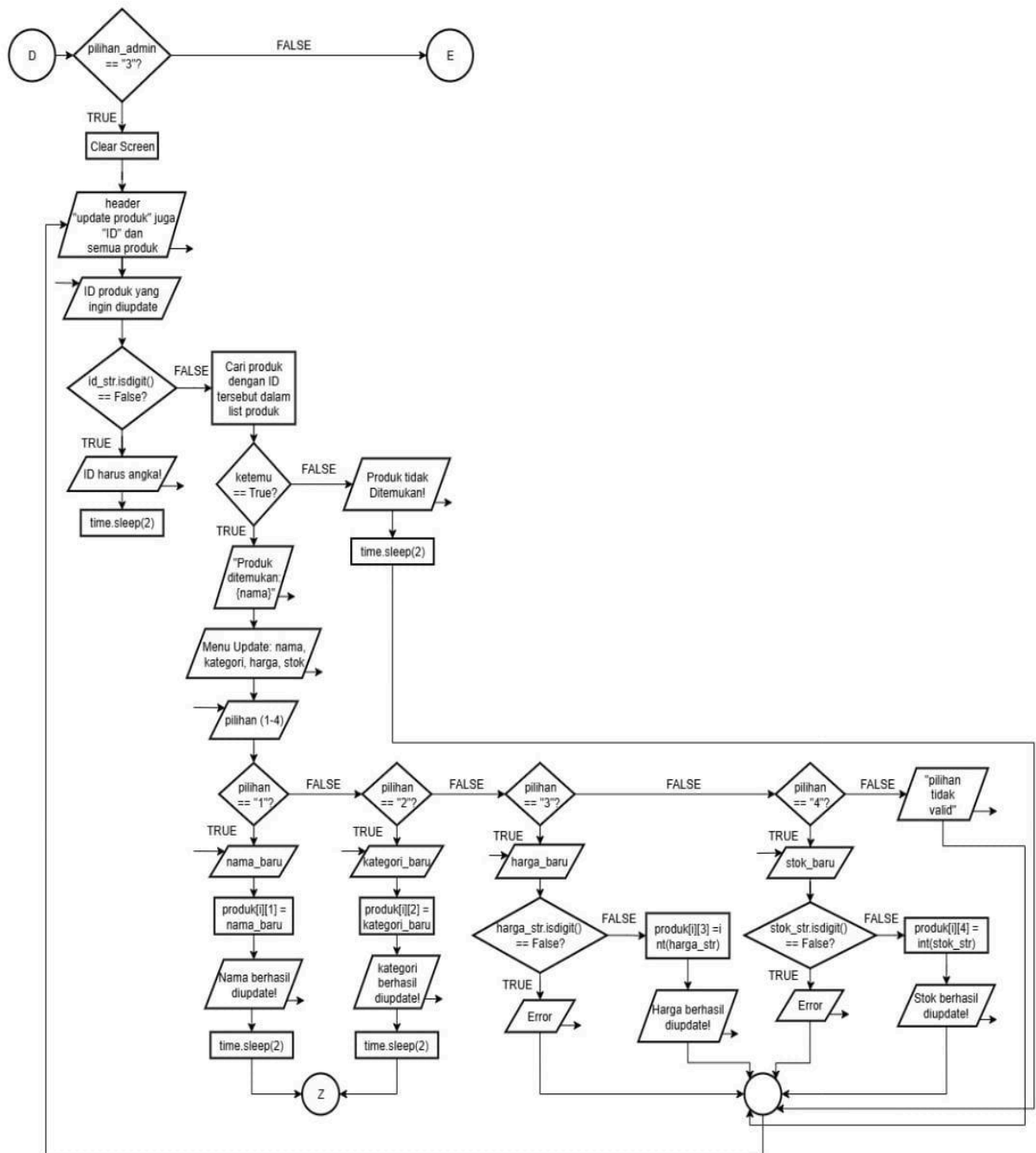
Gambar 1.1 Flowchart bagian awal program toko makanan kucing whinky



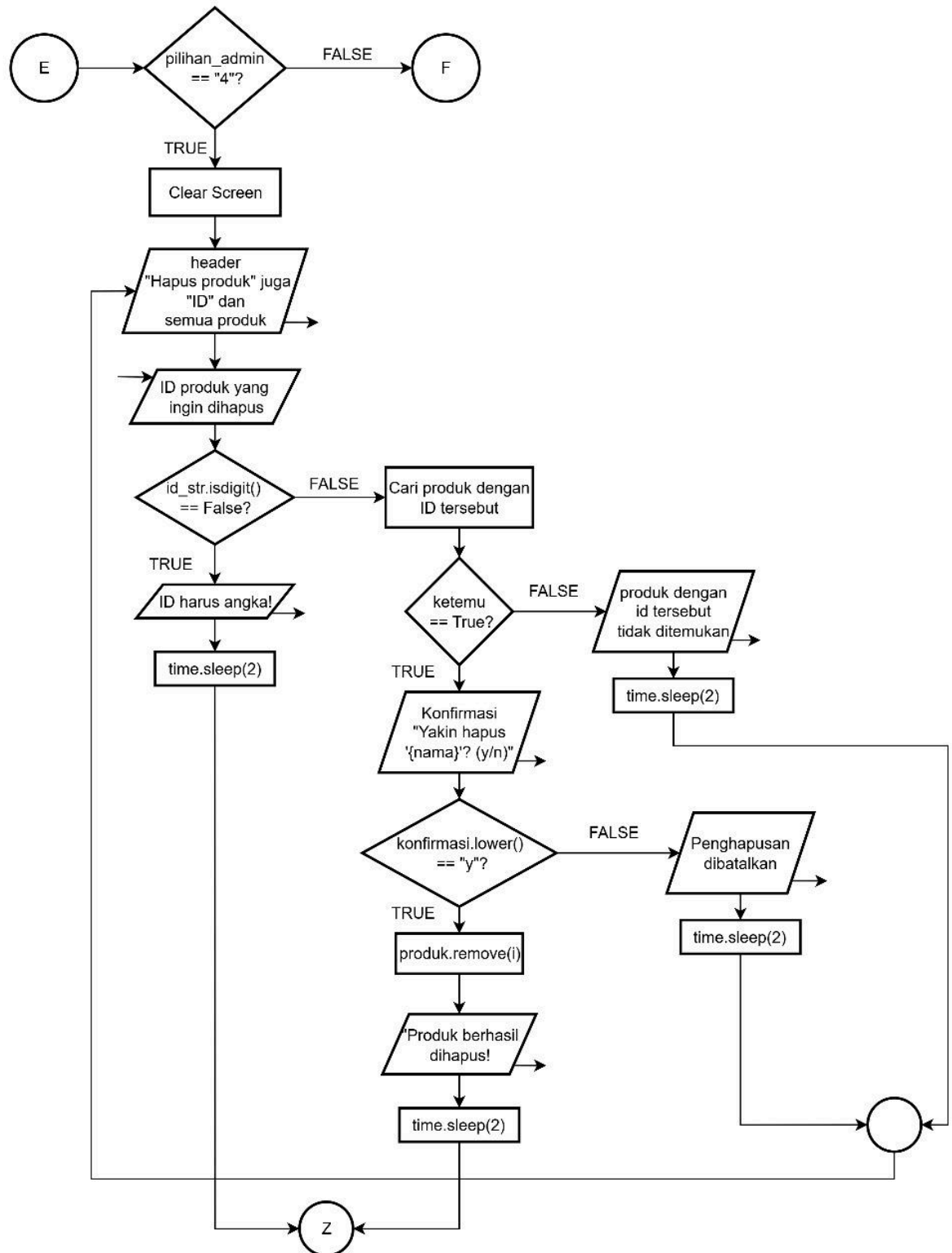
Gambar 1.2 Flowchart bagian on page A untuk program toko makanan kucing whinky



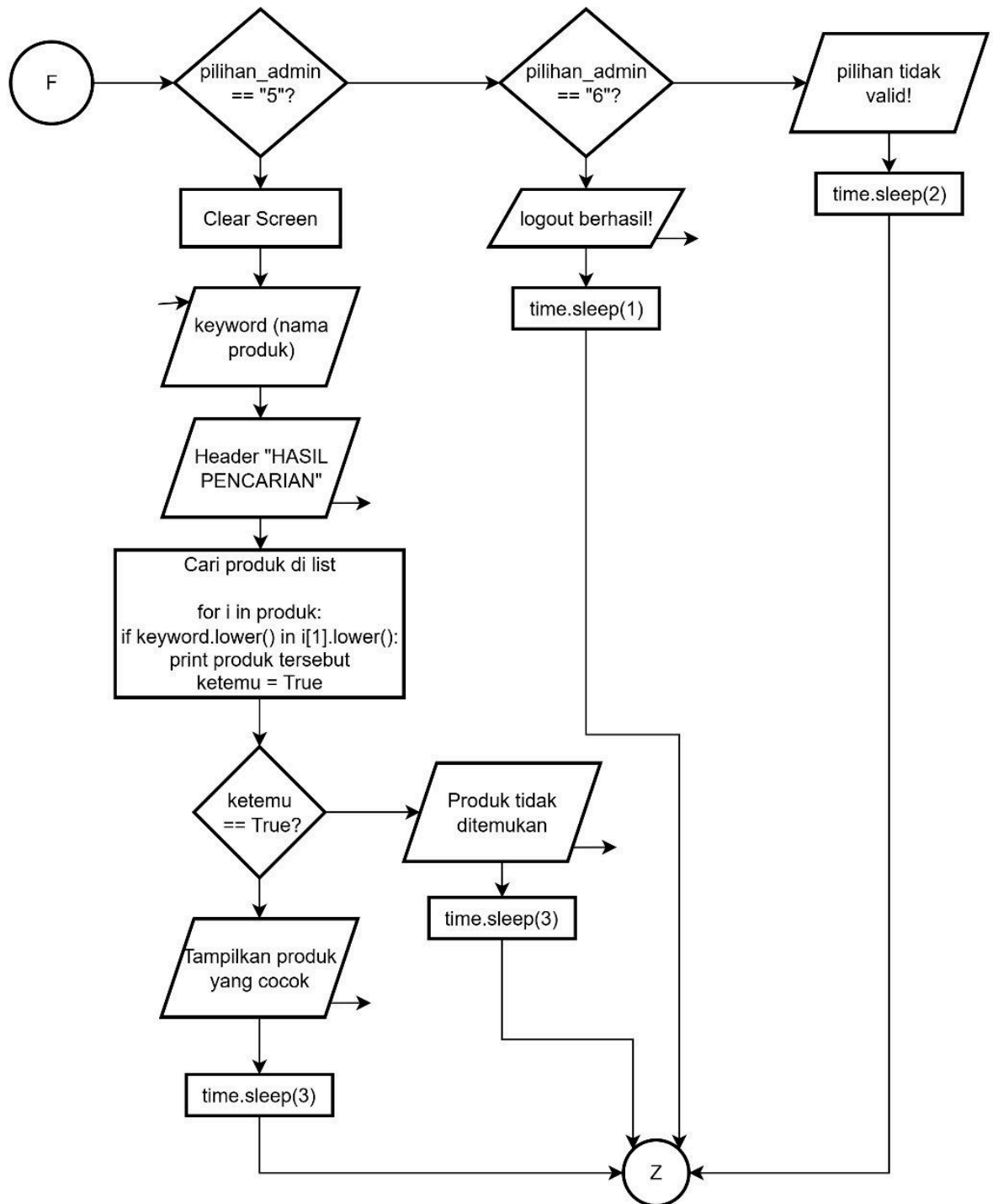
Gambar 1.3 Flowchart bagian on page C untuk program toko makanan kucing whinky



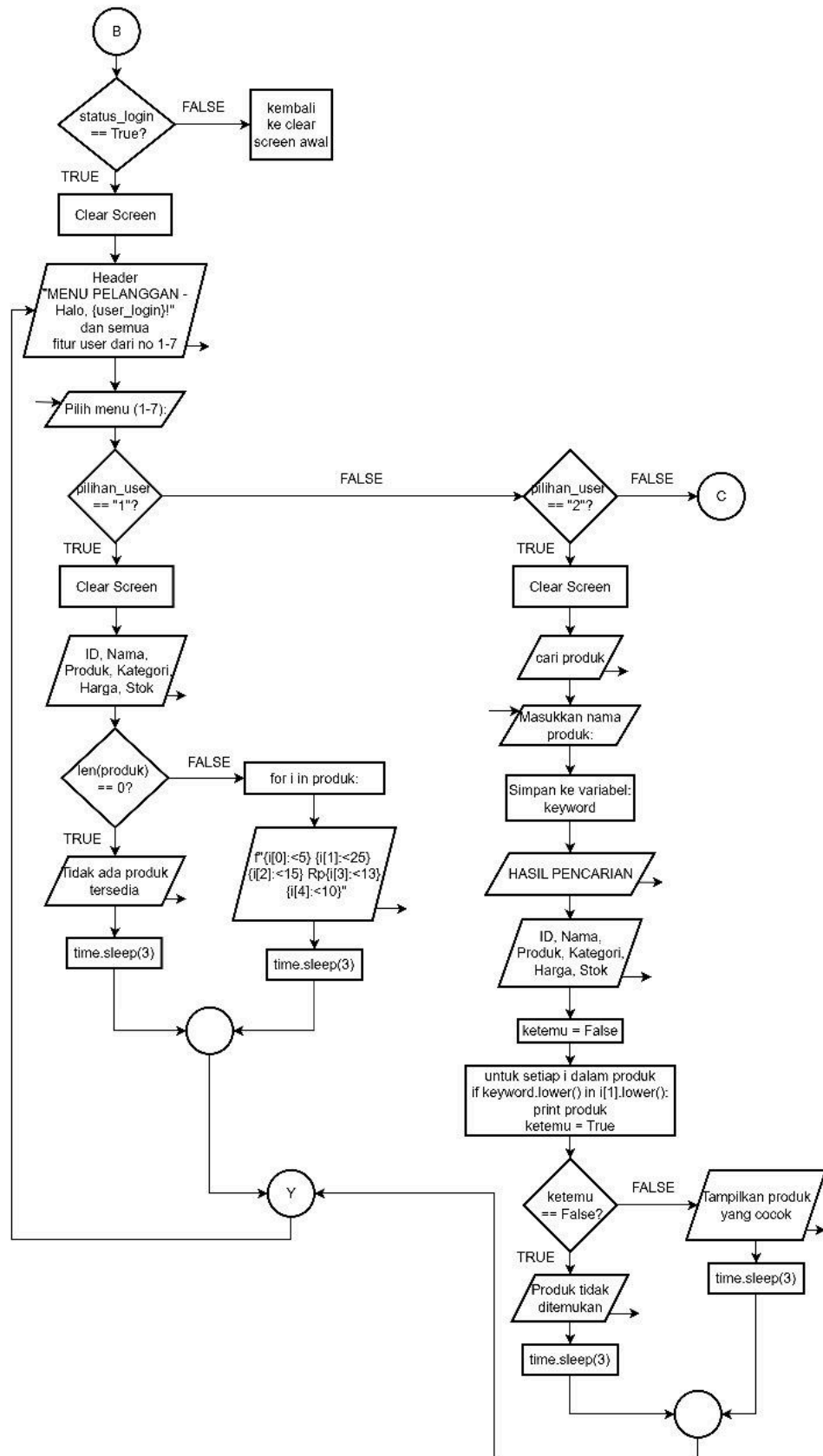
Gambar 1.4 Flowchart bagian on page D untuk program toko makanan kucing whinky



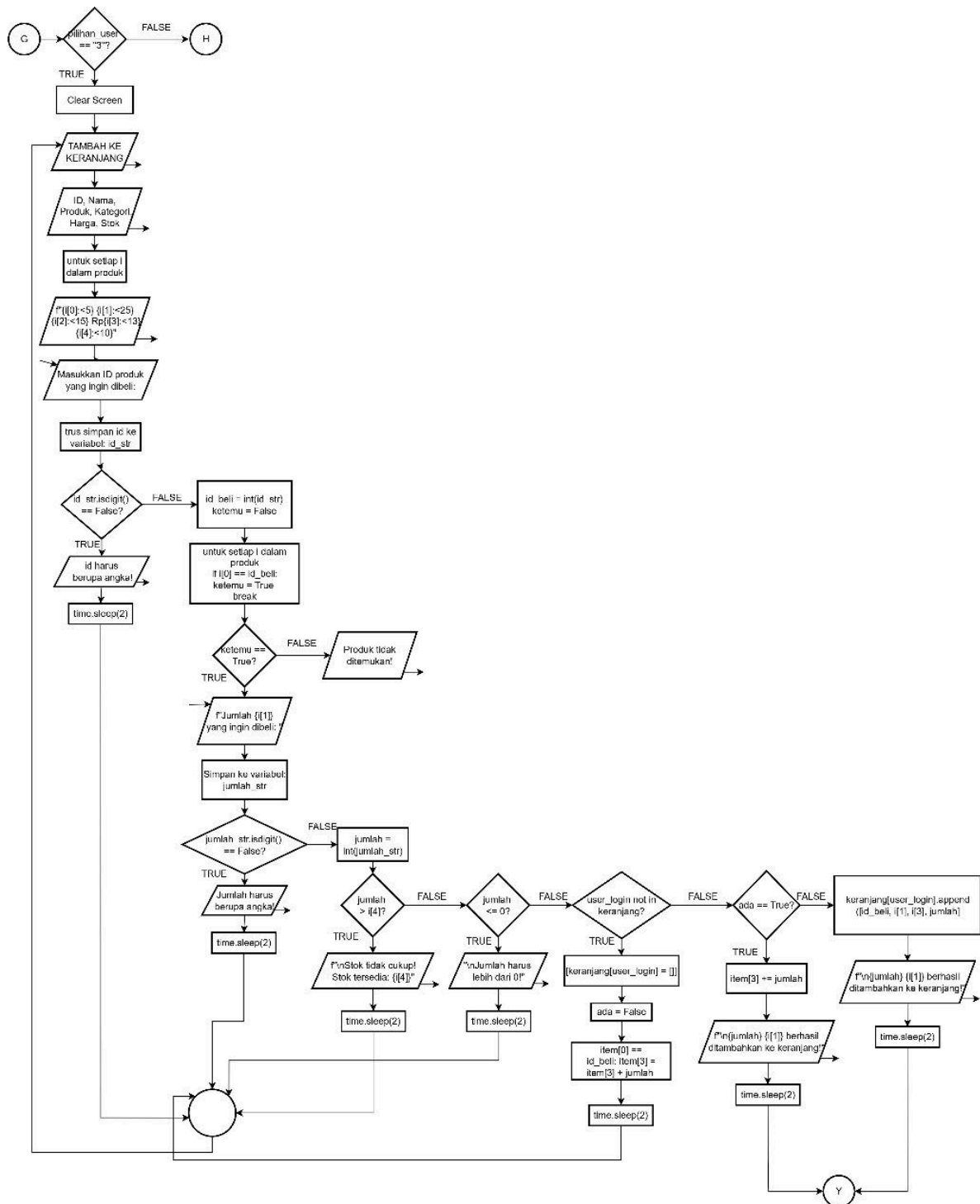
Gambar 1.5 Flowchart bagian on page E untuk program toko makanan kucing whinky



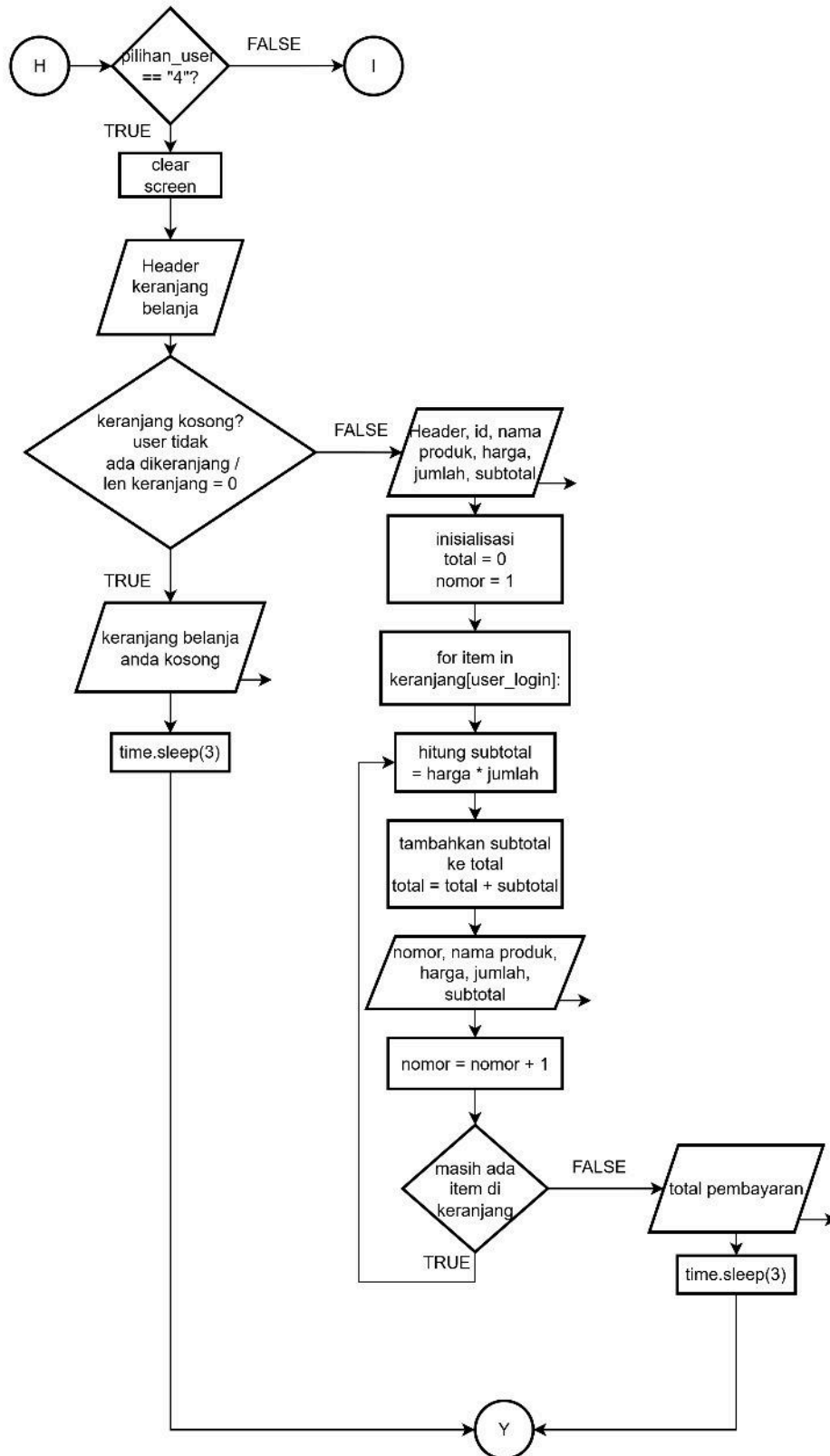
Gambar 1.6 Flowchart bagian on page F untuk program toko makanan kucing whinky



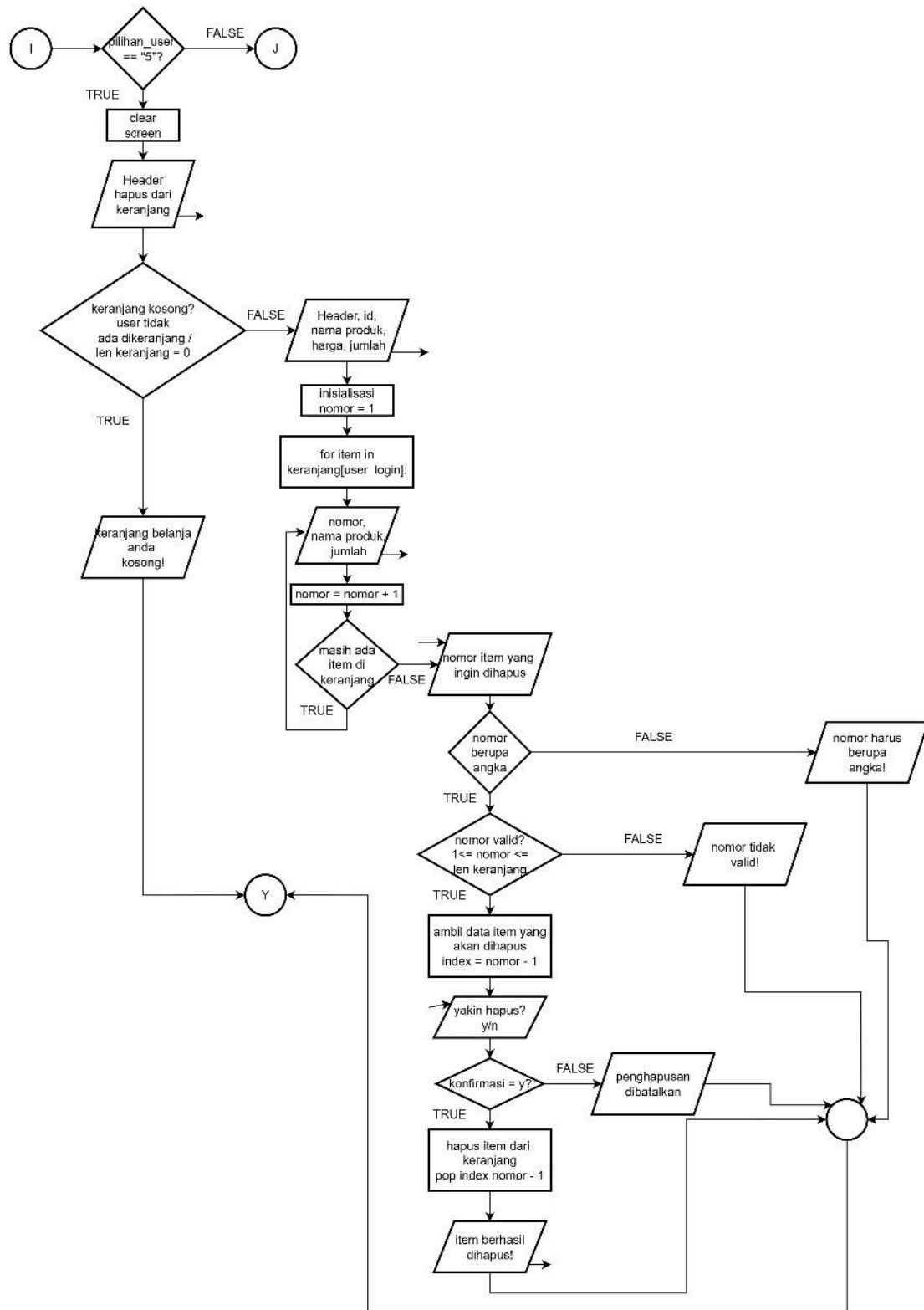
Gambar 1.7 Flowchart bagian on page B untuk program toko makanan kucing whinky



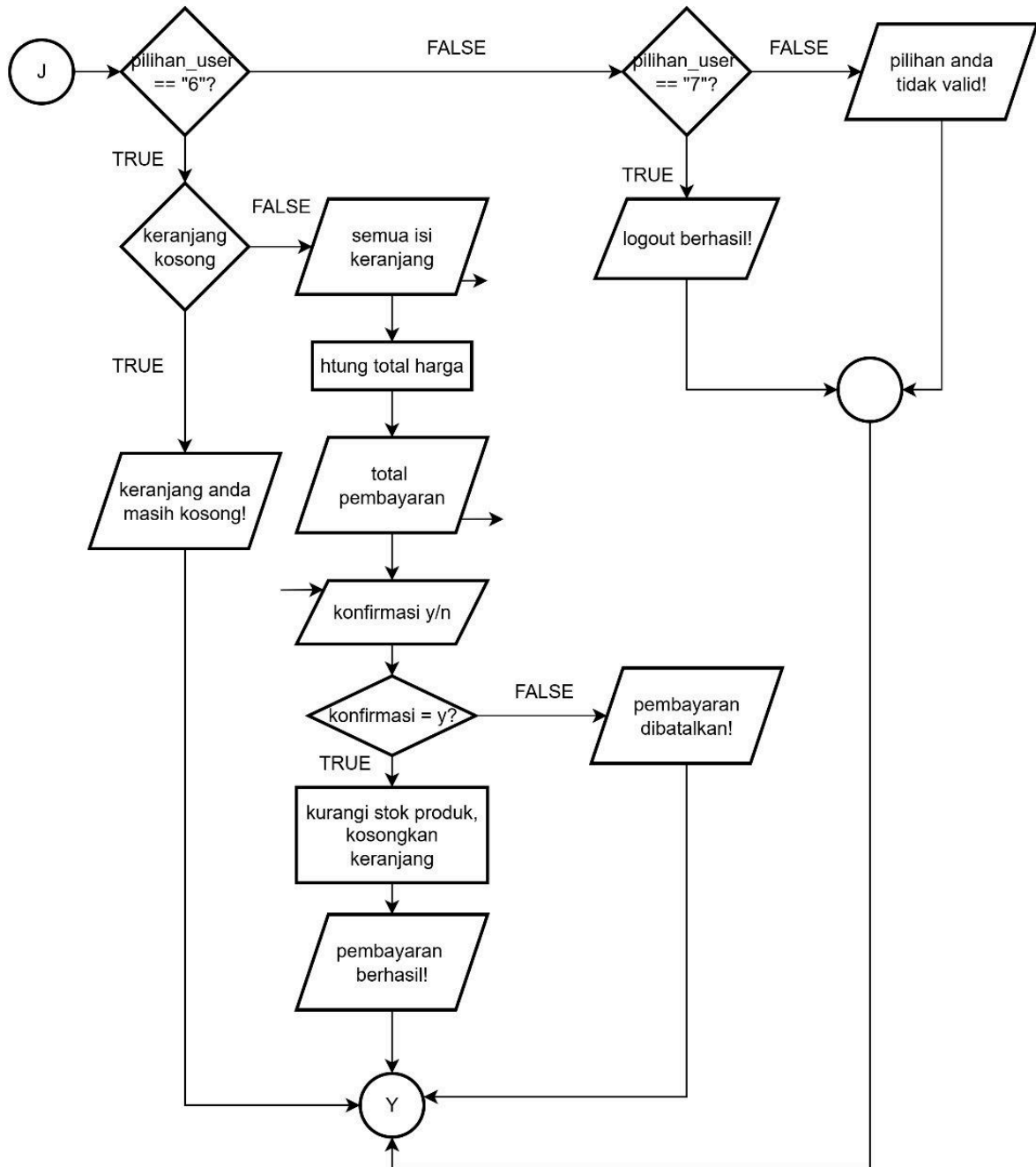
Gambar 1.8 Flowchart bagian on page G untuk program toko makanan kucing whinky



Gambar 1.9 Flowchart bagian on page H untuk program toko makanan kucing whinky



Gambar 1.10 Flowchart bagian on page I untuk program toko makanan kucing whinky



Gambar 1.11 Flowchart bagian on page J untuk program toko makanan kucing whinky

Jadi pertama tama yang muncul di tampilan program ini adalah ucapan selamat datang beserta tiga menu utama dari program ini yaitu 1. login (ini nanti bisa sebagai admin maupun user), 2. register (kita bisa buat nama username semau kita beserta passwordnya), 3. keluar. Lalu kita masuk ke pilihan 1, disitu nanti akan diminta input username dan password lalu program akan memproses buat dicek, kalau ketemu maka akan muncul output login berhasil selamat datang lalu program akan mengecek apakah yang memasukan inputan tadi merupakan admin atau user yang menentukan fitur kedepannya. jika pengguna memilih pilihan maka pengguna akan diminta untuk memberikan input berupa username dan

password baru sebagai data registrasi yang akan disimpan, lalu setelahnya akan muncul output registrasi berhasil, silahkan login untuk melanjutkan. Sedangkan di pilihan ketiga itu ya sudah jelas kalau hanya akan mendapatkan output terimakasih telah mengunjungi toko whinky atau jika pengguna iseng memilih pilihan selain 1-3 maka akan muncul output pilihan kamu tidak valid nih!.

Jika ternyata pengguna merupakan admin maka program akan memunculkan output dimana pengguna nanti diminta untuk memberikan inputan pilihan yang mau dilakukan oleh admin seperti pilihan 1 (lihat semua produk di pilihan ini nantinya program akan menampilkan tabel lengkap id nama produk stok nya dll), pilihan 2 (tambah produk, disini admin memiliki wewenang untuk menambahkan stok barang baru jadi nanti akan diminta input nama produk barunya, stoknya, harganya), pilihan 3 (update produk, nah disini admin dapat mengganti id nama jumlah stok harga dari semua barang yang sudah ada sebelumnya), pilihan 4 (menghapus produk, disini admin juga diberikan wewenang untuk menghapus etalase barang yang mana aja), pilihan 5 (cari produk, fitur ini sangat berguna apabila data barang nanti sudah mencapai ratusan atau ribuan karena dengan adanya fitur ini maka akan lebih cepat kita menemukan barang yang kita butuhkan atau kita cari cari.), dan yang terakhir fitur logout untuk keluar dari program.

Jika ternyata pengguna merupakan user maka program akan memunculkan output pilihan 1-7 yang dapat user pilih melalui input pilihan. Pilihan 1 (melihat semua produk yang tersedia di toko hari ini jadi tau stoknya sisa berapa dengan harga berapa), pilihan 2 (cari produk, sama halnya dengan fitur yang ada di admin ini yang berguna agar memudahkan user mencari nama barang yang sedang diinginkan), pilihan 3 (menambahkan produk ke keranjang, jadi user benar benar dapat membeli produk melalui program kita), pilihan 4 (lihat keranjang, fitur ini bukan hanya akan mempermudah user dari segi bisa melihat barang apa saja yang sudah terambil akan tetapi juga dapat sekaligus menghitung subtotal sementara dari belanjanya user), pilihan 5 (hapus dari keranjang disini maksudnya apabila user tidak jadi membeli produk di toko whinky namun hanya beberapa fitur ini hadir untuk mengatasi permasalahan itu dimana dengan program ini user dapat memilih barang spesifik apa yang ingin mereka kembalikan ke etalase atau sama saja dengan tidak membeli), pilihan 6 (checkout untuk fitur ini sendiri sudah jelas digunakan untuk bagian pembayaran ya, dan ini juga akan di konfirmasikan lagi apakah sudah benar pesannya serta langsung payment juga), dan yang terakhir itu udah pasti ada fitur logout untuk keluar dari program toko whinky

2. Deskripsi Singkat Program

Program sederhana mengenai toko makanan kucing winky ini dibuat dengan tujuan untuk mempermudah para pecinta kucing untuk dapat memenuhi kebutuhan pangan anak bulu mereka. Dengan adanya program sederhana berupa toko makanan winky ini para pengguna jadi tidak perlu cemas lagi untuk mencari toko makanan kucing terpercaya yang dapat menyediakan makanan kucing dengan berbagai merk yang mudah diakses hanya dari genggam tangan saja, bahkan jika pengguna sudah terdaftar sebagai member di toko

makanan kucing winky ini para pengguna akan mendapatkan benefit khusus berupa diskon sebesar 15% untuk semua item tanpa terkecuali. Terdapat beberapa fitur baru sebagai pembeda dari proyek sebelumnya yakni fitur yang lebih kompleks CRUD yang bisa dilakukan oleh admin dan juga pengguna (namun pengguna hanya dibatasi 7 fitur saja).

3. Source Code

Berikut ini merupakan source code yang saya buat untuk program toko makanan kucing whinky secara sederhana dengan fitur lebih kompleks (berstandar CRUD). Di posttest ke 6 ini saya hanya mengubah penggunaan nested list di posttest yang lalu menjadi dictionary dan selebihnya tidak ada fitur yang berubah.

```
import os
import time

# Database penggunaanya diganti jadi pakai dictionary sebelumnya kan pakai nested
list
# cara bacanya {username: {"password": password, "role": role}}
pengguna = {
    "gea": {
        "password": "pacar seokjin",
        "role": "admin"
    },
    "user": {
        "password": "user123",
        "role": "user"
    }
}

# cara bacanya: {id: {"nama": nama, "kategori": kategori, "harga": harga,
"stok": stok}}
produk = {
    1: {"nama": "Whiskas 1kg", "kategori": "Makanan", "harga": 50000, "stok":
20},
    2: {"nama": "Royal Canin 2kg", "kategori": "Makanan", "harga": 150000,
"stok": 15},
    3: {"nama": "Pasir Gumpal 5kg", "kategori": "Kebersihan", "harga": 45000,
"stok": 30},
    4: {"nama": "Sisir Kucing", "kategori": "Grooming", "harga": 25000, "stok":
50},
    5: {"nama": "Mainan Bola", "kategori": "Mainan", "harga": 15000, "stok": 40}
}

# cara bacanya: {username: {id_produk: {"nama": nama, "harga": harga, "jumlah":
jumlah}}}
keranjang = {}
```

```

# Variabel untuk ngecek user yang login
user_login = ""
role_login = ""
status_login = False

# ini program utamanya nda ada yang diubah cuma diubah bagian list kemarin jadi
dictionary aja
while True:
    os.system('cls || clear')
    print("=" * 60)
    print("|          SELAMAT DATANG DI TOKO PERALATAN KUCING WINGKY          |")
    print("=" * 60)
    print("\n1. Login")
    print("2. Register")
    print("3. Keluar")

    pilihan_utama = input("\nPilih menu (1-3): ")

    # ===== MENU LOGIN =====
    if pilihan_utama == "1":
        os.system('cls || clear')
        print("=" * 60)
        print("|          SELAMAT DATANG DI TOKO PERALATAN KUCING WINGKY          |")
        print("=" * 60)
        print("\n--- LOGIN ---")
        username = input("Username: ")
        password = input("Password: ")

        # Verifikasi login dulu disini
        if username in pengguna and pengguna[username]["password"] == password:
            user_login = username
            role_login = pengguna[username]["role"]
            status_login = True
            print(f"\nLogin berhasil! Selamat datang, {user_login}")
            time.sleep(1)

        # ===== MENU ADMIN =====
        if role_login == "admin":
            while status_login == True:
                os.system('cls || clear')
                print("=" * 50)
                print(f"|          MENU ADMIN - Halo, {user_login}!          |")

                print("=" * 50)
                print("\n1. Lihat Semua Produk")
                print("2. Tambah Produk")
                print("3. Update Produk")
                print("4. Hapus Produk")
                print("5. Cari Produk")
                print("6. Logout")

```



```

        pilihan_admin = input("\nPilih menu (1-6): ")

        # LIHAT PRODUK
        if pilihan_admin == "1":
            os.system('cls || clear')
            print("=" * 80)
            print("|                                     DAFTAR PRODUK
WINGKY                                     |")
            print("=" * 80)
            print(f"|{'ID':<5}| {'Nama Produk':<25}|
{'Kategori':<15}| {'Harga':<15}| {'Stok':<10}|")
            print("-" * 80)

            if len(produk) == 0:
                print("Tidak ada produk tersedia.")
            else:
                for id_produk, data in produk.items():
                    print(f"|{id_produk:<5}| {data['nama']:<25}|
{data['kategori']:<15}| Rp{data['harga']:<13}| {data['stok']:<10}|")

            print("=" * 80)
            time.sleep(10)

        # TAMBAH PRODUK
        elif pilihan_admin == "2":
            os.system('cls || clear')
            print("=" * 50)
            print("|                                     TAMBAH PRODUK BARU
|")

            print("=" * 50)

            if len(produk) == 0:
                id_baru = 1
            else:
                id_baru = max(produk.keys()) + 1

            nama = input("\nNama Produk: ")
            kategori = input("Kategori
(Makanan/Kebersihan/Grooming/Mainan): ")

            harga_str = input("Harga: ")
            if harga_str.isdigit() == False:
                print("\nHarga harus berupa angka!")
                time.sleep(2)
            else:
                harga = int(harga_str)

            stok_str = input("Stok: ")

```

```

        if stok_str.isdigit() == False:
            print("\nStok harus berupa angka!")
            time.sleep(2)
        else:
            stok = int(stok_str)
            produk[id_baru] = {
                "nama": nama,
                "kategori": kategori,
                "harga": harga,
                "stok": stok
            }
            print(f"\nProduk '{nama}' berhasil
ditambahkan!")

            time.sleep(2)

# UPDATE PRODUK
elif pilihan_admin == "3":
    os.system('cls || clear')
    print("=" * 50)
    print("|                                UPDATE PRODUK                                |")

    print("=" * 50)

    print(f"\n|{'ID':<5}| {'Nama Produk':<41}|")
    print("-" * 50)
    for id_produk, data in produk.items():
        print(f"|{id_produk:<5}| {data['nama']:<41}|")
        print("=" * 50)

    id_str = input("\nMasukkan ID produk yang ingin
diupdate: ")

    if id_str.isdigit() == False:
        print("\nID harus berupa angka!")
        time.sleep(2)
    else:
        id_update = int(id_str)

        if id_update in produk:
            print(f"\nProduk ditemukan:
{produk[id_update]['nama']}")

            print("\nPilih yang ingin diupdate:")
            print("1. Nama")
            print("2. Kategori")
            print("3. Harga")
            print("4. Stok")
            pilih = input("Pilihan: ")

            if pilih == "1":
                nama_baru = input("Nama baru: ")
                produk[id_update]["nama"] = nama_baru

```

```

        print("\nNama produk berhasil diupdate!")
    elif pilih == "2":
        kategori_baru = input("Kategori baru: ")
        produk[id_update]["kategori"] =
kategori_baru

        print("\nKategori produk berhasil
diupdate!")

    elif pilih == "3":
        harga_str = input("Harga baru: ")
        if harga_str.isdigit() == False:
            print("\nHarga harus berupa angka!")
        else:
            produk[id_update]["harga"] =
int(harga_str)

            print("\nHarga produk berhasil
diupdate!")

    elif pilih == "4":
        stok_str = input("Stok baru: ")
        if stok_str.isdigit() == False:
            print("\nStok harus berupa angka!")
        else:
            produk[id_update]["stok"] =
int(stok_str)

            print("\nStok produk berhasil
diupdate!")

    else:
        print("\nPilihan tidak valid!")
        time.sleep(2)
    else:
        print("\nProduk dengan ID tersebut tidak
ditemukan!")

        time.sleep(2)

# HAPUS PRODUK
elif pilihan_admin == "4":
    os.system('cls || clear')
    print("=" * 50)
    print("|                                HAPUS PRODUK
|")

    print("=" * 50)

    print(f"\n|{'ID':<5}| {'Nama Produk':<37}|")
    print("-" * 50)
    for id_produk, data in produk.items():
        print(f"|{id_produk:<5}| {data['nama']:<37}|")

    id_str = input("\nMasukkan ID produk yang ingin dihapus:
")

    if id_str.isdigit() == False:
        print("\nID harus berupa angka!")

```

```

        time.sleep(2)
    else:
        id_hapus = int(id_str)

        if id_hapus in produk:
            konfirmasi = input(f"\nYakin ingin menghapus
'{produk[id_hapus]['nama']}'? (y/n): ")
            if konfirmasi.lower() == "y":
                del produk[id_hapus]
                print("\nProduk berhasil dihapus!")
            else:
                print("\nPenghapusan dibatalkan.")
                time.sleep(2)
        else:
            print("\nProduk dengan ID tersebut tidak
ditemukan!")
            time.sleep(2)

# CARI PRODUK
elif pilihan_admin == "5":
    os.system('cls || clear')
    print("=" * 50)
    print("|                                CARI PRODUK                                |")
    print("=" * 50)
    keyword = input("\nMasukkan nama produk: ")

    print("\n" + "=" * 80)
    print("|                                HASIL PENCARIAN                                |")

    print("=" * 80)
    print(f"{'ID':<5} {'Nama Produk':<25} {'Kategori':<15}
{'Harga':<15} {'Stok':<10}")
    print("-" * 80)

    ketemu = False
    for id_produk, data in produk.items():
        if keyword.lower() in data['nama'].lower():
            print(f"{id_produk:<5} {data['nama']:<25}
{data['kategori']:<15} Rp{data['harga']:<13} {data['stok']:<10}")
            ketemu = True

    if ketemu == False:
        print("Produk tidak ditemukan.")

    print("=" * 80)
    time.sleep(3)

# LOGOUT
elif pilihan_admin == "6":
    print("\nLogout berhasil!")
    time.sleep(1)

```

```

        status_login = False

    else:
        print("\nPilihan tidak valid!")
        time.sleep(2)

# ===== MENU USER =====
else:
    while status_login == True:
        os.system('cls || clear')
        print("=" * 50)
        print(f"|                                MENU PELANGGAN - Halo, {user_login}!
|")

        print("=" * 50)
        print("\n1. Lihat Semua Produk")
        print("2. Cari Produk")
        print("3. Tambah ke Keranjang")
        print("4. Lihat Keranjang")
        print("5. Hapus dari Keranjang")
        print("6. Checkout")
        print("7. Logout")

        pilihan_user = input("\nPilih menu (1-7): ")

        # LIHAT PRODUK
        if pilihan_user == "1":
            os.system('cls || clear')
            print("=" * 80)
            print(f"|                                DAFTAR PRODUK
|")
            print("=" * 80)
            print(f"{'ID':<5} {'Nama Produk':<25} {'Kategori':<15}
{'Harga':<15} {'Stok':<10}")
            print("-" * 80)

            if len(produk) == 0:
                print("Tidak ada produk tersedia.")
            else:
                for id_produk, data in produk.items():
                    print(f"{id_produk:<5} {data['nama']:<25}
{data['kategori']:<15} Rp{data['harga']:<13} {data['stok']:<10}")

            print("=" * 80)
            time.sleep(7)

        # CARI PRODUK
        elif pilihan_user == "2":
            os.system('cls || clear')
            print("=" * 50)
            print(f"|                                CARI PRODUK
|")

```

```

        print("=" * 50)
        keyword = input("\nMasukkan nama produk: ")

        print("\n" + "=" * 80)
        print("|                                     HASIL Pencarian
|")

        print("=" * 80)
        print(f"{'ID':<5} {'Nama Produk':<25} {'Kategori':<15}
{'Harga':<15} {'Stok':<10}")
        print("-" * 80)

        ketemu = False
        for id_produk, data in produk.items():
            if keyword.lower() in data['nama'].lower():
                print(f"{'id_produk':<5} {'data['nama']':<25}
{data['kategori']:<15} Rp{data['harga']:<13} {data['stok']:<10}")
                ketemu = True

        if ketemu == False:
            print("Produk tidak ditemukan.")

        print("=" * 80)
        time.sleep(3)

    # TAMBAH KE KERANJANG
    elif pilihan_user == "3":
        os.system('cls || clear')
        print("=" * 80)
        print("|                                     TAMBAH KE KERANJANG
|")

        print("\n" + "=" * 80)
        print(f"| {'ID':<2} | {'Nama Produk':<26} |
{'Kategori':<15} | {'Harga':<15} | {'Stok':<10} |")
        print("-" * 80)
        for id_produk, data in produk.items():
            print(f"| {'id_produk':<2} | {data['nama']:<26} |
{data['kategori']:<15} | Rp{data['harga']:<13} | {data['stok']:<10} |")
            print("=" * 80)

        id_str = input("\nMasukkan ID produk yang ingin dibeli:
")

        if id_str.isdigit() == False:
            print("\nID harus berupa angka!")
            time.sleep(2)
        else:
            id_beli = int(id_str)

            if id_beli in produk:
                jumlah_str = input(f"Jumlah
{produk[id_beli]['nama']} yang ingin dibeli: ")

```

```

        if jumlah_str.isdigit() == False:
            print("\nJumlah harus berupa angka!")
            time.sleep(2)
        else:
            jumlah = int(jumlah_str)

            if jumlah > produk[id_beli]['stok']:
                print(f"\nStok tidak cukup! Stok
tersedia: {produk[id_beli]['stok']}")
                time.sleep(2)
            elif jumlah <= 0:
                print("\nJumlah harus lebih dari 0!")
                time.sleep(2)
            else:
                # Inisialisasi keranjang
                if user_login not in keranjang:
                    keranjang[user_login] = {}

                # Cek apakah produk sudah ada di
keranjang

                if id_beli in keranjang[user_login]:
keranjang[user_login][id_beli]["jumlah"] += jumlah
                else:
                    keranjang[user_login][id_beli] = {
                        "nama": produk[id_beli]["nama"],
                        "harga":
produk[id_beli]["harga"],
                        "jumlah": jumlah
                    }

                print(f"\n{jumlah}
{produk[id_beli]['nama']} berhasil ditambahkan ke keranjang!")
                time.sleep(2)
            else:
                print("\nProduk dengan ID tersebut tidak
ditemukan!")
                time.sleep(2)

        # LIHAT KERANJANG
        elif pilihan_user == "4":
            os.system('cls || clear')
            print("=" * 80)
            print("|
KERANJANG BELANJA
|")

            print("=" * 80)

            if user_login not in keranjang or
len(keranjang[user_login]) == 0:
                print("\nKeranjang belanja kosong.")

```

```

        print("=" * 80)
    else:
        print(f"| {'No':<5}| {'Nama Produk':<25}|
{'Harga':<15}| {'Jumlah':<10}| {'Subtotal':<14}|")
        print("-" * 80)

        total = 0
        nomor = 1
        for id_produk, item in
keranjang[user_login].items():
            subtotal = item["harga"] * item["jumlah"]
            total += subtotal
            print(f"| {'nomor':<5}| {item['nama']:<25}|
Rp{item['harga']:<13}| {item['jumlah']:<10}| Rp{subtotal:<14}|")
            nomor += 1

        print("-" * 80)
        print(f"| {'TOTAL':<56} Rp{total:<13}      |")
        print("=" * 80)

        time.sleep(7)

    # HAPUS DARI KERANJANG
    elif pilihan_user == "5":
        os.system('cls || clear')
        print("=" * 50)
        print("|                                HAPUS DARI KERANJANG
|")

        print("=" * 50)

        if user_login not in keranjang or
len(keranjang[user_login]) == 0:
            print("\nKeranjang belanja kosong.")
            time.sleep(2)
        else:
            print(f"\n  {'No':<5} {'Nama Produk':<25}
{'Jumlah':<10}")

            print("-" * 50)

            nomor = 1
            list_id = list(keranjang[user_login].keys())
            for id_produk in list_id:
                item = keranjang[user_login][id_produk]
                print(f"  {'nomor':<5} {item['nama']:<25}
{item['jumlah']:<10}")

                nomor += 1

            no_str = input("\nMasukkan nomor item yang ingin
dihapus: ")

            if no_str.isdigit() == False:

```



```

        print("\nNomor harus berupa angka!")
        time.sleep(2)
    else:
        no_hapus = int(no_str)

        if no_hapus < 1 or no_hapus >
len(keranjang[user_login]):
        print("\nNomor tidak valid!")
        time.sleep(2)
    else:
        id_hapus = list_id[no_hapus - 1]
        nama_item =
keranjang[user_login][id_hapus]["nama"]
        konfirmasi = input(f"\nYakin ingin menghapus
'{nama_item}' dari keranjang? (y/n): ")

        if konfirmasi.lower() == "y":
            del keranjang[user_login][id_hapus]
            print("\nItem berhasil dihapus dari
keranjang!")

        else:
            print("\nPenghapusan dibatalkan.")
            time.sleep(2)

# CHECKOUT
elif pilihan_user == "6":
    os.system('cls || clear')
    print("=" * 80)
    print("|
CHECKOUT
|")

    print("=" * 80)

    if user_login not in keranjang or
len(keranjang[user_login]) == 0:
        print("\nKeranjang belanja kosong. Tidak ada yang
bisa dicheckout.")
        time.sleep(2)
    else:
        print("\n" + "=" * 80)
        print(f"{'No':<5} {'Nama Produk':<25} {'Harga':<15}
{'Jumlah':<10} {'Subtotal':<15}")
        print("-" * 80)

        total = 0
        nomor = 1
        for id_produk, item in
keranjang[user_login].items():
            subtotal = item["harga"] * item["jumlah"]
            total += subtotal
            print(f"{nomor:<5} {item['nama']:<25}
Rp{item['harga']:<13} {item['jumlah']:<10} Rp{subtotal:<13}")

```

```

        nomor += 1

        print("-" * 80)
        print(f"{'TOTAL PEMBAYARAN':<56} Rp{total:<15}")
        print("=" * 80)

        konfirmasi = input("\nLanjutkan pembayaran? (y/n): ")

    ")

    if konfirmasi.lower() == "y":
        # Kurangi stok produk
        for id_produk, item in
keranjang[user_login].items():
            produk[id_produk]["stok"] -= item["jumlah"]

        # Kosongkan keranjang
        keranjang[user_login] = {}

        print("\n" + "=" * 50)
        print("                PEMBAYARAN BERHASIL!")
        print("        Terima kasih telah berbelanja di
Wingky!")

        print("=" * 50)
        time.sleep(8)
    else:
        print("\nPembayaran dibatalkan.")
        time.sleep(4)

    # LOGOUT
    elif pilihan_user == "7":
        print("\nLogout berhasil!")
        time.sleep(3)
        status_login = False

    else:
        print("\nPilihan tidak valid!")
        time.sleep(3)

else:
    print("\nUsername atau password salah!")
    time.sleep(3)

# ===== MENU REGISTER =====
elif pilihan_utama == "2":
    os.system('cls || clear')
    print("=" * 50)
    print("    SELAMAT DATANG DI TOKO PERALATAN KUCING WINGKY")
    print("=" * 50)
    print("\n--- REGISTER AKUN BARU ---")

    username_baru = input("Username: ")
    password_baru = input("Password: ")

```

```

    pengguna[username_baru] = {
        "password": password_baru,
        "role": "user"
    }
    print(f"\nRegistrasi berhasil! Akun '{username_baru}' telah dibuat.")
    print("Silakan login untuk melanjutkan.")
    time.sleep(4)

# ===== KELUAR =====
elif pilihan_utama == "3":
    os.system('cls || clear')
    print("=" * 60)
    print("|          Terima kasih telah mengunjungi Toko Wingky!          |")
    print("=" * 60)
    break

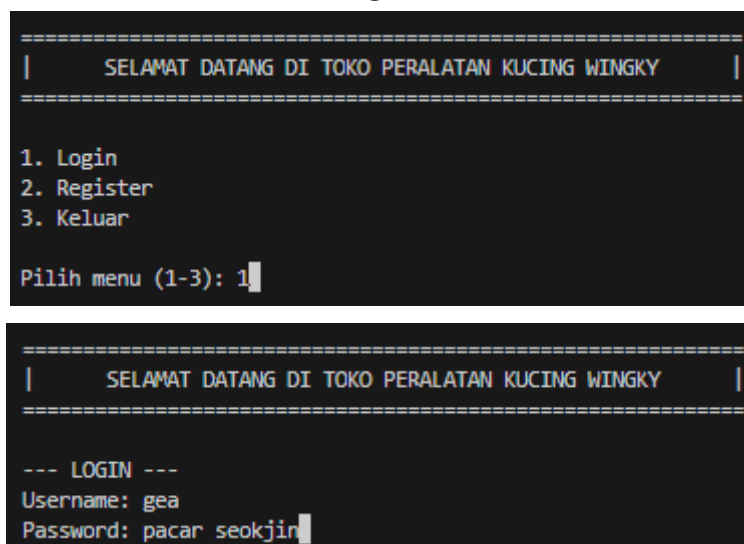
else:
    print("\nPilihan kamu tidak valid nih!")
    time.sleep(3)

```

4. Hasil Output

Berikut ini merupakan hasil output dari program toko makanan kucing whinky secara sederhana yang sudah saya buat.

4.1 Sebagai Admin



```

=====
|          SELAMAT DATANG DI TOKO PERALATAN KUCING WINGKY          |
=====

1. Login
2. Register
3. Keluar

Pilih menu (1-3): 1

=====
|          SELAMAT DATANG DI TOKO PERALATAN KUCING WINGKY          |
=====

--- LOGIN ---
Username: gea
Password: pacar seokjin

```

Gambar 4.1.1 Ketika login sebagai admin di pilihan 1 (login)

```
=====
| MENU ADMIN - Halo, gea! |
=====

1. Lihat Semua Produk
2. Tambah Produk
3. Update Produk
4. Hapus Produk
5. Cari Produk
6. Logout

Pilih menu (1-6): 1
```

Gambar 4.1.2 tampilan layar bila jadi admin

DAFTAR PRODUK WINGKY				
ID	Nama Produk	Kategori	Harga	Stok
1	Whiskas 1kg	Makanan	Rp50000	20
2	Royal Canin 2kg	Makanan	Rp150000	15
3	Pasir Gumpal 5kg	Kebersihan	Rp45000	30
4	Sisir Kucing	Grooming	Rp25000	50
5	Mainan Bola	Mainan	Rp15000	40

Gambar 4.1.3 output pilihan 1 (melihat semua produk)

```
=====
| TAMBAH PRODUK BARU |
=====

Nama Produk: litter box
Kategori (Makanan/Kebersihan/Grooming/Mainan): kebersihan
Harga: 65000
Stok: 5
```

Gambar 4.1.4 output pilihan 2 (menambahkan produk baru pada etalase toko)

```
=====
| UPDATE PRODUK |
=====

|ID | Nama Produk |
-----
|1 | Whiskas 1kg |
-----
|2 | Royal Canin 2kg |
-----
|3 | Pasir Gumpal 5kg |
-----
|4 | Sisir Kucing |
-----
|5 | Mainan Bola |
-----
|6 | litter box |
=====

Masukkan ID produk yang ingin diupdate: 6

Produk ditemukan: litter box

Pilih yang ingin diupdate:
1. Nama
2. Kategori
3. Harga
4. Stok
Pilihan: 4
Stok baru: 7
```

Gambar 4.1.5 output pilihan 3 (jadi bisa memperbarui nama/ kategori/ harga/ stok)

```
=====
| HAPUS PRODUK |
=====

|ID | Nama Produk |
-----

|1 | Whiskas 1kg |
|2 | Royal Canin 2kg |
|3 | Pasir Gumpal 5kg |
|4 | Sisir Kucing |
|5 | Mainan Bola |
|6 | litter box |
-----

Masukkan ID produk yang ingin dihapus: 1

Yakin ingin menghapus 'Whiskas 1kg'? (y/n): y
```

Gambar 4.1.6 output pilihan 4 (admin dapat menghapus apapun yang ada di etalase)

```
=====
Masukkan nama produk: sisir
=====
|                                     |
|             HASIL PENCARIAN             |
|                                     |
|-----|-----|-----|-----|
| ID  Nama Produk      Kategori    Harga    Stok |
|-----|-----|-----|-----|
| 4    Sisir Kucing    Grooming    Rp25000    50  |
|-----|-----|-----|-----|
|                                     |
|
```

Gambar 4.1.7 output pilihan 5 (admin bisa mencari produk lebih cepat dengan fitur ini)

4.2 sebagai user

```
=====
| SELAMAT DATANG DI TOKO PERALATAN KUCING WINGKY |
|=====|
1. Login
2. Register
3. Keluar
Pilih menu (1-3): 2
```

Gambar 4.2.1 output pilihan 2 sedari awal dengan tujuan daftar akun baru

```
=====
| SELAMAT DATANG DI TOKO PERALATAN KUCING WINGKY |
|=====|
--- REGISTER AKUN BARU ---
Username: seokjin
Password: ganteng

Registrasi berhasil! Akun 'seokjin' telah dibuat.
Silakan login untuk melanjutkan.
```

Gambar 4.2.1.a output pilihan 2 sedari awal dengan tujuan daftar akun baru

```

=====
|          MENU PELANGGAN - Halo, seokjin!          |
=====

1. Lihat Semua Produk
2. Cari Produk
3. Tambah ke Keranjang
4. Lihat Keranjang
5. Hapus dari Keranjang
6. Checkout
7. Logout

Pilih menu (1-7): 1

```

Gambar 4.2.2 Ketika login sebagai user (kalau registrasi itu akunnya nanti jadi user) di pilihan 1 (login)

```

=====
|          DAFTAR PRODUK WINGKY          |
=====
ID  Nama Produk      Kategori  Harga      Stok
-----
1   Whiskas 1kg      Makanan  Rp50000    20
2   Royal Canin 2kg   Makanan  Rp150000   15
3   Pasir Gumpal 5kg  Kebersihan  Rp45000    30
4   Sisir Kucing     Grooming  Rp25000    50
5   Mainan Bola       Mainan    Rp15000    40
=====

```

Gambar 4.2.3 output pilihan 1 (melihat semua produk)

```

=====
|          CARI PRODUK          |
=====

Masukkan nama produk: whiskas

=====
|          HASIL PENCARIAN          |
=====
ID  Nama Produk      Kategori  Harga      Stok
-----
1   Whiskas 1kg      Makanan  Rp50000    20
=====

```

Gambar 4.2.4 output pilihan 2 (fitur mencari produk)

```

=====
|          TAMBAH KE KERANJANG          |
=====

=====
ID   Nama Produk      Kategori   Harga      Stok
-----
1    Whiskas 1kg       Makanan   Rp50000     20
2    Royal Canin 2kg    Makanan   Rp150000    15
3    Pasir Gumpal 5kg   Kebersihan Rp45000     30
4    Sisir Kucing      Grooming  Rp25000     50
5    Mainan Bola         Mainan    Rp15000     40
=====

Masukkan ID produk yang ingin dibeli: 2
Jumlah Royal Canin 2kg yang ingin dibeli: 7

7 Royal Canin 2kg berhasil ditambahkan ke keranjang!

```

Gambar 4.2.5 output pilihan 3 (jadi user bisa nambahin barang kekeranjang sepuasnya
*asalkan stoknya cukup yaa)

```

=====
|          KERANJANG BELANJA          |
=====
No   Nama Produk      Harga      Jumlah   Subtotal
-----
1    Royal Canin 2kg    Rp150000   7         Rp1050000
2    Pasir Gumpal 5kg   Rp45000    2         Rp90000
-----
TOTAL                                Rp1140000
=====

```

Gambar 4.2.6 output pilihan 4 (fitur melihat isi keranjang sekaligus udah dihitung
kira kira subtotal belanjanya berapa)

```

=====
|          HAPUS DARI KERANJANG        |
=====

No   Nama Produk      Jumlah
-----
1    Royal Canin 2kg    7
2    Pasir Gumpal 5kg   2

Masukkan nomor item yang ingin dihapus: 2

Yakin ingin menghapus 'Pasir Gumpal 5kg' dari keranjang? (y/n): y

Item berhasil dihapus dari keranjang!

```


Gambar 4.2.7 output pilihan 5 (fitur menghapus produk pilihan yang ada di keranjang)

```
=====
|                               |
=====

=====
No    Nama Produk      Harga      Jumlah    Subtotal
=====
1     Royal Canin 2kg   Rp150000   7          Rp1050000
=====
TOTAL PEMBAYARAN                               Rp1050000
=====

Lanjutkan pembayaran? (y/n): y

=====
PEMBAYARAN BERHASIL!
Terima kasih telah berbelanja di Wingky!
=====
```

Gambar 4.2.8 output pilihan 6 (fitur checkout)

```
=====
|   Terima kasih telah mengunjungi Toko Wingky!   |
=====
```

Gambar 4.2.9 jika pengguna memilih pilihan 3 sejak awal

Jadi apabila sedari awal pengguna memilih registrasi pada pilihan 2 itu sudah pasti jadi user karna adminnya hanya 1 orang di program ini (bisa ditambah sih kalau mau), lalu kalau pengguna selalu memilih yang tidak ada di pilihan pasti akan keluar output eror yang beragam (seperti outputan eror yang ada di program toko peralatan kucing whinky ini) sebelum kembali ke looping awal lagi.

5. Langkah-langkah GIT

Git adalah sistem kontrol versi yang membantu kita melacak perubahan pada kode atau proyek kita, jadi dengan adanya git ini kita itu bisa ngeliat semua perubahan yang kita buat dari awal gitu, bahkan kita bisa kembali ke versi sebelumnya tanpa kehilangan jejak dari proyek yang sudah kita buat sebelumnya.

5.1 GIT Add

Git add digunakan untuk menambahkan file dan memasukan file ke daftar perubahan. Jadi kita bisa memindahkan file dari tempat asalnya ke tempat dimana file siap di commit dengan menggunakan Git add ini.

```
PS C:\Users\ASUS\OneDrive\Documents\praktikum-apd> git add .
```

Gambar 5.1.1 Penggunaan Git Add

5.2 GIT Commit

Git commit ini digunakan untuk menyimpan perubahan secara permanen, jadi kita bisa kembali lagi kesini kapan aja (udah seperti titik check point di game gitu dia).

```
PS C:\Users\ASUS\OneDrive\Documents\praktikum-apd> git commit -m "first commit"
[main ac10e00] first commit
2 files changed, 614 insertions(+)
create mode 100644 A1-2025/pertemuan 6.py
create mode 100644 post-test/post-test-apd-6/2509106007-Syarifah Anargya Rizky.py
```

Gambar 5.2.1 Penggunaan Git Commit

5.3 GIT Push

Git push digunakan untuk mengirim commit kita ke github nanti, jadi disambungkan gitu repository lokal kita ke github.

```
PS C:\Users\ASUS\OneDrive\Documents\praktikum-apd> git push
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 4 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (7/7), 5.03 KiB | 368.00 KiB/s, done.
Total 7 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/seokjinjuseyyo/praktikum-apd.git
ccf5724..ac10e00 main -> main
```

Gambar 5.3.1 penggunaan Git Push untuk menghubungkan repository lokal ke github