

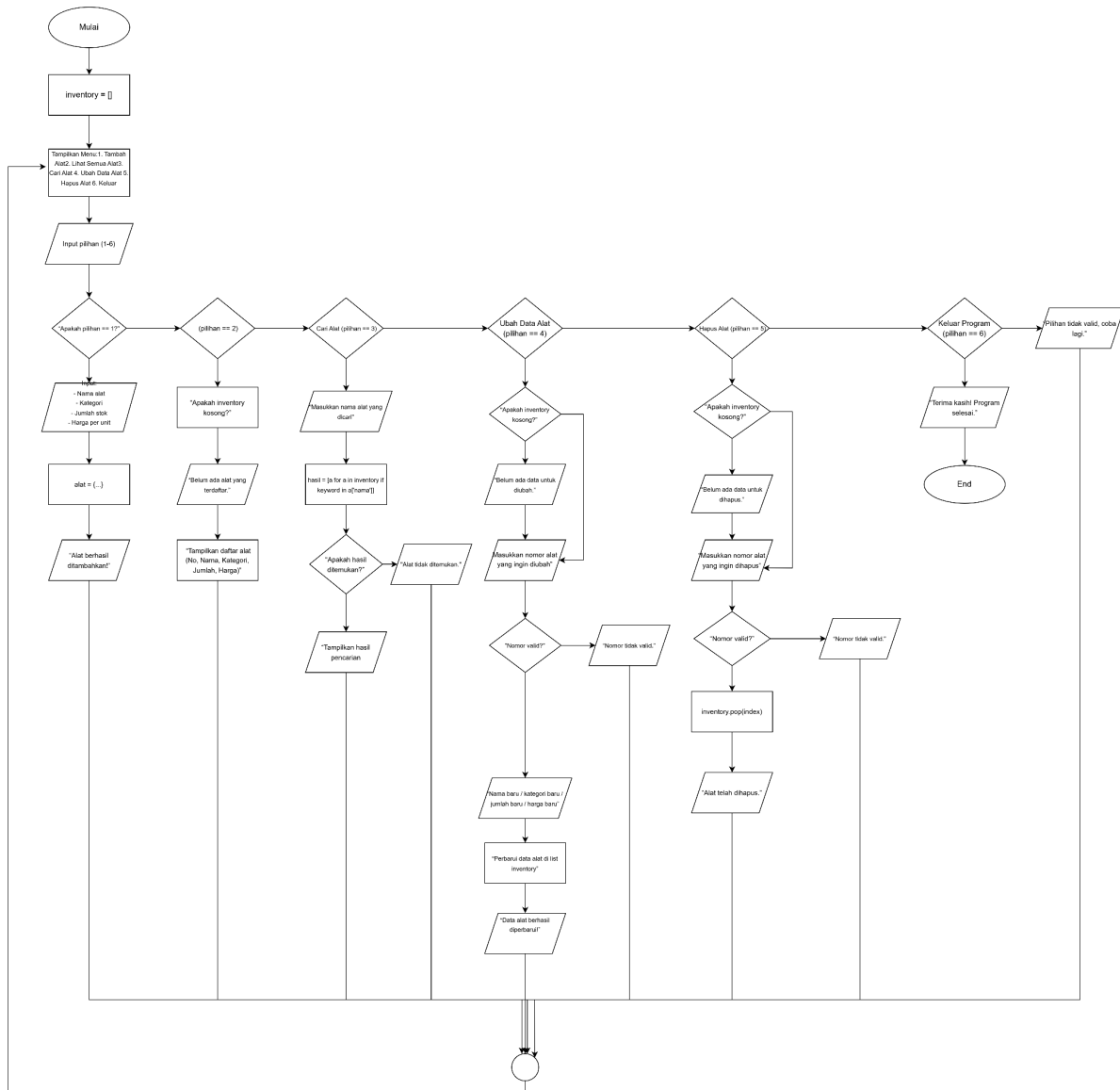
**LAPORAN PRAKTIKUM**  
**POSTTEST 7**  
**ALGORITMA PEMROGRAMAN DASAR**



**Disusun oleh:**  
**Fathur Rahman (2509106023)**  
**Kelas (A'25)**

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

## 1. Flowchart



**Gambar 1.1 Flowchart**

Flowchart Manajemen Inventory Alat Bangunan adalah diagram alur yang menggambarkan proses pengelolaan data alat bangunan secara sistematis dan terstruktur dalam sebuah program. Flowchart ini menunjukkan urutan langkah-langkah logika program, mulai dari memasukkan data alat, menyimpan informasi ke dalam sistem, menampilkan daftar alat, mencari, mengubah, hingga menghapus data alat yang tidak dibutuhkan lagi.

## 2. Deskripsi Singkat Program

Manajemen Inventory Alat Bangunan adalah sistem yang digunakan untuk mengatur, mencatat, dan memantau persediaan alat-alat bangunan seperti semen, kayu, cat, besi, paku, dan perlengkapan lainnya. Sistem ini membantu agar data alat tersimpan dengan rapi, mudah dicari, serta dapat diperbarui kapan saja jika ada perubahan stok atau harga.

## 3. Source Code

```
import os

users = {
    "admin": "1234",
    "kasir": "0000",
    "manager": "abcd"
}
inventory = {}
login_user = ""

def clear_screen():
    """Membersihkan layar terminal"""
    os.system("cls || clear")

def tampilkan_semua_alat():
    """Menampilkan semua data alat"""
    if not inventory:
        print("Belum ada alat yang terdaftar.")
    else:
        print(f"{'No':<4}{'Nama':<20}{'Kategori':<15}{'Jumlah':<10}{'Harga':<10}")
        for i, (nama, data) in enumerate(inventory.items(), start=1):
            print(f"{'i':<4}{nama:<20}{data['kategori']:<15}{data['jumlah']:<10}{data['harga']:<10.2f}")

print(f"{'i':<4}{nama:<20}{data['kategori']:<15}{data['jumlah']:<10}{data['harga']:<10.2f}")

def tambah_alat(nama, kategori, jumlah, harga):
    """Menambah alat baru ke inventory"""
    global inventory
    if nama in inventory:
        print("Alat ini sudah ada di inventory.")
    else:
        inventory[nama] = {"kategori": kategori, "jumlah": jumlah, "harga": harga}
        print(f"Alat '{nama}' berhasil ditambahkan!")

def cari_alat(keyword):
    """Mencari alat berdasarkan nama"""
    hasil = {n: d for n, d in inventory.items() if keyword.lower() in n.lower()}
    if hasil:
        print(f"\nDitemukan {len(hasil)} hasil:")
        for nama, data in hasil.items():
            print(f"- {nama} ({data['kategori']}), stok: {data['jumlah']}, harga: Rp{data['harga']:.2f}")
    else:
        print("Alat tidak ditemukan.")
```

```

def ubah_data_alat():
    """Prosedur untuk mengubah data alat"""
    global inventory
    if not inventory:
        print("Belum ada data untuk diubah.")
        return

    tampilkan_semua_alat()
    nama_alat = input("\nMasukkan nama alat yang ingin diubah: ")

    if nama_alat not in inventory:
        print("Alat tidak ditemukan.")
    else:
        data = inventory[nama_alat]
        nama_baru = input("Nama baru (kosongkan jika tidak diubah): ")
        kategori_baru = input("Kategori baru (kosongkan jika tidak diubah): ")
        jumlah_baru = input("Jumlah baru (kosongkan jika tidak diubah): ")
        harga_baru = input("Harga baru (kosongkan jika tidak diubah): ")

        if nama_baru:
            inventory[nama_baru] = inventory.pop(nama_alat)
            nama_alat = nama_baru
        if kategori_baru:
            data['kategori'] = kategori_baru
        if jumlah_baru:
            try:
                data['jumlah'] = int(jumlah_baru)
            except ValueError:
                print("Jumlah harus berupa angka!")
        if harga_baru:
            try:
                data['harga'] = float(harga_baru)
            except ValueError:
                print("Harga harus berupa angka!")

        print("Data alat berhasil diperbarui!")

def hapus_alat():
    """Prosedur untuk menghapus alat"""
    global inventory
    if not inventory:
        print("Belum ada data untuk dihapus.")
        return

    tampilkan_semua_alat()
    nama_hapus = input("\nMasukkan nama alat yang ingin dihapus: ")

    if nama_hapus in inventory:
        del inventory[nama_hapus]
        print(f"Alat '{nama_hapus}' telah dihapus.")
    else:
        print("Nama alat tidak ditemukan.")

def menu():
    """Menampilkan menu utama menggunakan rekursi"""
    clear_screen()
    print("\n=== MENU MANAJEMEN INVENTORY ALAT BANGUNAN ===")
    print("1. Tambah Alat")
    print("2. Lihat Semua Alat")

```

```

print("3. Cari Alat")
print("4. Ubah Data Alat")
print("5. Hapus Alat")
print("6. Keluar")

pilihan = input("Pilih menu (1-6): ")

try:
    if pilihan == "1":
        print("\n--- Tambah Alat Baru ---")
        nama = input("Nama alat: ")
        kategori = input("Kategori: ")
        try:
            jumlah = int(input("Jumlah stok: "))
            harga = float(input("Harga per unit: "))
            tambah_alat(nama, kategori, jumlah, harga)
        except ValueError:
            print("Input jumlah/harga tidak valid!")

    elif pilihan == "2":
        tampilkan_semua_alat()

    elif pilihan == "3":
        keyword = input("Masukkan nama alat yang dicari: ")
        cari_alat(keyword)

    elif pilihan == "4":
        ubah_data_alat()

    elif pilihan == "5":
        hapus_alat()

    elif pilihan == "6":
        print(f"Terima kasih, {login_user}! Program selesai.")
        return

    else:
        print("Pilihan tidak valid!")
except Exception as e:
    print(f"Terjadi kesalahan: {e}")

input("\nTekan Enter untuk kembali ke menu...")
menu()

def login():
    global login_user
    print("=== SISTEM LOGIN ===")
    percobaan = 0
    while percobaan < 3:
        username = input("Masukkan username: ")
        password = input("Masukkan password: ")

        if username in users and users[username] == password:
            print(f"\nSelamat datang, {username}!")
            login_user = username
            return True
        else:
            print("Username atau password salah!")
            percobaan += 1
    print("\nTerlalu banyak percobaan. Program berhenti.")

```

```
return False

if login():
    menu()
```

*Gambar 3.1 source code*

#### 4. Hasil Output

- Menu manajemen inventory alat bangunan

```
=== MENU MANAJEMEN INVENTORY ALAT BANGUNAN ===
1. Tambah Alat
2. Lihat Semua Alat
3. Cari Alat
4. Ubah Data Alat
5. Hapus Alat
6. Keluar
Pilih menu (1-6): 1

--- Tambah Alat Baru ---
Nama alat: 
```

*Gambar 4.1 Output menu manajemen inventory alat bangunan*

- Bagian untuk menambahkan alat baru

```
--- Tambah Alat Baru ---
Nama alat: kayu
Kategori (contoh: cat, semen, kayu, besi): kayu
Jumlah stok: 50
Harga per unit: 50000
Alat 'kayu' berhasil ditambahkan!
```

*Gambar 4.2 Output penambahan alat baru*

- **Daftar semua alat bangunan**

```
--- Daftar Semua Alat ---
```

No	Nama	Kategori	Jumlah	Harga
1	cat	cat	20	50000.00
2	semen	semen	30	90000.00
3	kayu	kayu	50	60000.00
4	besi	besi	60	80000.00

*Gambar 4.3 Output daftar semua alat bangunan*

- **Cari daftar alat bangunan**

```
--- Cari Alat ---  
Masukkan nama alat yang dicari: cat  
  
Ditemukan 1 hasil:  
- cat (cat), stok: 20, harga: Rp50000.00
```

*Gambar 4.4 Output cari daftar alat bangunan*

- Daftar alat yang ingin di ubah

```

=== MENU MANAJEMEN INVENTORY ALAT BANGUNAN ===
1. Tambah Alat
2. Lihat Semua Alat
3. Cari Alat
4. Ubah Data Alat
5. Hapus Alat
6. Keluar
Pilih menu (1-6): 4

--- Daftar Semua Alat ---
No  Nama           Kategori      Jumlah      Harga
1   cat            cat           20           50000.00
2   semen          semen         30           90000.00
3   kayu           kayu          50           60000.00
4   besi           besi          60           80000.00

Masukkan nomor alat yang ingin diubah: 4
Mengubah data 'besi'
Nama baru (biarkan kosong jika tidak diubah): paku
Kategori baru (biarkan kosong jika tidak diubah):
Jumlah baru (biarkan kosong jika tidak diubah): 90
Harga baru (biarkan kosong jika tidak diubah): 20000
Data alat berhasil diperbarui!

=== MENU MANAJEMEN INVENTORY ALAT BANGUNAN ===
1. Tambah Alat
2. Lihat Semua Alat
3. Cari Alat
4. Ubah Data Alat
5. Hapus Alat
6. Keluar
Pilih menu (1-6): 2

--- Daftar Semua Alat ---
No  Nama           Kategori      Jumlah      Harga
1   cat            cat           20           50000.00
2   semen          semen         30           90000.00
3   kayu           kayu          50           60000.00
4   paku           besi          90           20000.00

```

*Gambar 4.5 Output alat yang ingin di*



- daftar alat yang ingin dihapus

```
=== MENU MANAJEMEN INVENTORY ALAT BANGUNAN ===
1. Tambah Alat
2. Lihat Semua Alat
3. Cari Alat
4. Ubah Data Alat
5. Hapus Alat
6. Keluar
Pilih menu (1-6): 5

--- Daftar Semua Alat ---
No  Nama           Kategori    Jumlah  Harga
1   cat            cat         20      50000.00
2   semen          semen       30      90000.00
3   kayu           kayu        50      60000.00
4   paku           paku        90      20000.00

Masukkan nomor alat yang ingin dihapus: 4
Alat 'paku' telah dihapus.

=== MENU MANAJEMEN INVENTORY ALAT BANGUNAN ===
1. Tambah Alat
2. Lihat Semua Alat
3. Cari Alat
4. Ubah Data Alat
5. Hapus Alat
6. Keluar
Pilih menu (1-6): 2

--- Daftar Semua Alat ---
No  Nama           Kategori    Jumlah  Harga
1   cat            cat         20      50000.00
2   semen          semen       30      90000.00
3   kayu           kayu        50      60000.00
```

*gambar 4.6 Output alat yang ingin di hapus*

- Program telah selesai

```

=== MENU MANAJEMEN INVENTORY ALAT BANGUNAN ===
1. Tambah Alat
2. Lihat Semua Alat
3. Cari Alat
4. Ubah Data Alat
5. Hapus Alat
6. Keluar
Pilih menu (1-6): 6
Terima kasih! Program selesai.
PS D:\praktikum-apd>

```

*Gambar 4.7 Output program selesai*

## 5. GIT

### 5.1 GIT Add

```

PS D:\praktikum-apd> git add .

```

*Gambar 5.1 GIT add*

Git add itu perintah untuk memasukkan file atau perubahan file kedalam daftar antrian.

### 5.2 GIT Commit

```

PS D:\praktikum-apd> git commit -m "pos-test 7"
[main 12820e1] pos-test 7
11 files changed, 470 insertions(+)
create mode 100644 post-test/post-test-apd-5/# Program Manajemen Toko Apotek.py
create mode 100644 post-test/post-test-apd-5/# contoh dictionary.py
create mode 100644 post-test/post-test-apd-5/# contoh set modul 7.py
create mode 100644 post-test/post-test-apd-5/# jadwal bermain badminton.py
create mode 100644 post-test/post-test-apd-5/# penambahan, penghapusan, dan perulangan.py
create mode 100644 post-test/post-test-apd-5/# penambahan, penghapusan, dan perulangan.py
create mode 100644 post-test/post-test-apd-5/apotek = {'alpara', 'paracetamol', 'lodi.py
create mode 100644 post-test/post-test-apd-5/jadwal_kelas = {'senin', 'selasa', 'rabu.py
create mode 100644 post-test/post-test-apd-5/jenis_latihan = {'easy run', 'tempo run'.py
create mode 100644 post-test/post-test-apd-7/2509106023-Fathur Rahman-PT-7.py
create mode 100644 praktikum 7

```

*Gambar 5.2 GIT commit*

Git commit itu digunakan untuk **menyimpan perubahan (save)** yang sudah kamu buat di repositori lokal (komputer kamu sendiri).

Ibaratnya seperti **menekan tombol “save”** dengan catatan perubahan di proyekmu.

### 5.3 GIT Push

```
PS D:\praktikum-apd> git push
Enumerating objects: 19, done.
Counting objects: 100% (19/19), done.
Delta compression using up to 16 threads
Compressing objects: 100% (16/16), done.
Writing objects: 100% (16/16), 5.46 KiB | 931.00 KiB/s, done.
Total 16 (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/FathurRahmanA25/praktikum-apd.git
   0ead9c3..12820e1  main -> main
PS D:\praktikum-apd> █
```

**Gambar 5.3 GIT push**

Git push digunakan untuk **mengirim (upload)** perubahan yang sudah kamu *commit* ke **repositori online** seperti GitHub, GitLab, atau Bitbucket.