

LAPORAN PRAKTIKUM
POSTTEST 4
ALGORITMA PEMROGRAMAN LANJUT

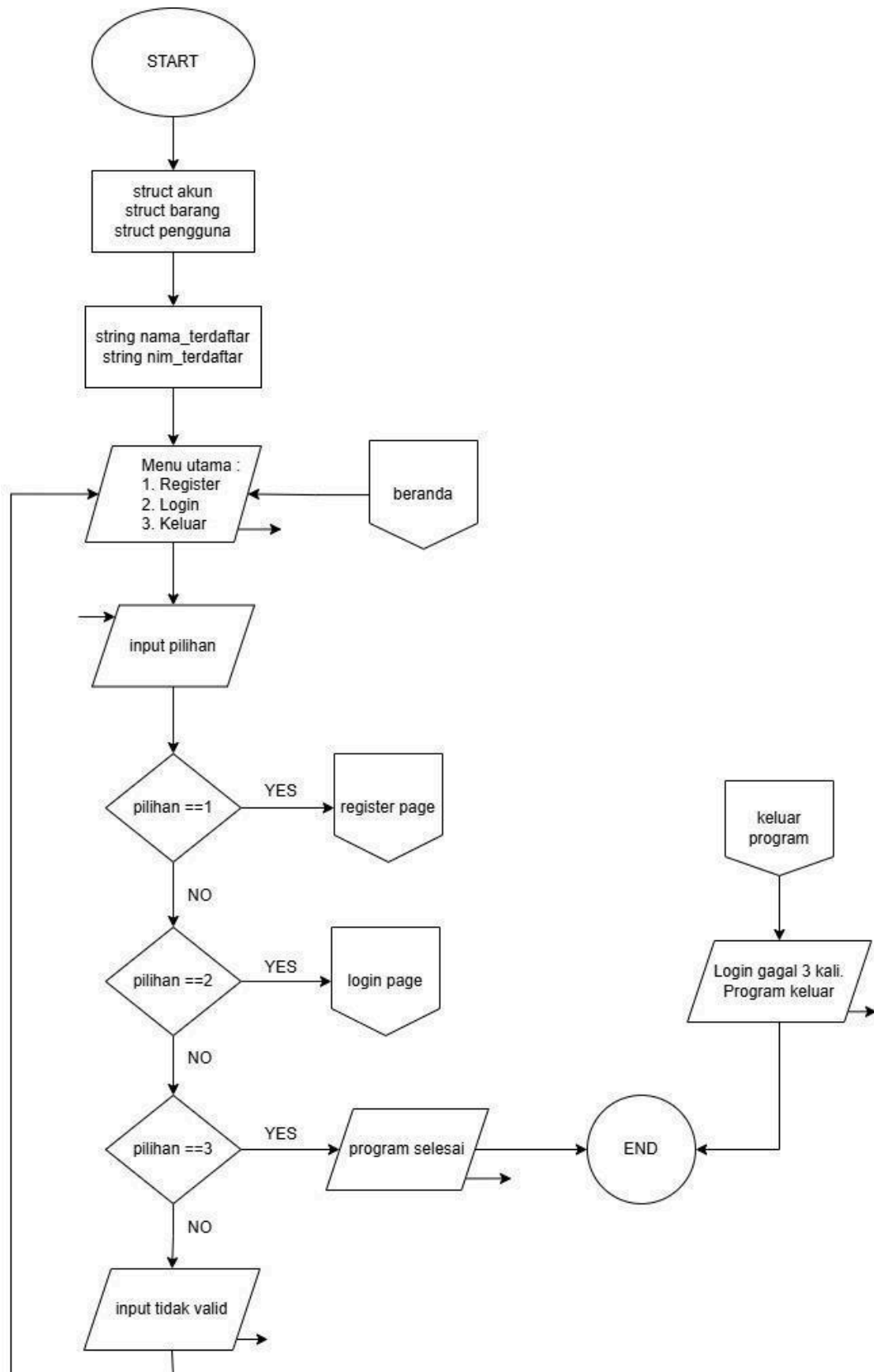


Disusun oleh:
WIDYA AYU ANGGRAINI (2409106011)
Kelas (A1 '24)

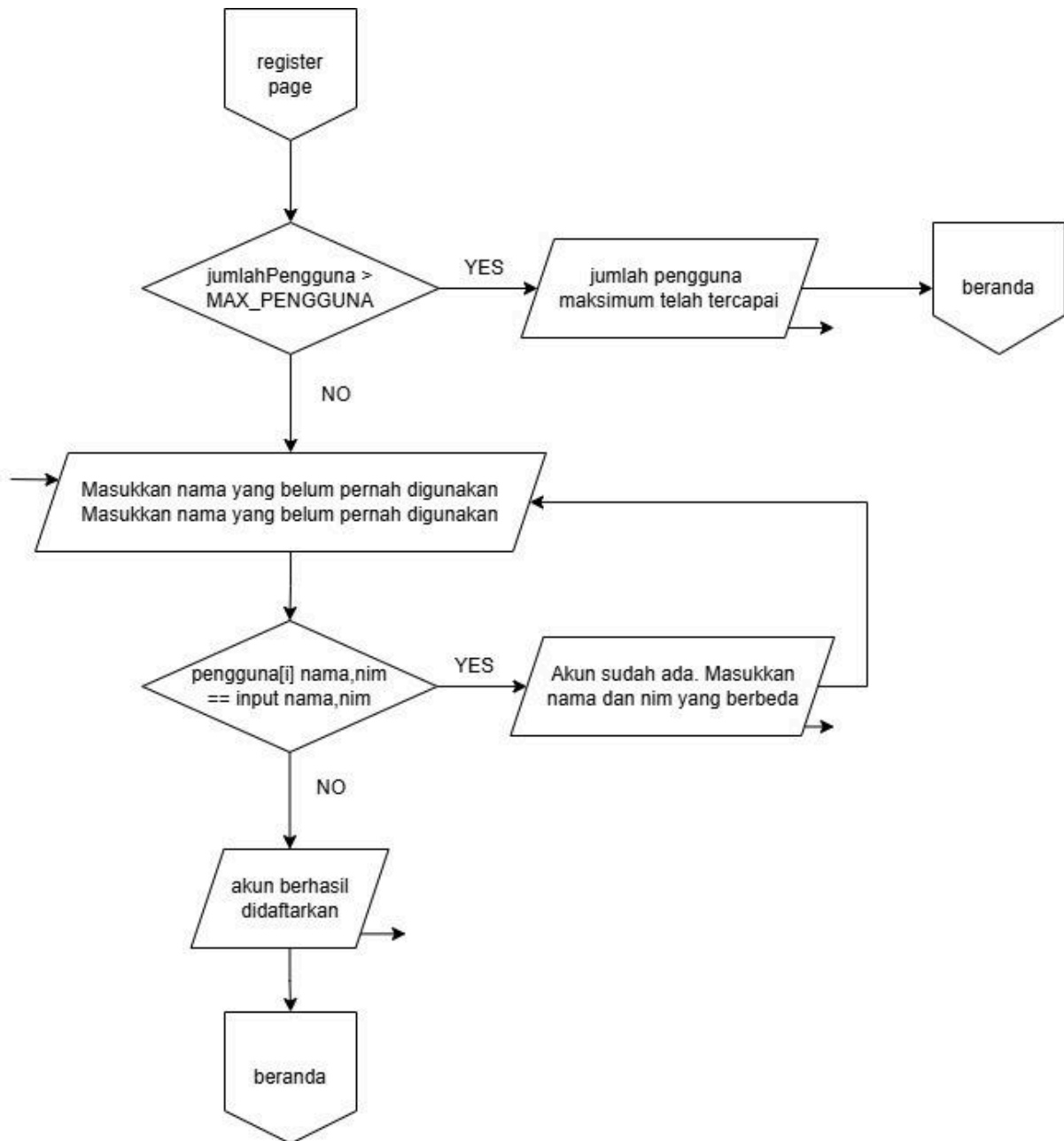
PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

1. Flowchart

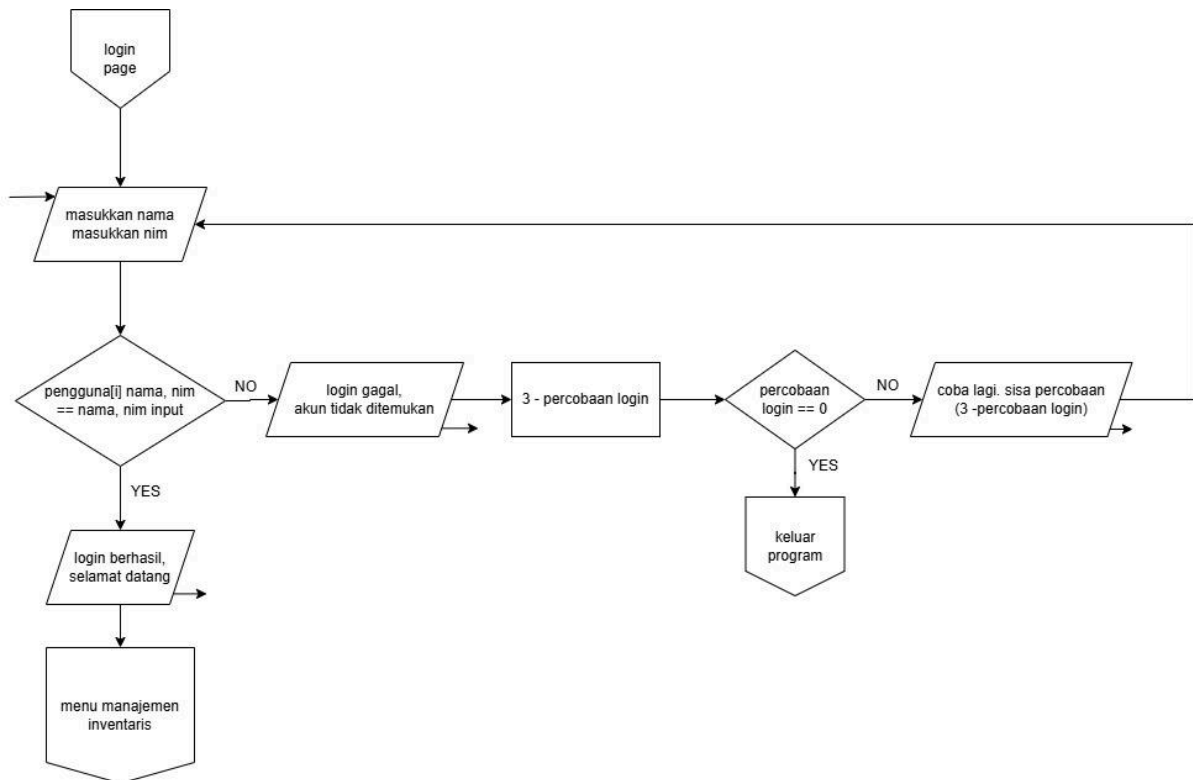
A. MENU UTAMA



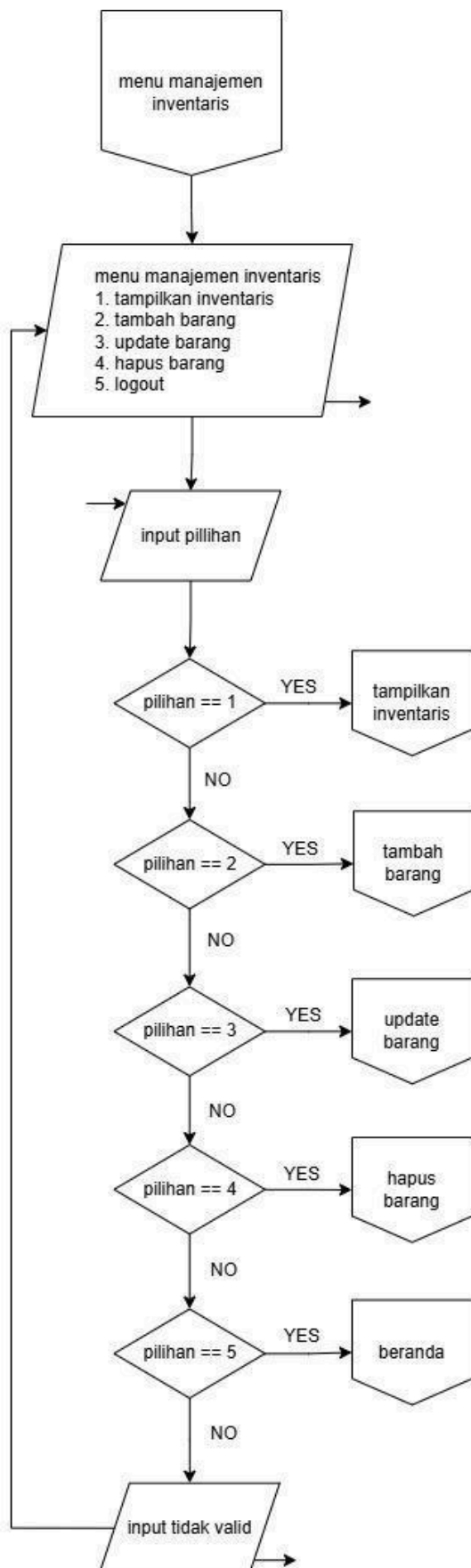
B. REGISTER



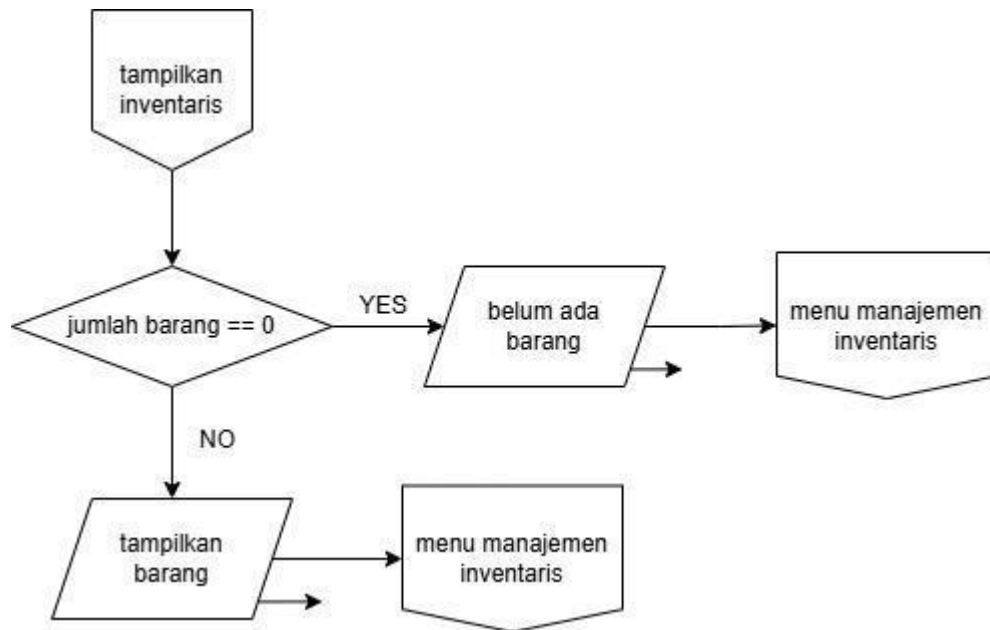
C. LOGIN



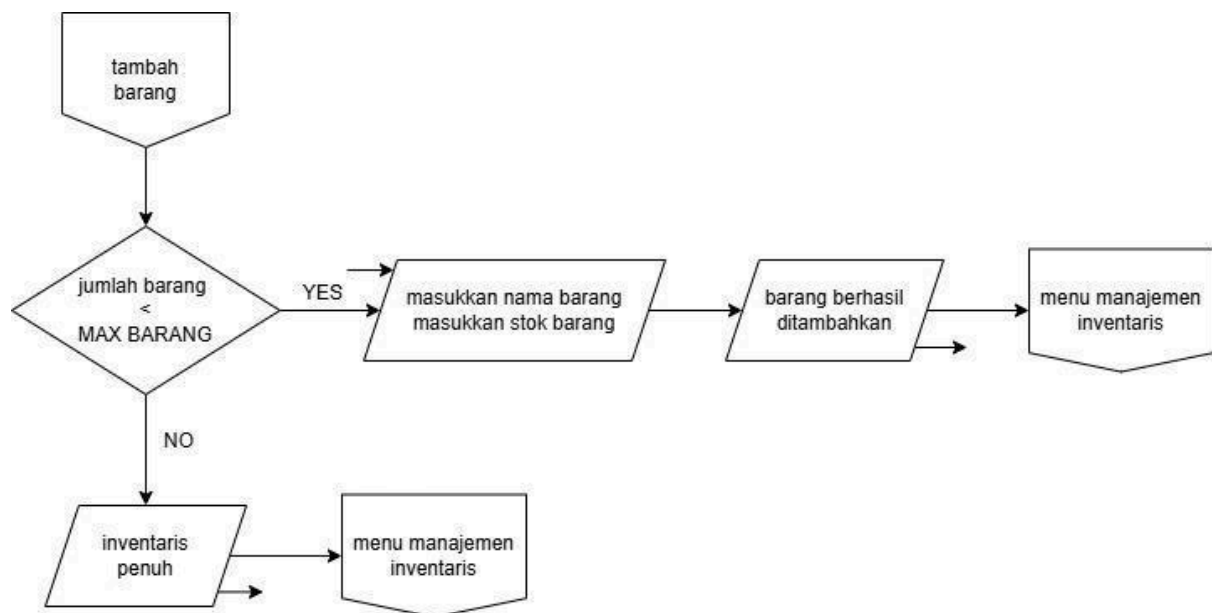
D. MENU MANAJEMEN INVENTARIS



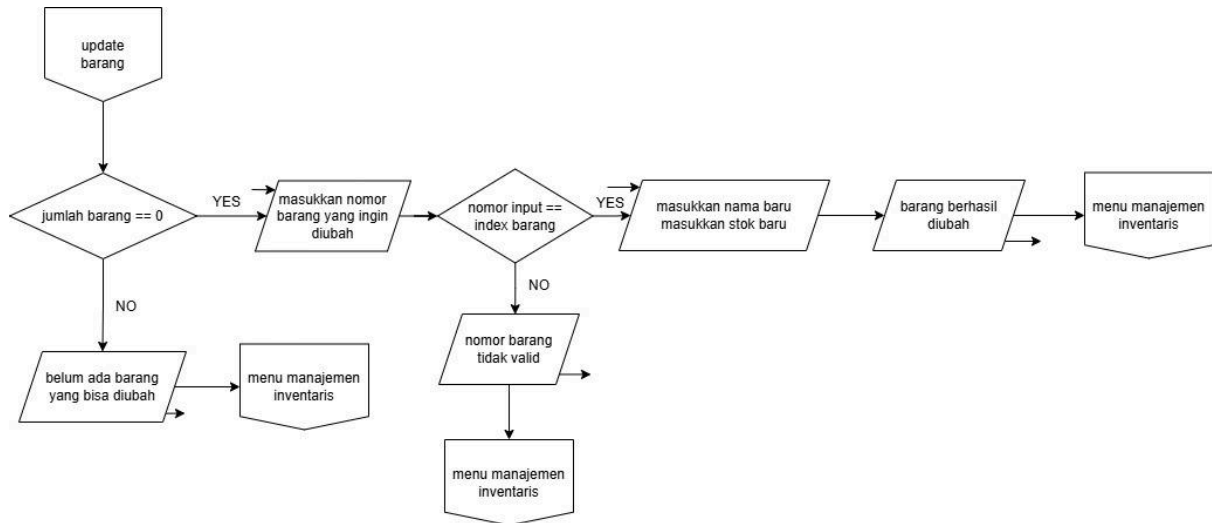
E. TAMPILKAN INVENTARIS



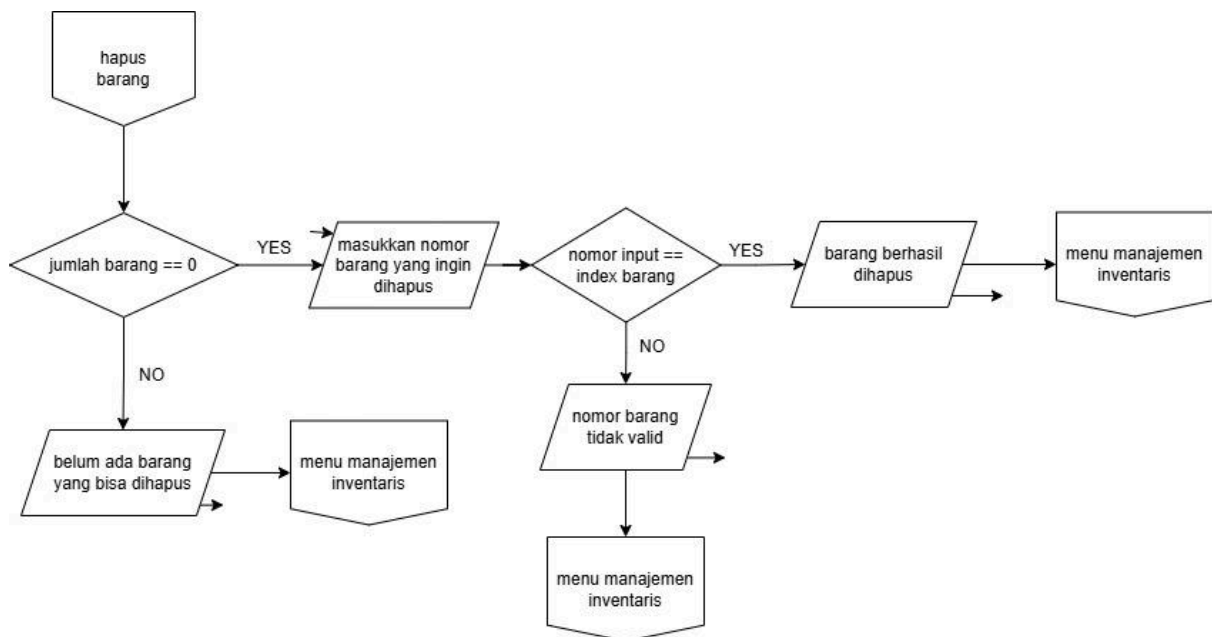
F. TAMBAH BARANG



G. UPDATE BARANG



H. HAPUS BARANG



2. Analisis Program

2.1 Deskripsi Singkat Program

Program ini adalah sistem manajemen inventaris sederhana yang ditulis dalam C++. Program memungkinkan pengguna untuk mendaftar, login, dan mengelola inventaris barang mereka. Program utama terdiri dari menu utama (registrasi, login, keluar) dan menu manajemen inventaris (tampilkan, tambah, update, hapus barang, logout). Setiap fitur dipisahkan menjadi subprogram (fungsi/prosedur) dengan parameter wajib, menggunakan pendekatan modular.

Deskripsi Fitur (Subprogram):

1. **Registrasi (void registrasi(Pengguna pengguna[], int &jumlahPengguna))**
 - Prosedur untuk mendaftarkan pengguna baru dengan nama dan NIM unik. Memeriksa duplikat akun dan membatasi hingga MAX_PENGGUNA (100).
2. **Login (int login(Pengguna pengguna[], int jumlahPengguna))**
 - Fungsi yang memverifikasi nama dan NIM pengguna. Mengembalikan indeks pengguna jika berhasil, -1 jika gagal, atau -2 jika gagal 3 kali (keluar program).
3. **Tampilkan Inventaris (void tampilInventaris(Pengguna &pengguna))**
 - Prosedur untuk menampilkan daftar barang dalam inventaris pengguna dalam format tabel, atau pesan jika inventaris kosong.
4. **Tambah Barang (void tambahBarang(Pengguna &pengguna))**
 - Prosedur untuk menambahkan barang baru (nama dan stok) ke inventaris pengguna, dengan batas MAX_BARANG (100).
5. **Update Barang (void updateBarang(Pengguna &pengguna))**
 - Prosedur untuk mengubah nama dan stok barang berdasarkan nomor yang dipilih pengguna.
6. **Hapus Barang (void hapusBarang(Pengguna &pengguna))**
 - Prosedur untuk menghapus barang dari inventaris berdasarkan nomor yang dipilih, dengan pergeseran data.
7. **Logout (void logout())**
 - Prosedur untuk keluar dari menu manajemen inventaris.
8. **Keluar Program (void keluarProgram())**
 - Prosedur untuk mengakhiri program dengan pesan perpisahan.

Struktur Program:

- **Struct:** Akun (nama, NIM), Barang (nama, stok), Pengguna (akun, inventaris, jumlahBarang).
- **Define:** MAX_PENGGUNA (100), MAX_BARANG (100).
- **Main:** Mengelola alur program dengan menu utama dan submenu setelah login.

3. Source Code

1. HEADER DAN STRUKTUR DATA

Bagian ini berisi include, define, dan definisi struct yang digunakan secara global.

SOURCE CODENYA :

```
#include <iostream>
#include <string>

#define MAX_PENGGUNA 100
#define MAX_BARANG 100

using namespace std;

struct Akun {
    string nama;
    string nim;
};

struct Barang {
    string nama;
    int stok;
};

struct Pengguna {
    Akun akun;
    Barang inventaris[MAX_BARANG];
    int jumlahBarang;
};
```

2. FUNGSI CLEAR SCREEN

Bagian ini berfungsi untuk menghilangkan riwayat di terminal agar program tersusun rapi

SOURCE CODENYA :

```
void clearScreen() {
    system("cls");
}
```

3. FUNGSI REGISTER

Bagian ini berfungsi untuk proses registrasi

SOURCE CODENYA :

```
void registrasi(Pengguna pengguna[], int &jumlahPengguna) {
    if (jumlahPengguna < MAX_PENGGUNA) {
        bool akun_sudah_ada = false;
        string nama_input, nim_input;
        do {
            cout << "Masukkan Nama Yang Belum Pernah Digunakan: ";
            getline(cin, nama_input);
            cout << "Masukkan NIM Yang Belum Pernah Digunakan: ";
            getline(cin, nim_input);

            akun_sudah_ada = false;
            for (int i = 0; i < jumlahPengguna; i++) {
                if (pengguna[i].akun.nama == nama_input &&
                pengguna[i].akun.nim == nim_input) {
                    akun_sudah_ada = true;
                    cout << "Akun sudah ada. Masukkan nama dan NIM yang
berbeda." << endl;
                    break;
                }
            }
        } while (akun_sudah_ada);

        pengguna[jumlahPengguna].akun.nama = nama_input;
        pengguna[jumlahPengguna].akun.nim = nim_input;
        pengguna[jumlahPengguna].jumlahBarang = 0;
        jumlahPengguna++;
        cout << "Akun berhasil didaftarkan." << endl;
    } else {
        cout << "Jumlah pengguna maksimum tercapai." << endl;
    }
    cout << "\nTekan Enter untuk melanjutkan..." << endl;
    cin.get();
    clearScreen();
}
```

4. FUNGSI LOGIN

Bagian ini berfungsi untuk proses login

SOURCE CODENYA :

```
int login(Pengguna pengguna[], int jumlahPengguna) {
    int percobaan_login = 0;
    bool login_berhasil = false;
    int index_pengguna = -1;
    string nama_input, nim_input;

    while (percobaan_login < 3 && !login_berhasil) {
        cout << "Masukkan Nama: ";
        getline(cin, nama_input);
        cout << "Masukkan NIM: ";
        getline(cin, nim_input);

        for (int i = 0; i < jumlahPengguna; i++) {
            if (pengguna[i].akun.nama == nama_input && pengguna[i].akun.nim
== nim_input) {
                login_berhasil = true;
                index_pengguna = i;
                break;
            }
        }

        if (!login_berhasil) {
            percobaan_login++;
            cout << "Login Gagal! Akun tidak ditemukan." << endl;
            if (percobaan_login < 3) {
                cout << "Coba lagi. Sisa percobaan: " << 3 - percobaan_login
<< endl;
            } else {
                cout << "Login gagal 3 kali. Program keluar." << endl;
                return -2; // Kode khusus untuk keluar program
            }
        }
    }

    if (login_berhasil) {
        cout << "Login Berhasil! Selamat datang, " <<
pengguna[index_pengguna].akun.nama << "!" << endl;
    }
    cout << "\nTekan Enter untuk melanjutkan..." << endl;
    cin.get();
    clearScreen();
    return index_pengguna;
}
```

5. FUNGSI TAMPILKAN INVENTARIS

Bagian ini berfungsi untuk menampilkan inventaris

SOURCE CODENYA :

```
void tampilInventaris(Pengguna &pengguna) {
    if (pengguna.jumlahBarang == 0) {
        cout << "Belum ada barang dalam inventaris." << endl;
    } else {
        cout << "No. | Nama Barang          | Stok" << endl;
        cout << "-----" << endl;
        for (int i = 0; i < pengguna.jumlahBarang; i++) {
            cout << i + 1 << ". | " << pengguna.inventaris[i].nama;
            for (int j = pengguna.inventaris[i].nama.length(); j < 22; j++)
            {
                cout << " ";
            }
            cout << "| " << pengguna.inventaris[i].stok << endl;
        }
    }
    cout << "\nTekan Enter untuk melanjutkan..." << endl;
    cin.get();
    clearScreen();
}
```

6. FUNGSI TAMBAH BARANG

Bagian ini berfungsi untuk menambah barang

SOURCE CODENYA :

```
void tambahBarang(Pengguna &pengguna) {
    if (pengguna.jumlahBarang < MAX_BARANG) {
        cout << "Masukkan nama barang: ";
        getline(cin, pengguna.inventaris[pengguna.jumlahBarang].nama);
        cout << "Masukkan stok barang: ";
        cin >> pengguna.inventaris[pengguna.jumlahBarang].stok;
        while (cin.get() != '\n');
        pengguna.jumlahBarang++;
        cout << "Barang berhasil ditambahkan." << endl;
    } else {
        cout << "Inventaris penuh." << endl;
    }
    cout << "\nTekan Enter untuk melanjutkan..." << endl;
    cin.get();
    clearScreen();
}
```

7. FUNGSI UPDATE BARANG

Bagian ini berfungsi untuk mengubah barang

SOURCE CODENYA :

```
void updateBarang(Pengguna &pengguna) {
    if (pengguna.jumlahBarang == 0) {
        cout << "Belum ada barang untuk diubah." << endl;
    } else {
        cout << "No. | Nama Barang | Stok" << endl;
        cout << "-----" << endl;
        for (int i = 0; i < pengguna.jumlahBarang; i++) {
            cout << i + 1 << ". | " << pengguna.inventaris[i].nama;
            for (int j = pengguna.inventaris[i].nama.length(); j < 22;
j++) {
                cout << " ";
            }
            cout << "| " << pengguna.inventaris[i].stok << endl;
        }
        int nomor_barang;
        cout << "\nMasukkan nomor barang yang ingin diubah: ";
        if (!(cin >> nomor_barang) || nomor_barang < 1 || nomor_barang >
pengguna.jumlahBarang) {
            cout << "Nomor barang tidak valid." << endl;
            cin.clear();
            while (cin.get() != '\n');
        } else {
            while (cin.get() != '\n');
            cout << "Masukkan nama baru barang: ";
            getline(cin, pengguna.inventaris[nomor_barang - 1].nama);
            cout << "Masukkan stok baru barang: ";
            cin >> pengguna.inventaris[nomor_barang - 1].stok;
            while (cin.get() != '\n');
            cout << "Barang berhasil diupdate." << endl;
        }
    }
    cout << "\nTekan Enter untuk melanjutkan..." << endl;
    cin.get();
    clearScreen();
}
```

8. FUNGSI HAPUS BARANG

Bagian ini berfungsi untuk menghapus barang

SOURCE CODENYA :

```

void hapusBarang(Pengguna &pengguna) {
    if (pengguna.jumlahBarang == 0) {
        cout << "Belum ada barang untuk dihapus." << endl;
    } else {
        cout << "No. | Nama Barang          | Stok" << endl;
        cout << "-----" << endl;
        for (int i = 0; i < pengguna.jumlahBarang; i++) {
            cout << i + 1 << ". | " << pengguna.inventaris[i].nama;
            for (int j = pengguna.inventaris[i].nama.length(); j < 22; j++)
            {
                cout << " ";
            }
            cout << "| " << pengguna.inventaris[i].stok << endl;
        }
        int nomor_barang;
        cout << "\nMasukkan nomor barang yang ingin dihapus: ";
        if (!(cin >> nomor_barang) || nomor_barang < 1 || nomor_barang >
pengguna.jumlahBarang) {
            cout << "Nomor barang tidak valid." << endl;
            cin.clear();
            while (cin.get() != '\n');
        } else {
            for (int i = nomor_barang - 1; i < pengguna.jumlahBarang - 1;
i++) {
                pengguna.inventaris[i] = pengguna.inventaris[i + 1];
            }
            pengguna.jumlahBarang--;
            cout << "Barang berhasil dihapus." << endl;
            while (cin.get() != '\n');
        }
    }
    cout << "\nTekan Enter untuk melanjutkan..." << endl;
    cin.get();
    clearScreen();
}

```

9. FUNGSI LOGOUT

Bagian ini berfungsi untuk logout dari program

SOURCE CODENYA :

```

void logout() {
    cout << "Logout berhasil." << endl;
    cout << "\nTekan Enter untuk melanjutkan..." << endl;
    cin.get();
    clearScreen();
}

```

10. FUNGSI KELUAR PROGRAM

Bagian ini berfungsi untuk keluar program

SOURCE CODENYA :

```
void keluarProgram() {
    cout << "Terima kasih telah menggunakan program ini." << endl;
    cout << "\nTekan Enter untuk melanjutkan..." << endl;
    cin.get();
    clearScreen();
}
```

11. FUNGSI MAIN

Bagian ini berfungsi untuk menjalankan program utama

SOURCE CODENYA :

```
int main() {
    Pengguna pengguna[MAX_PENGGUNA];
    int jumlahPengguna = 0;
    int pilihan_menu;

    // Tambahkan akun default
    pengguna[jumlahPengguna].akun.nama = "WidyaAyuAnggraini";
    pengguna[jumlahPengguna].akun.nim = "2409106011";
    pengguna[jumlahPengguna].jumlahBarang = 0;
    jumlahPengguna++;

    do {
        cout << "Menu Utama" << endl;
        cout << "1. Register" << endl;
        cout << "2. Login" << endl;
        cout << "3. Keluar" << endl;
        cout << "Pilihan: ";

        if (!(cin >> pilihan_menu)) {
            cout << "Input tidak valid. Masukkan angka." << endl;
            cin.clear();
            while (cin.get() != '\n');
            cout << "\nTekan Enter untuk melanjutkan..." << endl;
            cin.get();
            clearScreen();
            continue;
        }
        while (cin.get() != '\n');
```

```

switch (pilihan_menu) {
    case 1:
        registrasi(pengguna, jumlahPengguna);
        break;
    case 2: {
        int index_pengguna = login(pengguna, jumlahPengguna);
        if (index_pengguna == -2) {
            return 0; // Keluar program jika Login gagal 3 kali
        }
        if (index_pengguna != -1) {
            int pilihan_login;
            do {
                cout << "\nMenu Manajemen Inventaris" << endl;
                cout << "1. Tampilkan Inventaris" << endl;
                cout << "2. Tambah Barang" << endl;
                cout << "3. Update Barang" << endl;
                cout << "4. Hapus Barang" << endl;
                cout << "5. Logout" << endl;
                cout << "Pilihan: ";

                if (!(cin >> pilihan_login)) {
                    cout << "Input tidak valid. Masukkan angka." <<
endl;

                    cin.clear();
                    while (cin.get() != '\n');
                    cout << "\nTekan Enter untuk melanjutkan..." <<
endl;

                    cin.get();
                    clearScreen();
                    continue;
                }
                while (cin.get() != '\n');

                switch (pilihan_login) {
                    case 1:
tampilInventaris(pengguna[index_pengguna]); break;
                    case 2: tambahBarang(pengguna[index_pengguna]);
break;
                    case 3: updateBarang(pengguna[index_pengguna]);
break;
                    case 4: hapusBarang(pengguna[index_pengguna]);
break;
                    case 5: logout(); break;
                    default:
                        cout << "Pilihan tidak valid." << endl;
                        cout << "\nTekan Enter untuk melanjutkan..."
<< endl;

                        cin.get();

```



```

        clearScreen();
    }
    } while (pilihan_login != 5);
}
break;
}
case 3:
    keluarProgram();
    return 0;
default:
    cout << "Pilihan tidak valid." << endl;
    cout << "\nTekan Enter untuk melanjutkan..." << endl;
    cin.get();
    clearScreen();
}
} while (pilihan_menu != 3);

return 0;
}

```

4. Output

Hasil Output

```

PS C:\Users\silly\Downloads\praktikum-apl\post-test> cd "c:\Users\silly\Downloads\praktikum-apl\post-
i-PT-4.cpp -o 2409106011-WidyaAyuAnggraini-PT-4 } ; if ($?) { .\2409106011-WidyaAyuAnggraini-PT-4 }
Menu Utama
1. Register
2. Login
3. Keluar
Pilihan: 1
Masukkan Nama Yang Belum Pernah Digunakan: widya
Masukkan NIM Yang Belum Pernah Digunakan: 11
Akun berhasil didaftarkan.

Tekan Enter untuk melanjutkan...

```

Gambar 1 Registrasi

```

Menu Utama
1. Register
2. Login
3. Keluar
Pilihan: 2
Masukkan Nama: widya
Masukkan NIM: 11
Login Berhasil! Selamat datang, widya!

Tekan Enter untuk melanjutkan...

```

Gambar 2 Login

```

Menu Manajemen Inventaris
1. Tampilkan Inventaris
2. Tambah Barang
3. Update Barang
4. Hapus Barang
5. Logout
Pilihan: 2
Masukkan nama barang: kayu balok
Masukkan stok barang: 12
Barang berhasil ditambahkan.

Tekan Enter untuk melanjutkan...

```

Gambar 3 Tambah Barang

```

Menu Manajemen Inventaris
1. Tampilkan Inventaris
2. Tambah Barang
3. Update Barang
4. Hapus Barang
5. Logout
Pilihan: 1

```

| No. | Nama Barang | Stok |
|-----|-------------|------|
| 1. | kayu balok | 12 |

```

Tekan Enter untuk melanjutkan...

```

Gambar 4 Tampilkan Inventaris

```

Menu Manajemen Inventaris
1. Tampilkan Inventaris
2. Tambah Barang
3. Update Barang
4. Hapus Barang
5. Logout
Pilihan: 3
No. | Nama Barang          | Stok
-----
1.  | kayu balok             | 12

Masukkan nomor barang yang ingin diubah: 1
Masukkan nama baru barang: kayu bulat
Masukkan stok baru barang: 10
Barang berhasil diupdate.

Tekan Enter untuk melanjutkan...

```

Gambar 5 Update Barang

```

Menu Manajemen Inventaris
1. Tampilkan Inventaris
2. Tambah Barang
3. Update Barang
4. Hapus Barang
5. Logout
Pilihan: 4
No. | Nama Barang          | Stok
-----
1.  | kayu bulat           | 10

Masukkan nomor barang yang ingin dihapus: 1
Barang berhasil dihapus.

Tekan Enter untuk melanjutkan...

```

Gambar 6 Hapus Barang

```

Menu Manajemen Inventaris
1. Tampilkan Inventaris
2. Tambah Barang
3. Update Barang
4. Hapus Barang
5. Logout
Pilihan: 1
Belum ada barang dalam inventaris.

Tekan Enter untuk melanjutkan...

```

Gambar 7 Hasil Inventaris Akhir

5. GIT

5.1 Git Init (Menginisiasi Repository Git)

```
PS C:\Users\silly\Downloads\praktikum-apl\post-test> git init
Reinitialized existing Git repository in C:/Users/silly/Downloads/praktikum-apl/post-test/.git/
PS C:\Users\silly\Downloads\praktikum-apl\post-test> █
```

Gambar 5.1 Git Init

5.2 Git Add (Menambah File yang Ingin Dicommit)

```
PS C:\Users\silly\Downloads\praktikum-apl\post-test> git add .
PS C:\Users\silly\Downloads\praktikum-apl\post-test> █
```

Gambar 5.2 Git Add

5.3 Git Commit (CheckPoint)

```
PS C:\Users\silly\Downloads\praktikum-apl\post-test> git commit -m "kelar posttest 4"
[main ebe1811] kelar posttest 4
13 files changed, 1 insertion(+), 576 deletions(-)
delete mode 100644 post-test-3/2409106011-WidyaAyuAnggraini-PT-3.cpp
delete mode 100644 post-test-3/2409106011-WidyaAyuAnggraini-PT-3.exe
delete mode 100644 post-test-3/2409106011-WidyaAyuAnggraini-PT-3.pdf
delete mode 100644 post-test-4/2409106011-WidyaAyuAnggraini-PT-4.cpp
delete mode 100644 post-test-4/2409106011-WidyaAyuAnggraini-PT-4.exe
rename {post-test-1 => post-test/post-test-1}/2409106011-WidyaAyuAnggraini-PT-1.cpp (100%)
rename {post-test-1 => post-test/post-test-1}/2409106011-WidyaAyuAnggraini-PT-1.exe (100%)
rename {post-test-1 => post-test/post-test-1}/2409106011-WidyaAyuAnggraini-PT-1.pdf (100%)
rename {post-test-2 => post-test/post-test-2}/2409106011-WidyaAyuAnggraini-PT-2.cpp (100%)
rename {post-test-2 => post-test/post-test-2}/2409106011-WidyaAyuAnggraini-PT-2.exe (100%)
rename {post-test-2 => post-test/post-test-2}/2409106011-WidyaAyuAnggraini-PT-2.pdf (100%)
create mode 160000 post-test/post-test-4
PS C:\Users\silly\Downloads\praktikum-apl\post-test> █
```

Gambar 5.3 Git Commit

5.4 Git Push (Mengupdate Semua yang Ada di Repository Lokal)

```
PS C:\Users\silly\Downloads\praktikum-apl\post-test> git push -u origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 392 bytes | 392.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/WidyaRanyyy/praktikum-apl
7bae975..ebe1811 main -> main
branch 'main' set up to track 'origin/main'.
PS C:\Users\silly\Downloads\praktikum-apl\post-test> █
```

Gambar 5.4 Git Push