

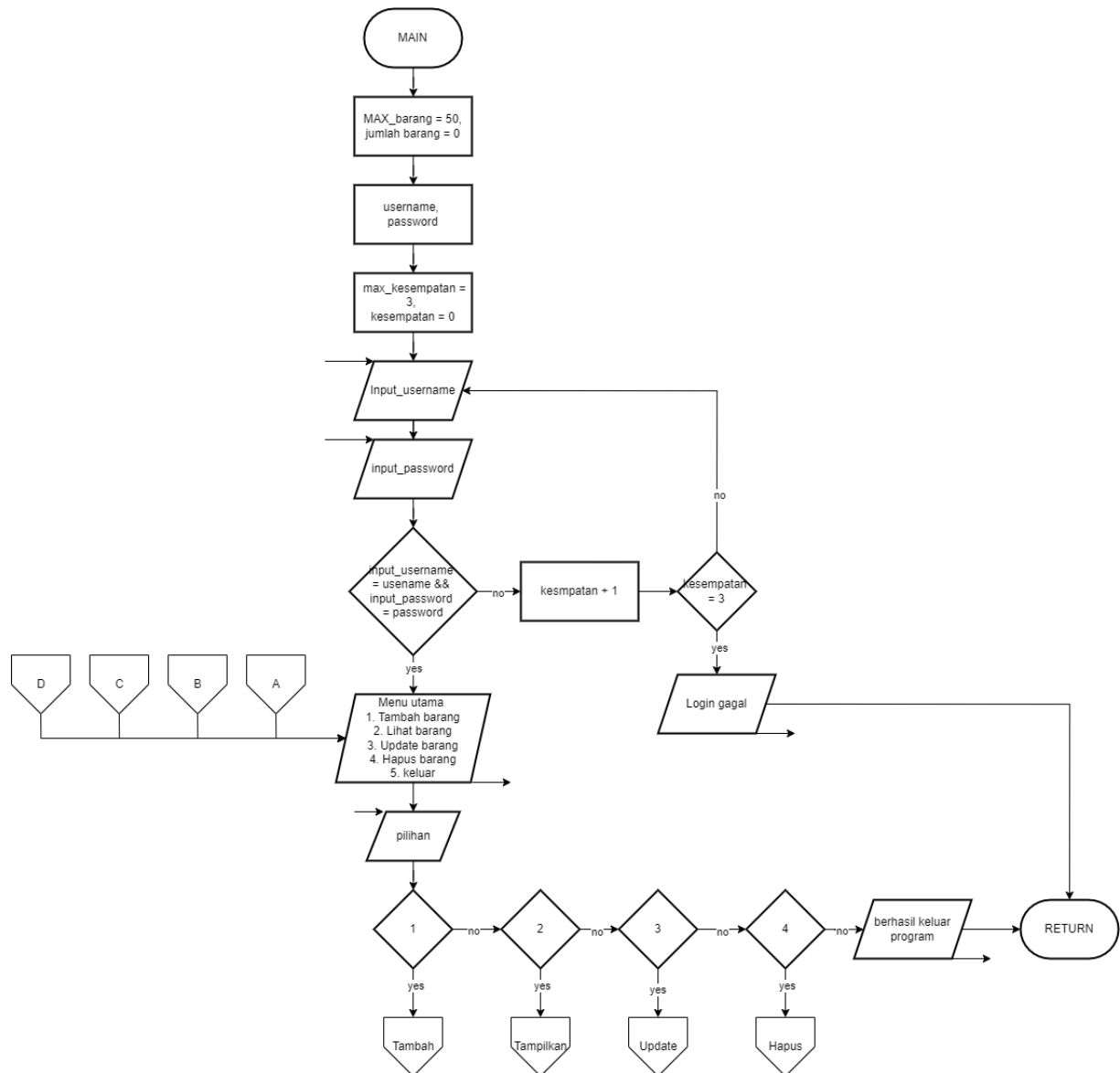
LAPORAN PRAKTIKUM
POSTTEST 2
ALGORITMA PEMROGRAMAN LANJUT



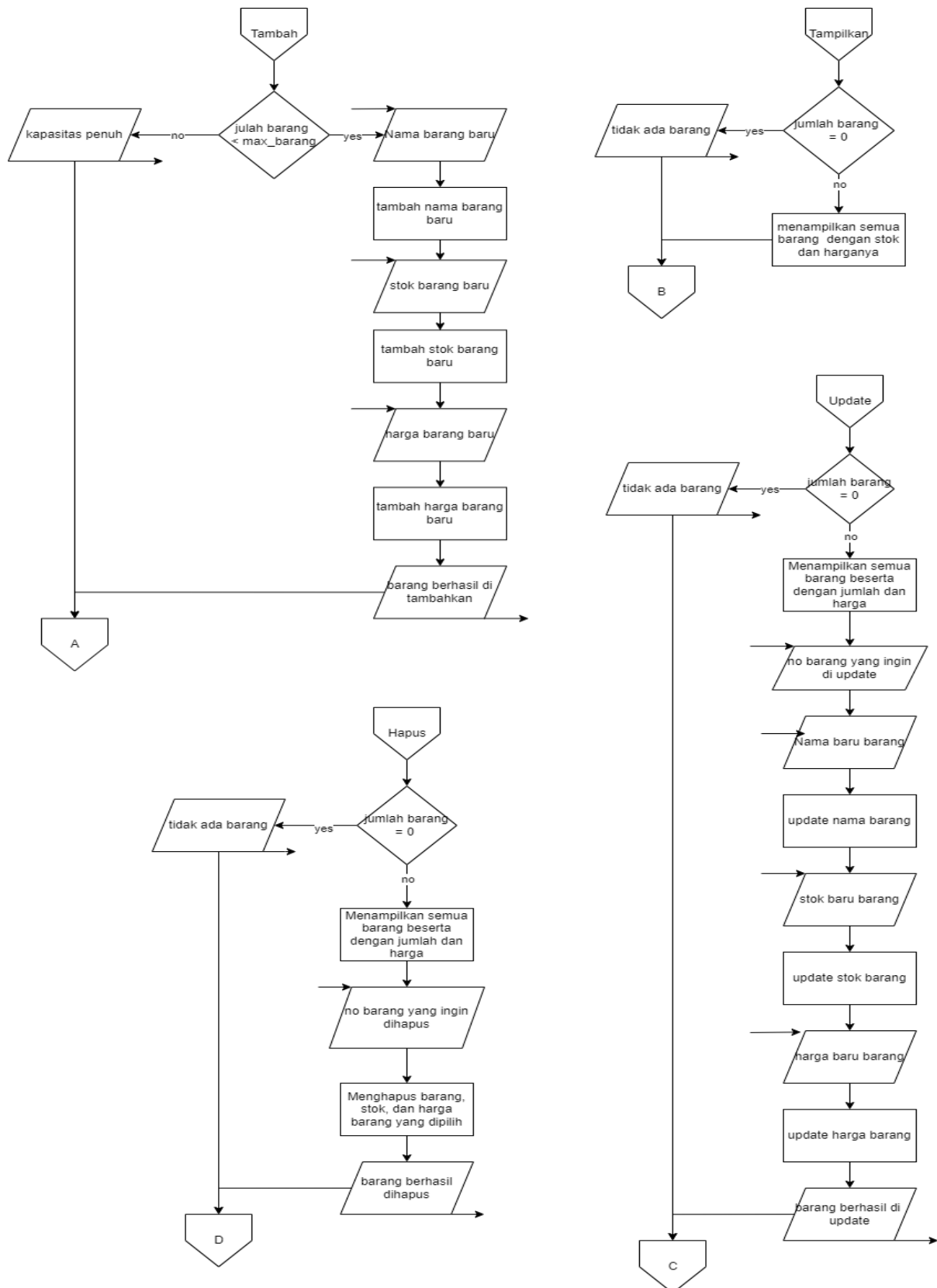
Disusun oleh:
Muhammad Ridwanansyah Rahman (2409106073)
Kelas (B2 '24)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

1. Flowchart



gambar 1.1 Flowchart menu login & CRUD



Gambar 1.2 Flowchart proses CURD

2. Analisis Program

2.1 Deskripsi Singkat Program

Program ini dirancang untuk membantu pengguna dalam mengelola stok barang sembako. Tujuan utama program ini adalah untuk memudahkan pengguna dalam melakukan operasi pada barang sembako, seperti menambahkan barang baru, melihat daftar barang yang tersedia, mengupdate informasi barang, dan menghapus barang yang tidak diperlukan lagi. Program ini juga memiliki fitur login yang memungkinkan pengguna untuk mengakses aplikasi dengan menggunakan username dan password yang telah ditentukan. Dengan demikian, program ini dapat membantu pengguna dalam mengelola stok barang sembako dengan lebih efektif dan efisien.

2.2 Penjelasan Alur & Algoritma

Pertama-tama program akan mendeklarasikan username yaitu “Muhammad Ridwanansyah Rahman”, password yaitu “2409106073”, max_kesempatan sebanyak 3 dan kesempatan awal 0. program meminta pengguna untuk melakukan input berupa username dan password. Setelah menginput program kemudian akan mengecek apakah yang diinput pengguna sesuai dengan username dan password yang telah ditentukan. Jika tidak maka program akan memberi output berupa “login gagal, sisa percobaan ... “ dan menyuruh pengguna untuk menginput ulang, pengguna diberi kesempatan sebanyak 3 kali percobaan, jika pengguna gagal sebanyak 3 kali maka program mengeluarkan output “Login gagal, anda telah login sebanyak 3 kali, program berhenti karena batas percobaan telah tercapai” dan otomatis keluar dari program. Sedangkan apabila yang diinput pengguna sesuai, maka program berikutnya akan mengarahkan pengguna ke menu utama. Pada menu utama program akan menampilkan menu utama yang berisi beberapa pilihan, yaitu menambahkan barang baru, melihat daftar barang yang tersedia, mengupdate informasi barang, menghapus barang yang tidak diperlukan lagi, serta keluar dari program. Pengguna dapat memilih salah satu pilihan menu yang tersedia. Apabila pengguna memilih menu pertama, program akan meminta pengguna untuk memasukkan nama barang, jumlah barang, dan harga barang. Setelah pengguna memasukkan data, program akan menambahkan data tersebut ke dalam array yang digunakan untuk menyimpan data barang. Apabila pengguna memilih menu kedua. Program akan menampilkan daftar barang yang tersedia, termasuk nama barang, jumlah barang, dan harga barang. Apabila pengguna memilih menu ketiga. Program akan meminta pengguna untuk memilih barang yang ingin diupdate,

kemudian meminta pengguna untuk memasukkan data baru untuk barang tersebut berupa nama baru, stok baru dan harga baru. Kemudian program akan mengupdate data barang yang ada sebelumnya. Apabila pengguna memilih menu keempat. Program akan meminta pengguna untuk memilih barang yang ingin dihapus, kemudian program akan menghapus data barang tersebut dari array berdasarkan pilihan dari pengguna. Dan yang terakhir apabila pengguna memilih menu kelima maka program akan menutup dan mengakhiri proses atau mengeluarkan pengguna dari program.

Blok Kode dan Fungsi

Berikut adalah penjelasan tentang beberapa blok kode dan fungsi yang digunakan dalam program:

- **Fungsi getline:** Fungsi ini digunakan untuk membaca input dari pengguna dan menyimpannya dalam variabel.
- **Fungsi cin:** Fungsi ini digunakan untuk membaca input dari pengguna dan menyimpannya dalam variabel.
- **Fungsi cout:** Fungsi ini digunakan untuk menampilkan output ke layar.
- **Fungsi system:** Fungsi ini digunakan untuk menjalankan perintah sistem, seperti membersihkan layar.
- **Array:** Array digunakan untuk menyimpan data barang, termasuk nama barang, jumlah barang, dan harga barang.
- **Looping:** Looping digunakan untuk mengulangi proses, seperti menampilkan menu utama dan meminta pengguna untuk memilih pilihan menu.

3. Source Code

A. Fitur Login

Fitur ini digunakan pengguna untuk masuk program dengan cara menginput username dan password yang sesuai dan nantinya pengguna akan diarahkan kemenu utama. Pengguna diberikan 3 kesempatan untuk memasukkan username dan password dengan benar, jika lewat dari 3 maka pengguna otomatis keluar dari program.

Source Code:

```
const string username = "Muhammad Ridwanansyah Rahman";
const string password = "2409106073";
const int max_kesempatan = 3;
int kesempatan = 0;

while (kesempatan < max_kesempatan) {
    string input_username, input_password;
    system("cls");
    cout << "\n===== " << endl;
    cout << "                Login Program                " << endl;
    cout << "===== " << endl;
    cout << "Username: ";
    getline(cin, input_username);
    cout << "Password: ";
    getline(cin, input_password);

    if (input_username == username && input_password == password) {
        cout << "Login berhasil!\n" << endl;
        break;
    } else {
        kesempatan++;
        cout << "Login gagal. Sisa percobaan: " << max_kesempatan -
kesempatan << "\n" << endl;
        cout << "Klik Enter...";
        cin.ignore();
        cin.get();
    }
}

if (kesempatan == max_kesempatan) {
    system("cls");
    cout << "\n===== " << endl;
    cout << "                Login Gagal                " << endl;
```

```

        cout << "=====" << endl;
        cout << "Anda telah mencoba login " << max_kesempatan << " kali." <<
endl;
        cout << "Program berhenti karena batas" << endl;
        cout << "percobaan telah tercapai " << endl;
        cout << "=====" << endl;
        return 0;
    }

```

B. Menu CURD

pengguna akan diberikan 4 opsi menu CURD (tambah barang, liat barang, update barang dan hapus barang) dan opsi keluar untuk keluar dari program.

Source Code:

```

while (running) {
    system("cls");
    cout << "\n=====" << endl;
    cout << "          MENU MANAJEMEN BARANG SEMBAKO          " << endl;
    cout << "=====" << endl;
    cout << "1. Tambah Barang" << endl;
    cout << "2. Lihat Barang" << endl;
    cout << "3. Update Barang" << endl;
    cout << "4. Hapus Barang" << endl;
    cout << "5. Keluar" << endl;
    cout << "=====" << endl;
    cout << "Pilih menu: ";
    int pilihan;
    cin >> pilihan;
}

```

C. Menambah Barang

kode dalam program ini digunakan untuk menambah barang sembako baru beserta dengan stok dan harganya.

Source Code:

```

case 1: // Nambah Barang
if (jumlahBarang < MAX_BARANG) {
    cout << "Masukkan Nama Barang: ";
    cin >> barang[jumlahBarang][0];

    int jumlah, harga;
    bool valid = false;

    while (!valid) {
        cout << "Masukkan Stok Barang: ";
        if (cin >> jumlah) {
            if (jumlah > 0) {
                valid = true;
                barang[jumlahBarang][1] = to_string(jumlah);
            } else {
                cout << "Stok barang tidak boleh negatif!!\n";
                cin.clear();
                cin.ignore(10000, '\n');
            }
        } else {
            cout << "Input harus berupa angka!\n";
            cin.clear();
            cin.ignore(10000, '\n');
        }
    }

    valid = false;

    while (!valid) {
        cout << "Masukkan Harga Barang: ";
        if (cin >> harga) {
            if (harga > 0) {
                valid = true;
                barang[jumlahBarang][2] = to_string(harga);
            } else {
                cout << "Harga barang tidak boleh negatif!!\n";
                cin.clear();
                cin.ignore(10000, '\n');
            }
        } else {
            cout << "Input harus berupa angka!\n";
            cin.clear();
            cin.ignore(10000, '\n');
        }
    }
}

```


D. Menampilkan Barang

Kode ini digunakan untuk menampilkan daftar barang sembako beserta stok dan harganya dalam format tabel dengan menggunakan perulangan for

Source Code:

```
system("cls");
cout << "\n===== " << endl;
cout << "          DAFTAR BARANG SEMBAKO          " << endl;
cout << "===== " << endl;
cout << setw(5) << "No" << setw(20) << "Nama Barang" << setw(10) <<
"Stok" << setw(10) << "Harga" << "\n";
for (int i = 0; i < jumlahBarang; i++) {
    cout << setw(5) << i + 1 << setw(20) << barang[i][0] << setw(10) <<
barang[i][1] << setw(10) << barang[i][2] << "\n";
}
cout << "===== " << endl;
cout << "Klik Enter untuk kembali ke menu...";
cin.ignore();
cin.get();
```

E. Update Barang

Kode ini memungkinkan pengguna untuk melakukan update/mengganti data nama, stok serta harga dari barang sembako yang dipilih dan sudah ada sebelumnya.

Source Code:

```
cout << "Masukkan nomor barang yang ingin diupdate: ";
if (cin >> index) {
    if (index > 0 && index <= jumlahBarang) {
        valid = true;
    } else {
        cout << "Nomor barang tidak valid. Silakan pilih nomor yang
tersedia.\n";
        cout << "Klik Enter untuk coba lagi...";
        cin.ignore();
        cin.get();
    }
} else {
    cout << "Input harus berupa angka. Silakan pilih nomor yang
```

```

tersedia.\n";
    cout << "Klik Enter untuk coba lagi...";
    cin.clear();
    cin.ignore(10000, '\n');
    cin.get();
}
}

cout << "Masukkan Nama Barang Baru: ";
cin >> barang[index - 1][0];

int jumlahBaru;
valid = false;
while (!valid) {
    cout << "Masukkan Jumlah Barang Baru: ";
    if (cin >> jumlahBaru) {
        if (jumlahBaru > 0) {
            valid = true;
            barang[index - 1][1] = to_string(jumlahBaru);
        } else {
            cout << "Jumlah barang tidak boleh negatif!!\n";
            cin.clear();
            cin.ignore(10000, '\n');
        }
    } else {
        cout << "Input harus berupa angka!\n";
        cin.clear();
        cin.ignore(10000, '\n');
    }
}

int hargaBaru;
valid = false;
while (!valid) {
    cout << "Masukkan Harga Barang Baru: ";
    if (cin >> hargaBaru) {
        if (hargaBaru > 0) {
            valid = true;
            barang[index - 1][2] = to_string(hargaBaru);
        } else {
            cout << "Harga barang tidak boleh negatif!!\n";
            cin.clear();
            cin.ignore(10000, '\n');
        }
    } else {
        cout << "Input harus berupa angka!\n";
        cin.clear();
        cin.ignore(10000, '\n');
    }
}

```

```
}
```

F. Hapus Barang

Kode ini memungkinkan pengguna untuk melakukan delete/menghapus data sembako yang dipilih pengguna, berupa nama, stok, serta harga barang sembako yang ada sebelumnya.

Source Code:

```
    cout << "Masukkan nomor barang yang ingin dihapus: ";
    if (cin >> index) {
        if (index > 0 && index <= jumlahBarang) {
            valid = true;
        } else {
            cout << "Nomor barang tidak valid. Silakan pilih nomor yang
tersedia.\n";

            cout << "Klik Enter untuk coba lagi...";
            cin.clear();
            cin.ignore(10000, '\n');
            cin.get();
        }
    } else {
        cout << "Input harus berupa angka. Silakan pilih nomor yang
tersedia.\n";
        cout << "Klik Enter untuk coba lagi...";
        cin.clear();
        cin.ignore(10000, '\n');
        cin.get();
    }
}

for (int i = index - 1; i < jumlahBarang - 1; i++) {
    barang[i][0] = barang[i + 1][0];
    barang[i][1] = barang[i + 1][1];
    barang[i][2] = barang[i + 1][2];
}
jumlahBarang--;
```

4. Uji Coba dan Hasil Output

4.1 Uji Coba

a. Menu login

Pengguna menginput username “Muhammad Ridwanansyah Rahman” dan password “2409106073” outputnya Login Berhasil. Pengguna menginput diluar username “Muhammad Ridwanansyah Rahman” dan password “2409106073” outputnya Login Gagal.

b. Menu Utama

Pengguna menginput opsi pilihan yang tersedia outputnya masuk ke menu sesuai dengan opsi yang dipilih. Pengguna menginput angka diluar opsi atau menginput huruf outputnya “Input tidak valid”

c. Menu CURD

Pengguna memilih opsi CURD yang tersedia. Pada menu tambah pengguna menginput nama barang baru, stok baru serta harga baru, maka program akan menampilkan barang berhasil ditambahkan beserta detail barangnya. Apabila pengguna ingin menambahkan barang baru tetapi telah melewati kapasitas maka akan mengeluarkan output “kapasitas barang penuh”. Pengguna memilih menu tampilkan maka program akan menampilkan barang beserta stok dan harganya dalam format tabel. Pada menu update pengguna menginput nomer barang yang tersedia. Lalu pengguna menginput nama baru, stok baru, dan harga baru. Program akan mengeluarkan output berupa “barang berhasil diupdate”. Pada menu menghapus barang pengguna menginput nomer barang yang tersedia maka program akan mengeluarkan output berupa “barang berhasil dihapus,” beserta detail barang yang dihapus. Apabila pengguna menginput angka negative atau huruf pada saat input stok dan harga maka akan mengeluarkan output “input tidak valid”.

d. Keluar program

Pengguna menginput 5 pada menu utama outputnya “Anda telah keluar dari program” dan keluar dari program.

4.2 Hasil Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

=====
                        Login Program
=====
Username: Muhammad Ridwanansyah Rahman
Password: 2409106073

=====
                        Login Berhasil
=====
Selamat datang, Muhammad Ridwanansyah Rahman!
Anda telah berhasil login.
=====
Klik Enter untuk melanjutkan...|
```

Gambar 4.1 Menu Login

```
=====
MENU MANAJEMEN BARANG SEMBAKO
=====
1. Tambah Barang
2. Lihat Barang
3. Update Barang
4. Hapus Barang
5. Keluar
=====
Pilih menu: 1
Masukkan Nama Barang: masako
Masukkan Stok Barang: -98
Stok barang tidak boleh negatif!!
Masukkan Stok Barang: delapan
Input harus berupa angka!
Masukkan Stok Barang: 20
Masukkan Harga Barang: -3400
Harga barang tidak boleh negatif!!
Masukkan Harga Barang: sembilan
Input harus berupa angka!
Masukkan Harga Barang: 3000

=====
BARANG BERHASIL DITAMBAHKAN
=====
Barang dengan nama masako
Jumlah barang: 20
Harga barang: 3000
=====
Klik Enter untuk kembali ke menu utama...|
```

Gambar 4.2 Menu Tambah Barang dan kemungkinan inputnya

```
=====
MENU MANAJEMEN BARANG SEMBAKO
=====
1. Tambah Barang
2. Lihat Barang
3. Update Barang
4. Hapus Barang
5. Keluar
=====
Pilih menu: 2

=====
DAFTAR BARANG SEMBAKO
=====
No      Nama Barang      Stok      Harga
1        Beras           50       12000
2    Minyak Goreng     30       25000
3          Gula       40       15000
4        masako       20        3000
=====
Klik Enter untuk kembali ke menu...|
```

Gambar 4.3 Menu menampilkan daftar barang

```
=====
DAFTAR BARANG SEMBAKO
=====
No      Nama Barang      Stok      Harga
1        Beras           50       12000
2    Minyak Goreng     30       25000
3          Gula       40       15000
4        masako       20        3000
=====
Masukkan nomor barang yang ingin diupdate: 4
Masukkan Nama Barang Baru: Garam
Masukkan Jumlah Barang Baru: -23
Jumlah barang tidak boleh negatif!!
Masukkan Jumlah Barang Baru: dua
Input harus berupa angka!
Masukkan Jumlah Barang Baru: 23
Masukkan Harga Barang Baru: -5000
Harga barang tidak boleh negatif!!
Masukkan Harga Barang Baru: duar
Input harus berupa angka!
Masukkan Harga Barang Baru: 5000

=====
BARANG BERHASIL DIUPDATE
=====
Klik Enter untuk kembali ke menu...
```

Gambar 4.4 Menu Update Barang dan kemungkinan inputnya

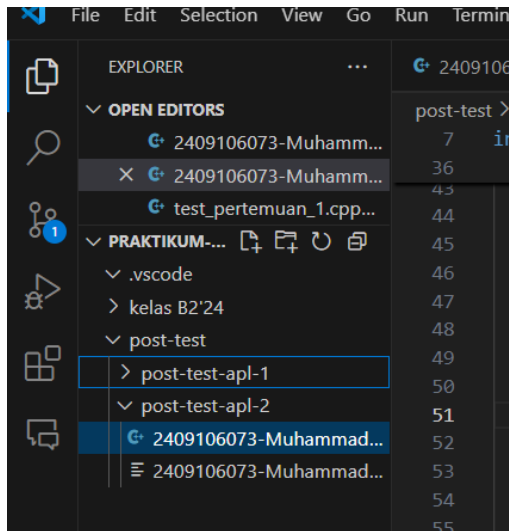
```
=====
                        DAFTAR BARANG SEMBAKO
=====
No          Nama Barang      Stok      Harga
  1          Beras           50       12000
  2      Minyak Goreng       30       25000
  3          Gula            40       15000
=====
Masukkan nomor barang yang ingin dihapus: 3
=====
                        BARANG BERHASIL DIHAPUS
=====
Barang dengan nama Gula
Jumlah barang : 40
Harga barang : 15000
=====
Klik Enter untuk kembali ke menu...|
```

Gambar 4.5 Menu Menghapus Barang

```
=====
                        MENU MANAJEMEN BARANG SEMBAKO
=====
1. Tambah Barang
2. Lihat Barang
3. Update Barang
4. Hapus Barang
5. Keluar
=====
Pilih menu: 5
=====
                        ANDA TELAH KELUAR DARI PROGRAM
=====
PS A:\praktikum-apl\post-test\post-test-apl-2>|
```

Gambar 4.6 Keluar Program

5. Langkah-langkah Git



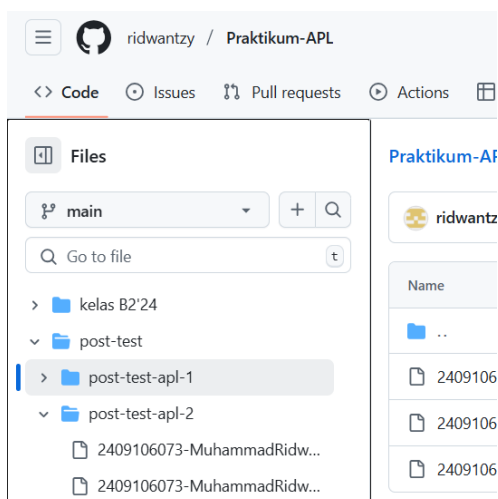
Gambar 5.1 Membuat folder posttest 2 di VSCode

```
PS A:\praktikum-apl> git add .
PS A:\praktikum-apl> git commit -m "posttest2 selesai"
[main ef84b2f] posttest2 selesai
6 files changed, 370 insertions(+), 6 deletions(-)
rename post-test/{ => post-test-apl-1}/2409106073-MuhammadRidwanansyahRahman-PT-1.cpp (100%)
rename post-test/{ => post-test-apl-1}/2409106073-MuhammadRidwanansyahRahman-PT-1.exe (100%)
create mode 100644 post-test/post-test-apl-2/2409106073-MuhammadRidwanansyahRahman-PT-2.cpp
create mode 100644 post-test/post-test-apl-2/2409106073-MuhammadRidwanansyahRahman-PT-2.exe
```

Gambar 5.2 melakukan git add dan git commit

```
PS A:\praktikum-apl> git push -u origin main
Enumerating objects: 22, done.
Counting objects: 100% (20/20), done.
Delta compression using up to 12 threads
Compressing objects: 100% (12/12), done.
Writing objects: 100% (14/14), 949.84 KiB | 3.02 MiB/s, done.
```

Gambar 5.3 melakukan git push



Gambar 5.4 Posttest 2 berhasil ditambahkan