

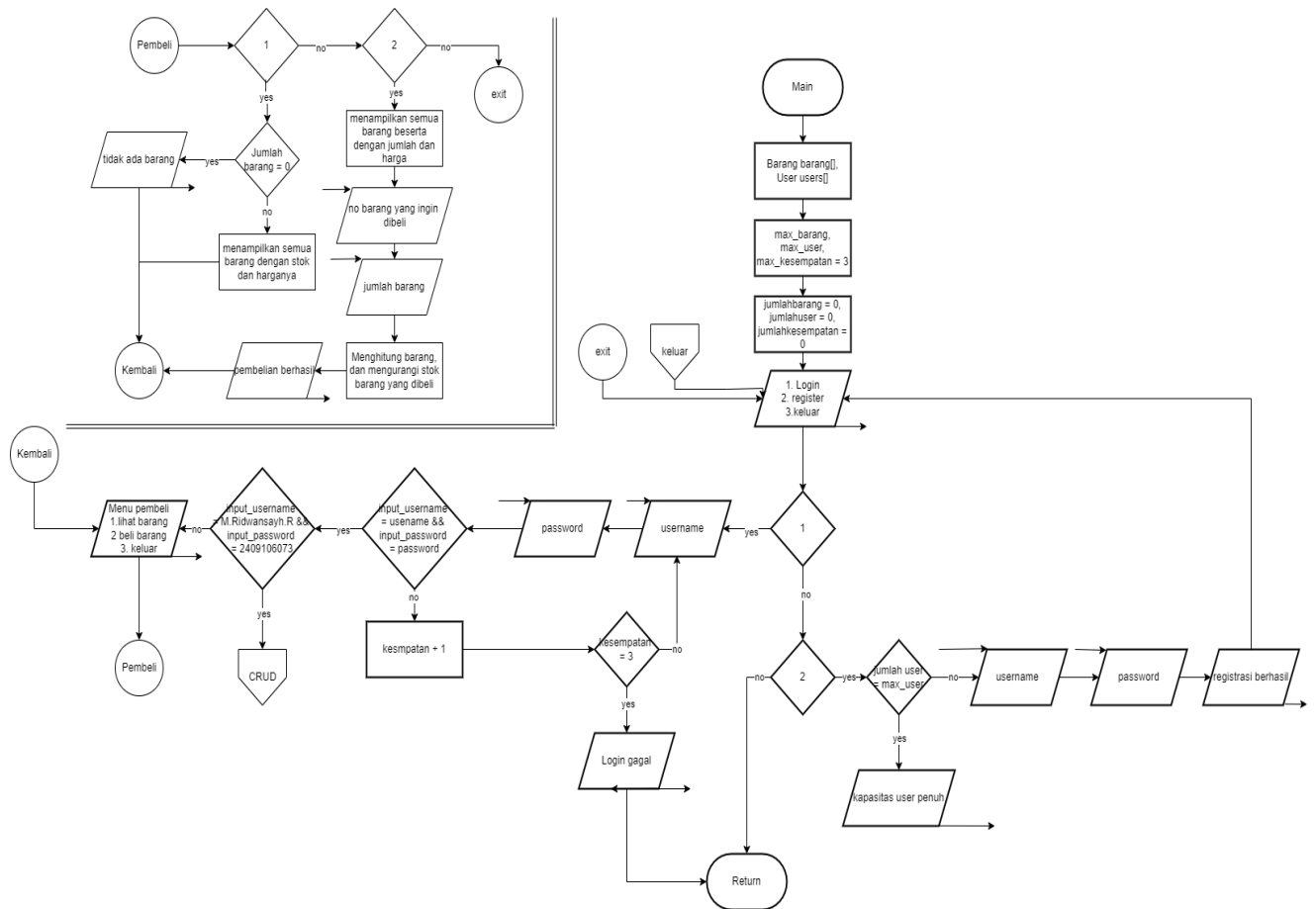
**LAPORAN PRAKTIKUM**  
**POSTTEST 3**  
**ALGORITMA PEMROGRAMAN LANJUT**



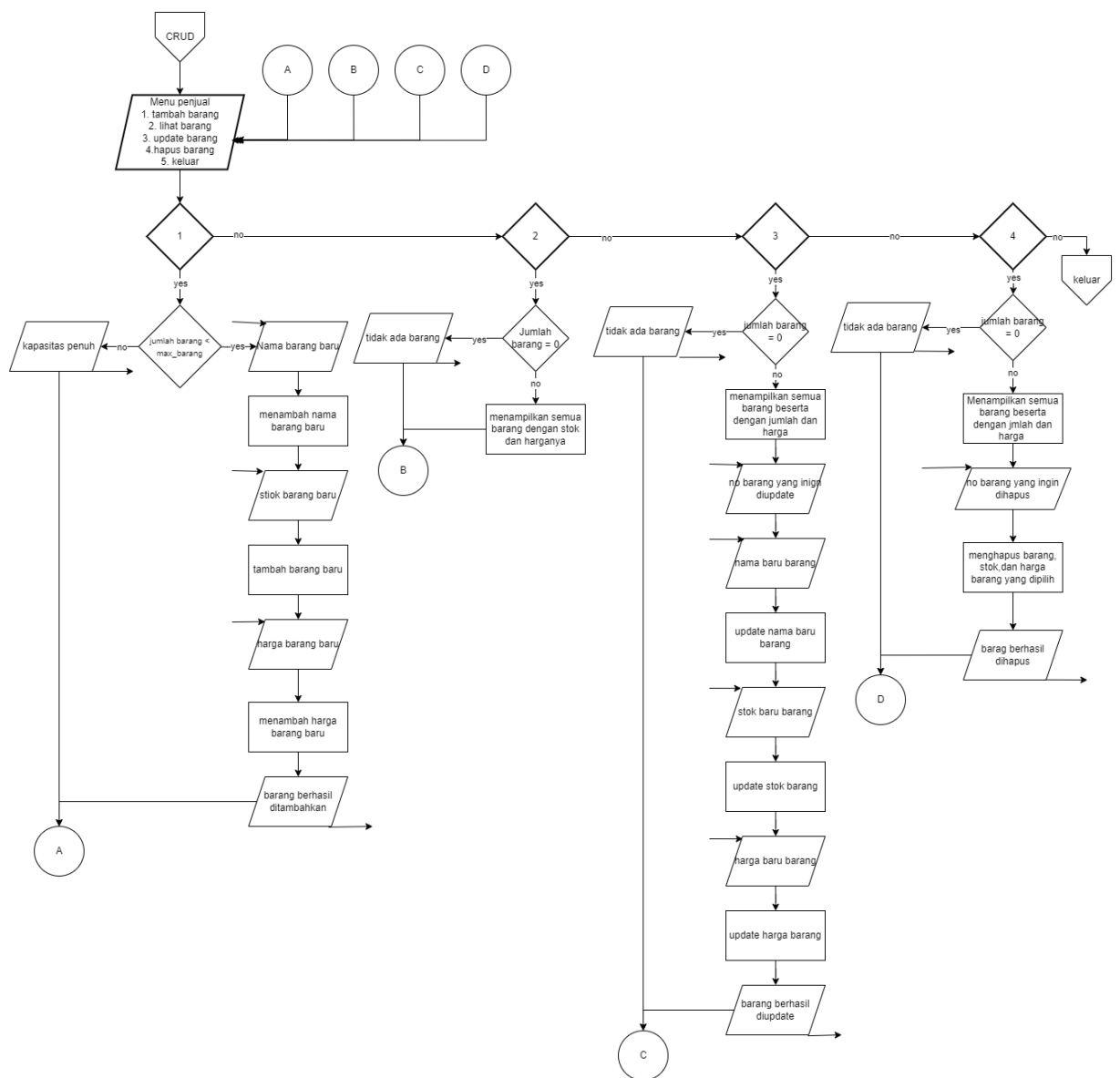
**Disusun oleh:**  
**Muhammad Ridwanansyah Rahman (2409106073)**  
**Kelas (B2 '24)**

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

# 1. Flowchart



Gambar 1.1 Flowchart Menu Login, registrasi, dan Pembeli



Gambar 1.2 Flowchart menu CRUD

## **2. Analisis Program**

Tujuan utama program ini adalah membangun sistem yang membantu dalam manajemen barang sembako dan dapat digunakan oleh penjual maupun pembeli.

Fungsi utama program ini adalah:

- **Manajemen Barang:** Program ini memungkinkan penjual untuk menambah, melihat, mengupdate, dan menghapus barang sembako.
- **Pembelian Barang:** Program ini memungkinkan pembeli untuk melihat daftar barang sembako yang tersedia dan melakukan pembelian.
- **Manajemen Akun:** Program ini memungkinkan pengguna untuk mendaftar dan login ke dalam sistem.

Dengan demikian, program ini dapat membantu penjual dalam mengelola stok barang sembako dan memudahkan pembeli dalam melakukan pembelian.

### 3. Source Code

#### A. Fitur Registrasi

Fitur ini digunakan pembeli untuk membuat akun pertama kali, yang dimana program pertama-tama mengecek jumlah user yang terdaftar, kemudian program meminta input berupa username dan password yang kemudian akan mengeluarkan pesan output “Registrasi Berhasil” dan mengarahkan pembeli ke menu awal untuk melakukan login.

#### Source Code:

```
if (jumlahUser < MAX_USER) {
    cin.ignore();
    system("cls");
    cout << "\n===== " << endl;
    cout << "          REGISTRASI AKUN          " << endl;
    cout << "===== " << endl;
    cout << "Username: ";
    getline(cin, users[jumlahUser].username);
    cout << "Password: ";
    getline(cin, users[jumlahUser].password);
    users[jumlahUser].role = "pembeli";
    jumlahUser++;
    system("cls");
    cout << "\n===== " << endl;
    cout << "          REGISTRASI BERHASIL          " << endl;
    cout << "===== " << endl;
    cout << "Selamat, Anda telah berhasil mendaftar!" << endl;
    cout << "===== " << endl;
    cout << "Klik Enter untuk kembali ke menu...";
    cin.ignore();
}
```

#### B. Fitur Login

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

#### Source Code:

```

while (kesempatan < max_kesempatan && !loginBerhasil) {
    cin.ignore();
    system("cls");
    cout << "\n===== " << endl;
    cout << "          Login Program          " << endl;
    cout << "===== " << endl;
    cout << "Username: ";
    getline(cin, username);
    cout << "Password: ";
    getline(cin, password);

    if (username == "Muhammad Ridwanansyah Rahman" && password ==
"2409106073") {
        loginBerhasil = true;
        role = "penjual";
    } else {
        for (int i = 0; i < jumlahUser; i++) {
            if (users[i].username == username && users[i].password ==
password) {
                loginBerhasil = true;
                role = users[i].role;
                break;
            }
        }
    }
    if (loginBerhasil) {
        system("cls");
        cout << "\n===== " << endl;
        cout << "          Login Berhasil          " << endl;
        cout << "===== " << endl;
        cout << "Selamat datang, " << username << "!" << endl;
        cout << "Anda login sebagai " << role << "." << endl;
        cout << "===== " << endl;
        cout << "Klik Enter untuk melanjutkan...";
        cin.ignore();
    }
}

```

### C. Menambah Barang

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

**Source Code:**

```

if (jumlahBarang < MAX_BARANG) {
    cin.ignore();
}

```

```

cout << "Masukkan Nama Barang: ";
getline(cin, barang[jumlahBarang].nama);

int stok, harga;
bool valid = false;

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

jumlahBarang++;
system("cls");
cout << "\n===== " << endl;
cout << "          BARANG BERHASIL DITAMBAHKAN          " << endl;
cout << "===== " << endl;
cout << "Barang dengan nama " << barang[jumlahBarang-1].nama << endl;
cout << "Jumlah barang: " << barang[jumlahBarang-1].detail.stok << endl;
cout << "Harga barang: " << barang[jumlahBarang-1].detail.harga << endl;
cout << "===== " << endl;

```

```

        cout << "Klik Enter untuk kembali ke menu utama..." ;
        cin.ignore();
        cin.get();
    } else {
        cout << "Kapasitas barang penuh!\n";
        cout << "Klik Enter untuk kembali ke menu...";
        cin.ignore();
        cin.get();
    }
    break;

```

#### 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].nama <<
    setw(10) << barang[i].detail.stok << setw(10) << barang[i].detail.harga <<
    "\n";
}
cout << "===== " << endl;
}
cout << "Klik Enter untuk kembali ke menu...";

```

#### 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.ignore();
    getline(cin, barang[index - 1].nama);

    int stokBaru;
    valid = false;
    while (!valid) {
        cout << "Masukkan Stok Barang Baru: ";
        if (cin >> stokBaru) {
            if (stokBaru > 0) {
                valid = true;
                barang[index - 1].detail.stok = stokBaru;
            } else {
                cout << "Stok barang tidak boleh 0 atau kurang!!\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].detail.harga = hargaBaru;
    } else {
        cout << "Harga barang tidak boleh 0 atau kurang!!\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.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();
    }
}

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

```

## G. Membeli Barang

Pada kode ini program akan menampilkan daftar barang yang beserta sisa stok dan harganya, yang kemudian pembeli disuruh untuk menginput nomor barang yang ingin dibeli serta banyaknya. Kemudian program akan mengeluarkan output berupa total harga yang harus dibayar pembeli.

### Source Code:

```
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].nama << setw(10) <<
barang[i].detail.stok << setw(10) << barang[i].detail.harga << "\n";
    }
    cout << "===== " << endl;

    cout << "Masukkan nomor barang yang ingin dibeli: ";
    if (cin >> index) {
        if (index > 0 && index <= jumlahBarang) {
            cout << "Masukkan jumlah yang ingin dibeli: ";
            if (cin >> jumlahBeli) {
                if (jumlahBeli > 0 && jumlahBeli <= barang[index -
1].detail.stok) {
                    valid = true;
                    barang[index - 1].detail.stok -= jumlahBeli;
                    int totalHarga = jumlahBeli * barang[index -
1].detail.harga;
                    system("cls");
                    cout << "\n===== "
<< endl;

                    cout << "          PEMBELIAN BERHASIL          " << endl;
                    cout << "===== " <<
endl;

                    cout << "Total harga: Rp " << totalHarga << ",-" << endl;
                    cout << "Silahkan lakukan pembayaran..." << endl;
                    cout << "===== " <<
endl;

                    cout << "Klik Enter untuk kembali ke menu...";
```

#### 4. Uji Coba dan Hasil Output

```
PS E:\kerja posttest> cd "e:\kerja posttest\" ; if ($?) {
=====
                        SELAMAT DATANG
=====
1. Login
2. Register
3. Keluar
=====
Pilih menu: 2

=====
                        REGISTRASI AKUN
=====
Username: ridwan
Password: 12345

=====
                        REGISTRASI BERHASIL
=====
Selamat, Anda telah berhasil mendaftar!
=====
Klik Enter untuk kembali ke menu...
=====
```

Gambar 4.1 Registrasi akun pembeli

```
=====
                        SELAMAT DATANG
=====
1. Login
2. Register
3. Keluar
=====
Pilih menu: 1

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

=====
                        Login Berhasil
=====
Selamat datang, Muhammad Ridwanansyah Rahman!
Anda login sebagai penjual.
=====
Klik Enter untuk melanjutkan...
=====
```

Gambar 4.2 Login penjual

```
=====
                        SELAMAT DATANG
=====
1. Login
2. Register
3. Keluar
=====
Pilih menu: 1

=====
                        Login Program
=====
Username: ridwan
Password: 12345

=====
                        Login Berhasil
=====
Selamat datang, ridwan!
Anda login sebagai pembeli.
=====
Klik Enter untuk melanjutkan...|
```

Gambar 4.3 Login pembeli

```
=====
                        MENU MANAJEMEN BARANG SEMBAKO
=====
1. Tambah Barang
2. Lihat Barang
3. Update Barang
4. Hapus Barang
5. Keluar
=====
Pilih menu: 1
Masukkan Nama Barang: Cuka manis
Masukkan Stok Barang: 25
Masukkan Harga Barang: 15000

=====
                        BARANG BERHASIL DITAMBAHKAN
=====
Barang dengan nama Cuka manis
Jumlah barang: 25
Harga barang: 15000
=====
Klik Enter untuk kembali ke menu utama...|
```

Gambar 4.4 Tambah barang

```
=====
                        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      100   5000
2       Gula Pasir  50    2000
3       Minyak     20    15000
4       Cuka manis  25    15000
=====
Klik Enter untuk kembali ke menu...|
```

Gambar 4.5 Tampilkan data barang

```
=====
MENU MANAJEMEN BARANG SEMBAKO
=====

1. Tambah Barang
2. Lihat Barang
3. Update Barang
4. Hapus Barang
5. Keluar

=====
Pilih menu: 3

=====
DAFTAR BARANG SEMBAKO
=====

No      Nama Barang      Stok      Harga
1        Beras          100       5000
2        Gula Pasir        50        2000
3        Minyak Goreng    20        15000
4        Cuka manis        25        15000

=====
Masukkan nomor barang yang ingin diupdate: 4
Masukkan Nama Barang Baru: Gula Pahit
Masukkan Stok Barang Baru: 25
Masukkan Harga Barang Baru: 10000

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

Gambar 4.6 Update data barang

```
=====
MENU MANAJEMEN BARANG SEMBAKO
=====

1. Tambah Barang
2. Lihat Barang
3. Update Barang
4. Hapus Barang
5. Keluar

=====
Pilih menu: 4

=====
DAFTAR BARANG SEMBAKO
=====

No      Nama Barang      Stok      Harga
1        Beras          100       5000
2        Gula Pasir        50        2000
3        Minyak Goreng    20        15000
4        Gula Pahit        25        10000

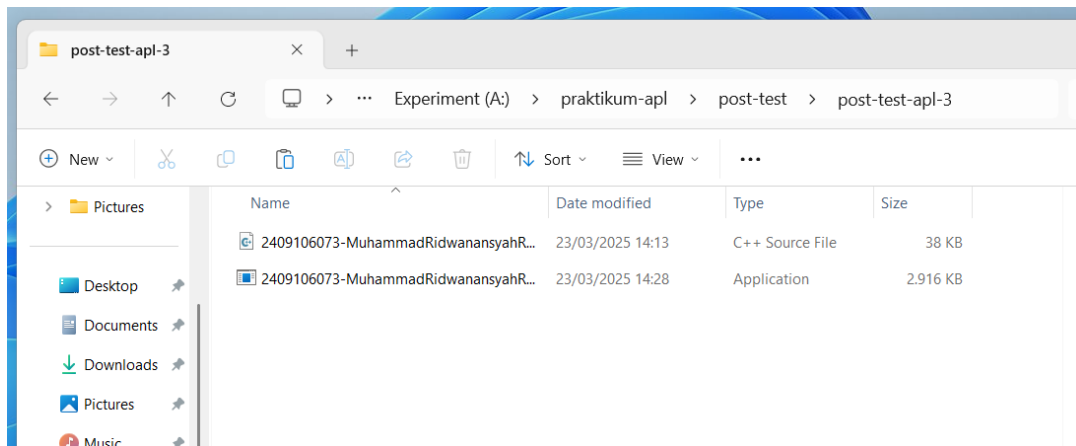
=====
Masukkan nomor barang yang ingin dihapus: 4

=====
BARANG BERHASIL DIHAPUS
=====
Barang dengan nama Gula Pahit
Jumlah barang : 25
Harga barang : 10000

=====
Klik Enter untuk kembali ke menu...s|
```

Gambar 4.7 Hapus data barang

## 5. Langkah-Langkah Git pada VSCode



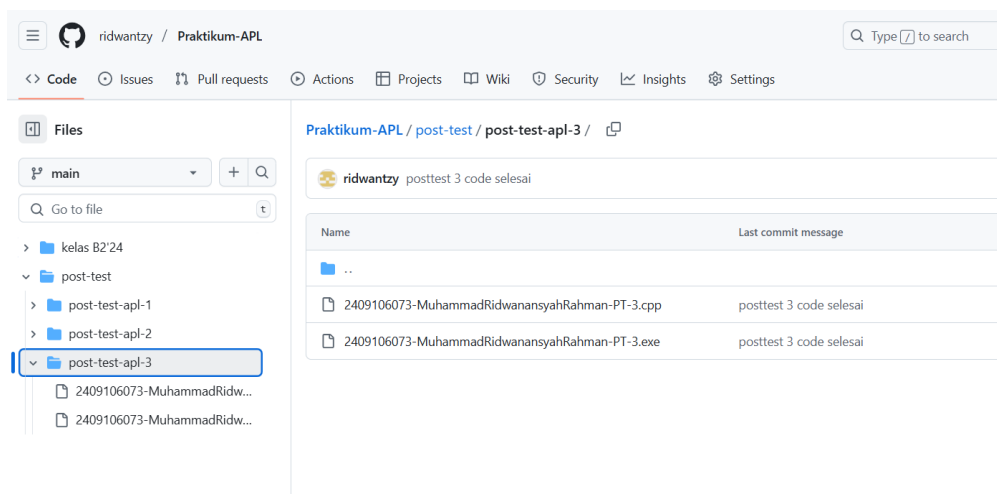
Gambar 5.1 Membuat folder `post-test-apl-3` yang berisi file `.cpp` dan `.exe` dalam file lokal `praktikum-apl`

```
PS A:\praktikum-apl> git add .
PS A:\praktikum-apl> git commit -m "posttest 3 code selesai"
[main bea8311] posttest 3 code selesai
5 files changed, 691 insertions(+)
create mode 100644 kelas B2'24/test_pertemuan_3.cpp
create mode 100644 kelas B2'24/test_pertemuan_3.exe
create mode 100644 post-test/post-test-apl-3/2409106073-MuhammadRidwanansyahRahman-PT-3.cpp
create mode 100644 post-test/post-test-apl-3/2409106073-MuhammadRidwanansyahRahman-PT-3.exe
```

Gambar 5.2 Buka folder dengan terminal lalu melakukan `git add` dan `git commit`

```
PS A:\praktikum-apl> git push -u origin main
Enumerating objects: 16, done.
Counting objects: 100% (16/16), done.
Delta compression using up to 12 threads
Compressing objects: 100% (11/11), done.
Writing objects: 100% (11/11), 944.00 KiB | 4.84 MiB/s, done.
Total 11 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), completed with 2 local objects.
To https://github.com/ridwantzy/Praktikum-APL.git
aa517df..bea8311 main -> main
branch 'main' set up to track 'origin/main'.
```

Gambar 5.3 Melakukan `git push`



Gambar 5.4 Posttest 3 berhasil ditambahkan