

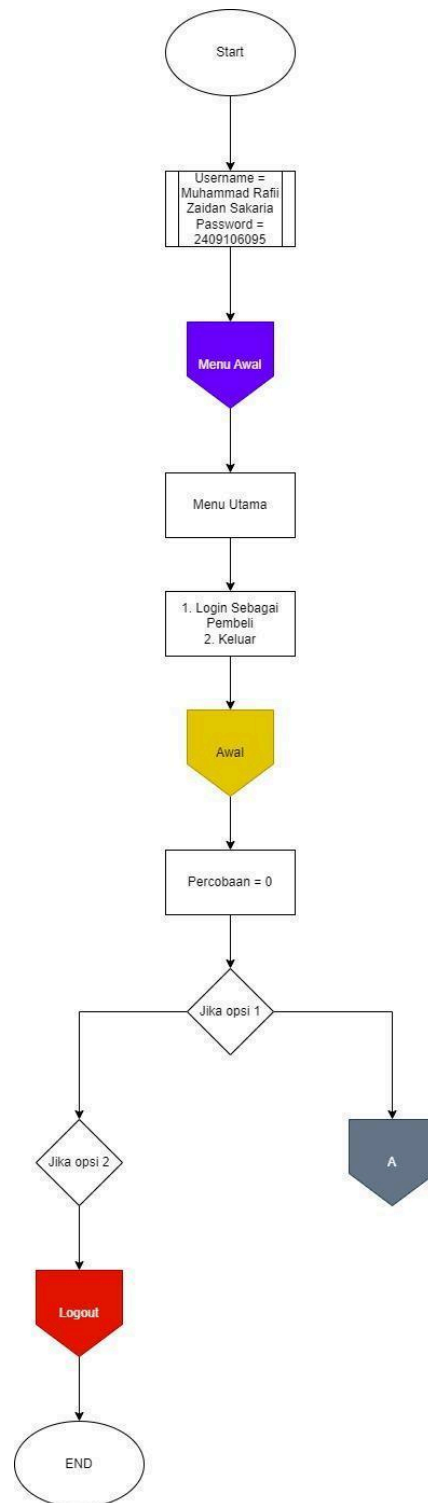
LAPORAN PRAKTIKUM
POSTTEST 4
ALGORITMA PEMROGRAMAN LANJUT

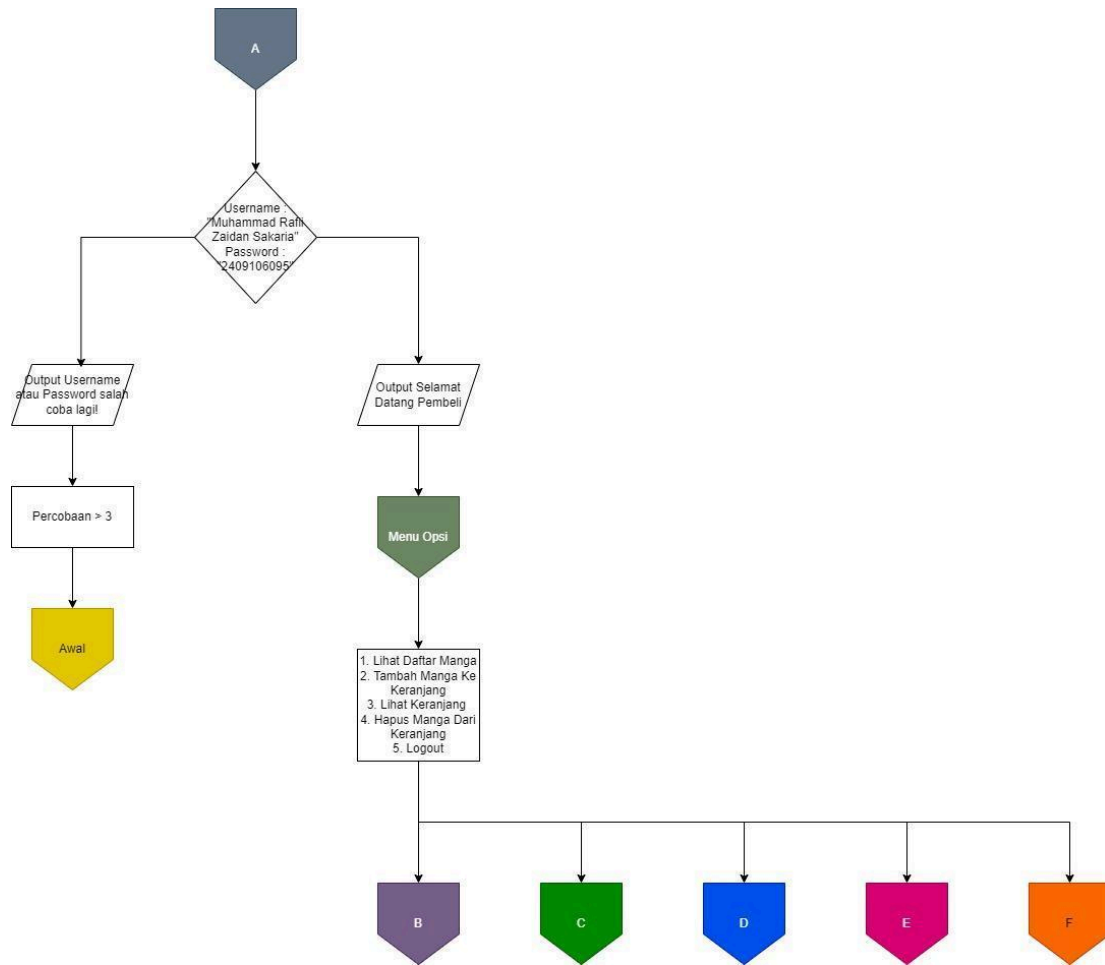


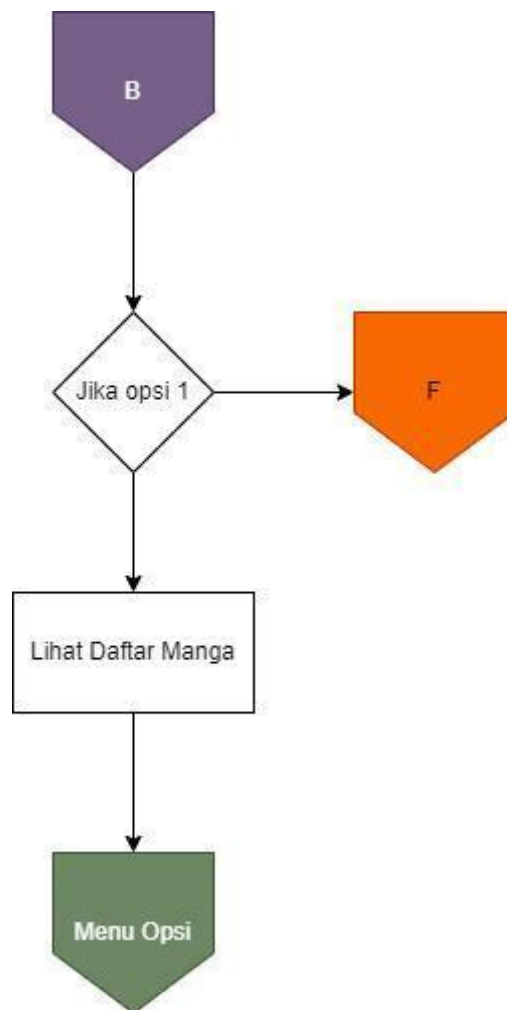
Disusun oleh:
Muhammad Rafii' Zaidan Sakaria 2409106095
Kelas C1 '24

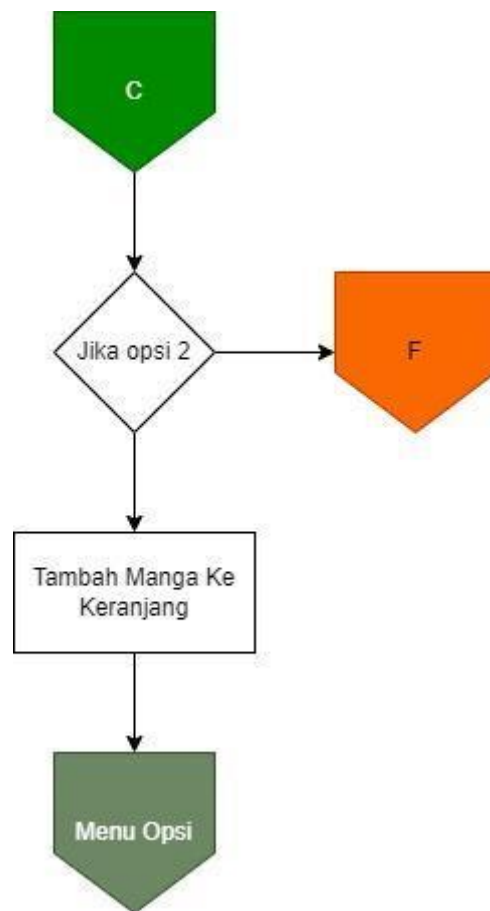
PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

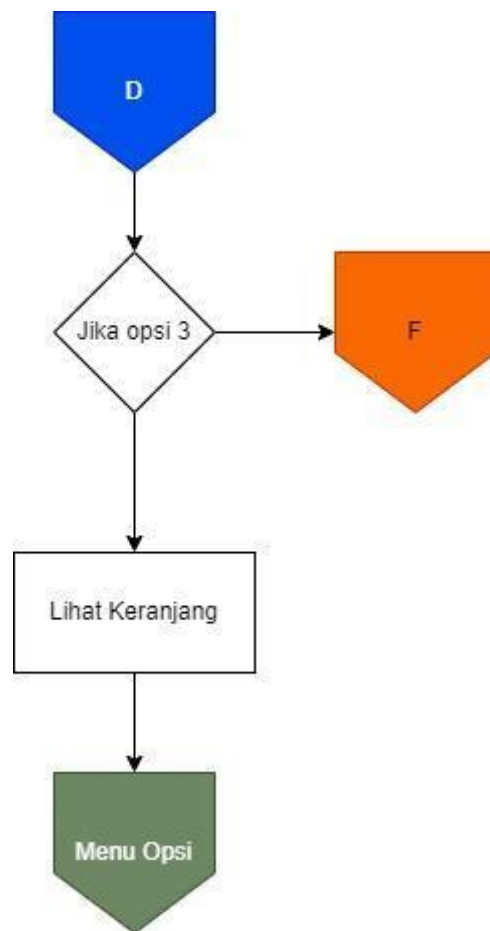
1. Flowchart

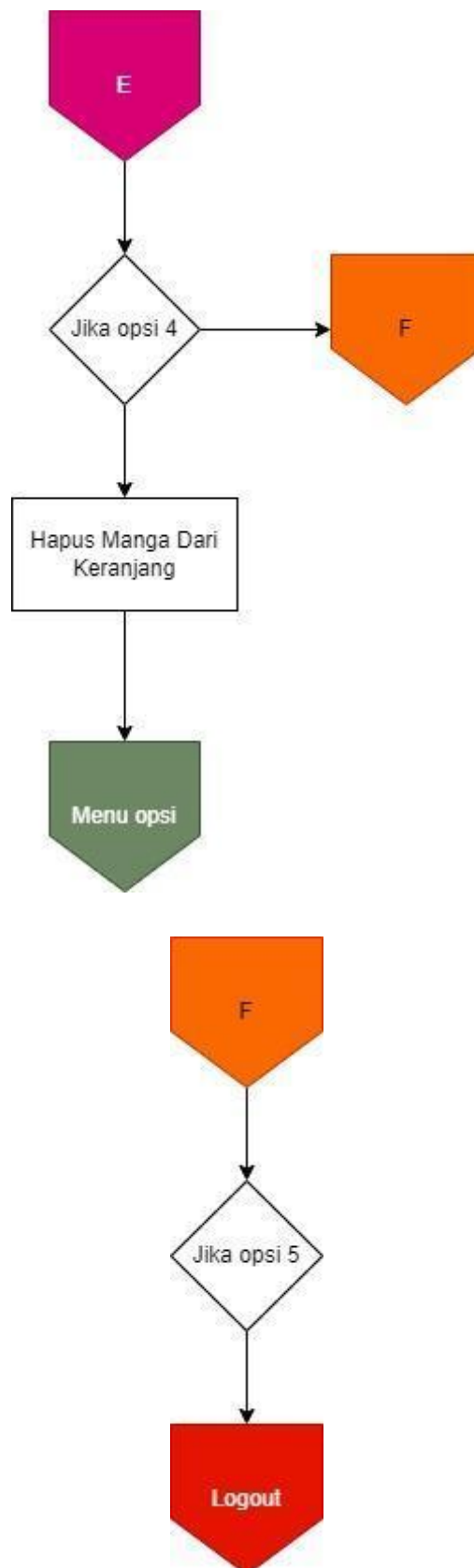












2. Analisis Program

Program ini adalah sistem pembelian Manga dengan fitur login dan keranjang belanja program ini juga punya fitur-fitur nya yaitu :

1. Pengguna (Pembeli)
2. Sistem keamanan login dengan 3x percobaan kalau sudah dari 3x percobaan program akan berhenti
3. Melihat daftar Manga yang tersedia
 - menampilkan daftar manga dengan harga masing-masing
4. Menambah Manga ke keranjang
 - Pengguna memilih manga berdasar kan nomor
 - Jika manga sudah ada di keranjang, jumlah bertambah
 - Jika belum ada, manga baru ditambah ke dalam keranjang
5. Melihat isi Keranjang
 - Menampilkan daftar manga yg telah di beli
 - Menampilkan harga
6. Menghapus isi keranjang
 - dapat menghapus atau delete manga yg kita mau
7. Logout
 - Mengembalikan pengguna ke menu utama
8. Keluar dari Program
 - Jika pengguna memilih opsi “Keluar”, maka program akan berhenti

3. Source Code

```
#include <iostream>
using namespace std;

struct User {
    string username;
    string password;
};

struct Manga {
    string judul;
    int harga;
};

struct Keranjang {
    string judul;
    int jumlah;
    int totalHarga;
};

// Fungsi untuk menampilkan daftar manga
void tampilkanDaftarManga(Manga mangaList[], int jumlahmanga) {
    cout << "\n=====";
    cout << "\n          DAFTAR MANGA";
    cout << "\n=====";
    cout << "\nNo.   Judul Manga          Harga";
    cout << "\n-----";
    for (int i = 0; i < jumlahmanga; i++) {
        cout << "\n" << i + 1 << ".   " << mangaList[i].judul;
        if (mangaList[i].judul.length() < 15) {
            cout << string(15 - mangaList[i].judul.length(), ' ');
        }
        cout << "Rp" << mangaList[i].harga;
    }
    cout << "\n=====";
}

// Fungsi untuk menampilkan keranjang
void tampilkanKeranjang(Keranjang keranjang[], int jumlahKeranjang) {
    if (jumlahKeranjang == 0) {
        cout << "\n=====";
        cout << "\n          KERANJANG";
        cout << "\n=====";
        cout << "\nKeranjang Anda masih kosong.";
        cout << "\n=====";
    } else {
        cout << "\n=====";
    }
}
```

```

        cout << "\n                KERANJANG";
        cout << "\n=====";
        cout << "\nNo.   Judul Manga           Jumlah   Total Harga";
        cout << "\n-----";
        for (int i = 0; i < jumlahKeranjang; i++) {
            cout << "\n" << i + 1 << ".   " << keranjang[i].judul;
            if (keranjang[i].judul.length() < 15) {
                cout << string(15 - keranjang[i].judul.length(), ' ');
            }
            cout << keranjang[i].jumlah << "        Rp" <<
keranjang[i].totalHarga;
        }
        cout << "\n=====";
    }
}

// Fungsi untuk menambah manga ke keranjang
void tambahKeKeranjang(Keranjang keranjang[], int &jumlahKeranjang, Manga
mangaList[], int jumlahmanga) {
    int pilihManga, Jumlahbeli;
    cout << "\n=====";
    cout << "\n        TAMBAH KE KERANJANG";
    cout << "\n=====";
    cout << "\nPilih manga yang ingin dibeli (nomor): ";
    cin >> pilihManga;

    if (pilihManga < 1 || pilihManga > jumlahmanga) {
        cout << "\nERROR: Nomor manga tidak valid!";
    } else {
        cout << "Masukkan jumlah: ";
        cin >> Jumlahbeli;

        if (Jumlahbeli <= 0) {
            cout << "\nERROR: Jumlah tidak valid!";
        } else {
            int ditemukan = 0;
            for (int i = 0; i < jumlahKeranjang; i++) {
                if (keranjang[i].judul == mangaList[pilihManga - 1].judul) {
                    keranjang[i].jumlah += Jumlahbeli;
                    keranjang[i].totalHarga += mangaList[pilihManga -
1].harga * Jumlahbeli;
                    ditemukan = 1;
                    break;
                }
            }

            if (ditemukan == 0) {
                keranjang[jumlahKeranjang].judul = mangaList[pilihManga -

```

```

1].judul;
        keranjang[jumlahKeranjang].jumlah = Jumlahbeli;
        keranjang[jumlahKeranjang].totalHarga = mangaList[pilihManga
- 1].harga * Jumlahbeli;
        jumlahKeranjang++;
    }

    cout << "\nBERHASIL: Manga berhasil ditambahkan ke keranjang!";
}
}
cout << "\n=====";
}

// Fungsi untuk menghapus manga dari keranjang
void hapusDariKeranjang(Keranjang keranjang[], int &jumlahKeranjang) {
    if (jumlahKeranjang == 0) {
        cout << "\n=====";
        cout << "\n        HAPUS DARI KERANJANG";
        cout << "\n=====";
        cout << "\nKeranjang Anda masih kosong.";
        cout << "\n=====";
    } else {
        tampilkanKeranjang(keranjang, jumlahKeranjang);

        int hapus;
        cout << "\nPilih nomor manga yang ingin dihapus: ";
        cin >> hapus;
        if (hapus < 1 || hapus > jumlahKeranjang) {
            cout << "\nERROR: Nomor tidak valid!";
        } else {
            for (int i = hapus - 1; i < jumlahKeranjang - 1; i++) {
                keranjang[i] = keranjang[i + 1];
            }
            jumlahKeranjang--;
            cout << "\nBERHASIL: Manga berhasil dihapus dari keranjang!";
        }
        cout << "\n=====";
    }
}

// MAIN FUNCTION
int main() {
    int pilihan;
    User userakun = {"Muhammad Rafii Zaidan Sakaria", "2409106095"};
    string username, password;
    int percobaan = 3;

    string judulmanga[] = {"One Piece", "Naruto", "Bleach", "Attack on

```

```

Titan", "Dragon Ball"};
int hargaManga[] = {50000, 45000, 48000, 55000, 52000};
int jumlahmanga = sizeof(judulmanga) / sizeof(judulmanga[0]);

Manga mangalist[5];
for (int i = 0; i < jumlahmanga; i++) {
    mangalist[i].judul = judulmanga[i];
    mangalist[i].harga = hargaManga[i];
}

Keranjang keranjang[10];
int jumlahKeranjang = 0;

while (true) {
    cout << "\n=====";
    cout << "\n          MENU UTAMA";
    cout << "\n=====";
    cout << "\n1. Login sebagai Pembeli";
    cout << "\n2. Keluar";
    cout << "\n-----";
    cout << "\nPilihan: ";
    cin >> pilihan;
    cin.ignore();

    if (pilihan == 1) {
        while (percobaan > 0) {
            cout << "\n=====";
            cout << "\n          LOGIN PEMBELI";
            cout << "\n=====";
            cout << "\nUsername: ";
            getline(cin, username);
            cout << "Password: ";
            cin >> password;
            cin.ignore();

            if (username == userakun.username && password ==
userakun.password) {
                while (true) {
                    cout <<
"\n=====";
                    cout << "\n          SELAMAT DATANG PEMBELI";
                    cout <<
"\n=====";
                    cout << "\n1. Lihat Daftar Manga";
                    cout << "\n2. Tambah Manga ke Keranjang";
                    cout << "\n3. Lihat Keranjang";
                    cout << "\n4. Hapus Manga dari Keranjang";
                    cout << "\n5. Logout";

```

```

        cout <<
"\n-----";
        cout << "\nPilihan: ";
        cin >> pilihan;
        cin.ignore();

        if (pilihan == 1) {
            tampilkanDaftarManga(mangaList, jumlahmanga);
        } else if (pilihan == 2) {
            tambahKeKeranjang(keranjang, jumlahKeranjang,
mangaList, jumlahmanga);
        } else if (pilihan == 3) {
            tampilkanKeranjang(keranjang, jumlahKeranjang);
        } else if (pilihan == 4) {
            hapusDariKeranjang(keranjang, jumlahKeranjang);
        } else if (pilihan == 5) {
            cout << "\nBERHASIL: Anda telah logout.";
            cout <<
"\n=====";
            break;
        } else {
            cout << "\nERROR: Pilihan tidak valid!";
        }
    }
    break;
} else {
    percobaan--;
    if (percobaan == 0) {
        cout <<
"\n=====";
        cout << "\n  LOGIN GAGAL 3x! PROGRAM BERHENTI";
        cout <<
"\n=====";
        return 0;
    }
    cout << "\nERROR: Login gagal! Username atau password
salah.";

    cout << "\nSisa percobaan: " << percobaan;
    cout << "\n=====";
}
}

} else if (pilihan == 2) {
    cout << "\n=====";
    cout << "\n          PROGRAM SELESAI";
    cout << "\n=====";
    return 0;
} else {
    cout << "\nERROR: Pilihan tidak valid!";

```

```
    }  
}  
  
return 0;  
}
```

4. Uji Coba dan Hasil Output

4.1 Uji Coba

1. Pengguna memilih membuat akun, pengguna di minta masukkan username dan password
2. Pengguna memilih login, jika username dan password benar maka login berhasil, jika username dan password salah maka program memberikan 3x percobaan jika sudah 3x percobaan maka program akan berhenti
3. Ketika pengguna berhasil login pengguna akan ke menu utama dan terdapat opsi, Lihat Daftar Manga, Tambah Manga Ke Keranjang, Lihat Keranjang, Hapus Manga Dari Keranjang, Logout
4. Dan di opsi itu ketika pengguna memilih opsi Lihat Daftar Manga maka pengguna bisa melihat daftar manga yang tersedia, jika pengguna memilih opsi Tambah Manga Ke Keranjang maka pengguna dapat menambahkan Manga yang mau di beli, jika pengguna memilih opsi Lihat Keranjang Maka pengguna dapat melihat apa saja yang dia beli, Jika pengguna memilih opsi Hapus Manga Dari Keranjang maka pengguna dapat menghapus manga apa yang dia beli, dan jika pengguna memilih opsi Logout maka pengguna akan ke menu

4.2 Hasil Output

```
=====
                        MENU UTAMA
=====
1. Login sebagai Pembeli
2. Keluar
-----
Pilihan: 1
=====
                        LOGIN PEMBELI
=====
Username: Muhammad Rafii Zaidan Sakaria
Password: 2409106095
```

Gambar 4.1 Contoh

```
=====
                        SELAMAT DATANG PEMBELI
=====

1. Lihat Daftar Manga
2. Tambah Manga ke Keranjang
3. Lihat Keranjang
4. Hapus Manga dari Keranjang
5. Logout
-----
Pilihan: 1

=====
                        DAFTAR MANGA
=====

No.   Judul Manga           Harga
-----
1.    One Piece             Rp50000
2.    Naruto                Rp45000
3.    Bleach                 Rp48000
4.    Attack on TitanRp55000
5.    Dragon Ball           Rp52000
=====
```

Gambar 4.2


```
=====
                        SELAMAT DATANG PEMBELI
=====
1. Lihat Daftar Manga
2. Tambah Manga ke Keranjang
3. Lihat Keranjang
4. Hapus Manga dari Keranjang
5. Logout
-----
Pilihan: 2

=====
                        TAMBAH KE KERANJANG
=====
Pilih manga yang ingin dibeli (nomor): 1
Masukkan jumlah: 2

BERHASIL: Manga berhasil ditambahkan ke keranjang!
=====
```

Gambar 4.3

```
=====
                        SELAMAT DATANG PEMBELI
=====
1. Lihat Daftar Manga
2. Tambah Manga ke Keranjang
3. Lihat Keranjang
4. Hapus Manga dari Keranjang
5. Logout
-----
Pilihan: 3

=====
                        KERANJANG
=====
No.   Judul Manga           Jumlah   Total Harga
-----
1.    One Piece             2        Rp100000
=====
```

Gambar 4.5

```

=====
                        SELAMAT DATANG PEMBELI
=====
1. Lihat Daftar Manga
2. Tambah Manga ke Keranjang
3. Lihat Keranjang
4. Hapus Manga dari Keranjang
5. Logout
-----
Pilihan: 4

=====
                        HAPUS DARI KERANJANG
=====
No.  Judul Manga          Jumlah  Total Harga
-----
1.   One Piece           2       Rp100000
=====
Pilih nomor manga yang ingin dihapus: 1

BERHASIL: Manga berhasil dihapus dari keranjang!
=====

```

Gambar 4.6

```
=====
                        SELAMAT DATANG PEMBELI
=====
1. Lihat Daftar Manga
2. Tambah Manga ke Keranjang
3. Lihat Keranjang
4. Hapus Manga dari Keranjang
5. Logout
-----
Pilihan: 5

BERHASIL: Anda telah logout.
=====
=====
                        MENU UTAMA
=====
1. Login sebagai Pembeli
2. Keluar
-----
Pilihan: 2

=====
                        PROGRAM SELESAI
=====
```

Gambar 4.7

5. Langkah-Langkah Git

```
ACER@LAPTOP-P702V07L MINGW64 /d/GITHUB/Praktikum-APL (main)
$ git add .

ACER@LAPTOP-P702V07L MINGW64 /d/GITHUB/Praktikum-APL (main)
$ git commit -m "post-test-4"
[main 26ff0a8] post-test-4
4 files changed, 232 insertions(+)
create mode 100644 Kelas/pertemuan ke 4.cpp
create mode 100644 Post-test/Post-test-4/2409106095-MuhammadRafiiZaidanSakaria-PT-4.cpp
create mode 100644 Post-test/Post-test-4/2409106095-MuhammadRafiiZaidanSakaria-PT-4.exe
create mode 100644 Post-test/Post-test-4/Template Laporan APL (4).docx

ACER@LAPTOP-P702V07L MINGW64 /d/GITHUB/Praktikum-APL (main)
$ git push
Enumerating objects: 11, done.
Counting objects: 100% (11/11), done.
Delta compression using up to 8 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (8/8), 1.09 MiB | 638.00 KiB/s, done.
Total 8 (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/Rafiizs/Praktikum-APL.git
   c856bf0..26ff0a8  main -> main

ACER@LAPTOP-P702V07L MINGW64 /d/GITHUB/Praktikum-APL (main)
$ |
```

1. Mengatur Email Global untuk Git

\$ git config --global user.email "123raffimhmd@gmail.com"

- Perintah ini digunakan untuk menetapkan email global dalam git, yang akan ditetapkan pada commit yang dibuat oleh user
- Email ini berfungsi untuk mengidentifikasi pengguna dalam repository git

2. Menginisialisasi Repository Git

\$ git init

- Perintah ini akan membuat repository git dalam folder bernama praktikum-Apl.
- Jika di dalam folder tersebut sudah terdapat repository Git, maka Git hanya akan memberikan notifikasi bahwa repository telah ada sebelum nya (reinitialized).

3. Menambahkan File ke Staging Area

\$ git add .

- Perintah ini berfungsi untuk memasukkan seluruh file yg ada di praktikum-Apl ke dalam staging Area
- Staging Area adalah tempat penyimpanan sementara sebelum file dikomit ke repository

4. Menambahkan Remote Repository

\$ git remote add origin <https://github.com/Rafiizs/Praktikum-APL.git>

- Perintah ini digunakan untuk menghubungkan repository lokal dengan repository jarak jauh (remote) yang diberikan nama origin.
- Jika muncul error "remote origin already exist", itu berarti repository remote sudah pernah di tambah kan sebelum nya

5. Membuat Commit dengan pesan "post-test-1"

\$ git commit -m "post-test-1"

- Perintah ini berfungsi untuk menyimpan perubahan yang telah ditambahkan ke staging dalam bentuk commit dengan pesan "post-test-1"

6. Mengunggah Push perubahan ke Repository Remote