

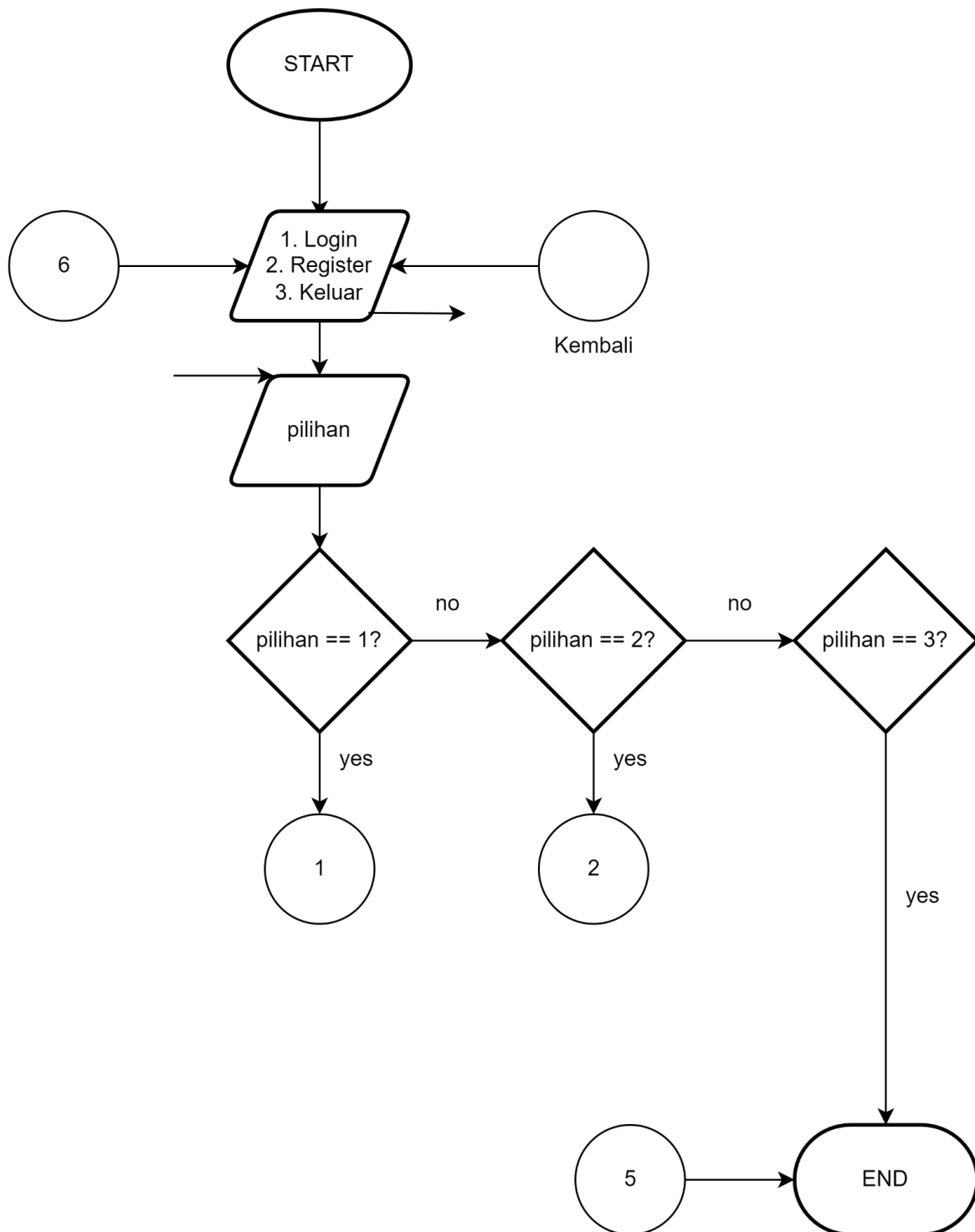
**LAPORAN PRAKTIKUM**  
**POSTTEST 5**  
**ALGORITMA PEMROGRAMAN LANJUT**



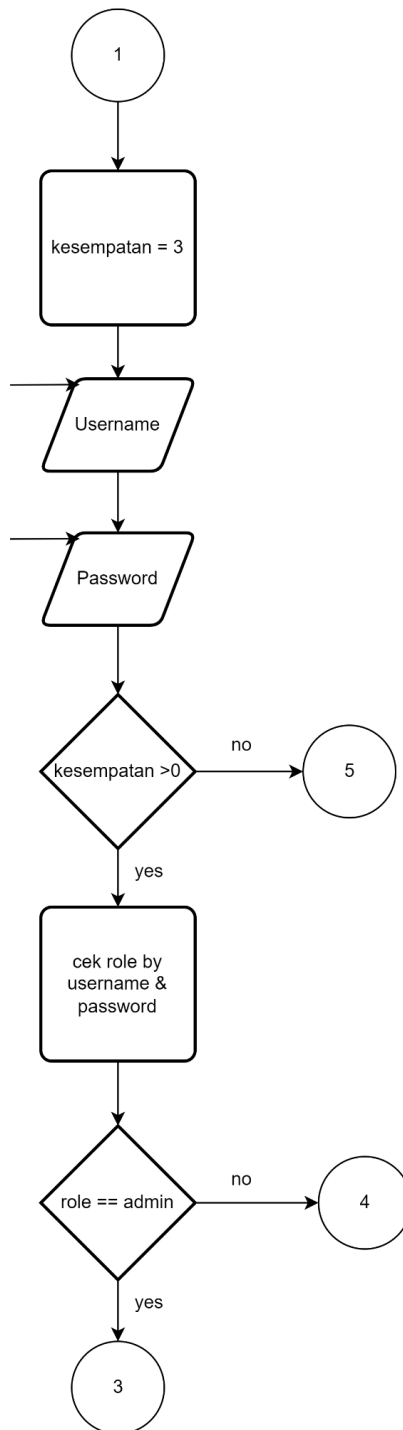
**Disusun oleh:**  
**Muhammad Riva Fachrodhiya 2409106053**  
**Kelas B1 '24**

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

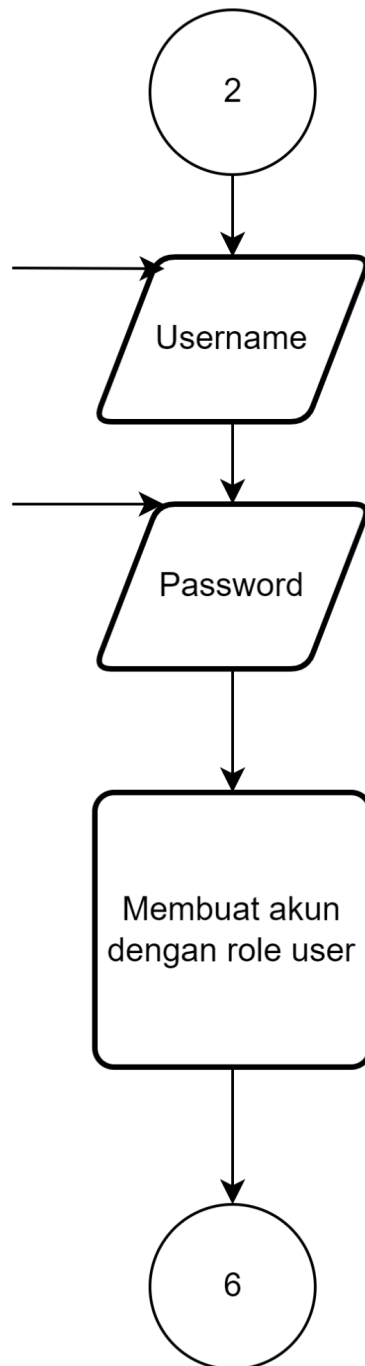
## 1. Flowchart



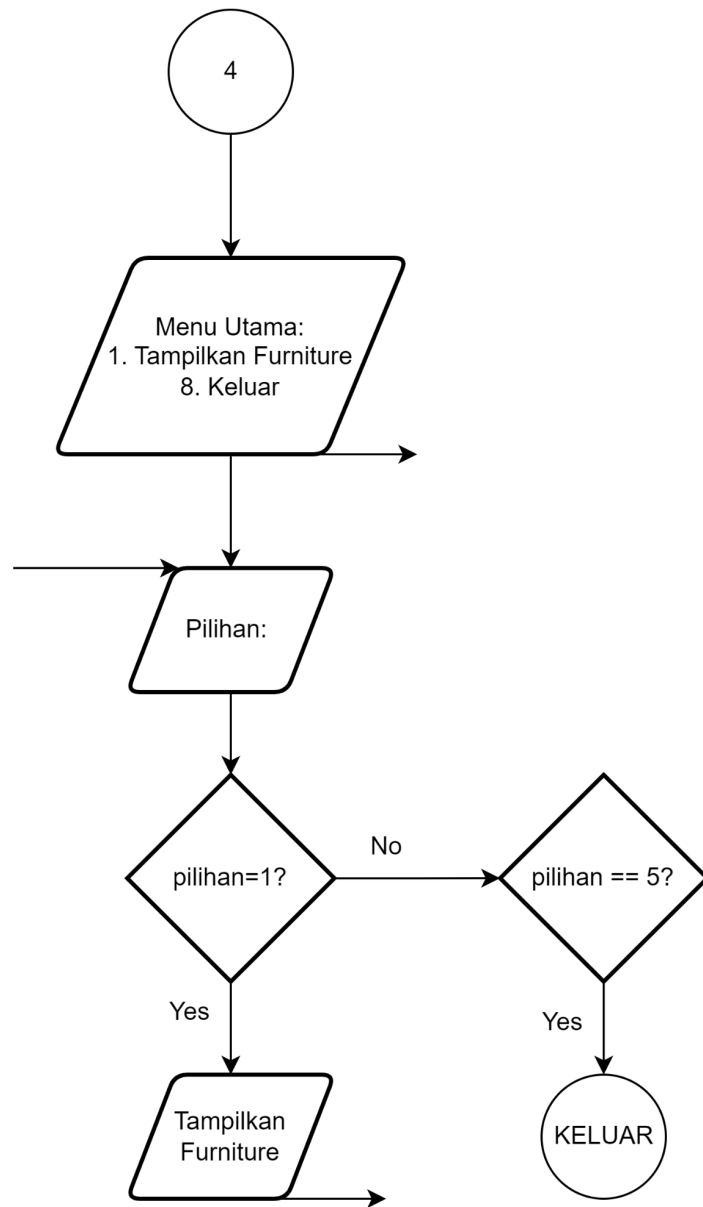
Gambar 1.1 Menu Akun



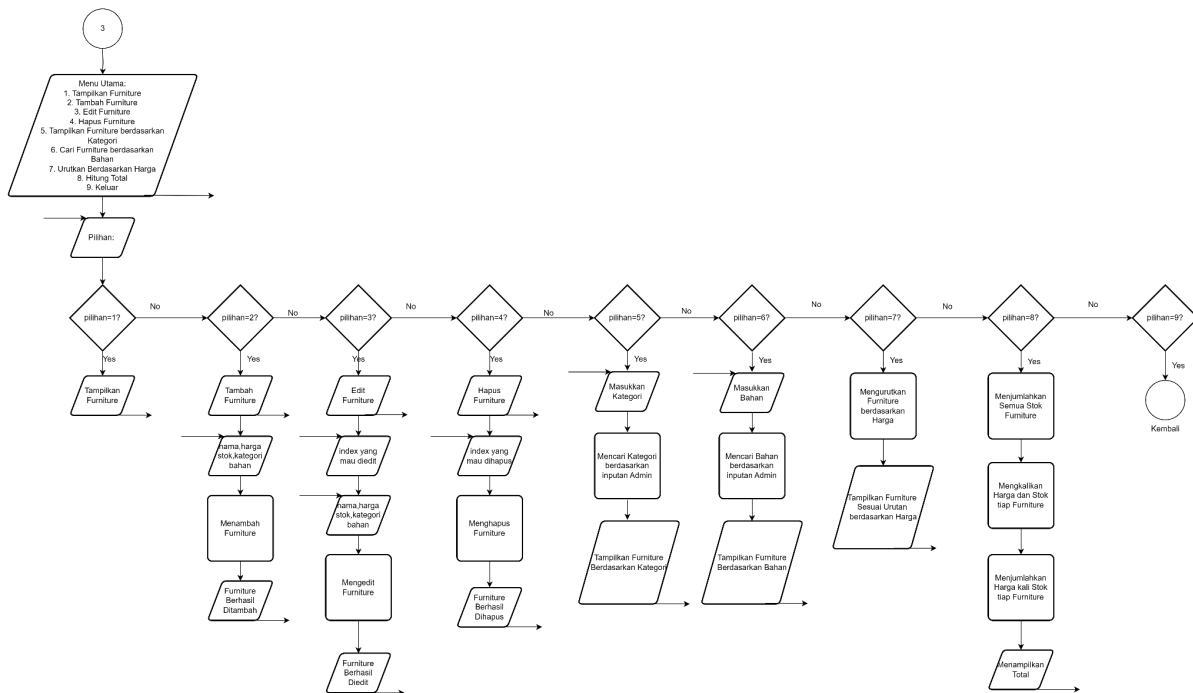
Gambar 1.2 Menu Login



Gambar 1.3 Menu Register



Gambar 1.4 Menu User



Gambar 1.5 Menu Admin

## 2. Penjelasan Singkat Program

Program Manajemen Tiket Kapal memiliki beberapa fitur, yaitu Login, Register, CRUD, Sorting, dan Searching. Program ini memiliki 2 role, yaitu Admin dan User.

## 3. Source Code

### A. Fitur Login

Fitur ini digunakan untuk pengguna dapat login dengan role masing-masing.

Source Code:

```
bool login(int attempts, string& role) {
    if (attempts == 0) {
        cout << "Anda telah salah memasukkan password 3 kali."
    }
}
```

```

Program berhenti.\n";
    return false;
}

string username, password;
cout << "Username: "; cin >> username;
cout << "Password: "; cin >> password;

for (int i = 0; i < jumlahUsers; i++) {
    if (username == users[i].username && password ==
users[i].password) {
        cout << "Login berhasil!\n";
        role = users[i].role;
        return true;
    }
}

cout << "Login gagal! Sisa percobaan: " << attempts - 1 <<
endl;
return login(attempts - 1, role);
}

```

## B. Fitur Register

Fitur ini digunakan agar pengguna baru dapat register dan akan mendapatkan role user.

```

void registrasi(User* userArray, int* jumlahPtr) {
    clear();
    if (*jumlahPtr >= MAX_USERS) {
        cout << "Pendaftaran penuh! Tidak bisa menambah akun baru.\n";
    } else {
        cin.ignore();
        cout << "Username: "; getline(cin, (userArray +
*jumlahPtr)->username);
        cout << "Password: "; getline(cin, (userArray +
*jumlahPtr)->password);
        (*(userArray + *jumlahPtr)).role = "user";
        (*jumlahPtr)++;
        cout << "Registrasi berhasil!\n";
    }
}

```

```
    paus();
}
```

### C. Fitur Lihat Furniture

Fitur ini digunakan agar user dan admin dapat melihat furniture yang tersedia.

```
void tampilkanFurniture() {
    cout <<
    "=====
    << endl;
    cout << "| No |      Nama      |      Harga (Rp)      | Stok | Kategori |
    Bahan |" << endl;
    cout <<
    "=====
    << endl;
    for (int i = 0; i < jumlahFurniture; i++) {
        cout << "| " << setw(2) << i + 1 << " | " << setw(14) <<
        furnitures[i].nama << " | "
        << setw(16) << furnitures[i].harga << " | " << setw(4) <<
        furnitures[i].stok << " | "
        << setw(10) << furnitures[i].kategori.nama << " | " << setw(6)
        << furnitures[i].kategori.bahan << " |" << endl;
    }
    cout <<
    "=====
    << endl;
}
```

### D. Fitur Tambah Furniture

Fitur ini digunakan agar admin dapat menambah furniture yang baru dengan memasukkan nama furniture, harga furniture, stok furniture, kategori furniture, dan bahan furniture.

```
void tambahFurniture(int& jumlahRef) {
    if (jumlahRef >= MAX_FURNITURE) {
        cout << "Kapasitas penuh!\n";
        return;
    }
    cin.ignore();
    cout << "Nama\t\t: "; getline(cin, furnitures[jumlahRef].nama);
    cout << "Harga\t\t: "; cin >> furnitures[jumlahRef].harga;
    cout << "Stok\t\t: "; cin >> furnitures[jumlahRef].stok;
    cin.ignore();
    cout << "Kategori\t: "; getline(cin,
```



```
furnitures[jumlahRef].kategori.nama);
    cout << "Bahan\t\t: "; getline(cin,
furnitures[jumlahRef].kategori.bahan);
    jumlahRef++;
    cout << "Furniture berhasil ditambahkan!\n";
}
```

## E. Fitur Edit Furniture

Fitur ini digunakan agar admin dapat mengedit furniture dengan memasukkan indeks furniture. Setelah itu, admin dapat mengganti nama furniture, harga furniture, stok furniture, kategori furniture, dan bahan furniture.

```
void editFurniture(Furniture* furnitureArray) {
    tampilkanFurniture();
    int index;
    cout << "Masukkan nomor furniture yang ingin diedit: ";
    cin >> index;
    index--;
    if (index >= 0 && index < jumlahFurniture) {
        cin.ignore();
        cout << "Nama: "; getline(cin, (furnitureArray + index)->nama);
        cout << "Harga: "; cin >> (*(furnitureArray + index)).harga;
        cout << "Stok: "; cin >> furnitureArray[index].stok;
        cin.ignore();
        cout << "Kategori: "; getline(cin, (*(furnitureArray +
index)).kategori.nama);
        cout << "Bahan: "; getline(cin, (furnitureArray +
index)->kategori.bahan);
        cout << "Data berhasil diperbarui!\n";
    } else {
        cout << "Nomor furniture tidak valid!\n";
    }
}
```

## F. Fitur Hapus Furniture

Fitur ini digunakan agar admin dapat menghapus furniture dengan memasukkan indeks furniture.

```
void hapusFurniture(Furniture* furnitureArray, int* jumlahPtr) {
    tampilkanFurniture();
    int index;
    cout << "Masukkan nomor furniture yang ingin dihapus: ";
    cin >> index;
    index--;
```

```

    if (index >= 0 && index < *jumlahPtr) {
        for (int i = index; i < *jumlahPtr - 1; i++) {
            *(furnitureArray + i) = *(furnitureArray + i + 1);
        }
        (*jumlahPtr)--;
        cout << "Furniture berhasil dihapus!\n";
    } else {
        cout << "Nomor furniture tidak valid!\n";
    }
}
}

```

## G. Fitur Tampilkan Furniture berdasarkan Kategori

Fitur ini digunakan agar admin dapat mencari furniture berdasarkan kategori Furniture.

```

void cariFurnitureBerdasarkanBahan() {
    cin.ignore();
    string bahan;
    cout << "Masukkan bahan yang ingin dicari: "; getline(cin, bahan);
    bool ditemukan = false;
    cout <<
    "=====
    << endl;
    cout << "| No |      Nama      |      Harga (Rp)      | Stok | Kategori |
    Bahan |" << endl;
    cout <<
    "=====
    << endl;
    int nomor = 1;
    for (int i = 0; i < jumlahFurniture; i++) {
        if (furnitures[i].kategori.bahan == bahan) {
            ditemukan = true;
            cout << "| " << setw(2) << nomor++ << " | " << setw(14) <<
            furnitures[i].nama << " | "
                << setw(16) << furnitures[i].harga << " | " << setw(4) <<
            furnitures[i].stok << " | "
                << setw(10) << furnitures[i].kategori.nama << " | " <<
            setw(6) << furnitures[i].kategori.bahan << " |" << endl;
        }
    }
    if (!ditemukan) {
        cout << "|
        |" << endl;
    }
    cout <<
    "=====

```

```
<< endl;
}
```

## H. Fitur Urutkan berdasarkan Harga

Fitur ini digunakan agar admin dapat melihat furniture sesuai urutan berdasarkan harga.

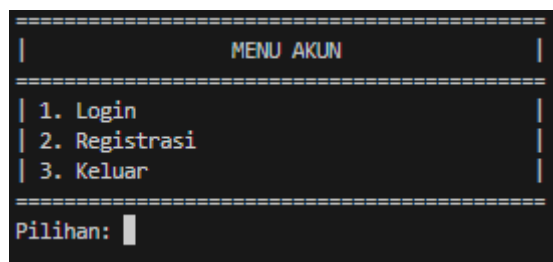
```
void urutkanFurnitureBerdasarkanHarga() {
    sort(furnitures, furnitures + jumlahFurniture, [](Furniture a, Furniture
b) {
        return a.harga < b.harga;
    });
    cout << "Data berhasil diurutkan berdasarkan harga (termurah ke
termahal)!\n";
    tampilkanFurniture();
}
```

## I. Fitur Hitung Total

Fitur ini digunakan agar admin dapat melihat semua stok furniture dan jumlah harga semua furniture dengan mengkalikan jumlah stok dengan harga furniture.

```
void hitungTotal(int& totalStok, int& totalNilai) {
    totalStok = 0;
    totalNilai = 0;
    for (int i = 0; i < jumlahFurniture; i++) {
        totalStok += furnitures[i].stok;
        totalNilai += furnitures[i].stok * furnitures[i].harga;
    }
    cout << "Total Stok Furniture    : " << totalStok << endl;
    cout << "Total Nilai Persediaan : Rp. " << totalNilai << endl;
}
```

## 4. Hasil Output



Gambar 4.1 Menu Akun

```

=====
                        LOGIN
=====
Username: admin
Password: admin053

```

Gambar 4.2 Menu Login

```

Username: Ripa
Password: 053

```

Gambar 4.3 Menu Register

```

=====
                        MENU UTAMA
=====
1. Tampilkan Semua Furniture
2. Kembali ke Menu Awal
=====
Pilihan:

```

Gambar 4.4 Menu User

```

=====
                        MENU UTAMA
=====
1. Tampilkan Semua Furniture
2. Tambah Furniture
3. Edit Furniture
4. Hapus Furniture
5. Tampilkan Furniture berdasarkan kategori
6. Cari Furniture berdasarkan bahan
7. Urutkan berdasarkan harga
8. Hitung total
9. Kembali ke Menu Awal
=====
Pilihan:

```

Menu 4.5 Menu Admin

No	Nama	Harga (Rp)	Stok	Kategori	Bahan
1	Meja Kayu	500000	10	Meja	Kayu
2	Kursi Rotan	250000	15	Kursi	Rotan
3	Lemari Jati	1500000	5	Lemari	Jati
4	Sofa Minimalis	2000000	3	Sofa	Kain
5	Rak Buku	300000	8	Rak	Kayu

Tekan enter untuk kembali ke menu...

Gambar 4.6 Menu Tampilkan Furniture

```

Nama      : Kursi Besi
Harga     : 150000
Stok      : 13
Kategori  : Kursi
Bahan     : Besi
Furniture berhasil ditambahkan!
Tekan enter untuk kembali ke menu...

```

Gambar 4.7 Menu Tambah Furniture

```

=====
| No | Nama | Harga (Rp) | Stok | Kategori | Bahan |
=====
| 1 | Meja Kayu | 500000 | 10 | Meja | Kayu |
| 2 | Kursi Rotan | 250000 | 15 | Kursi | Rotan |
| 3 | Lemari Jati | 1500000 | 5 | Lemari | Jati |
| 4 | Sofa Minimalis | 2000000 | 3 | Sofa | Kain |
| 5 | Rak Buku | 300000 | 8 | Rak | Kayu |
| 6 | Kursi Besi | 150000 | 13 | Kursi | Besi |
=====
Masukkan nomor furniture yang ingin diedit: 6
Nama: Meja Besi
Harga: 200000
Stok: 11
Kategori: Meja
Bahan: Besi

```

Gambar 4.8 Menu Edit Furniture

```

=====
| No | Nama | Harga (Rp) | Stok | Kategori | Bahan |
=====
| 1 | Meja Kayu | 500000 | 10 | Meja | Kayu |
| 2 | Kursi Rotan | 250000 | 15 | Kursi | Rotan |
| 3 | Lemari Jati | 1500000 | 5 | Lemari | Jati |
| 4 | Sofa Minimalis | 2000000 | 3 | Sofa | Kain |
| 5 | Rak Buku | 300000 | 8 | Rak | Kayu |
| 6 | Meja Besi | 200000 | 11 | Meja | Besi |
=====
Masukkan nomor furniture yang ingin dihapus: 5
Furniture berhasil dihapus!
Tekan enter untuk kembali ke menu...

```

Gambar 4.9 Menu Hapus Furniture

```

Masukkan kategori: Meja
=====
| No | Nama | Harga (Rp) | Stok | Kategori | Bahan |
=====
| 1 | Meja Kayu | 500000 | 10 | Meja | Kayu |
| 2 | Meja Besi | 200000 | 11 | Meja | Besi |
=====
Tekan enter untuk kembali ke menu...

```

Gambar 4.10 Menu Menampilkan Furniture berdasarkan Kategori

```

Masukkan bahan yang ingin dicari: Besi
=====
| No | Nama | Harga (Rp) | Stok | Kategori | Bahan |
=====
| 1 | Meja Besi | 200000 | 11 | Meja | Besi |
=====
Tekan enter untuk kembali ke menu...

```

Gambar 4.11 Menu Cari Furniture berdasarkan Bahan

```

Data berhasil diurutkan berdasarkan harga (termurah ke termahal)!
=====
| No | Nama | Harga (Rp) | Stok | Kategori | Bahan |
=====
| 1 | Meja Besi | 200000 | 11 | Meja | Besi |
| 2 | Kursi Rotan | 250000 | 15 | Kursi | Rotan |
| 3 | Meja Kayu | 500000 | 10 | Meja | Kayu |
| 4 | Lemari Jati | 1500000 | 5 | Lemari | Jati |
| 5 | Sofa Minimalis | 2000000 | 3 | Sofa | Kain |
=====
Tekan enter untuk kembali ke menu...

```

Gambar 4.12 Menu Mengurutkan Furniture berdasarkan Harga

```

Total Stok Furniture : 44
Total Nilai Persediaan : Rp. 24450000
Tekan enter untuk kembali ke menu...

```

Gambar 4.13 Menu Hitung Total

## 5. Git

### 1. Git Add

Git add berfungsi digunakan menandai file mana yang ingin kita lakukan perubahan. Gunakan git add . untuk menandai semua file.

```
PS C:\posttestapl\praktikum-apl\post-test\post-test-5> git add .
```

Gambar 5.1 Git Add

### 2. Git Commit

Git commit digunakan untuk melakukan perubahan pada file kita tandai dan memberi komentar.

```

PS C:\posttestapl\praktikum-apl\post-test\post-test-5> git commit -m "Finish PT5"
[main b5de813] Finish PT5
2 files changed, 320 insertions(+)
create mode 100644 post-test/post-test-5/2409106053-MuhammadRivaFachrodhiya-PT5.cpp
create mode 100644 post-test/post-test-5/2409106053-MuhammadRivaFachrodhiya-PT5.exe

```

Gambar 5.2 Git Commit

### 3. Git Push

Git push berfungsi untuk mengupload file yang kita tandai ke GitHub

```

PS C:\posttestapl\praktikum-apl\post-test\post-test-5> git push -u origin main
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 12 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 686.13 KiB | 4.57 MiB/s, done.
Total 6 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/MuhammadRivaFachrodhiya/praktikum-apl.git
887c2c8..b5de813 main -> main
branch 'main' set up to track 'origin/main'.

```

Gambar 5.3 Git Push