

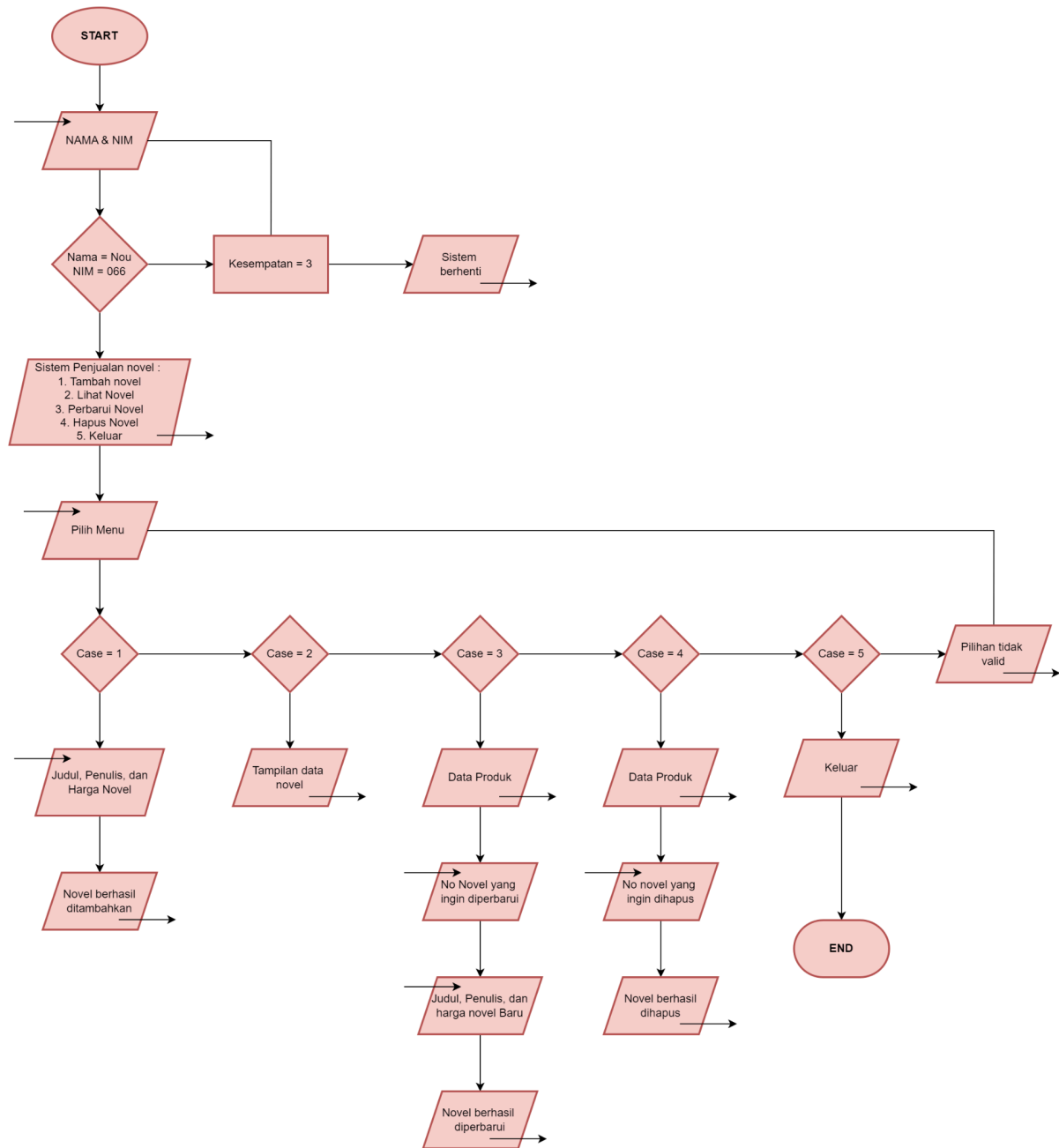
**LAPORAN PRAKTIKUM**  
**POSTTEST 5**  
**ALGORITMA PEMROGRAMAN LANJUT**



**Disusun oleh:**  
**Nou Julyanah Mazuwa**  
**B1 24**

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

# 1. Flowchart



## **2. Analisis Program**

### **Deskripsi Singkat Program**

Program ini dibuat untuk menjelaskan tentang sistem management penjualan novel dengan berbagai fungsi utama, seperti menambah, menampilkan, memperbarui, dan menghapus data novel. Sebelum mengakses sistem, pengguna harus melakukan login dengan memasukkan nama dan NIM yang benar.

### 3. Source Code

## Autentikasi Login

## Fungsi Lihat daftar novel

```
void tampilkanDaftarNovel(Novel daftar_novel[], int jumlah_novel) {
    if (jumlah_novel == 0) {
        cout << "Silahkan masukkan daftar novel terlebih dahulu.\n";
    } else {
        cout << "\nDaftar Novel:\n";
        cout <<
            "+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+\n";
            cout << "| No | Judul | Penulis | Tahun Terbit | Harga (Rp) |\n";
            cout <<
                "+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+\n";
                for (int i = 0; i < jumlah_novel; i++) {
                    cout << "| " << setw(2) << i + 1 << " | "
                        << left << setw(20) << daftar_novel[i].judul << " | "
                        << left << setw(20) << daftar_novel[i].penulis << " | "
                        << right << setw(12) << daftar_novel[i].tahun_terbit << " | "
                        << right << setw(12) << daftar_novel[i].harga << " |\n";
                }
                cout <<

```

```
"-----+-----+-----+-----+-----+-----+-----  
-----+\n";  
    }  
    cout << "\nTekan enter untuk kembali ke menu...";  
    cin.get();  
}
```

## Fungsi tambah novel

```
void tambahNovel(Novel daftar_novel[], int* jumlah_novel) {
    if (*jumlah_novel < MAX_NOVEL) {
        cout << "Masukkan judul novel: ";
        getline(cin, daftar_novel[*jumlah_novel].judul);

        cout << "Masukkan nama penulis: ";
        getline(cin, daftar_novel[*jumlah_novel].penulis);

        cout << "Masukkan tahun terbit: ";
        cin >> daftar_novel[*jumlah_novel].tahun_terbit;

        cout << "Masukkan harga: Rp. ";
        cin >> daftar_novel[*jumlah_novel].harga;
        cin.ignore();

        (*jumlah_novel)++;
        cout << "Novel berhasil ditambahkan!\n";
    } else {
        cout << "Daftar novel sudah penuh!\n";
    }
    cout << "\nTekan enter untuk kembali ke menu...";
    cin.get();
}
```

## Fungsi perbarui novel

```
void perbaruiNovel(Novel daftar_novel[], int jumlah_novel) {
    if (jumlah_novel == 0) {
        cout << "Belum ada novel untuk diperbarui.\n";
    } else {
        int index;
        cout << "Masukkan nomor novel yang ingin diperbarui: ";
        cin >> index;
        cin.ignore();

        if (index < 1 || index > jumlah_novel) {
            cout << "Nomor tidak valid!\n";
        } else {
            index--;
        }
    }
}
```

```

        cout << "Masukkan judul baru: ";
        getline(cin, daftar_novel[index].judul);

        cout << "Masukkan nama penulis baru: ";
        getline(cin, daftar_novel[index].penulis);

        cout << "Masukkan tahun terbit baru: ";
        cin >> daftar_novel[index].tahun_terbit;

        cout << "Masukkan harga baru: Rp. ";
        cin >> daftar_novel[index].harga;
        cin.ignore();

        cout << "Novel berhasil diperbarui!\n";
    }
}
cout << "\nTekan enter untuk kembali ke menu...";
cin.get();
}

```

### Fungsi hapus novel

```

void hapusNovel(Novel daftar_novel[], int* jumlah_novel) {
    if (*jumlah_novel == 0) {
        cout << "Belum ada novel untuk dihapus.\n";
    } else {
        int index;
        cout << "Masukkan nomor novel yang ingin dihapus: ";
        cin >> index;

        if (index < 1 || index > *jumlah_novel) {
            cout << "Nomor tidak valid!\n";
        } else {
            index--;
            for (int i = index; i < *jumlah_novel - 1; i++) {
                daftar_novel[i] = daftar_novel[i + 1];
            }
            (*jumlah_novel)--;
            cout << "Novel berhasil dihapus!\n";
        }
    }
    cin.ignore();
    cout << "\nTekan enter untuk kembali ke menu...";
    cin.get();
}

```

## Menu utama

```
int main() {
    if (!login()) {
        cout << "Anda gagal login 3 kali! Program berhenti.\n";
        return 0;
    }

    Novel daftar_novel[MAX_NOVEL] = {
        {"Bumi", "Tere Liye", 2017, 120000},
        {"Laskar Pelangi", "Andrea Hirata", 2016, 75000},
        {"Dilan 1990", "Pidi Baiq", 2020, 85000},
        {"Negeri 5 Menara", "Ahmad Fuadi", 2002, 95000},
        {"Hujan", "Tere Liye", 2015, 135000}
    };

    int jumlah_novel = 5;
    int pilihan;

    do {
        cls();
        cout << "\n=== Sistem Penjualan Novel ===\n";
        cout << "1. Lihat Daftar Novel\n";
        cout << "2. Tambah Novel\n";
        cout << "3. Perbarui Novel\n";
        cout << "4. Hapus Novel\n";
        cout << "5. Keluar\n";
        cout << "Masukkan pilihan anda: ";
        cin >> pilihan;
        cin.ignore();

        switch (pilihan) {
            case 1:
                cls();
                tampilkanDaftarNovel(daftar_novel, jumlah_novel);
                break;
            case 2:
                cls();
                tambahNovel(daftar_novel, &jumlah_novel);
                break;
            case 3:
                cls();
                perbaruiNovel(daftar_novel, jumlah_novel);
                break;
            case 4:
                cls();
                hapusNovel(daftar_novel, &jumlah_novel);
                break;
        }
    } while (pilihan != 5);
}
```

```
        case 5:
            cout << "Terima kasih telah menggunakan program ini.\n";
            break;
        default:
            cout << "Pilihan tidak valid!\n";
            cout << "\nTekan enter untuk kembali ke menu...";
            cin.get();
    }
} while (pilihan != 5);

return 0;
}
```



## 4. Tampilan Output

Menu utama

```
=== Sistem Penjualan Novel ===
1. Lihat Daftar Novel
2. Tambah Novel
3. Perbarui Novel
4. Hapus Novel
5. Keluar
Silahkan masukkan pilihan menu anda: █
```

Tampilan output jika memilih Menu 1

```
Daftar Novel:
+---+-----+-----+-----+-----+
| No | Judul           | Penulis           | Tahun Terbit | Harga (Rp) |
+---+-----+-----+-----+-----+
| 1  | Bumi            | Tere Liye         | 2017         | 120000     |
| 2  | Laskar Pelangi  | Andrea Hirata     | 2016         | 75000      |
| 3  | Dilan 1990      | Pidi Baiq         | 2020         | 85000      |
| 4  | Negeri 5 Menara | Ahmad Fuadi       | 2002         | 95000      |
| 5  | Hujan           | Tere Liye         | 2015         | 135000     |
+---+-----+-----+-----+-----+

Tekan enter untuk kembali ke menu... █
```

Tampilan output jika memilih Menu 2

```
Masukkan judul novel: rinjani
Masukkan nama penulis: tere liye
Masukkan tahun terbit: 2019
Masukkan harga: Rp. 128000
Novel telah berhasil ditambahkan!

Tekan enter untuk kembali ke menu... █
```

Tampilan output jika memilih menu 3

```
Masukkan nomor novel yang ingin diperbarui: 5
Masukkan judul baru: hujan
Masukkan nama penulis baru: tere liye
Masukkan tahun terbit baru: 2020
Masukkan harga baru: Rp. 123000
Novel berhasil diperbarui!

Tekan enter untuk kembali ke menu...
```

Tampilan output jika memilih menu 4

```
Masukkan nomor novel yang ingin dihapus: 5
Novel berhasil dihapus!

Tekan enter untuk kembali ke menu...
```

Tampilan output jika memilih menu 5

```
=== Sistem Penjualan Novel ===
1. Lihat Daftar Novel
2. Tambah Novel
3. Perbarui Novel
4. Hapus Novel
5. Keluar
Silahkan masukkan pilihan menu anda: 5
Terima kasih telah menggunakan sistem ini.
PS C:\pratikum-apl\post-test\post-test-4>
```

## 5. Git

### 4.1 git add

```
MyBook Hype@Mazuwa MINGW64 /c/pratikum-apl (main)
$ git add .
```

Git add berfungsi untuk menambahkan file atau perubahan ke dalam staging area sebelum di commit.

### 4.2 git commit

```
MyBook Hype@Mazuwa MINGW64 /c/pratikum-apl (main)
$ git commit -m "pt 5 done"
[main 7aaa0dd] pt 5 done
6 files changed, 407 insertions(+)
create mode 100644 kelas b1 2024/Pertemuan-5/mahasiswa.cpp
create mode 100644 kelas b1 2024/Pertemuan-5/main.cpp
create mode 100644 kelas b1 2024/Pertemuan-5/main.exe
create mode 160000 post-test/post-test-2
create mode 100644 post-test/post-test/post-test-5/2409106066-NouJulyanahMazuwa-PT-5.cpp
create mode 100644 post-test/post-test/post-test-5/2409106066-NouJulyanahMazuwa-PT-5.exe
```

Git commit berfungsi untuk menyimpan perubahan yang sudah ditambahkan ke staging area ke dalam repository lokal git.

### 4.3 git push origin main

```
MyBook Hype@Mazuwa MINGW64 /c/pratikum-apl (main)
$ git push -u origin main
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 8 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 2.85 MiB | 121.46 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/Njlynhmzw/praktikum-apl.git
 e0fb002..7254174 main -> main
branch 'main' set up to track 'origin/main'.
```

git push origin main berfungsi untuk mengunggah(push) perubahan dari branch main di repository lokal ke repository github