

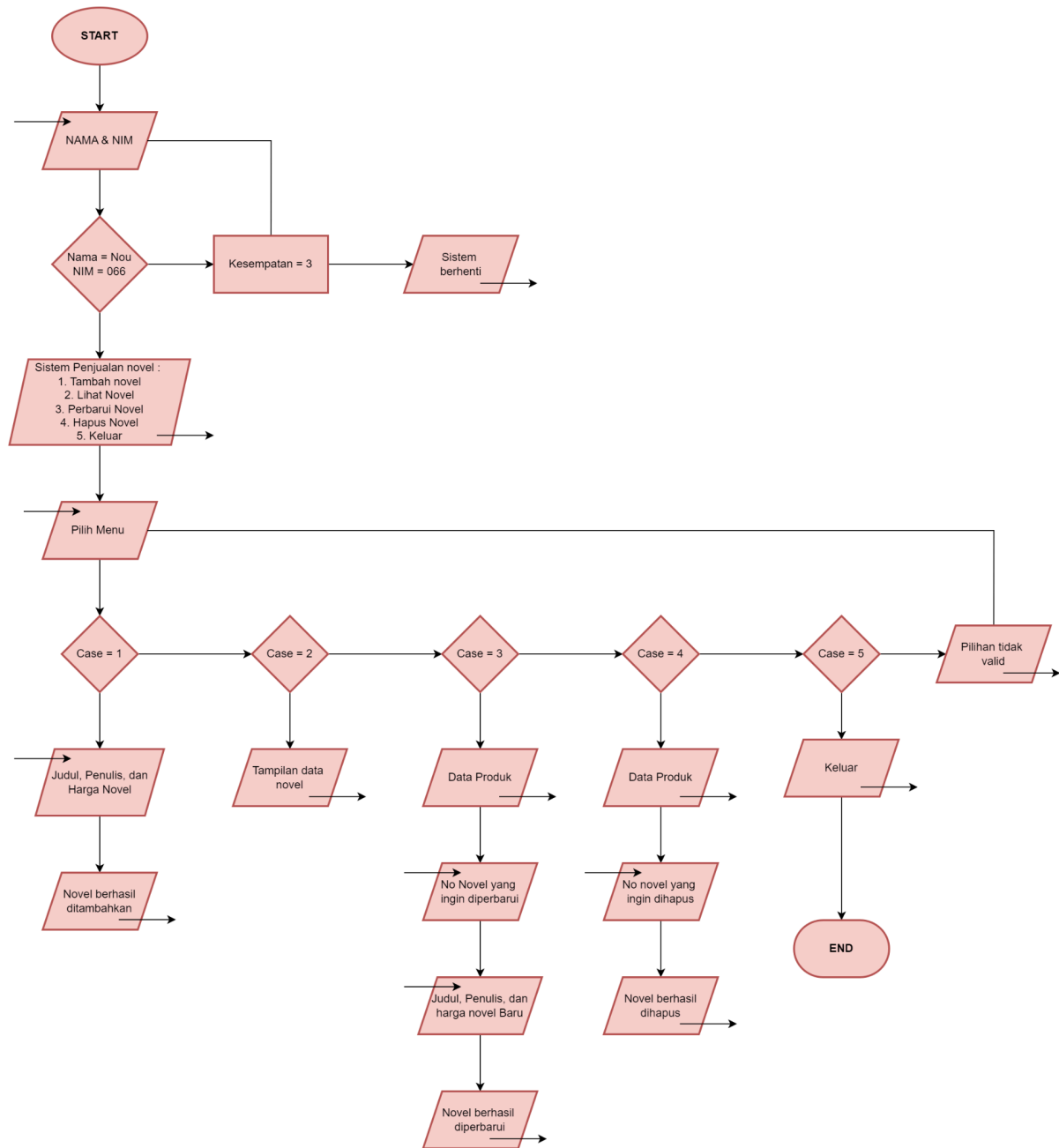
LAPORAN PRAKTIKUM
POSTTEST 3
ALGORITMA PEMROGRAMAN LANJUT



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

```
int login() {
    string nama, NIM;
    int kesempatan = 3;

    while (kesempatan > 0) {
        cout << "Silahkan masukkan Nama anda: ";
        cin >> nama;
        cout << "Silahkan masukkan NIM anda: ";
        cin >> NIM;

        if (nama == "nou" && NIM == "066") {
            return 1;
        }

        kesempatan--;
        cout << "Login anda telah gagal! Sisa percobaan login: " <<
kesempatan << "\n";
    }
    return 0;
}
```

CRUD

```
int main() {
    if (!login()) {
        cout << "Anda telah gagal login sebanyak 3 kali! Sistem
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 << "Silahkan masukkan pilihan menu anda: ";
cin >> pilihan;
cin.ignore();
```

case 2:

```
    cls();
    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 telah berhasil ditambahkan!\n";
    } else {
        cout << "Daftar novel telah penuh!\n";
    }
    cout << "\nTekan enter untuk kembali ke menu...";
    cin.get();
    break;
```

case 3:

```
    cls();
    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 yang anda masukkan 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();
break;

```

```

case 4:

    cls();
    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";
        }
    }
    cout << "\nTekan enter untuk kembali ke menu...";
    cin.get();
    break;

```

```

case 5:

    cout << "Terima kasih telah menggunakan sistem ini.\n";
    break;

default:
    cout << "Pilihan tidak valid! Silahkan mencoba kembali.\n";
    cout << "\nTekan enter untuk kembali ke menu...";
    cin.get();

```

4. 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 "up code"
[main 2580cc5] up code
7 files changed, 380 insertions(+)
create mode 100644 .vscode/settings.json
create mode 100644 kelas b1 2024/pertemuan-3/crud.cpp
create mode 100644 kelas b1 2024/pertemuan-3/crud.exe
create mode 100644 kelas b1 2024/pertemuan-3/main.cpp
create mode 100644 kelas b1 2024/pertemuan-3/main.exe
create mode 100644 post-test/post-test-3/2409106066-NouJulyanahMazuwa-PT-3.cpp
create mode 100644 post-test/post-test-3/2409106066-NouJulyanahMazuwa-PT-3.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: 17, done.
Counting objects: 100% (17/17), done.
Delta compression using up to 8 threads
Compressing objects: 100% (13/13), done.
Writing objects: 100% (14/14), 3.83 MiB | 22.92 MiB/s, done.
Total 14 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), done.
To https://github.com/Njlynhmzw/praktikum-apl.git
   2680ebd..2580cc5  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