

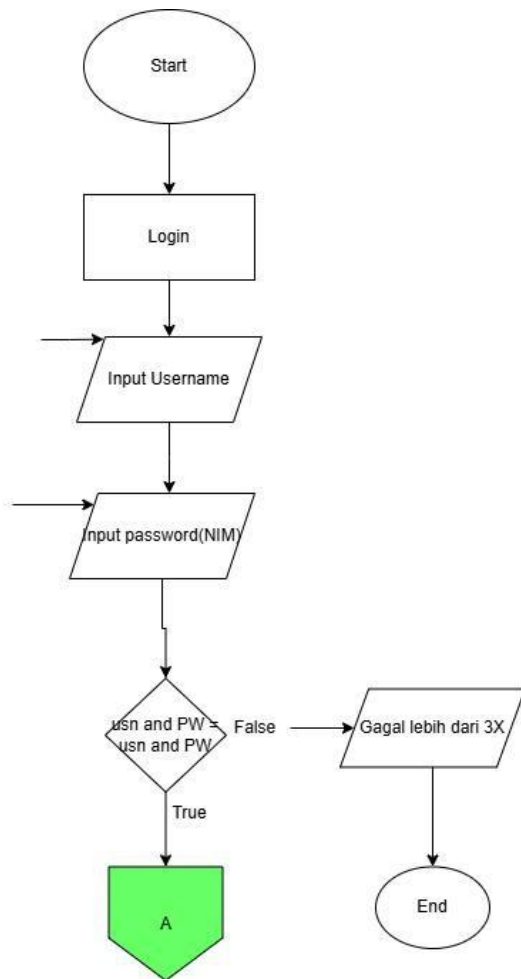
LAPORAN PRAKTIKUM
POSTTEST 3
ALGORITMA PEMROGRAMAN LANJUT



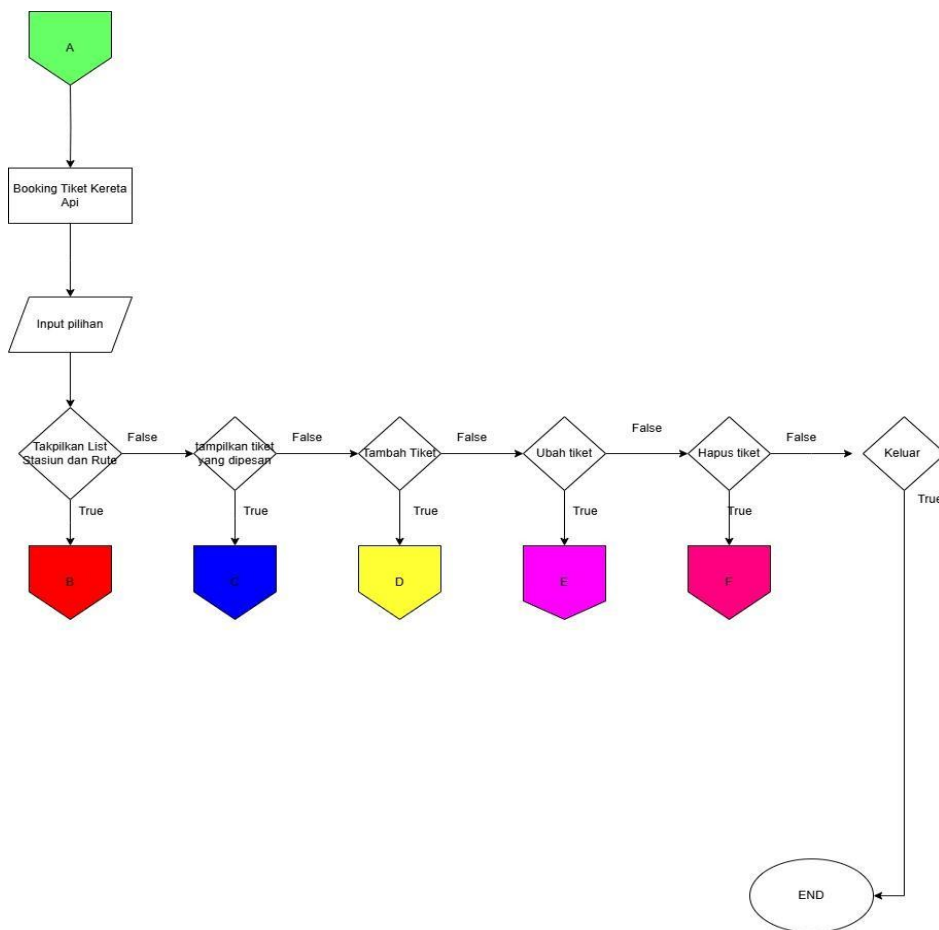
Disusun oleh:
Sayid Annashir Ikhwan (2409106097)
Kelas (C '24)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

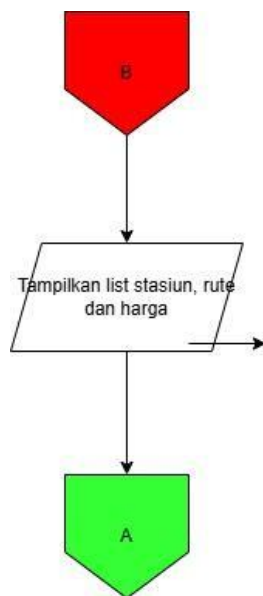
1. Flowchart



Gambar 1.1. Flowchart



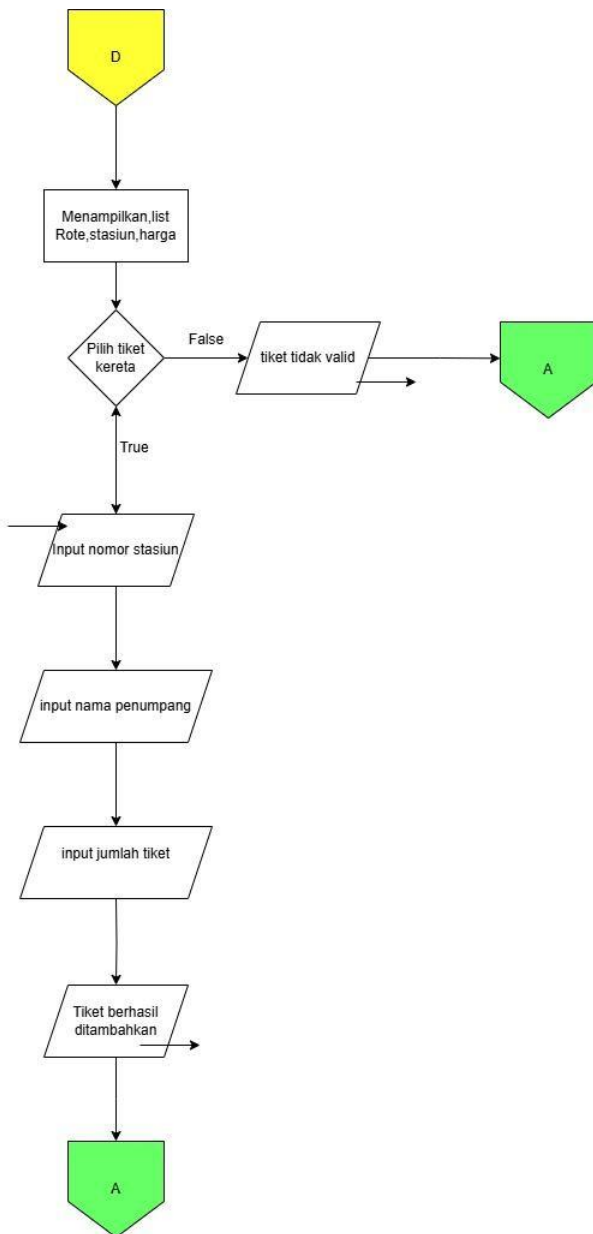
Gambar 1.2. <flowchart tampilan menu utama>



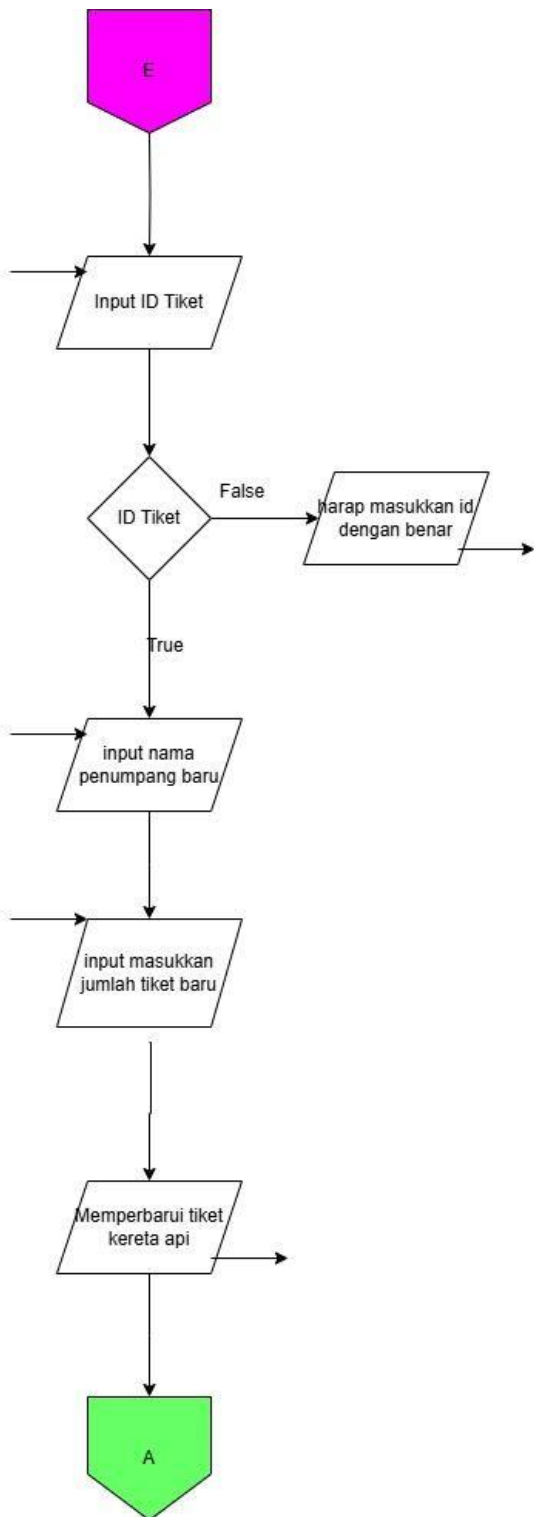
Gambar 1.3. <manampilkan list stasiun>



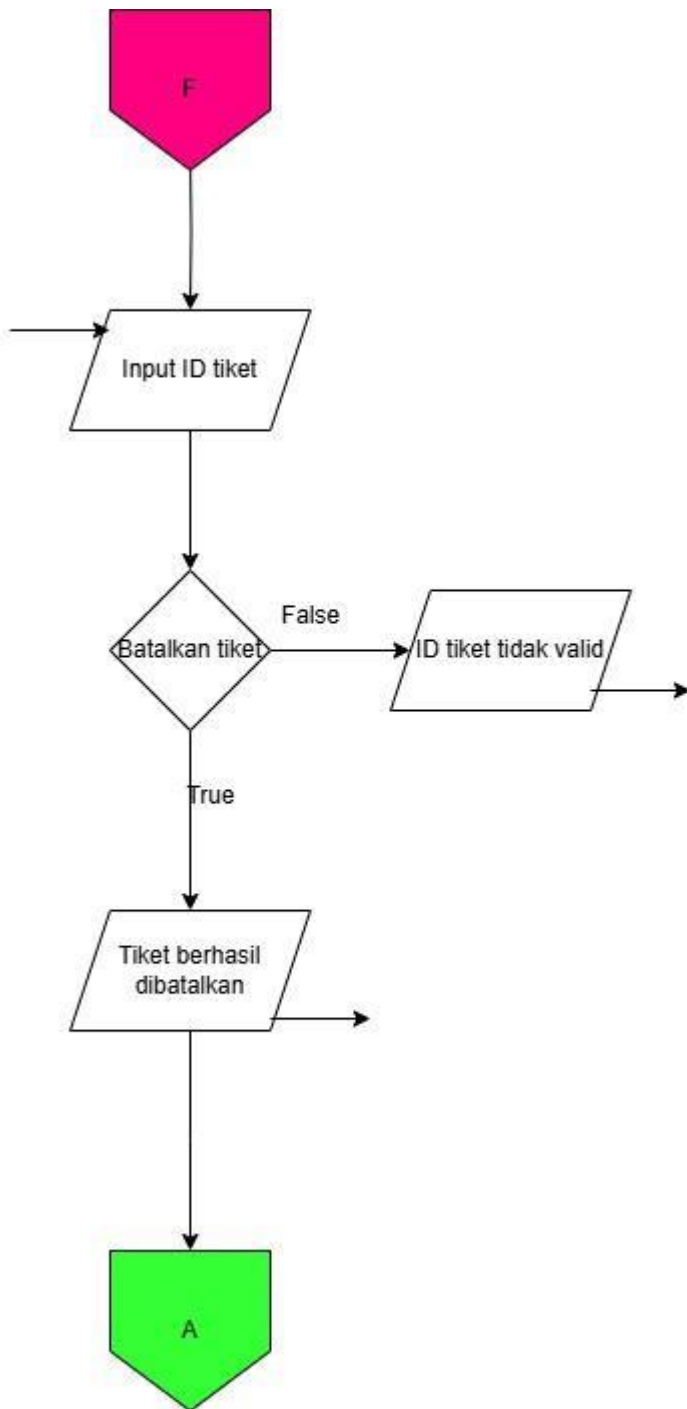
Gambar 1.4 menampilkan tiket yang sudah dipesan



Gambar 1.5 Tambah tiket



Gambar 1.6 ubah tiket



Gambar 1.7 Hapus tiket

2. Analisis Program

2.1 Deskripsi Singkat Program

Program ini adalah sebuah sistem pemesanan tiket kereta api berbasis konsol yang memungkinkan pengguna untuk melakukan pemesanan tiket, melihat daftar stasiun dan rute, serta mengelola tiket yang telah dipesan. Program ini dirancang untuk memenuhi kebutuhan pengguna dalam memesan tiket kereta api dengan antarmuka yang sederhana dan mudah digunakan. Dalam program ini, struct digunakan untuk menyimpan informasi tentang tiket yang dipesan oleh user. **Struct** ini menggantikan 3 array terpisah (namaPenumpang, stasiunDipilih, dan jumlahTiket) menjadi satu struktur data yang lebih terorganisir.

2.2 Penjelasan Alur & Algoritma

Alur Program:

1. Login :

- User diminta memasukkan nama dan NIM.
- Jika nama dan NIM sesuai, user berhasil login dan masuk ke menu utama.
- Jika salah, user diberi 3 kesempatan. Jika gagal 3 kali, program berhenti.

2. Menu Utama:

- Setelah login berhasil, program menampilkan menu utama dengan 6 opsi:
 1. Tampilkan List Stasiun dan Rute.
 2. Tampilkan Tiket yang Dipesan.
 3. Tambah Tiket.
 4. Ubah Tiket.
 5. Hapus Tiket.
 6. Keluar.
- Program akan berulang sampai user memilih opsi "Keluar".

3. CRUD:

- Create (Tambah Tiket): User dapat menambahkan tiket baru dengan memilih stasiun, memasukkan nama penumpang, dan jumlah tiket.
- Read (Tampilkan Tiket): Menampilkan semua tiket yang sudah dipesan.
- Update (Ubah Tiket): User dapat mengubah informasi tiket yang sudah dipesan.
- Delete (Hapus Tiket): User dapat menghapus tiket yang sudah dipesan.

4. Keluar:

- Jika user memilih opsi "Keluar", program akan menampilkan pesan terima kasih dan berhenti.

3. Source Code

```
// Definisi struct untuk menyimpan data tiket
struct Tiket {
    string namaPenumpang;
    int stasiunDipilih;
    int jumlahTiket;
    int totalHarga;
};
```

Gambar 3.1 (fitur struct)

```
string namaUser, nimUser;
int loginAttempts[1] = {0};
int loggedIn[1] = {0}; // Menggunakan array untuk loggedIn (0 = false, 1 = true)

// Login System
while (loginAttempts[0] < 3 && loggedIn[0] == 0) {
    cout << "Masukkan Nama: ";
    getline(cin, namaUser);
    cout << "Masukkan NIM: ";
    getline(cin, nimUser);
    if (namaUser == "Sayid Annashir Ikhwan" && nimUser == "2409106097") {
        loggedIn[0] = 1; // Login berhasil
        cout << "Hore Login berhasil!" << endl;
    } else {
        loginAttempts[0]++;
        cout << "Maaf Login gagal. Percobaan " << loginAttempts[0] << " dari 3." << endl;
    }
}
if (loggedIn[0] == 0) {
    cout << "Anda telah gagal login 3 kali. Program berhenti." << endl;
    return 0;
}
```

Gambar 3.2 (Fitur login)

```

int pilihan, index;
do {
    system("cls"); // Membersihkan layar
    cout << "======" << endl;
    cout << "Selamat datang di Program Booking Tiket Kereta Api" << endl;
    cout << "======" << endl;
    cout << "1. Tampilkan List Stasiun dan Rute" << endl;
    cout << "2. Tampilkan Tiket yang Dipesan" << endl;
    cout << "3. Tambah Tiket" << endl;
    cout << "4. Ubah Tiket" << endl;
    cout << "5. Hapus Tiket" << endl;
    cout << "6. Keluar" << endl;
    cout << "Pilihan: ";
    cin >> pilihan;
    cin.ignore(); // Membersihkan buffer
}

```

Gambar 3.3 (menu utama)

```

switch (pilihan) {
    case 1: // Tampilkan List Stasiun dan Rute
        cout << "Daftar Stasiun dan Rute:" << endl;
        cout << "-----" << endl;
        for (int i = 0; i < JUMLAH_STASIUN; i++) {
            cout << "Stasiun: " << stasiun[i] << endl;
            cout << "Rute: " << rute[i] << endl;
            cout << "Jumlah Tiket Tersedia: " << jumlahTiketStasiun[i] << endl;
            cout << "Harga: Rp " << harga[i] << endl;
            cout << "-----" << endl;
        }
        break;
}

```

Gambar 3.4 (daftar list stasiun dan rute)

```

case 2: // Tampilkan Tiket yang Dipesan
    if (panjang[0] == 0) {
        cout << "Belum ada tiket yang dipesan." << endl;
    } else {
        cout << "Daftar Tiket yang Dipesan:" << endl;
        cout << "-----" << endl;
        for (int i = 0; i < panjang[0]; i++) {
            cout << "Tiket ke-" << i + 1 << ":" << endl;
            cout << "Nama Penumpang: " << tiket[i].namaPenumpang << endl;
            cout << "Stasiun: " << stasiun[tiket[i].stasiunDipilih] << endl;
            cout << "Rute: " << rute[tiket[i].stasiunDipilih] << endl;
            cout << "Jumlah Tiket: " << tiket[i].jumlahTiket << endl;
            cout << "Total Harga: Rp " << tiket[i].totalHarga << endl;
            cout << "-----" << endl;
        }
    }
    break;
}

```

Gambar 3.5 (tampilan tiket yang dipesan)

```
case 3: // Tambah Tiket
    if (panjang[0] < MAX_TIKET) {
        cout << "Pilih Stasiun:" << endl;
        for (int i = 0; i < JUMLAH_STASIUN; i++) {
            cout << i + 1 << ". " << stasiun[i] << " (" << rute[i] << ")" << endl;
        }
        cout << "Masukkan nomor stasiun: ";
        cin >> index;
        cin.ignore(); // Membersihkan buffer

        if (index > 0 && index <= JUMLAH_STASIUN) {
            if (jumlahTiketStasiun[index - 1] > 0) {
                cout << "Masukkan Nama Penumpang: ";
                getline(cin, tiket[panjang[0]].namaPenumpang);
                cout << "Masukkan Jumlah Tiket: ";
                cin >> tiket[panjang[0]].jumlahTiket;
                cin.ignore(); // Membersihkan buffer
                if (tiket[panjang[0]].jumlahTiket <= jumlahTiketStasiun[index - 1]) {
                    tiket[panjang[0]].stasiunDipilih = index - 1;
                    tiket[panjang[0]].totalHarga = tiket[panjang[0]].jumlahTiket * harga[index - 1];
                    jumlahTiketStasiun[index - 1] -= tiket[panjang[0]].jumlahTiket;
                    panjang[0]++;
                    cout << "Tiket berhasil ditambahkan!" << endl;
                } else {
                    cout << "Jumlah tiket tidak mencukupi." << endl;
                }
            } else {
                cout << "Tiket untuk stasiun ini sudah habis." << endl;
            }
        } else {
            cout << "Nomor stasiun tidak benar. Harap masukkan yang betul." << endl;
        }
    } else {
        cout << "Kapasitas tiket penuh. Tidak bisa menambah tiket lagi." << endl;
    }
    break;
```

Gambar 3.6 (tambah tiket)

```
case 4: // Ubah Tiket
    if (panjang[0] == 0) {
        cout << "Belum ada tiket untuk diubah." << endl;
    } else {
        cout << "Daftar Tiket yang Dipesan:" << endl;
        cout << "-----" << endl;
        for (int i = 0; i < panjang[0]; i++) {
            cout << "Tiket ke-" << i + 1 << ":" << endl;
            cout << "Nama Penumpang: " << tiket[i].namaPenumpang << endl;
            cout << "Stasiun: " << stasiun[tiket[i].stasiunDipilih] << endl;
            cout << "Rute: " << rute[tiket[i].stasiunDipilih] << endl;
```

```

        cout << "Jumlah Tiket: " << tiket[i].jumlahTiket << endl;
        cout << "Total Harga: Rp " << tiket[i].totalHarga << endl;
        cout << "-----" << endl;
    }
    cout << "Masukkan nomor tiket yang akan diubah: ";
    cin >> index;
    cin.ignore(); // Membersihkan buffer

    if (index > 0 && index <= panjang[0]) {
        cout << "Masukkan Nama Penumpang baru: ";
        getline(cin, tiket[index - 1].namaPenumpang);
        cout << "Masukkan Jumlah Tiket baru: ";
        cin >> tiket[index - 1].jumlahTiket;
        cin.ignore(); // Membersihkan buffer
        tiket[index - 1].totalHarga = tiket[index - 1].jumlahTiket * harga[tiket[index - 1].stasiunDipilih];
        cout << "Tiket berhasil diubah!" << endl;
    } else {
        cout << "Nomor tiket tidak valid." << endl;
    }
}
break;

```

Gambar 3.7 (ubah tiket)

```

case 5: // Hapus Tiket
    if (panjang[0] == 0) {
        cout << "Belum ada tiket untuk dihapus." << endl;
    } else {
        cout << "Daftar Tiket yang Dipesan:" << endl;
        cout << "-----" << endl;
        for (int i = 0; i < panjang[0]; i++) {
            cout << "Tiket ke-" << i + 1 << ":" << endl;
            cout << "Nama Penumpang: " << tiket[i].namaPenumpang << endl;
            cout << "Stasiun: " << stasiun[tiket[i].stasiunDipilih] << endl;
            cout << "Rute: " << rute[tiket[i].stasiunDipilih] << endl;
            cout << "Jumlah Tiket: " << tiket[i].jumlahTiket << endl;
            cout << "Total Harga: Rp " << tiket[i].totalHarga << endl;
            cout << "-----" << endl;
        }
        cout << "Masukkan nomor tiket yang akan dihapus: ";
        cin >> index;
        cin.ignore(); // Membersihkan buffer

        if (index > 0 && index <= panjang[0]) {
            jumlahTiketStasiun[tiket[index - 1].stasiunDipilih] -= tiket[index - 1].jumlahTiket;
            for (int i = index - 1; i < panjang[0] - 1; i++) {
                tiket[i] = tiket[i + 1];
            }
            panjang[0]--;
            cout << "Tiket berhasil dihapus!" << endl;
        } else {
            cout << "Nomor tiket tidak valid. Harap masukkan dengan benar." << endl;
        }
    }
}
break;

```

Gambar 3.8 (Hapus tiket)

```
ase 6: // Keluar
        cout << "Terima kasih sudah menggunakan program, sampai berjumpa kembali!" << endl;
        break;

    default:
        cout << "Pilihan tidak valid. Silakan coba lagi." << endl;
        break;
    }
    if (pilihan != 6) {
        cout << "Tekan Enter untuk melanjutkan...";
        cin.ignore();
    }
} while (pilihan != 6);
return 0;
}
```

Gambar 3.9 (logout/keluar)

4. Uji Coba dan Hasil Output

4.1

```
Maaf Login gagal. Percobaan 1 dari 3.
Masukkan Nama: asd
Masukkan NIM: asd
Maaf Login gagal. Percobaan 2 dari 3.
Masukkan Nama: asd
Masukkan NIM: asd
Maaf Login gagal. Percobaan 3 dari 3.
Anda telah gagal login 3 kali. Program berhenti.
PS C:\Users\LOQ>
```

Gambar 4.1

```

=====
Selamat datang di Program Booking Tiket Kereta Api
=====
1. Tampilkan List Stasiun dan Rute
2. Tampilkan Tiket yang Dipesan
3. Tambah Tiket
4. Ubah Tiket
5. Hapus Tiket
6. Keluar
Pilihan: |

```

Gambar 4.2

```

Daftar Stasiun dan Rute:
-----
Stasiun: Stasiun UNMUL
Rute: Samarinda-Balikpapan
Jumlah Tiket Tersedia: 10
Harga: Rp 500000
-----
Stasiun: Stasiun AI
Rute: Samarinda-Tenggarong
Jumlah Tiket Tersedia: 10
Harga: Rp 250000
-----
Stasiun: Stasiun ISO
Rute: Balikpapan-Berau
Jumlah Tiket Tersedia: 10
Harga: Rp 350000
-----
Stasiun: Stasiun APL
Rute: Balikpapan-Tanah Grogot
Jumlah Tiket Tersedia: 10
Harga: Rp 150000
-----
Stasiun: Stasiun Masjaya
Rute: Samarinda-Bontang
Jumlah Tiket Tersedia: 10
Harga: Rp 300000
-----
Tekan Enter untuk melanjutkan...|

```

Gambar 4.3

```

Daftar Tiket yang Dipesan:
-----
Tiket ke-1:
Nama Penumpang: Deni
Stasiun: Stasiun AI
Rute: Samarinda-Tenggarong
Jumlah Tiket: 5
Total Harga: Rp 1250000
-----
Tekan Enter untuk melanjutkan...|

```

Gambar 4.4

```
Tiket ke-1:
Nama Penumpang: Deni
Stasiun: Stasiun AI
Rute: Samarinda-Tenggarong
Jumlah Tiket: 5
Total Harga: Rp 1250000
-----
Masukkan nomor tiket yang akan dihapus: 1
Tiket berhasil dihapus!
Tekan Enter untuk melanjutkan...|
```

Gambar 4.5

5. GIT

```
LOQ@LAPTOP-4B4VUDGG MINGW64 ~/Desktop/GITHUB/Praktikum-Apl (main|MERGING)
$ git add .

LOQ@LAPTOP-4B4VUDGG MINGW64 ~/Desktop/GITHUB/Praktikum-Apl (main|MERGING)
$ git commit -m "post-test-3"
[main 9ba4129] post-test-3

LOQ@LAPTOP-4B4VUDGG MINGW64 ~/Desktop/GITHUB/Praktikum-Apl (main)
$ git commit -m "Post-test-3"
On branch main
Your branch is ahead of 'origin/main' by 2 commits.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean

LOQ@LAPTOP-4B4VUDGG MINGW64 ~/Desktop/GITHUB/Praktikum-Apl (main)
$ git push
Enumerating objects: 22, done.
Counting objects: 100% (22/22), done.
Delta compression using up to 20 threads
Compressing objects: 100% (13/13), done.
Writing objects: 100% (15/15), 1.65 MiB | 952.00 KiB/s, done.
Total 15 (delta 4), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (4/4), completed with 2 local objects.
To https://github.com/wizker12/Praktikum-Apl.git
   1d2a16d..9ba4129  main -> main

LOQ@LAPTOP-4B4VUDGG MINGW64 ~/Desktop/GITHUB/Praktikum-Apl (main)
$ |
```

1. git add .

- Perintah ini digunakan untuk menambahkan semua perubahan (file yang baru dibuat, diubah, atau dihapus) ke dalam staging area.

- Titik (`.`) menunjukkan bahwa semua file yang ada dalam direktori kerja akan ditambahkan.

2. git commit -m "Posttest 3"

- Perintah ini membuat commit dengan pesan `"Posttest 3"`, yang mendokumentasikan perubahan yang telah

ditambahkan ke staging area.

- Commit ini menyimpan snapshot dari perubahan dalam repository lokal.

3. git push

- Perintah ini mengunggah commit yang telah dibuat ke repository di GitHub pada branch `main`.

- `-u` (atau `--set-upstream`) digunakan untuk menghubungkan branch lokal `main` dengan branch `main` di repository remote (`origin`).

- `origin` adalah alias default untuk repository remote yang dikloning atau diatur sebelumnya.