

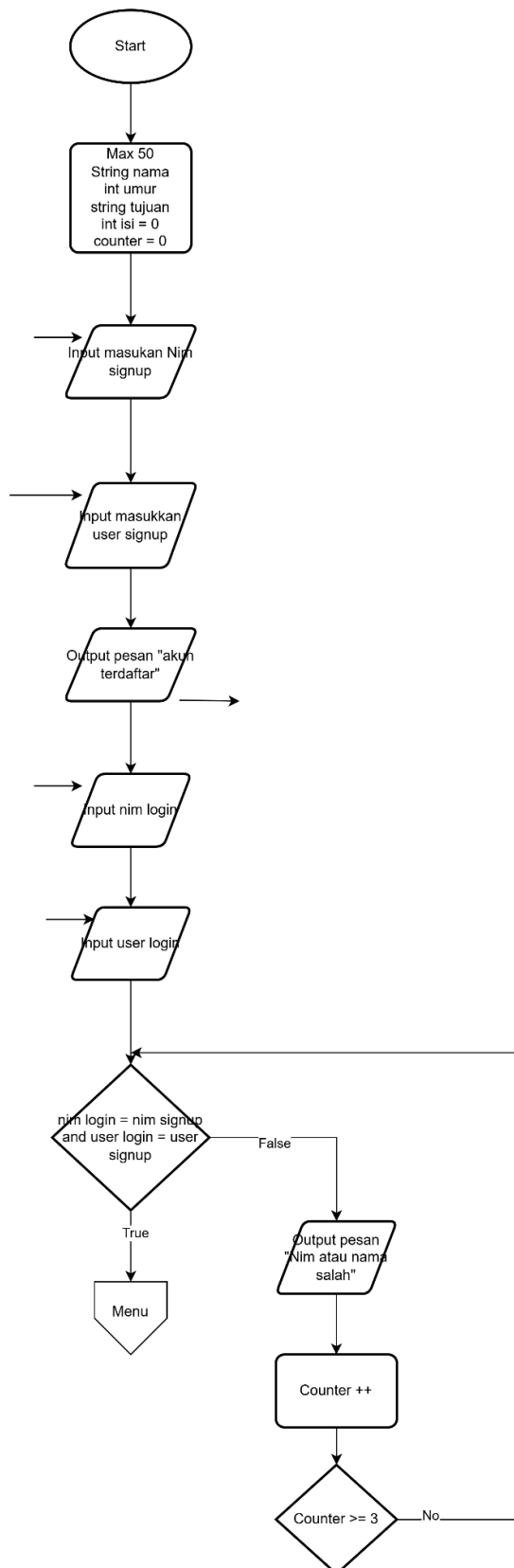
LAPORAN PRAKTIKUM
POSTTEST (2)
ALGORITMA PEMROGRAMAN LANJUT



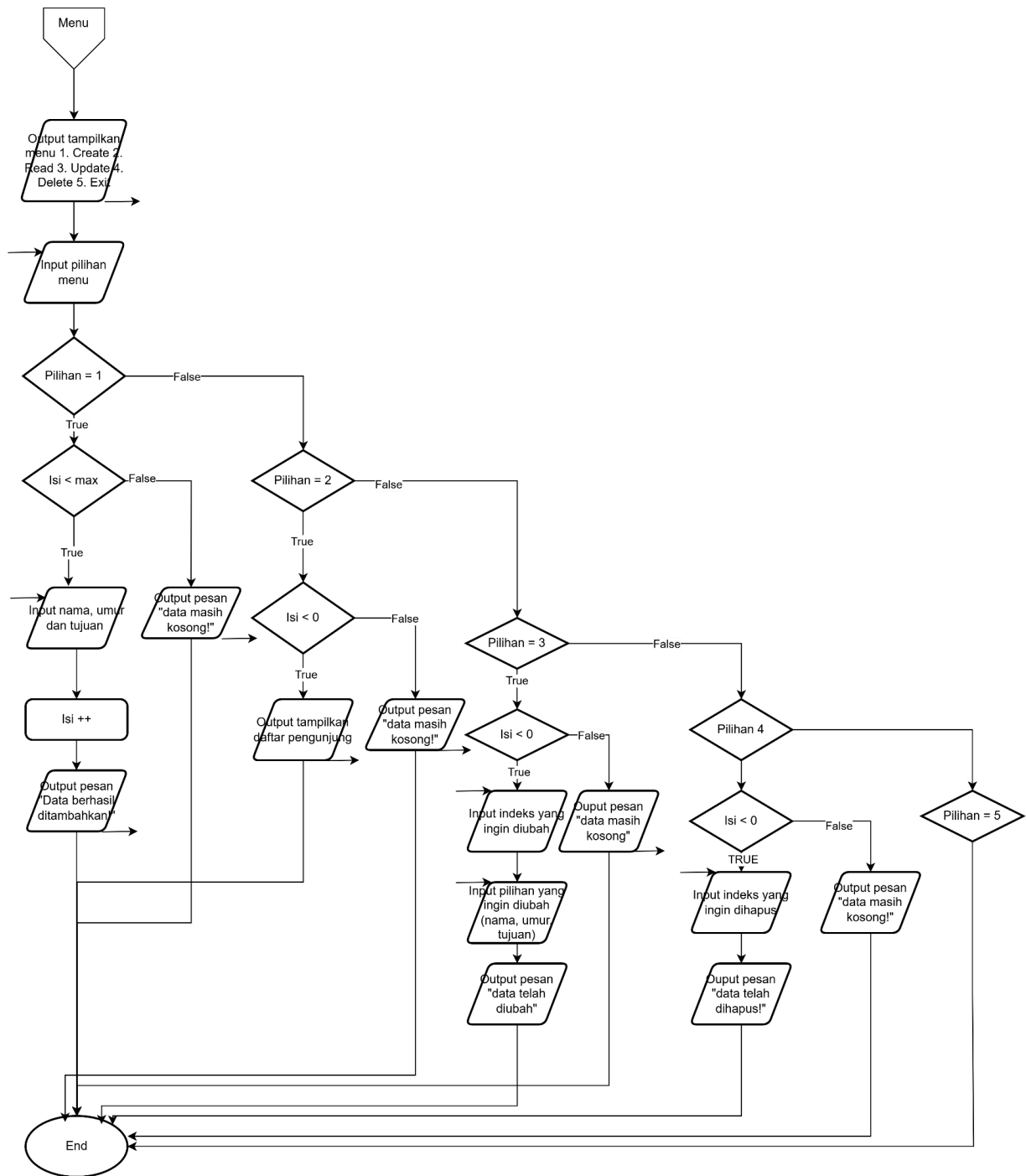
Disusun oleh:
Richard Dante Gunawan (2409106061)
Kelas (B1 '24)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

1. Flowchart



Gambar 1.1 Flowchart login dan signup



Gambar 1.2 Flowchart menu utama

2. Analisis Program

2.1 Deskripsi Singkat Program

Program manajemen data pengunjung mall dibuat dengan tujuan mensimulasikan konsep CRUD seperti menambah data pengunjung, membaca data pengunjung, mengubah data pengunjung serta menghapus data pengunjung.

2.2 Penjelasan Alur & Algoritma

Algoritma program manajemen data pengunjung mall

1. Pendaftaran akun
 - Pengguna akan diminta untuk memasukkan NIM dan Nama untuk mendaftar akun.
 - Data ini disimpan dalam variabel nimsignup dan usersignup.
 - Program menampilkan pesan bahwa akun telah terdaftar.
2. Login akun
 - Pengguna harus memasukkan kembali NIM dan Nama yang telah didaftarkan.
 - Jika data cocok, login berhasil dan pengguna masuk ke menu utama.
 - Jika salah, pengguna diberi kesempatan hingga 3 kali percobaan. Jika masih salah, program akan berhenti.
3. Menu utama

Setelah berhasil login, program akan menampilkan 5 menu utama, yaitu:

- 1) Tambah data pengunjung
 - Pengguna memasukkan Nama, Umur, dan Tujuan Kunjungan.
 - Data disimpan dalam array nama, umur, dan tujuan.
 - Jika daftar sudah penuh, program akan menampilkan peringatan.
- 2) Baca data pengunjung
 - Program menampilkan daftar pengunjung yang sudah tersimpan.
 - Jika belum ada data, program menampilkan pesan “Daftar pengunjung masih kosong!”.
- 3) Update data pengunjung
 - Pengguna memilih nomor indeks pengunjung yang ingin diperbarui.

- Program memberikan tiga opsi pembaruan yaitu nama, umur, tujuan kunjungan
- Jika indeks tidak valid, program akan menampilkan peringatan.

4) Hapus data pengunjung

- Pengguna memilih nomor indeks pengunjung yang ingin dihapus.
- Jika indeks valid, data tersebut dihapus dengan cara menggeser data setelahnya ke indeks sebelumnya.
- Jika daftar kosong, program menampilkan pesan “Tidak ada data pengunjung yang bisa dihapus!”

5) exit

- Program berhenti.

3. Source Code

A. Signup

Fitur ini digunakan untuk mendaftarkan akun pengguna menggunakan nim dan nama.

Source Code:

```
cout << "=== Signup Akun ===" << endl;
cout << "Silahkan masukkan NIM anda: ";
cin >> nimsignup;
cin.ignore();

cout << "Silahkan masukkan nama anda: ";
getline(cin, usersignup);

cout << "\nAkun berhasil terdaftar!\n" << endl;
```

B. Login

Fitur ini digunakan untuk memvalidasi akun pengguna. Pengguna akan diberi kesempatan sebanyak 3 kali untuk mencoba.

Source Code:

```
int counter = 0;
while (counter < 3) {
    cout << "=== Login Akun ===" << endl;
    cout << "Silahkan masukkan NIM anda: ";
    cin >> nimlogin;
    cin.ignore();

    cout << "Silahkan masukkan nama Anda: ";
    getline(cin, userlogin);

    if (nimsignup == nimlogin && usersignup == userlogin) {
        cout << "\nLogin berhasil! Selamat datang, " <<
usersignup << "!\n" << endl;
        break;
    } else {
        cout << "\nLogin gagal! NIM atau Nama tidak sesuai.\n"
```

```

<< endl;
        counter++;
    }
}

    if (counter == 3) {
        cout << "Anda telah mencoba login sebanyak 3 kali. Program
berhenti." << endl;
        return 0;
    }
}

```

C. Menu

Fitur ini digunakan untuk menampilkan menu utama yang terdiri dari tambah data, baca data, update data, hapus data dan exit.

Source Code:

```

    cout << "\n==== Silahkan pilih aksi yang ingin dilakukan! ==== "
<< endl;
        cout << "1. Tambah Data Pengunjung" << endl;
        cout << "2. Lihat Data Pengunjung" << endl;
        cout << "3. Perbarui Data Pengunjung" << endl;
        cout << "4. Hapus Data Pengunjung" << endl;
        cout << "5. Exit" << endl;
        cout << "Pilih menu: ";
        cin >> pilihan;
        cin.ignore();

```

D.Tambah data

Fitur ini digunakan untuk menambahkan data baru ke dalam array.

Source Code:

```

    if (isi < Max) {
        cout << "Silahkan masukkan nama pengunjung: ";
        getline(cin, nama[isi]);

        cout << "Masukkan umur pengunjung: ";

```

```

        cin >> umur[isi];
        cin.ignore();

        cout << "Masukkan tujuan kunjungan: ";
        getline(cin, tujuan[isi]);

        isi++;
        cout << "Data pengunjung berhasil
ditambahkan!\n" << endl;
    } else {
        cout << "Daftar pengunjung sudah penuh!\n" <<
endl;
    }
    break;

```

E. Baca data

Fitur ini digunakan untuk membaca data yang telah tersimpan di dalam array.

Source Code:

```

    if (isi == 0) {
        cout << "Daftar pengunjung masih kosong!\n" <<
endl;
    } else {
        cout << "Daftar Pengunjung" << endl;
        for (int i = 0; i < isi; i++) {
            cout << "Pengunjung ke-" << i + 1 << ": "
<< nama[i]
                << " | Umur: " << umur[i]
                << " | Tujuan: " << tujuan[i] <<
endl;
        }
    }
    break;

```

F. Update data

Fitur ini digunakan untuk memperbarui data di dalam array. Pengguna diberi 3 pilihan untuk mengubah data yaitu nama, umur atau tujuan kunjungan.

Source Code:


```

if (isi == 0) {
    cout << "Belum ada pengunjung untuk diperbarui!\n"
    << endl;
} else {
    int index;
    cout << "Masukkan index pengunjung yang ingin
diperbarui: ";

    cin >> index;
    cin.ignore();
    if (index > 0 && index <= isi) {
        int pilihanEdit;
        cout << "Pilih data yang ingin diperbarui:" <<
endl;

        cout << "1. Nama" << endl;
        cout << "2. Umur" << endl;
        cout << "3. Tujuan Kunjungan" << endl;
        cout << "Pilih: ";
        cin >> pilihanEdit;
        cin.ignore();

        switch (pilihanEdit) {
            case 1:
                cout << "Masukkan nama baru: ";
                getline(cin, nama[index - 1]);
                cout << "Nama berhasil diperbarui!\n";
                break;
            case 2:
                cout << "Masukkan umur baru: ";
                cin >> umur[index - 1];
                cin.ignore();
                cout << "Umur berhasil diperbarui!\n";
                break;
            case 3:
                cout << "Masukkan tujuan kunjungan baru:
";

                getline(cin, tujuan[index - 1]);
                cout << "Tujuan kunjungan berhasil
diperbarui!\n";

                break;
            default:
                cout << "Pilihan tidak valid!\n";
                break;
        }
    } else {
        cout << "Index tidak valid!\n" << endl;
    }
}

```

```
    }  
  }  
  break;
```

G. Hapus data

Fitur ini digunakan untuk menghapus data dari indeks yang dipilih.

Source Code:

```
if (isi == 0) {  
    cout << "Tidak ada data pengunjung untuk dihapus!\n"  
<< endl;  
} else {  
    int index;  
    cout << "Masukkan index pengunjung yang ingin  
dihapus: ";  
    cin >> index;  
    if (index > 0 && index <= isi) {  
        for (int i = index - 1; i < isi - 1; i++) {  
            nama[i] = nama[i + 1];  
            umur[i] = umur[i + 1];  
            tujuan[i] = tujuan[i + 1];  
        }  
        isi--;  
        cout << "Data pengunjung berhasil dihapus!\n" <<  
endl;  
    } else {  
        cout << "Index tidak valid!\n" << endl;  
    }  
}  
break;
```

4. Uji Coba dan Hasil Output

4.1 Uji Coba

1. Skenario 1

Andi adalah seorang petugas yang mencatat data pengunjung mall tiap harinya. Ia telah bertugas selama 3 tahun. Ia telah mendaftarkan akunnya dan kemudian memasukkan username andi dan NIM 061 ke dalam program dan program berhasil terbuka. Ia lalu menjalankan tugasnya dengan menginput data salah seorang pengunjung bernama Budi yang berusia 25 tahun dengan tujuan Matahari. Ia kemudian melihat kembali data Budi tersebut dan mengubah tujuannya ke Uniqlo. Selanjutnya Ia tak sengaja menghapus data pengunjung itu dan keluar dari program.

2. Skenario 2

Maria adalah seorang fresh graduate yang menggantikan Andi setelah Andi dipecat atas kelalaiannya. Sebagai orang awam, ia tidak mengerti mengoperasikan program ini. Ia adalah orang yang teledor dan sampai salah 3 kali saat login yang membuat program tak terbuka. Saat program terbuka setelah ia mencoba mengganti username dan NIM, kerjaannya juga berantakan.

4.2 Hasil Output

```
=== Signup Akun ===  
Silahkan masukkan NIM anda: 061  
Silahkan masukkan nama anda: Andi  
  
Akun berhasil terdaftar!  
  
=== Login Akun ===  
Silahkan masukkan NIM anda: 061  
Silahkan masukkan nama Anda: Andi  
  
Login berhasil! Selamat datang, Andi!
```

Gambar 4.1 Akun Andi berhasil terdaftar dan bisa login

```
==== Silahkan pilih aksi yang ingin dilakukan! ====  
1. Tambah Data Pengunjung  
2. Lihat Data Pengunjung  
3. Perbarui Data Pengunjung  
4. Hapus Data Pengunjung  
5. Exit  
Pilih menu: 1  
Silahkan masukkan nama pengunjung: Budi  
Masukkan umur pengunjung: 25  
Masukkan tujuan kunjungan: matahari  
Data pengunjung berhasil ditambahkan!
```

Gambar 4.2 Andi menambah data Budi

```

==== Silahkan pilih aksi yang ingin dilakukan! ====
1. Tambah Data Pengunjung
2. Lihat Data Pengunjung
3. Perbarui Data Pengunjung
4. Hapus Data Pengunjung
5. Exit
Pilih menu: 3
Masukkan index pengunjung yang ingin diperbarui: 1
Pilih data yang ingin diperbarui:
1. Nama
2. Umur
3. Tujuan Kunjungan
Pilih: 3
Masukkan tujuan kunjungan baru: uniqlo
Tujuan kunjungan berhasil diperbarui!

```

Gambar 4.3 Andi memperbarui data Budi

```

=== Login Akun ===
Silahkan masukkan NIM anda: 099
Silahkan masukkan nama Anda: maria

Login gagal! NIM atau Nama tidak sesuai.

=== Login Akun ===
Silahkan masukkan NIM anda: 099
Silahkan masukkan nama Anda: mari

Login gagal! NIM atau Nama tidak sesuai.

Anda telah mencoba login sebanyak 3 kali. Program berhenti.

```

Gambar 4.4 Maria tidak dapat memasuki program

5.GIT

```

PS C:\Users\PC\Desktop\praktikum-apl\B1'24\post-test\post-test-apl-2> git status
On branch main
Your branch is up to date with 'origin/main'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
  ./

nothing added to commit but untracked files present (use "git add" to track)
PS C:\Users\PC\Desktop\praktikum-apl\B1'24\post-test\post-test-apl-2> git commit -m "Finish Post Test 2"
[main a48a491] Finish Post Test 2
2 files changed, 170 insertions(+)
 create mode 100644 B1'24/post-test/post-test-apl-2/2409106061-RICHARDDANTEGUNAWAN-PT-2.cpp
 create mode 100644 B1'24/post-test/post-test-apl-2/2409106061-RICHARDDANTEGUNAWAN-PT-2.exe
PS C:\Users\PC\Desktop\praktikum-apl\B1'24\post-test\post-test-apl-2> git push -u origin main
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 4 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (7/7), 676.83 KiB | 2.08 MiB/s, done.
Total 7 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/404-Dev/NotFound/praktikum-apl.git
 826672b..a48a491 main -> main
branch 'main' set up to track 'origin/main'.
PS C:\Users\PC\Desktop\praktikum-apl\B1'24\post-test\post-test-apl-2> git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
PS C:\Users\PC\Desktop\praktikum-apl\B1'24\post-test\post-test-apl-2>

```

Git status: perintah untuk menampilkan informasi tentang status dari repositori Git yang sedang digunakan.

Git add: perintah untuk menambahkan file ke staging area

Git commit: perintah untuk menyimpan perubahan yang telah ditambahkan ke staging area ke dalam repository lokal.

Git push: perintah untuk mengirim commit dari repository lokal ke repository remote.