**LAPORAN PRAKTIKUM POSTTEST 4**

**ALGORITMA PEMROGRAMAN LANJUT**

|  |  |
| --- | --- |
|  |  |
|  |  |

**Disusun oleh:**

**Gilbert Finsen Zakaria Lian (2409106110) Kelas (C‘2)**

**PROGRAM STUDI INFORMATIKA UNIVERSITAS MULAWARMAN SAMARINDA**

**2025**

# **1.** **Flowchart**

|  |  |
| --- | --- |
|  |  |
|  |  |

**2.** **Analisis Program**

*(Tuliskan secara ringkas tujuan dan fungsi/manfaat utama dari program yang dibuat.)*

**Tujuan Program**

Program ini bertujuan untuk mengelola data aset dengan fitur multiuser menggunakan konsep array of struct dan nested struct. Setiap user dapat memiliki daftar aset pribadi yang dapat dikelola secara mandiri.

# **Fungsi dan Manfaat Utama**

1. **Registrasi dan Login**: Memungkinkan pengguna untuk membuat akun baru dan masuk dengan validasi hingga 3 kali percobaan.

2. **Manajemen Aset**: Pengguna dapat menambah, melihat, mengubah, dan menghapus data aset yang mereka miliki.

3. **Penggunaan Struct**: Mengorganisir data aset dan pengguna secara efisien menggunakan nested struct.

4. **Data Terpisah per User**: Setiap user memiliki data aset yang tersimpan secara terpisah, menjaga keamanan dan privasi data.

5. **Antarmuka Sederhana**: Menyediakan menu navigasi yang mudah dipahami untuk mengelola aset.

# 

# **3.** **Source Code**

FUNGSI DAN PROSEDUR

1. **Registrasi**

|  |
| --- |
| void registrasi(User users[], int &totalUsers) |

* Menerima array users dan jumlahnya.
* Input nama dan NIM.
* Menambahkan user ke array jika belum penuh.

2. **Login**

|  |
| --- |
| int login(User users[], int totalUsers) |

* Meminta input nama & NIM.
* Memeriksa apakah cocok dengan data di array.
* Maksimal 3 percobaan.
* Return index user jika berhasil, user.daftarAset jika gagal.

3. **Fungsi Overloading: Menampilkan Aset**

|  |
| --- |
| void tampilkanAset(const Aset &aset, int index); void tampilkanAset(const Aset &aset); |

* Dua versi:
  + Menampilkan aset dengan nomor urut (index).
  + Menampilkan satu aset saja tanpa format tabel.

4. **Fungsi Rekursif: Total Nilai Aset**

|  |
| --- |
| double hitungTotalNilaiAset(const Aset daftar[], int jumlah) |

* Menghitung total nilai dari semua aset user.
* Rekursif: user.daftarAset aset terakhir + total sebelumnya.

5. **Prosedur Tambah Aset**

|  |
| --- |
| void tambahAset(User &user) |

* Meminta nama, jumlah, dan harga dari aset baru.
* Disimpan di array user.daftarAset.

6. **Prosedur Lihat Aset**

|  |
| --- |
| void lihatAset(User &user) |

* Menampilkan semua aset user dalam bentuk tabel.
* Menampilkan total nilai aset (pakai fungsi rekursif).

7. **Prosedur Ubah Aset**

|  |
| --- |
| void ubahAset(User &user) |

* Meminta nomor aset yang ingin diubah.
* Input data baru (nama, jumlah, harga).

8. **Prosedur Hapus Aset**

|  |
| --- |
| void hapusAset(User &user) |

* Meminta nomor aset yang akan dihapus.
* Menggeser data ke kiri (menghapus dengan cara overwrite).

9. **Menu Aset**

|  |
| --- |
| void menuAset(User &user) |

* Menyediakan 5 pilihan: Tambah, Lihat, Ubah, Hapus, Logout.
* Memanggil subprogram sesuai pilihan user.

# **4.** **Uji Coba dan Hasil Output**

*(Sertakan tangkapan layar atau hasil output dari program setelah dijalankan.)*

**4.1**

|  |  |
| --- | --- |
|  |  |
|  |  |

**Halaman Untuk Registrasi**

**Gambar 4.1**

**4.2** **Halaman Login**

|  |  |
| --- | --- |
|  |  |
|  |  |

**Gambar 4.2**

**4.3** **Pilihan Menu**

|  |  |
| --- | --- |
|  |  |
|  |  |

**Gambar 4.3**

**4.4** **Tambah Aset**

|  |  |
| --- | --- |
|  |  |
|  |  |

**Gambar 4.4**

**4.5** **Lihat Aset**

|  |  |
| --- | --- |
|  |  |
|  |  |

**Gambar 4.5**

**4.6** **Ubah Aset**

|  |  |
| --- | --- |
|  |  |
|  |  |

**Gambar 4.6**

**4.7** **Hapus Aset**

|  |  |
| --- | --- |
|  |  |
|  |  |

Gambar 4.7

**4.8** **Logout dan keluar program**

|  |  |
| --- | --- |
|  |  |
|  |  |

Gambar 4.8

**4.9** **Akses Di Tolak Saat Gagal Login**

|  |  |
| --- | --- |
|  |  |
|  |  |

Gambar 4.9

Gambar 4.1 Contoh

Gambar <nomor bab>.[index] <judul-gambar>

# **5.** **Langkah-Langkah Git**

**Git add**

|  |  |
| --- | --- |
|  |  |
|  |  |

|  |  |
| --- | --- |
|  |  |
|  |  |

**Git commit –m**

|  |  |
| --- | --- |
|  |  |
|  |  |

**Git push**