

LAPORAN PRAKTIKUM ALGORITMA PEMOGRAMAN LANJUT



NAMA:Muhammad febriansyah

KELAS:B2 24 informatika

NIM:2409106090

TUGAS:laporan posttes 4

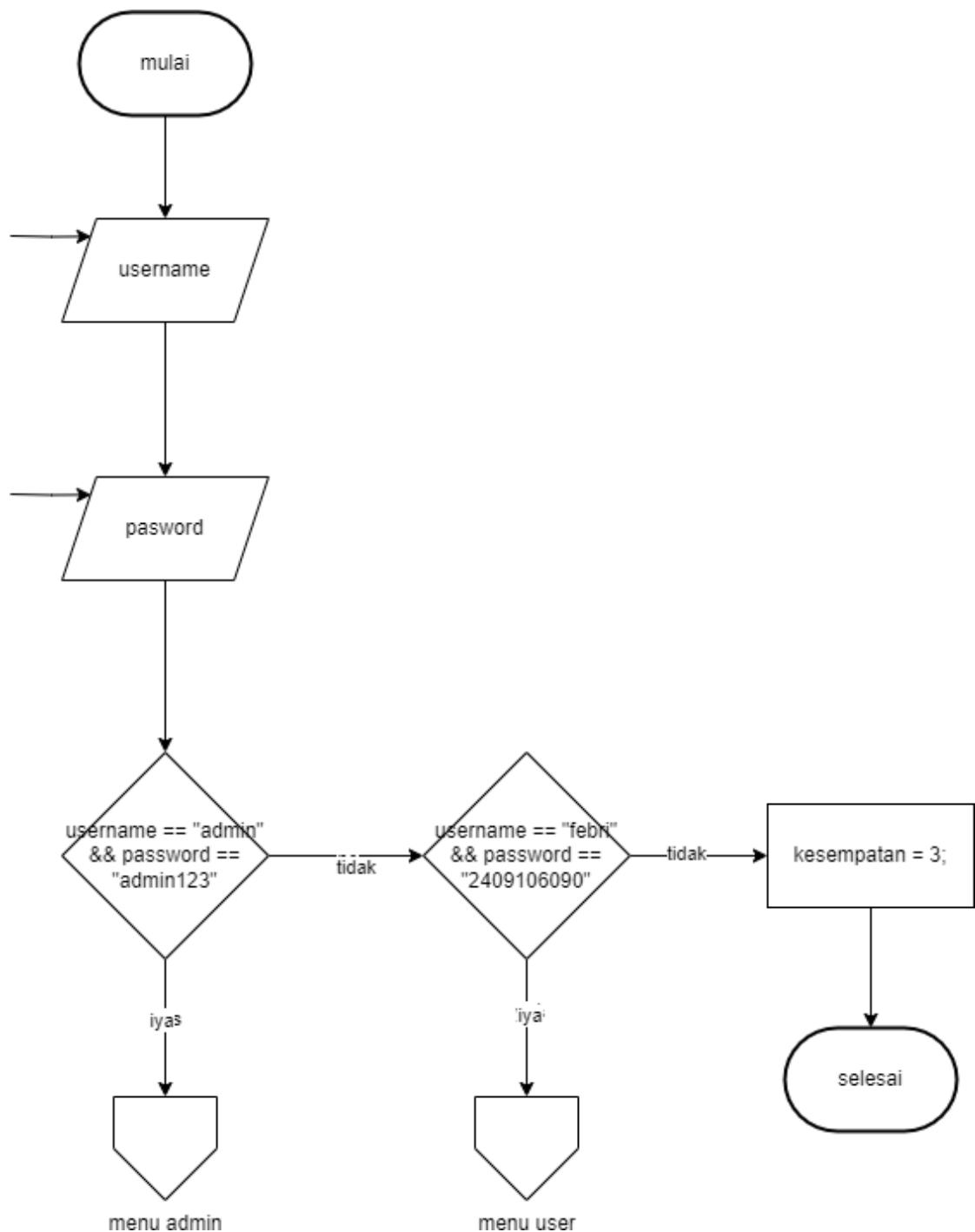
PROGRAM STUDI INFORMATIKA

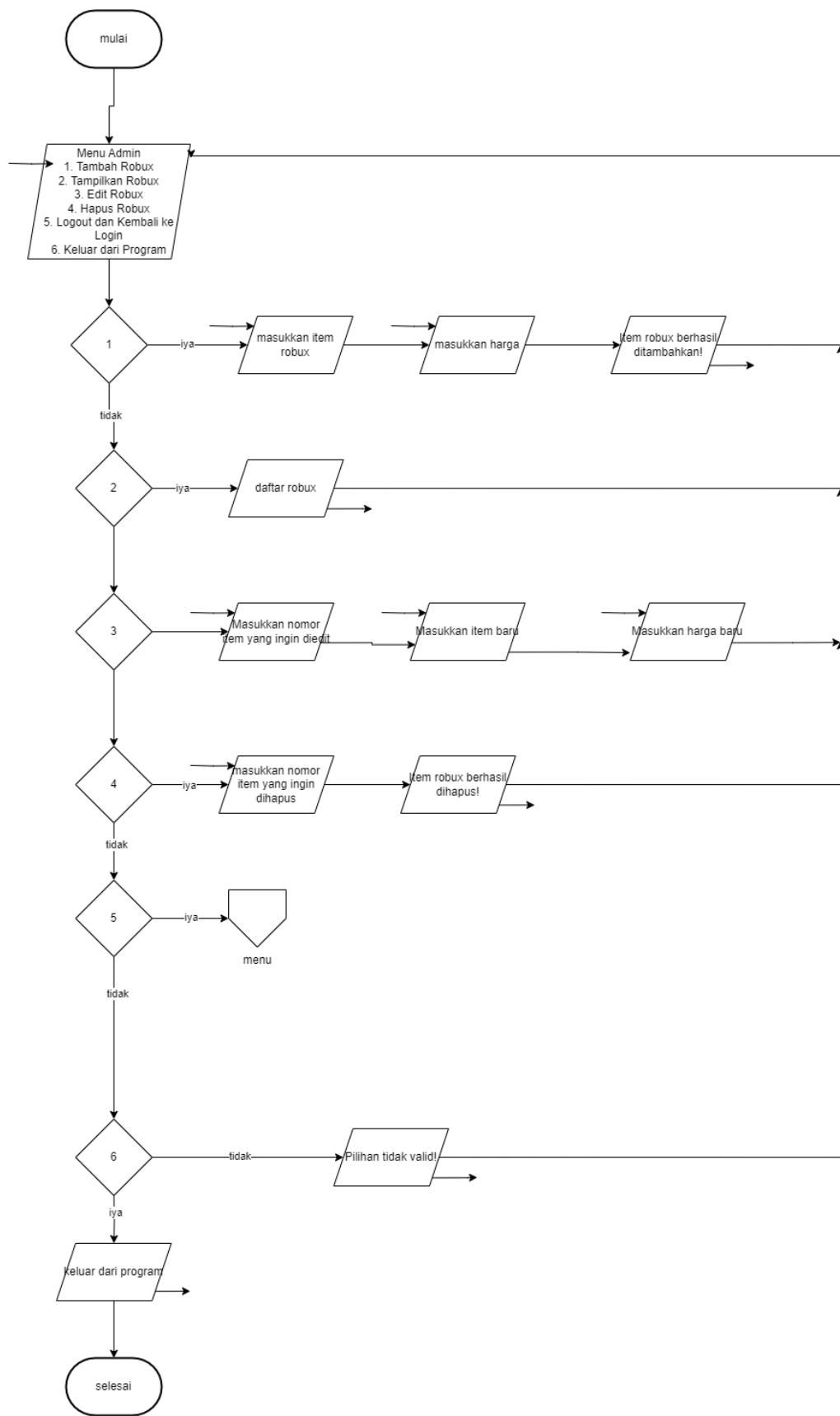
UNIVERSITAS MULAWARMAN

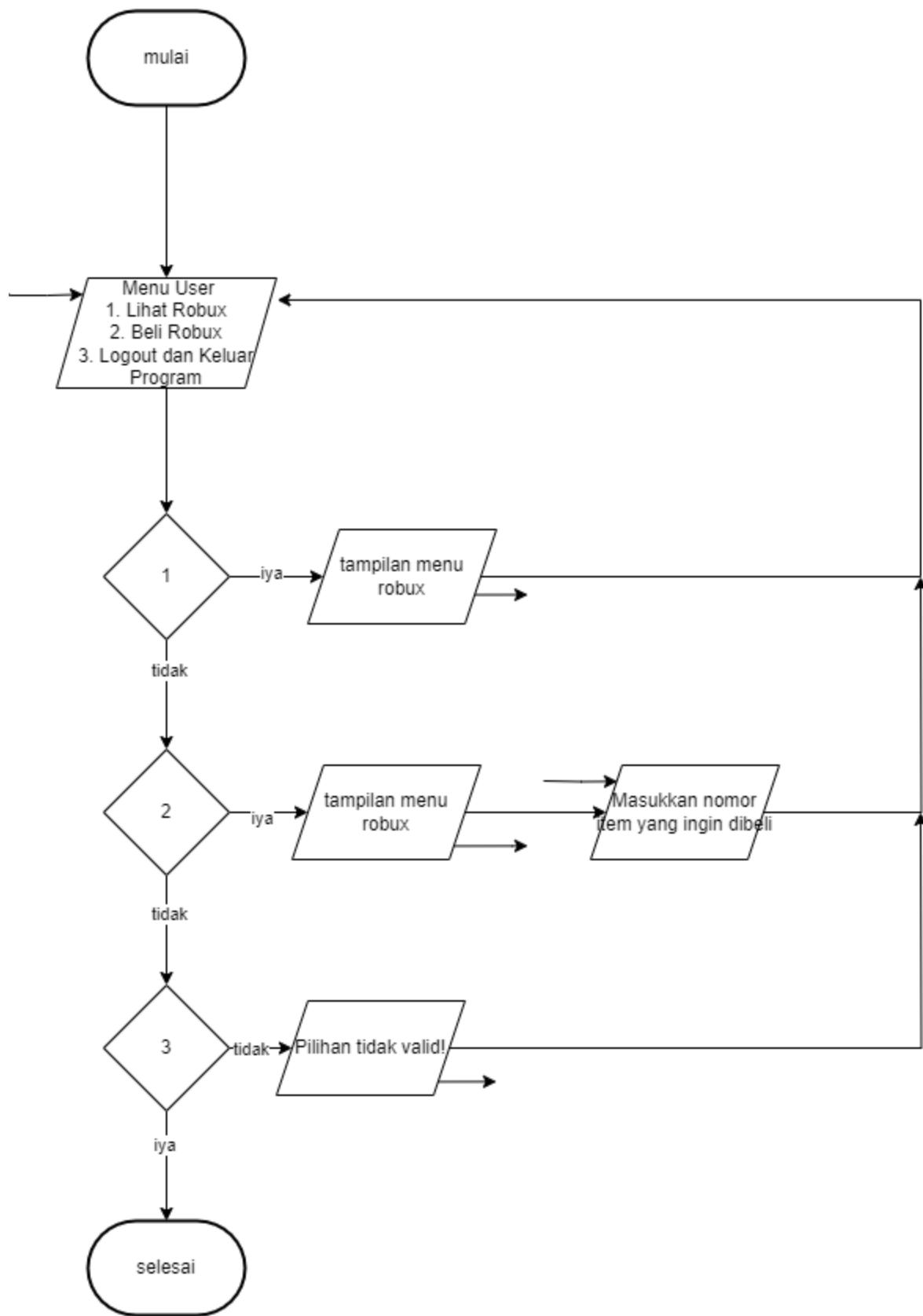
SAMARINDA

2025

1.FLOWCHART:







2. ANALISIS PROGRAM

Program ini adalah sistem manajemen penjualan Robux berbasis CLI (Command Line Interface). Program memiliki dua jenis pengguna:

Admin: Dapat menambahkan, menampilkan, mengedit, dan menghapus item Robux.

User: Hanya dapat melihat daftar item Robux dan melakukan pembelian

Sistem Login: Memastikan hanya pengguna yang memiliki kredensial yang benar dapat mengakses program.

A. Deklarasi Struktur dan Variabel

Struct RobuxItem digunakan untuk menyimpan informasi item Robux, yang terdiri dari:

nama: Nama item Robux.

harga: Harga per unit (tipe int).

stok: Jumlah stok tersedia.

Array robux[MAX_DATA] menyimpan semua item Robux dalam bentuk struct.

Variabel totalRobux menyimpan jumlah item yang tersedia.

Variabel login digunakan untuk autentikasi pengguna.

B. Sistem Login

Program meminta **username dan password**.

Jika salah, pengguna memiliki **3 kesempatan** untuk login.

Terdapat **dua jenis pengguna**:

Admin (admin, admin123): Bisa mengelola data.

User (febri, 2409106090): Hanya bisa melihat dan membeli.

Jika gagal login 3 kali, program keluar.

C. Menu Admin

1. Tambah Robux

Meminta input **nama, harga, dan stok**.

Menambah data ke dalam array robux.

Mengecek apakah array penuh sebelum menambah.

2. Tampilkan Robux

Menampilkan daftar item yang tersedia dalam format tabel.

3. Edit Robux

Memilih item berdasarkan nomor indeks.

Memungkinkan admin mengubah **nama, harga, dan stok**.

4. Hapus Robux

Memilih item berdasarkan nomor indeks.

Menggeser data untuk menghapus item.

5. Logout

Kembali ke halaman login.

6. Keluar Program

Menghentikan program.

D. Menu User

1. Lihat Robux

Sama seperti fitur admin, hanya bisa melihat daftar item.

2. Beli Robux

Memilih item berdasarkan nomor indeks.

Memasukkan jumlah yang ingin dibeli.

Jika stok mencukupi, stok akan dikurangi sesuai jumlah pembelian.

3. Logout dan Keluar

Keluar dari program.

3. source code



```
17 // ===== PROSEDUR DAN FUNGSI =====
18 void tambahRobux(RobuxItem data[], int &total) {
19     if (total >= MAX_DATA) {
20         cout << "Data penuh! Tidak bisa menambahkan lagi.\n";
21         return;
22     }
23     cout << "\nMasukkan item robux: ";
24     cin >> data[total].nama;
25     cout << "Masukkan harga: ";
26     cin >> data[total].harga;
27     cout << "Masukkan stok: ";
28     cin >> data[total].stok;
29     total++;
30     cout << "Item robux berhasil ditambahkan!\n";
31 }
32
33
34 void tampilanRobux(const RobuxItem data[], int total) {
35     if (total == 0) {
36         cout << "\nTidak ada item robux tersedia.\n";
37         return;
38     }
39     cout << "\n-----\n";
40     cout << "| No | Item | Harga | Stok |\n";
41     cout << "-----\n";
42     for (int i = 0; i < total; i++) {
43         cout << "| " << setw(2) << i + 1 << " | "
44             << setw(12) << data[i].nama << " | "
45             << setw(8) << data[i].harga << " | "
46             << setw(7) << data[i].stok << " |\n";
47     }
48     cout << "-----\n";
49 }
```

1. Fungsi dan Prosedur Manajemen Robux

void tambahRobux(RobuxItem data [], int total)

Menambahkan item baru ke dalam array robux.

Parameter:

Data [] → array Robux

total → jumlah total item saat ini (dengan referensi)

Input dari user:

Nama robux

Harga

Stok

Setelah input, item ditambahkan ke array dan total bertambah 1.

void tampilanRobux(const RobuxItem data[], int total)

Menampilkan semua data robux yang tersedia dalam bentuk tabel.

Hanya menampilkan, tidak bisa mengubah data.

Parameter:

Data [] → array Robux

total → jumlah data robux yang aktif

void editRobux(RobuxItem data[], int total)

Mengedit data robux berdasarkan nomor urut.

Langkah:

1. Minta input nomor item
 2. Minta input data baru (nama, harga, stok)
 3. Update elemen pada array
-

void hapusRobux(RobuxItem data[], int &total)

Menghapus item dari array berdasarkan nomor.

Setelah item dihapus, semua elemen di bawahnya digeser ke atas.

total dikurangi satu.

void beliRobux(RobuxItem data[], int total)

Untuk melakukan pembelian robux oleh user biasa.

Langkah:

1. Tampilkan semua item robux
2. User memilih item & jumlah yang ingin dibeli
3. Stok akan berkurang, total harga ditampilkan

Validasi:

Nomor item valid

Jumlah beli \leq stok

2. Fungsi Login

bool login(string user, string pass, bool &isAdmin)

Mengecek apakah kombinasi username dan password valid.

Jika cocok dengan:

admin/admin123 \rightarrow isAdmin = true

febri/2409106090 \rightarrow isAdmin = false

Return true jika login berhasil, false jika gagal.

3. Main Program (Fungsi main())

Inisialisasi login, kemudian masuk ke menu Admin atau User berdasarkan siapa yang login.

Menu berbentuk perulangan (do-while) sampai pengguna logout atau keluar program.

