

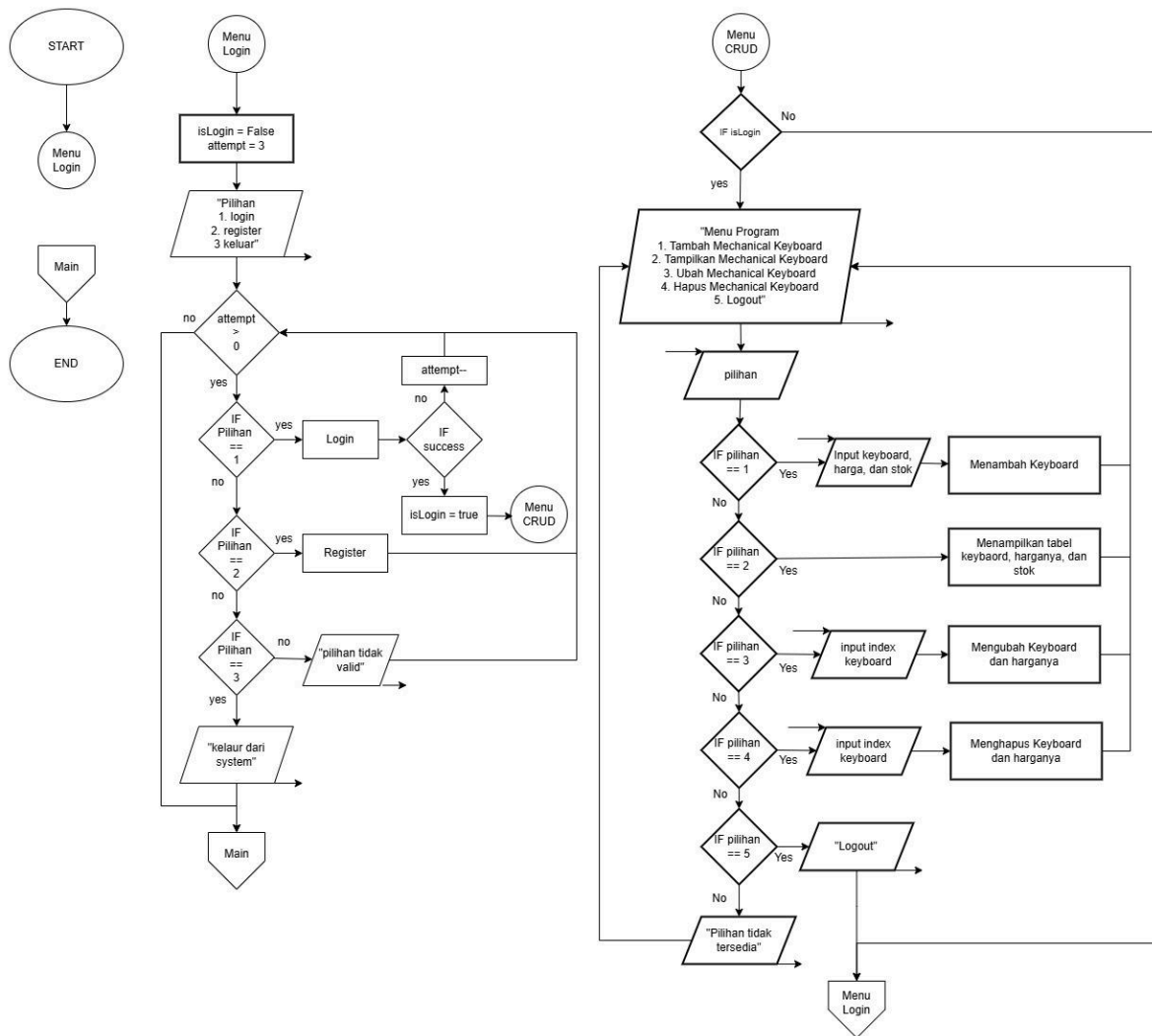
**LAPORAN PRAKTIKUM**  
**POSTTEST 6**  
**ALGORITMA PEMROGRAMAN LANJUT**



**Disusun oleh:**  
**Muhammad Haykal Makhmud 2409106005**  
**Kelas A1 '24**

**PROGRAM STUDI INFORMATIKA**  
**UNIVERSITAS MULAWARMAN**  
**SAMARINDA**  
**2025**

# 1. Flowchart



Gambar 1.1 flowchart

## 2. Analisis Program

### 2.1 Deskripsi Singkat Program

Program ini bertujuan untuk mengelola list produk mechanical keyboard. Program ini dapat menambahkan mechanical keyboard baru, dapat menampilkan tabel keyboard dan harganya, dapat mengubah mechanical keyboard yang ada pada list, serta dapat menghapus mechanical keyboard yang diinginkan. Program ini mengharuskan user untuk login terlebih dahulu untuk menjalankan program. User dapat melakukan register terlebih dahulu jika belum memiliki akun. User hanya diberi 3x kesempatan untuk melakukan login. Jika gagal maka sistem akan berhenti.

## 3. Source Code

### A. Menu Login

Fitur ini digunakan untuk menampilkan menu login atau register

Source Code:

```
cout << "WELCOME" << endl;
cout << "1. Login" << endl;
cout << "2. Register" << endl;
cout << "3. Keluar" << endl;
cout << "Pilihan: ";
cin >> pilihan_login;
```

### B. Menu Utama

Fitur ini menampilkan menu utama untuk memilih fitur pada sistem

Source Code:

```
cout << "MENU" << endl;
cout << "1. Tambah Keyboard" << endl;
cout << "2. Tampilkan Keyboard" << endl;
cout << "3. Ubah Keyboard" << endl;
cout << "4. Hapus Keyboard" << endl;
```

```

cout << "5. Sort Stok Descending" << endl;
cout << "6. Sort Nama Ascending" << endl;
cout << "7. Sort Harga Ascending" << endl;
cout << "8. Logout" << endl;
cout << "Pilihan: ";

```

### C. Login

Fitur ini untuk melakukan login untuk mengakses sistem

#### Source Code:

```

void Login(){
    User *userPtr = user;

    cout << "LOGIN" << endl;
    cout << "Masukkan Username : ";
    cin >> username;
    cout << "Masukkan Password : ";
    cin >> password;

    for (int i = 0; i < panjang_user; i++) {
        if (username == (userPtr + i)->name && password ==
(userPtr + i)->NIM) {
            cout << "Login Berhasil" << endl;
            isLogin = true;
            break;
        }
    }

    if (!isLogin) {
        system("cls");
        cout << "Login Gagal" << endl;
        attempt--;
    }
}

```

### D. Register

Fitur ini untuk melakukan register untuk membuat user baru

#### Source Code:

```
void Register(){
    cout << "REGISTER" << endl;
    cout << "Masukkan Username : ";
    cin >> username;
    cout << "Masukkan Password : ";
    cin >> password;

    user[panjang_user].name = username;
    user[panjang_user].NIM = password;

    panjang_user++;

    system("cls");
    cout << "Register Berhasil" << endl;
}
```

#### E. Tambah Keyboard

Fitur ini untuk menambahkan keyboard baru ke dalam array

#### Source Code:

```
bool AddKeyboard(Keyboard newKeyboard){
    keyboard[panjang_keyboard].nama = newKeyboard.nama;
    keyboard[panjang_keyboard].harga = newKeyboard.harga;
    keyboard[panjang_keyboard].stok = newKeyboard.stok;

    panjang_keyboard++;

    return true;
}
```

```
if (panjang_keyboard < MAX_ARRAY) {
    Keyboard newKeyboard;
    cin.ignore();
    cout << "Masukkan Nama Keyboard : ";
    getline(cin, newKeyboard.nama);
}
```

```

    cout << "Masukkan Harga Keyboard : ";
    cin >> newKeyboard.harga;
    cout << "Masukkan Stok Keyboard : ";
    cin >> newKeyboard.stok;

    if(AddKeyboard(newKeyboard)){
        system("cls");
        cout << "Keyboard berhasil ditambahkan" << endl;
    }

} else {
    system("cls");
    cout << "Kapasitas penuh! Tidak bisa menambah keyboard
lagi."<< endl;
}

```

## F. Menampilkan keyboard

Fitur ini untuk menampilkan tabel berisi keyboard, harga, dan stoknya

### Source Code:

```

void ShowKeyboard(Keyboard *keyboardPtr, int &panjang_keyboard){
    system("cls");
    if (panjang_keyboard == 0) {
        cout << "Belum ada Keyboard" << endl;
    } else {

        cout <<
        "+-----+-----+-----+-----+" << endl;
        cout << "| No   | Keyboard           | Harga (Rp) | Stok"
        "|" << endl;
        cout <<
        "+-----+-----+-----+-----+" << endl;

        for (int i = 0; i < panjang_keyboard; i++) {
            cout << "| " << setw(3) << i + 1 << " | "
            << left << setw(22) << (keyboardPtr + i )->nama << " | "
            << right << setw(12) << (keyboardPtr + i )->harga << " | "
            << right << setw(7) << (keyboardPtr + i )->stok << " | " <<
            endl;
        }
    }
}

```

```

        cout <<
        "+-----+-----+-----+-----+" << endl;
    }
}

```

## G. Mengubah Keyboard

Fitur ini untuk mengubah keyboard dan harganya berdasarkan keinginan user

### Source Code:

```

bool UpdateKeyboard(int index, Keyboard newKeyboard){
    keyboard[index-1].nama = newKeyboard.nama;
    keyboard[index-1].harga = newKeyboard.harga;
    keyboard[index-1].stok = newKeyboard.stok;

    return true;
}

```

```

if (panjang_keyboard == 0) {
    cout << "Belum ada Keyboard" << endl;
} else {
    ShowKeyboard(keyboard, panjang_keyboard);
    cout << "Masukkan nomor keyboard yang akan diubah: ";
    cin >> index;

    if (index > 0 && index <= panjang_keyboard) {

        Keyboard newKeyboard;

        cout << "Masukkan Nama Keyboard : ";
        cin.ignore();
        getline(cin, newKeyboard.nama);
        cout << "Masukkan Harga Keyboard : ";
        cin >> newKeyboard.harga;
        cout << "Masukkan Stok Keyboard : ";
        cin >> newKeyboard.stok;

        if(UpdateKeyboard(index, newKeyboard)){

```

```

        system("cls");
        cout << "Keyboard berhasil diubah" << endl;
    }

    } else {
        system("cls");
        cout << "Nomor keyboard tidak valid" << endl;
    }
}

```

## H. Menghapus Keyboard

Fitur ini untuk menghapus data keyboard dan harganya berdasarkan keinginan user

### Source Code:

```

bool DeleteKeyboard(int index){
    for (int i = index - 1; i < panjang_keyboard - 1; i++) {
        keyboard[i] = keyboard[i + 1];
    }
    panjang_keyboard--;

    return true;
}

```

```

if(panjang_keyboard == 0){
    cout << "Belum ada Keyboard" << endl;
}else{
    ShowKeyboard(keyboard, panjang_keyboard);
    cout << "Masukkan nomor keyboard yang akan dihapus: ";
    cin >> index;

    if (index > 0 && index <= panjang_keyboard) {
        if(DeleteKeyboard(index)){
            system("cls");
            cout << "Keyboard berhasil dihapus" << endl;
        }
    } else {
        system("cls");
        cout << "Nomor keyboard tidak valid" << endl;
    }
}

```



```
}  
}
```

## I. Mengurutkan Stok Keyboard Secara Descending

Fitur ini untuk mengurutkan data keyboard berdasarkan stok secara descending (besar ke kecil) dengan menggunakan algoritma bubble sort

### Source Code:

```
void BubbleSortStokDesc(){  
    for (int i = 0; i < panjang_keyboard; i++) {  
        for (int j = 0; j < (panjang_keyboard-i-1); j++) {  
            if(keyboard[j].stok < keyboard[j+1].stok) {  
                Keyboard temp = keyboard[j];  
                keyboard[j] = keyboard[j+1];  
                keyboard[j+1] = temp;  
            }  
        }  
        ShowKeyboard(keyboard, panjang_keyboard);  
    }  
}
```

## J. Mengurutkan Nama Keyboard Secara Ascending

Fitur ini untuk mengurutkan data keyboard berdasarkan nama secara ascending (kecil ke besar) dengan menggunakan algoritma selection sort

### Source Code:

```
void SelectionSortNameAsc(){  
    for (int i = 0; i < panjang_keyboard - 1; i++){  
        int min = i;  
        for (int j = i + 1; j < panjang_keyboard; j++){  
            if (keyboard[j].nama < keyboard[min].nama){  
                min = j;  
            }  
        }  
    }  
}
```

```

        Keyboard temp = keyboard[i];
        keyboard[i] = keyboard[min];
        keyboard[min] = temp;
    }
    ShowKeyboard(keyboard, panjang_keyboard);
}

```

## K. Mengurutkan Harga Keyboard Secara Ascending

Fitur ini untuk mengurutkan data keyboard berdasarkan harga secara ascending (kecil ke besar) dengan menggunakan algoritma quick sort

### Source Code:

```

void QuickSortHargaAsc(int low, int high){
    if (low >= high) return;

    int mid = low + (high - low) / 2;
    int pivot = keyboard[mid].harga;
    int i = low, j = high;
    while (i <= j) {
        while (keyboard[i].harga < pivot) {
            i++;
        }
        while (keyboard[j].harga > pivot) {
            j--;
        }
        if (i <= j) {
            swap(keyboard[i], keyboard[j]);
            i++;
            j--;
        }
    }

    if (low < j) {
        QuickSortHargaAsc(low, j);
    }
    if (i < high) {
        QuickSortHargaAsc( i, high);
    }
}

```

```
}
```

## 4. Hasil Output

### 4.1 Hasil Output

```
WELCOME
1. Login
2. Register
3. Keluar
Pilihan: 
```

Gambar 4.1 Menu\_login

```
Login Gagal
WELCOME
1. Login
2. Register
3. Keluar
Pilihan: 1
LOGIN
Masukkan Username : salah
Masukkan Password : salah
```

Gambar 4.2 Login\_gagal

```
WELCOME
1. Login
2. Register
3. Keluar
Pilihan: 2
REGISTER
Masukkan Username : userBaru
Masukkan Password : 240000001
```

Gambar 4.3 Register

```

MENU
1. Tambah Keyboard
2. Tampilkan Keyboard
3. Ubah Keyboard
4. Hapus Keyboard
5. Logout
Pilihan: █

```

Gambar 4.4 Menu\_utama

```

Masukkan Nama Keyboard : Maxfit61
Masukkan Harga Keyboard : 450000
Masukkan Stok Keyboard : 30█

```

Gambar 4.5 Tambah\_keyboard

No	Keyboard	Harga (Rp)	Stok
1	Keychron Q1 Max	4195000	10
2	Noir Timeless82 v2	1370000	10

Gambar 4.6 Menampilkan\_keyboard

```

+-----+-----+-----+-----+
| No | Keyboard | Harga (Rp) | Stok |
+-----+-----+-----+-----+
| 1 | Keychron Q1 Max | 4195000 | 10 |
| 2 | Noir Timeless82 v2 | 1370000 | 10 |
+-----+-----+-----+-----+
Masukkan nomor keyboard yang akan diubah: 1
Masukkan Nama Keyboard : maxfit61
Masukkan Harga Keyboard : 600000
Masukkan Stok Keyboard : 21█

```

Gambar 4.7 Ubah\_keyboard

No	Keyboard	Harga (Rp)	Stok
1	maxfit61	600000	21
2	Noir Timeless82 v2	1370000	10

Masukkan nomor keyboard yang akan dihapus: 1

Gambar 4.8 Hapus\_keyboard

No	Keyboard	Harga (Rp)	Stok
1	Keychron Q1 Max	1370000	10
2	Noir Timeless82 v2	4195000	5
3	Ajazz AK820	840000	5

Gambar 4.9 sorting\_stok\_secara\_descending

No	Keyboard	Harga (Rp)	Stok
1	Ajazz AK820	840000	5
2	Keychron Q1 Max	1370000	10
3	Noir Timeless82 v2	4195000	5

Gambar 4.10 sorting\_nama\_secara\_ascending

No	Keyboard	Harga (Rp)	Stok
1	Ajazz AK820	840000	5
2	Keychron Q1 Max	1370000	10
3	Noir Timeless82 v2	4195000	5

Gambar 4.11 sorting\_harga\_secara\_ascending

## 5. Langkah-Langkah Penggunaan Git

### 1. git add .

Perintah git add . digunakan untuk menambahkan seluruh file atau perubahan yang telah dibuat ke dalam staging area

```
PS C:\Users\oru\Documents\Kuliah\praktikum-apl> git add .
```

Gambar 5.2 git\_add

### 2. git commit

Perintah ini menyimpan perubahan yang telah ditambahkan ke staging area ke dalam repository dengan pesan commit yang menjelaskan perubahan yang dilakukan. flag m digunakan untuk menambahkan pesan commit. penggunaan 'feat: ' bertujuan untuk memberitahu bahwa hasil commit merupakan penambahan fitur baru pada program

```
praktikum-apl> git commit -m "feat: menambahkan sorting"
```

Gambar 5.3 git\_commit

### 3. git push

Perintah ini mengunggah commit yang telah dibuat ke branch main di repository pada server github

```
PS C:\Users\oru\Documents\Kuliah\praktikum-apl\post-test> git push
```

Gambar 5.5 git\_push