

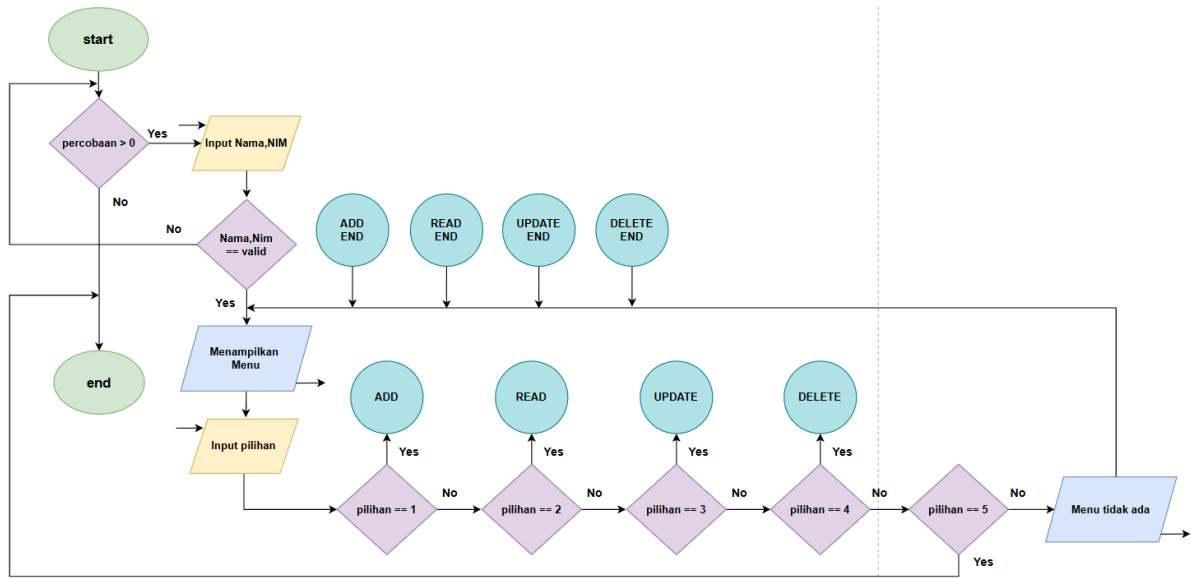
LAPORAN PRAKTIKUM
POSTTEST 2
ALGORITMA PEMROGRAMAN LANJUT



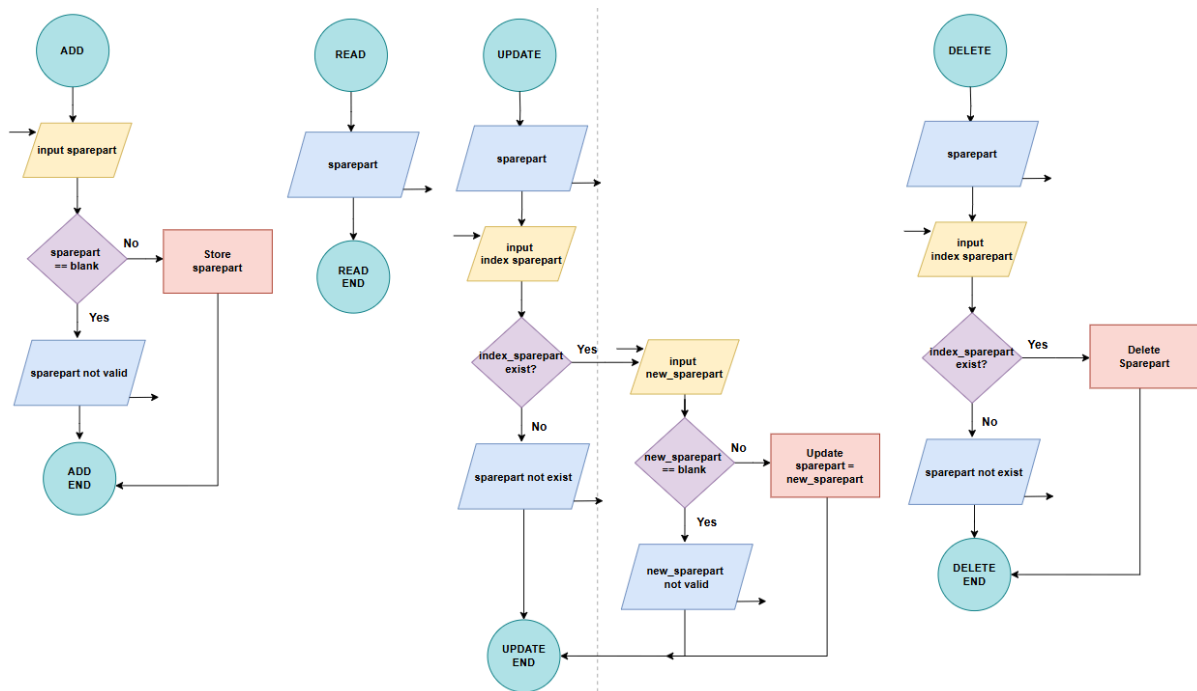
Disusun oleh:
Miftahul Fauzan 2409106048
Kelas B1 '24

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

1. Flowchart



Gambar 1.1 Flowcahrt 1



Gambar 1.2 Flowcahrt 2

2. Analisis Program

2.1 Deskripsi Singkat Program

Program yang saya buat adalah Manajemen Inventaris Sparepart Motor berbasis terminal. Program Manajemen Inventaris Sparepart Motor ini memungkinkan pengguna untuk melakukan beberapa hal. Di dalam program ini terdapat beberapa fitur-fitur seperti tambah sparepart, tampilkan sparepart, ubah stok sparepart, dan hapus sparepart. Pengguna bisa menggunakan fitur tambah sparepart jika pengguna ingin menambahkan data sparepart baru ke inventaris, menggunakan fitur tampilkan sparepart ingin mengetahui data sparepart yang ada di inventaris, menggunakan fitur update stok sparepart jika pengguna ingin mengubah stok sparepart yang ada di inventaris, dan pengguna juga bisa menggunakan fitur hapus sparepart jika ingin menghapus data sparepart yang ada di inventaris. Program ini dibuat dengan tujuan untuk memudahkan pihak bengkel motor dalam melakukan manajemen terhadap data sparepart motor. Kekurangan dari program ini adalah tidak memiliki tampilan yang menarik, hanya berbasis terminal.

2.2 Penjelasan Alur & Algoritma

Program dimulai dengan meminta pengguna melakukan proses login, pengguna diminta untuk memasukkan username dan password mereka. Jika username dan password yang dimasukkan benar, pengguna dapat mengakses menu utama. Namun, jika pengguna salah memasukkan username dan password, mereka diberikan tiga kali kesempatan sebelum sistem melarang mereka untuk melakukan login selama 15 menit.

Setelah berhasil masuk dengan memasukkan username dan password yang benar, program akan menampilkan menu dengan lima pilihan yaitu tambah sparepart, tampilkan sparepart, ubah stok sparepart, dan hapus sparepart, dan keluar. Jika pengguna memilih tambah sparepart, mereka dapat memasukkan beberapa data tentang sparepart seperti nama, jumlah, dan harga yang kemudian data tersebut akan ditambahkan ke sistem. Jika data sparepart yang dimasukkan tidak valid, seperti angka negatif untuk data harga, sistem akan menolak memasukkan data ke inventaris. Setelah melakukan tambah sparepart, program akan menampilkan menu utama lagi sampai pengguna memilih opsi keluar.

Jika pengguna memilih untuk tampilkan sparepart, sistem akan langsung menampilkan data sparepart ke terminal. Jika pengguna memilih untuk ubah stok sparepart,

pengguna diminta untuk memasukkan nomor sparepart dan stok sparepart yang baru, kemudian sistem akan mengecek sparepart berdasarkan indeks tersebut. Jika data sparepart ditemukan dalam sistem, maka sistem akan mengubah data stok, jika tidak sistem akan memberitahu bahwa data sparepart tidak ada.

Jika pengguna memilih untuk menghapus sparepart pengguna diminta untuk memasukkan nomor sparepart, kemudian sistem akan mengecek sparepart berdasarkan indeks tersebut. Jika data sparepart ditemukan dalam sistem, maka sistem akan menghapus data sparepart, jika tidak sistem akan memberitahu bahwa data sparepart tidak ditemukan. Setelah melakukan tambah sparepart, tampilkan sparepart, ubah stok sparepart, dan hapus sparepart, program akan menampilkan menu utama lagi sampai pengguna memilih opsi keluar. Jika pengguna telah selesai melakukan manajemen data sparepart, pengguna dapat memilih opsi keluar. Program akan berhenti jika pengguna memilih opsi keluar.

3. Source Code

A. Tambah Sparepart

Fitur ini digunakan untuk melakukan penyimpanan beberapa data sparepart ke dalam sistem. Di dalam fitur ini pengguna diminta menginputkan beberapa data, seperti nama, jumlah, dan harga sparepart.

Source Code:

```
// Tambah Sparepart
if (currentAmount < MAX_SPAREPART){
    // Deklarasi variabel nameOfSparepart untuk menampung nama sparepart
    string nameOfSparepart;
    // Deklarasi variabel quantity untuk menampung kuantitas sparepart
    int quantity;
    // Deklarasi variabel price untuk menampung harga sparepart
    int price;

    // Melakukan input nama sparepart
    cout << "Masukkan nama sparepart : ";
    cin.ignore();
    getline(cin, nameOfSparepart);

    // Melakukan input jumlah sparepart
    cout << "Masukkan jumlah sparepart : ";
    cin >> quantity;

    // Melakukan input harga sparepart
    cout << "Masukkan total harga per item : ";
    cin >> price;

    // Melakukan validasi semua field
    if (price < 0 || quantity < 0 || nameOfSparepart.empty()) {
        cout << "Masukkan nama, jumlah, total harga sparepart yang valid!" << endl;
    } else {
        // Append semua field ke dalam array masing-masing
        sparepart[currentAmount] = nameOfSparepart;
        quantities[currentAmount] = quantity;
        prices[currentAmount] = price;
        currentAmount++;
        cout << "\nBerhasil menambahkan sparepart!" << endl;
    }
}
```

```

} else {
    cout << "Tidak bisa menambahkan sparepart, inventaris sudah penuh!" << endl;
}

```

Gambar 3.1 Fitur Tambah Sparepart

B. Fitur Tampilkan Sparepart

Fitur ini digunakan untuk menampilkan data spare part dalam format tabel.

Source Code:

```

// Tampilkan Sparepart

// Menampilkan header tabel
cout << "+-----+-----+-----+-----+" << endl;
cout << "| No | Sparepart | Stok | Harga |" << endl;
cout << "+-----+-----+-----+-----+" << endl;
// Menampilkan data sparepart dalam tabel
for (int i = 0; i < currentAmount; i++) {
    cout << "| " << i + 1;
    i + 1 < 10 ? cout << " " : cout << " ";
    cout << "| " << sparepart[i];
    for (int j = sparepart[i].length(); j < 40; j++) {
        cout << " ";
    }
    cout << " | " << quantities[i];
    if (quantities[i] < 10 ) {
        cout << " ";
    } else if (quantities[i] < 100) {
        cout << " ";
    } else {
        cout << " ";
    }
    cout << " | " << prices[i];
    if (prices[i] < 10000) {
        cout << " ";
    } else if (prices[i] < 100000) {
        cout << " ";
    } else {
        cout << " ";
    }
    cout << " |" << endl;
}

```

```

}
// Menutup tabel
if (currentAmount != 0) {
    cout << "+-----+-----+-----+-----+" << endl;
}

```

Gambar 3.2 Fitur Tampilkan Sparepart

C. Fitur Ubah Stok Sparepart

Fitur ini digunakan untuk mengubah data stok dari inventaris. Di dalam fitur ini pengguna diminta memasukkan nomor sparepart data stok sparepart yang baru.

Source Code:

```

int indexSparepart;
// Melakukan input index sparepart
cout << "Masukkan No sparepart : ";
cin >> indexSparepart;
// Memvalidasi index sparepart
if (indexSparepart > 0 && indexSparepart <= currentAmount) {
    int newStcok;
    cout << "Masukkan stok sparepart : ";
    cin >> newStcok;
    if (newStcok < 0) {
        cout << "Gagal memperbarui stok sparepart!";
    } else {
        // Mengubah stok sparepart
        quantities[indexSparepart - 1] = newStcok;
        cout << "Berhasil mengubah stok!" << endl;
    }
} else {
    cout << "Nomor sparepart tidak valid" << endl;
}

```

Gambar 3.3 Fitur Ubah Stok Sparepart

D. Fitur Hapus Sparepart

Fitur ini digunakan untuk menghapus semua data sparepart dari inventaris. Di dalam fitur ini pengguna diminta memasukkan nomor.

Source Code:

```
int indexSparepart;
// Melakukan input index sparepart
cout << "Masukkan No sparepart : ";
cin >> indexSparepart;
// Memvalidasi index sparepart
if (indexSparepart > 0 && indexSparepart <= currentAmount) {
    for (int i = indexSparepart - 1; i < currentAmount - 1; i++) {
        sparepart[i] = sparepart[i + 1];
        quantities[i] = quantities[i + 1];
        prices[i] = prices[i + 1];
    }
    currentAmount--;
    cout << "Berhasil menghapus sparepart!" << endl;
} else {
    cout << "Nomor sparepart tidak valid" << endl;
}
```

Gambar 3.4 Fitur Hapus Sparepart

4. Uji Coba Dan Output

4.1 Uji Coba

1. Skenario 1, pengguna memasukkan username dan password yang benar.
2. Skenario 2, pengguna memasukkan username dan password yang salah sebanyak tiga kali.
3. Skenario 3, pengguna memasukkan username dan password yang benar dan pengguna memasukkan pilihan “1” pada menu utama program.
4. Skenario 4, pengguna memasukkan username dan password yang benar dan pengguna memasukkan pilihan “2” pada menu utama program.
5. Skenario 5, pengguna memasukkan username dan password yang benar dan pengguna memasukkan pilihan “3” pada menu utama program.
6. Skenario 6, pengguna memasukkan username dan password yang benar dan pengguna memasukkan pilihan “4” pada menu utama program.
7. Skenario 7, pengguna memasukkan username dan password yang benar dan pengguna memasukkan pilihan “5” pada menu utama program.
8. Skenario 8, pengguna memasukkan username dan password yang benar dan pengguna memasukkan pilihan yang tidak terdapat pada menu utama program.

4.2 Hasil Output

```
Masukkan Username : fauzan
Masukkan Password : 2409106048

MENU INVENTARIS
1. Tambah Sparepart
2. Tampilkan Sparepart
3. Ubah Stok Sparepart
4. Hapus Sparepart
5. Keluar
Masukkan pilihan Anda : █
```

Gambar 4.1 Skenario 1

```
Masukkan Username : a
Masukkan Password : a
Username atau Password yang Anda masukkan salah, tersisa 2 percobaan
Masukkan Username : a
Masukkan Password : a
Username atau Password yang Anda masukkan salah, tersisa 1 percobaan
Masukkan Username : a
Masukkan Password : a
Kesempatan Anda memasukkan untuk login telah habis, tunggu 15 menit untuk login kembali!
PS C:\Users\VICTUS\Documents\semester 2\praktikum-apl\post-test\post-test-2> █
```

Gambar 4.2 Skenario 2

```
MENU INVENTARIS
1. Tambah Sparepart
2. Tampilkan Sparepart
3. Ubah Stok Sparepart
4. Hapus Sparepart
5. Keluar
Masukkan pilihan Anda : 1
Masukkan nama sparepart : Oli Shell Advance 10w 40 Ultra
Masukkan jumlah sparepart : 20
Masukkan total harga per item : 55000

Berhasil menambahkan sparepart!
```

Gambar 4.3 Skenario 3

```

MENU INVENTARIS
1. Tambah Sparepart
2. Tampilkan Sparepart
3. Ubah Stok Sparepart
4. Hapus Sparepart
5. Keluar
Masukkan pilihan Anda : 2
+---+-----+-----+-----+
| No | Sparepart                | Stok | Harga  |
+---+-----+-----+-----+
| 1  | OLI AHM 800ML           | 100  | 50000  |
| 2  | KAMPAS REM              | 10   | 150000 |
| 3  | Oli Shell Advance 10w 40 Ultra | 20   | 55000  |
+---+-----+-----+-----+

```

Gambar 4.4 Skenario 4

```

MENU INVENTARIS
1. Tambah Sparepart
2. Tampilkan Sparepart
3. Ubah Stok Sparepart
4. Hapus Sparepart
5. Keluar
Masukkan pilihan Anda : 3
+---+-----+-----+-----+
| No | Sparepart                | Stok | Harga  |
+---+-----+-----+-----+
| 1  | OLI AHM 800ML           | 100  | 50000  |
| 2  | KAMPAS REM              | 10   | 150000 |
| 3  | Oli Shell Advance 10w 40 Ultra | 20   | 55000  |
+---+-----+-----+-----+
Masukkan No sparepart : 3
Masukkan stok sparepart : 40
Berhasil mengubah stok!

```

Gambar 4.5 Skenario 5

```

MENU INVENTARIS
1. Tambah Sparepart
2. Tampilkan Sparepart
3. Ubah Stok Sparepart
4. Hapus Sparepart
5. Keluar
Masukkan pilihan Anda : 4
+---+-----+-----+-----+
| No | Sparepart                | Stok | Harga  |
+---+-----+-----+-----+
| 1  | OLI AHM 800ML           | 100  | 50000  |
| 2  | KAMPAS REM              | 10   | 150000 |
| 3  | Oli Shell Advance 10w 40 Ultra | 40   | 55000  |
+---+-----+-----+-----+
Masukkan No sparepart : 2
Berhasil menghapus sparepart!

```

Gambar 4.6 Skenario 6

```

MENU INVENTARIS
1. Tambah Sparepart
2. Tampilkan Sparepart
3. Ubah Stok Sparepart
4. Hapus Sparepart
5. Keluar
Masukkan pilihan Anda : 5
Keluar dari MENU INVENTARIS!
PS C:\Users\VICTUS\Documents\semester 2\praktikum-apl\post-test\post-test-2>

```

Gambar 4.7 Skenario 7

```

MENU INVENTARIS
1. Tambah Sparepart
2. Tampilkan Sparepart
3. Ubah Stok Sparepart
4. Hapus Sparepart
5. Keluar
Masukkan pilihan Anda : 6
Maaf, pilihan Anda tidak tersedia!

```

Gambar 4.8 Skenario 8

5. Git

1. Git Init

```
VICTUS@LAPTOP-L9P3J8CL MINGW64 ~/Documents/semester 2/praktikum-apl
$ git init
Initialized empty Git repository in C:/Users/VICTUS/Documents/Semester 2/praktikum-apl/.git/
```

Gambar 5.1 Git Init

Git init digunakan untuk menginisialisasi sebuah repository git di dalam sebuah direktori

2. Git Add

```
VICTUS@LAPTOP-L9P3J8CL MINGW64 ~/Documents/semester 2/praktikum-apl (main)
$ git add .
```

Gambar 5.2 Git Add

Git Add digunakan untuk menambahkan perubahan file ke dalam staging area sebelum dilakukan commit. “Git add .” digunakan untuk menambahkan semua perubahan file ke dalam staging area.

3. Git Commit

```
VICTUS@LAPTOP-L9P3J8CL MINGW64 ~/Documents/semester 2/praktikum-apl (main)
$ git commit -m "Finish Post Test 2"
[main 7db215d] Finish Post Test 2
2 files changed, 285 insertions(+)
create mode 100644 post-test/post-test-2/2409106048-MiftahulFauzan-PT-2.cpp
create mode 100644 post-test/post-test-2/2409106048-MiftahulFauzan-PT-2.exe
```

Gambar 5.3 Git Commit

Git Commit digunakan untuk menyimpan perubahan yang telah ditambahkan ke staging area ke dalam repository Git.

4. Git Remote

```
VICTUS@LAPTOP-L9P3J8CL MINGW64 ~/Documents/semester 2/praktikum-apl (main)
$ git remote add origin https://github.com/mfthfzn/praktikum-apl
```

Gambar 5.4 Git Remote

Git Remote digunakan untuk mengelola remote repository. Git remote add origin digunakan untuk menambahkan remote repository dengan nama origin ke repository lokal. Remote repository berada di platform github

5. Git Push

```
VICTUS@LAPTOP-L9P3J8CL MINGW64 ~/Documents/semester 2/praktikum-apl (main)
$ git push
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 12 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 677.79 KiB | 8.58 MiB/s, done.
Total 6 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/mfthfzn/praktikum-apl
   b7e9050..7db215d  main -> main
```

Gambar 5.5 Git Push

Git Push digunakan untuk mengupload commit dari repository lokal ke repository remote.