

LAPORAN PRAKTIKUM
POSTTEST 2
ALGORITMA PEMROGRAMAN LANJUT

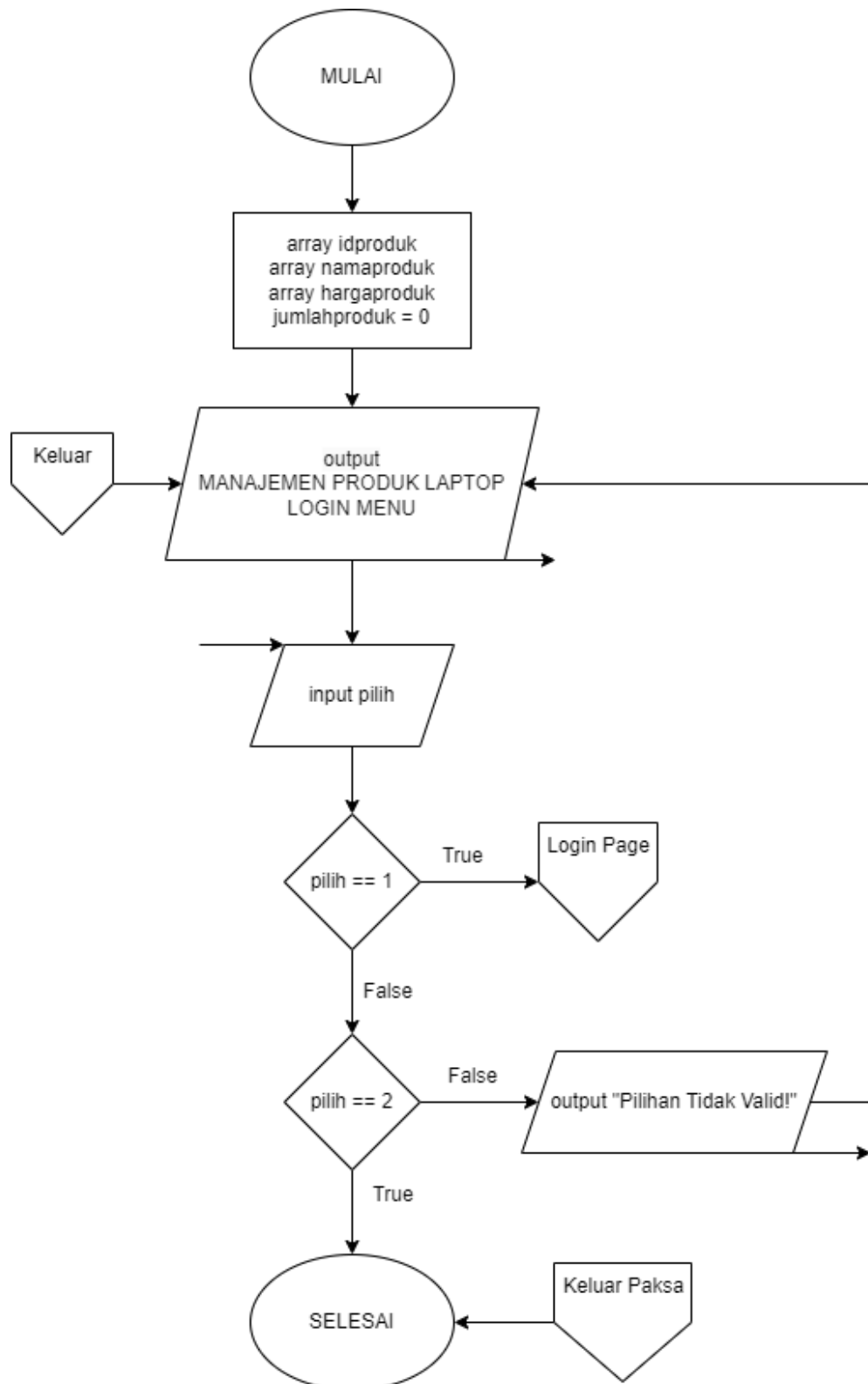


Disusun oleh:
Zeydan Fazle Mawla (2409106010)
Kelas (A1 '24)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

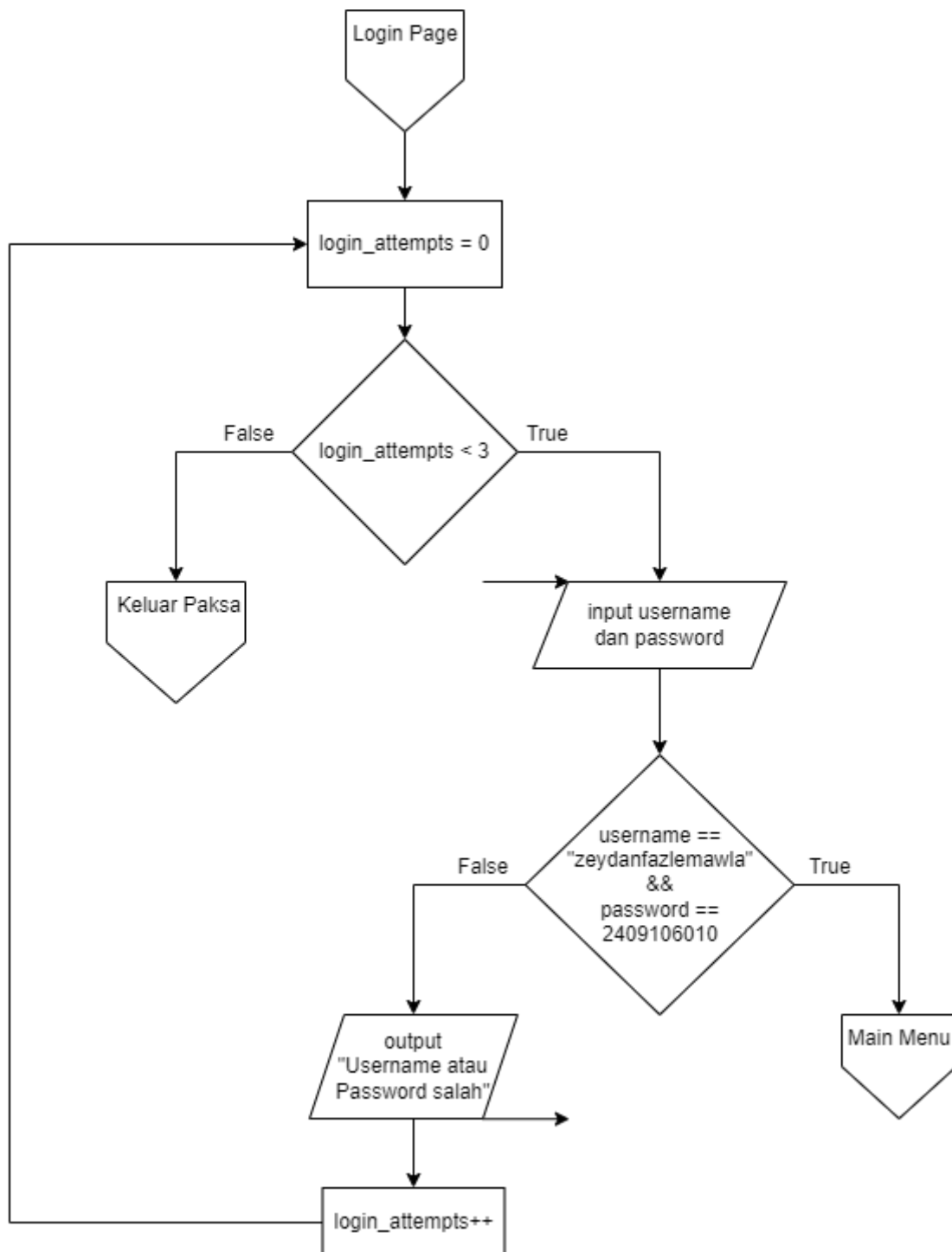
1. Flowchart

1.1 Menu Login



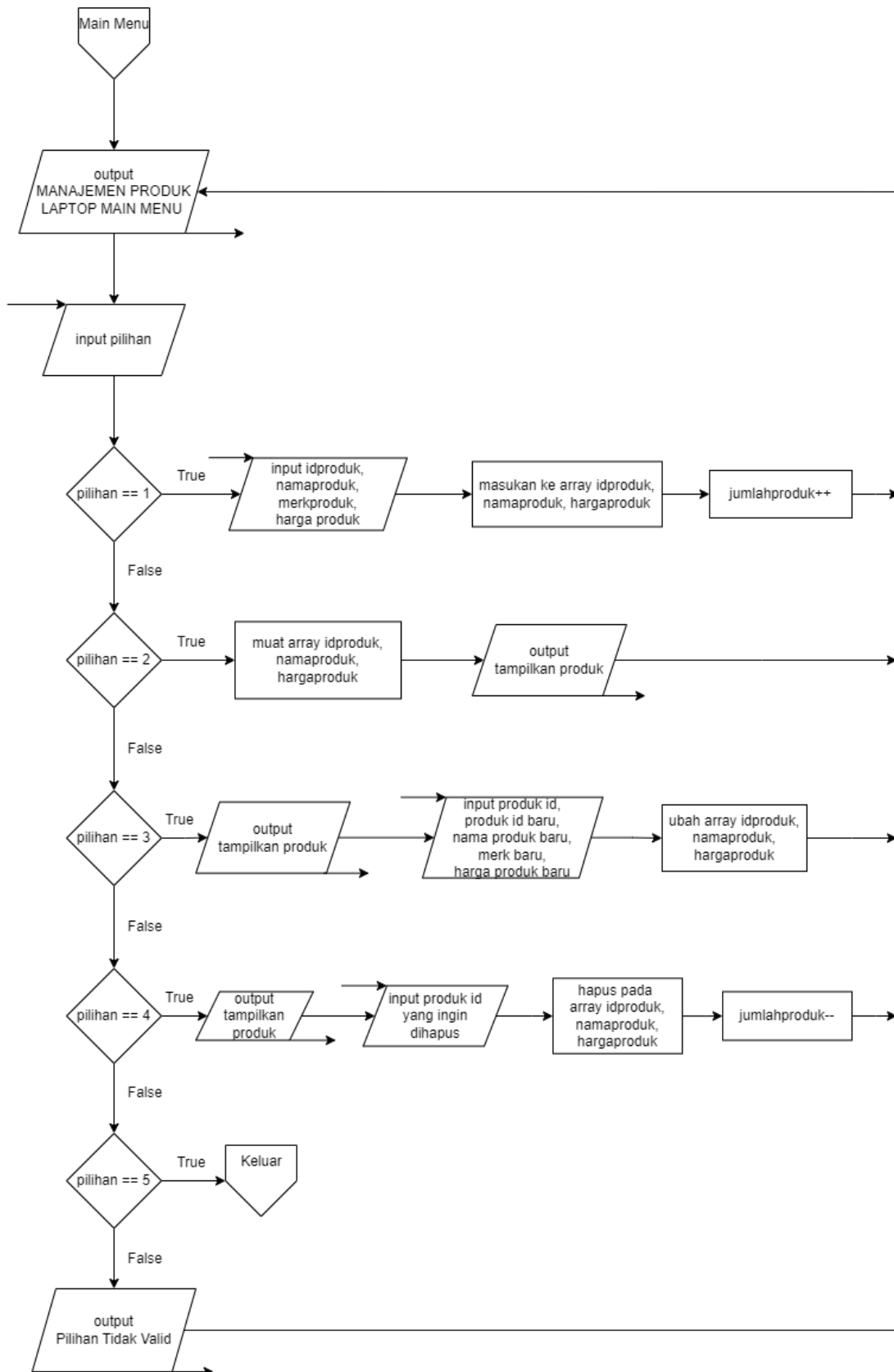
Gambar 1.1 Menu Login

1.2 Menu Validasi Login



Gambar 1.2 Menu Validasi Login

1.3 Menu Utama



Gambar 1.3 Menu Utama

2. Analisis Program

2.1 Deskripsi Singkat Program

Program dibuat untuk manajemen produk laptop yang dapat menambah laptop, melihat daftar laptop, mengubah laptop, dan menghapus laptop setelah pengguna berhasil login ke menu utama. Data produk yang ditambahkan disimpan kedalam masing-masing array yang memuat, id, nama, merek dan harga.

2.2 Penjelasan Alur & Algoritma

1. Tampilan Awal
 - a. Program menampilkan menu berupa pilihan 1 untuk login dan pilihan 2 untuk keluar
 - b. Jika pengguna menginput 1, maka akan masuk ke menu login dan jika menginput 2, maka akan langsung keluar program
 - c. Jika pengguna menginput selain 1 atau 2, maka akan muncul pesan “Pilihan Tidak Valid”
2. Validasi Login
 - a. Pada menu login program meminta untuk menginput username dan password
 - b. Jika username = “zeydanfazlemawla” dan password = 2409106010, maka akan lanjut masuk ke menu utama
 - c. Jika username dan password tidak sesuai maka percobaan login bertambah 1
 - d. Ketika percobaan login = 3 maka program akan langsung terhenti
3. Menu Utama
 - a. Pada menu ini terdapat 5 pilihan yaitu
 - i. Tambah Laptop
 - ii. Lihat Laptop
 - iii. Ubah Laptop
 - iv. Hapus Laptop
 - v. Keluar
 - b. Jika pengguna memilih 1, maka pengguna diminta menginput id, nama, merek dan produk laptop, lalu sistem menyimpannya kedalam array id, nama, merek dan harga

- c. Jika pengguna memilih 2, maka sistem akan menampilkan produk dengan format id, nama, merek, harga yang sebelumnya disimpan didalam array
- d. Jika pengguna memilih 3, maka pengguna diminta menginput id yang akan diubah, selanjutnya menginput id baru, nama baru, merek baru, dan harga baru
- e. Jika pengguna memilih 4, maka pengguna diminta menginput id yang akan dihapus, lalu sistem akan menghapus produk yang sesuai dengan id input pengguna
- f. Jika pengguna memilih 5, maka program akan kembali ke tampilan awal
- g. Jika pengguna menginput selain 1-5, maka akan muncul pesan “Pilihan Tidak Valid”

Fungsi Pada Blok Kode

```
const int MAX_PRODUK = 100;
int idProduk[MAX_PRODUK];
string namaProduk[MAX_PRODUK][2];
int hargaProduk[MAX_PRODUK];
int jumlahProduk = 0;
```

1. `const int MAX_PRODUK = 100;` Mendefinisikan jumlah maksimum produk yang bisa disimpan, yaitu 100 produk. Menggunakan `const` agar nilai tidak bisa diubah
2. `int idProduk[MAX_PRODUK];` Menyimpan id produk
3. `string namaProduk[MAX_PRODUK][2];` Menyimpan nama dan merek produk
4. `int hargaProduk[MAX_PRODUK];` Menyimpan harga produk
5. `int jumlahProduk = 0;` Total produk didalam array

```
int main() {
    string username, password;
    string pilih, pilihan;
    int login_attempts = 0;
    bool login_success = false;
```

1. `int main()` adalah fungsi utama program.
2. Variabel yang digunakan dalam login:
 - a. `username` dan `password` untuk menyimpan kredensial pengguna.
 - b. `login_attempts` untuk membatasi percobaan login.
 - c. `login_success` untuk menentukan apakah login berhasil atau tidak.
3. Variabel `pilih` dan `pilihan` digunakan dalam navigasi menu login dan menu utama

```

while (true) {
    cout << "\n=====\\n";
    cout << "|   MANAJEMEN PRODUK LAPTOP   |\\n";
    cout << "=====\\n";
    cout << "| 1. Login                       |\\n";
    cout << "| 2. Keluar                      |\\n";
    cout << "=====\\n";
    cout << "Masukkan Pilihan: ";
    cin >> pilih;
    cin.ignore();
}

```

1. while (true) membuat loop tak terbatas yang akan terus berjalan sampai dihentikan dengan break.
2. cout digunakan untuk menampilkan menu login pada layar.
3. cin >> pilih digunakan untuk membaca input pengguna dan menyimpannya dalam variabel pilih.
4. cin.ignore(); digunakan untuk mengabaikan karakter yang tersisa di buffer input setelah penggunaan cin >> agar tidak menyebabkan masalah saat membaca input berikutnya.

```

if (pilih == "1") {
    while (login_attempts < 3) {
        cout << "\n=====\\n";
        cout << "\\n|               LOGIN               |\\n";
        cout << "\\n=====\\n";
        cout << "\\nMasukkan Username: ";
        cin >> username;
        cout << "Masukkan Password: ";
        cin >> password;

        if (username == "zeydanfazlemawla" && password == "2409106010")
        {
            login_success = true;
            cout << "Login Berhasil.\\n";
            break;
        } else {
            cout << "Username atau Password salah! Percobaan ke-" <<
login_attempts + 1 << " dari 3\\n";
            login_attempts++;
        }
    }

    if (!login_success) {
        cout << "Kamu gagal login 3 kali. Program berhenti." << endl;
        return 0;
    }
}

```

1. if (pilih == "1") memeriksa apakah pengguna memilih opsi login.
2. while (login_attempts < 3) memastikan pengguna hanya bisa mencoba login maksimal 3 kali.
3. cin >> username dan cin >> password menerima input username dan password dari pengguna.
4. if (username == "zeydanfazlemawla" && password == "2409106010"), Memeriksa apakah username dan password cocok.
5. Jika benar, login_success diatur ke true dan loop berhenti dengan break.
6. Jika salah, program menampilkan peringatan dan menambah jumlah percobaan login (login_attempts++).
7. if (!login_success), jika pengguna gagal login 3 kali, program akan menampilkan pesan gagal dan berhenti (return 0;).

```
while (login_success) {
    cout << "\n=====\\n";
    cout << "|  MANAJEMEN PRODUK LAPTOP MAIN MENU  |\\n";
    cout << "=====\\n";
    cout << "| 1. Tambah Laptop                |\\n";
    cout << "| 2. Lihat Laptop                 |\\n";
    cout << "| 3. Ubah Laptop                  |\\n";
    cout << "| 4. Hapus Laptop                 |\\n";
    cout << "| 5. Logout                       |\\n";
    cout << "=====\\n";
    cout << "Masukkan Pilihan: ";
    cin >> pilihan;
    cin.ignore();
}
```

1. while (true) membuat loop tak terbatas yang akan terus berjalan sampai pengguna memilih logout.
2. cout digunakan untuk menampilkan menu utama
3. cin >> pilihan digunakan untuk menerima input pengguna terkait konversi satuan atau logout.
4. cin.ignore(); digunakan untuk mengabaikan karakter yang tersisa di buffer input setelah penggunaan cin >> agar tidak menyebabkan masalah saat membaca input berikutnya.


```

        if (pilihan == "1") {
            if (jumlahProduk >= MAX_PRODUK) {
                cout << "Data penuh! Tidak bisa menambahkan produk
lagi." << endl;
            } else {
                int idBaru;
                cout << "Masukkan ID Produk: ";
                cin >> idBaru;
                cin.ignore();

                bool idExists = false;
                for (int i = 0; i < jumlahProduk; i++) {
                    if (idProduk[i] == idBaru) {
                        cout << "ID Produk sudah ada! Gunakan ID lain."
<< endl;

                        idExists = true;
                    }
                }
                if (!idExists) {
                    idProduk[jumlahProduk] = idBaru;
                    cout << "Masukkan Merek Produk: ";
                    getline(cin, namaProduk[jumlahProduk][0]);
                    cout << "Masukkan Nama Produk: ";
                    getline(cin, namaProduk[jumlahProduk][1]);
                    cout << "Masukkan Harga Produk: ";
                    cin >> hargaProduk[jumlahProduk];
                    cin.ignore();
                    jumlahProduk++;
                    cout << "Produk berhasil ditambahkan!\n";
                }
            }
        }
    }
}

```

1. `if (jumlahProduk >= MAX_PRODUK);` Memeriksa apakah masih ada ruang untuk menambah produk.
2. `cin >> idBaru; cin.ignore();` Meminta pengguna memasukkan id produk dan membersihkan buffer input agar tidak terjadi kesalahan saat membaca string selanjutnya.
3. `for (int i = 0; i < jumlahProduk; i++);` Melakukan perulangan untuk memeriksa apakah id yang dimasukkan sudah ada dalam daftar produk yang tersimpan.
4. `if (idProduk[i] == idBaru);` Jika id ditemukan, tampilkan pesan bahwa id sudah ada
5. `idProduk[jumlahProduk] = idBaru;` Menyimpan ID produk ke dalam array.
6. `getline(cin, namaProduk[jumlahProduk][0]);` Meminta input merek produk.
7. `getline(cin, namaProduk[jumlahProduk][1]);` Meminta input nama produk.

8. `cin >> hargaProduk[jumlahProduk]; cin.ignore();` Meminta input harga produk dan membersihkan buffer.
9. `jumlahProduk++;` Menambah jumlah produk dalam sistem setelah data berhasil disimpan.
10. `cout << "Produk berhasil ditambahkan!\n";` Memberi tahu pengguna bahwa produk baru berhasil ditambahkan ke dalam sistem.

```

        else if (pilihan == "2") {
            if (jumlahProduk == 0) {
                cout << "Tidak ada produk.\n";
            } else {
                cout << "\nDaftar Produk:\n";
                cout << left << setw(10) << "ID" << setw(20) << "Merek"
<< setw(25) << "Nama" << setw(15) << "Harga" << endl;
                cout <<
                "-----\n";
                for (int i = 0; i < jumlahProduk; i++) {
                    cout << left << setw(10) << idProduk[i] << setw(20)
<< namaProduk[i][0] << setw(25) << namaProduk[i][1] << "Rp " << hargaProduk[i]
<< endl;
                }
            }
        }
    }

```

1. `if (jumlahProduk == 0):` Memeriksa apakah ada produk dalam daftar. Jika tidak ada, tampilkan pesan "Tidak ada produk."
2. `cout << left << setw(10) << "ID" << setw(20) << "Merek" << setw(25) << "Nama" << setw(15) << "Harga";` Mengatur format tampilan tabel dengan lebar kolom yang rapi menggunakan `setw()`.
3. `for (int i = 0; i < jumlahProduk; i++):` Melakukan perulangan untuk menampilkan semua produk yang tersimpan dalam array.
4. `cout << left << setw(10) << idProduk[i] << setw(20) << namaProduk[i][0] << setw(25) << namaProduk[i][1] << "Rp " << hargaProduk[i];` Menampilkan ID, merek, nama, dan harga produk dengan format tabel yang rapi.

```

        else if (pilihan == "3") {
            if (jumlahProduk == 0) {
                cout << "Tidak ada produk yang dapat diubah!\n";
            } else {
                cout << "\nDaftar Produk:\n";
                cout << left << setw(10) << "ID" << setw(20) << "Merek"
<< setw(25) << "Nama" << setw(15) << "Harga" << endl;
                cout <<

```

```

"-----\n";
        for (int i = 0; i < jumlahProduk; i++) {
            cout << left << setw(10) << idProduk[i] << setw(20)
<< namaProduk[i][0] << setw(25) << namaProduk[i][1] << "Rp " << hargaProduk[i]
<< endl;
        }
        int id, index = -1;
        cout << "\nMasukkan ID produk yang ingin diubah: ";
        cin >> id;
        cin.ignore();

        for (int i = 0; i < jumlahProduk; i++) {
            if (idProduk[i] == id) {
                index = i;
            }
        }
        if (index != -1) {
            int idBaru;
            cout << "Masukkan ID baru: ";
            cin >> idBaru;

            bool idSudahAda = false;
            for (int i = 0; i < jumlahProduk; i++) {
                if (idProduk[i] == idBaru && i != index) {
                    cout << "ID baru sudah digunakan oleh produk
lain!\n";

                    idSudahAda = true;
                    break;
                }
            }

            if (!idSudahAda) {
                idProduk[index] = idBaru;
                cin.ignore();
                cout << "Masukkan Merek baru: ";
                getline(cin, namaProduk[index][0]);
                cout << "Masukkan Nama baru: ";
                getline(cin, namaProduk[index][1]);
                cout << "Masukkan Harga baru: ";
                cin >> hargaProduk[index];
                cin.ignore();
                cout << "Produk berhasil diperbarui!\n";
            }
        } else {
            cout << "Produk tidak ditemukan!\n";
        }
    }
}

```

1. if (jumlahProduk == 0) → Memeriksa apakah ada produk yang bisa diubah. Jika tidak ada, tampilkan pesan "Tidak ada produk yang dapat diubah!"
2. Menampilkan tabel dengan ID, merek, nama, dan harga produk menggunakan setw() agar format rapi.
3. for (int i = 0; i < jumlahProduk; i++): Perulangan untuk menampilkan semua produk.
4. cin >> id; cin.ignore(): Meminta pengguna memasukkan id produk dan mengabaikan karakter newline agar tidak mengganggu input selanjutnya.
5. Jika produk ditemukan (index != -1)
 - a. Meminta ID baru dan mengecek apakah id sudah digunakan oleh produk lain.
 - b. Jika ID belum digunakan:
 - c. Memasukkan data baru (merek, nama, harga) dengan getline(cin, namaProduk[index][0]), getline(cin, namaProduk[index][1]), dan cin >> hargaProduk[index].
 - d. Tampilkan pesan sukses: "Produk berhasil diperbarui!"

```

else if (pilihan == "4") {
    if (jumlahProduk == 0) {
        cout << "Tidak ada produk yang dapat dihapus!\n";
    } else {
        cout << "\nDaftar Produk:\n";
        cout << left << setw(10) << "ID" << setw(20) << "Merek"
<< setw(25) << "Nama" << setw(15) << "Harga" << endl;
        cout <<
"-----\n";
        for (int i = 0; i < jumlahProduk; i++) {
            cout << left << setw(10) << idProduk[i] << setw(20)
<< namaProduk[i][0] << setw(25) << namaProduk[i][1] << "Rp " << hargaProduk[i]
<< endl;
        }

        int id, index = -1;
        cout << "\nMasukkan ID produk yang ingin dihapus: ";
        cin >> id;
        cin.ignore();

        for (int i = 0; i < jumlahProduk; i++) {
            if (idProduk[i] == id) {
                index = i;
                break;
            }
        }

        if (index != -1) {
            for (int i = index; i < jumlahProduk - 1; i++) {

```

```

        idProduk[i] = idProduk[i + 1];
        namaProduk[i][0] = namaProduk[i + 1][0];
        namaProduk[i][1] = namaProduk[i + 1][1];
        hargaProduk[i] = hargaProduk[i + 1];
    }

    jumlahProduk--;
    cout << "Produk berhasil dihapus!\n";
} else {
    cout << "Produk tidak ditemukan!\n";
}
}
}

```

1. `cin >> id; cin.ignore();`: Meminta input id produk dan mengabaikan karakter newline.
2. Mencari produk dengan id yang dimasukkan menggunakan perulangan for dan menyimpan indeksinya di variabel index.
3. Jika produk ditemukan (`index != -1`)
 - a. Menggeser elemen array ke kiri mulai dari indeks produk yang dihapus hingga elemen terakhir.
 - b. Mengurangi jumlah produk (`jumlahProduk--`) untuk menghapus data terakhir yang sudah digeser.
 - c. Menampilkan pesan sukses: "Produk berhasil dihapus!"
4. Jika produk tidak ditemukan maka tampilkan pesan "Produk tidak ditemukan!"

```

        else if (pilihan == "5") {
            login_success = false;
        } else {
            cout << "Pilihan tidak valid!\n";
        }
    }
} else if (pilih == "2") {
    cout << "Program selesai." << endl;
    break;
} else {
    cout << "Pilihan tidak valid!" << endl;
}
}
return 0;
}

```

1. `else if (pilihan == "5")`: Keluar dari menu produk dengan mengatur `login_success = false`, yang kemungkinan akan mengembalikan pengguna ke menu login atau keluar dari sesi produk.
2. `else` (Jika input tidak valid): Menampilkan pesan "Pilihan tidak valid!" jika pengguna memasukkan angka selain 1-5.
3. `else if (pilih == "2")`: Menghentikan program dengan `break` jika pengguna memilih opsi keluar dari program.
4. `else` (Jika input utama tidak valid): Jika input utama bukan "1" atau "2", tampilkan pesan "Pilihan tidak valid!"
5. `return 0`: Menandakan bahwa program telah selesai dijalankan tanpa error.

3. Source Code

A. Fitur Menu Login

Fitur ini digunakan untuk menampilkan menu login

Source Code:

```
while (true) {
    cout << "\n=====\\n";
    cout << "|   MANAJEMEN PRODUK LAPTOP   |\\n";
    cout << "=====\\n";
    cout << "| 1. Login                       |\\n";
    cout << "| 2. Keluar                      |\\n";
    cout << "=====\\n";
    cout << "Masukkan Pilihan: ";
    cin >> pilih;
    cin.ignore();
}
```

B. Fitur Validasi Login

Fitur ini digunakan untuk input username dan password dan memvalidasinya

Source Code:

```
if (pilih == "1") {
    while (login_attempts < 3) {
        cout << "\n=====\\n";
        cout << "\\n|               LOGIN               |\\n";
        cout << "\\n=====\\n";
        cout << "\\nMasukkan Username: ";
        cin >> username;
        cout << "Masukkan Password: ";
        cin >> password;

        if (username == "zeydanfazlemawla" && password == "2409106010")
        {
            login_success = true;
            cout << "Login Berhasil.\\n";
            break;
        } else {
            cout << "Username atau Password salah! Percobaan ke-" <<
login_attempts + 1 << " dari 3\\n";
            login_attempts++;
        }
    }
}
```

```

if (!login_success) {
    cout << "Kamu gagal login 3 kali. Program berhenti." << endl;
    return 0;
}

```

C. Fitur Menu Utama

Fitur ini digunakan untuk menampilkan menu utama

Source Code:

```

while (login_success) {
    cout << "\n=====\\n";
    cout << "|  MANAJEMEN PRODUK LAPTOP MAIN MENU  |\\n";
    cout << "=====\\n";
    cout << "| 1. Tambah Laptop                      |\\n";
    cout << "| 2. Lihat Laptop                       |\\n";
    cout << "| 3. Ubah Laptop                        |\\n";
    cout << "| 4. Hapus Laptop                       |\\n";
    cout << "| 5. Logout                            |\\n";
    cout << "=====\\n";
    cout << "Masukkan Pilihan: ";
    cin >> pilihan;
    cin.ignore();
}

```

D. Fitur Tambah Laptop

Fitur ini digunakan untuk menambahkan produk laptop dan memastikan tidak ada id yang sama

Source Code:

```

if (pilihan == "1") {
    if (jumlahProduk >= MAX_PRODUK) {
        cout << "Data penuh! Tidak bisa menambahkan produk
lagi." << endl;
    } else {
        int idBaru;
        cout << "Masukkan ID Produk: ";
        cin >> idBaru;
        cin.ignore();

        bool idExists = false;
        for (int i = 0; i < jumlahProduk; i++) {
            if (idProduk[i] == idBaru) {
                cout << "ID Produk sudah ada! Gunakan ID lain."
<< endl;

                idExists = true;
            }
        }
    }
}

```



```

    }
    if (!idExists) {
        idProduk[jumlahProduk] = idBaru;
        cout << "Masukkan Merek Produk: ";
        getline(cin, namaProduk[jumlahProduk][0]);
        cout << "Masukkan Nama Produk: ";
        getline(cin, namaProduk[jumlahProduk][1]);
        cout << "Masukkan Harga Produk: ";
        cin >> hargaProduk[jumlahProduk];
        cin.ignore();
        jumlahProduk++;
        cout << "Produk berhasil ditambahkan!\n";
    }
}
}

```

E. Fitur Lihat Laptop

Fitur ini digunakan untuk menampilkan produk laptop dalam bentuk tabel

Source Code:

```

        else if (pilihan == "2") {
            if (jumlahProduk == 0) {
                cout << "Tidak ada produk.\n";
            } else {
                cout << "\nDaftar Produk:\n";
                cout << left << setw(10) << "ID" << setw(20) << "Merek"
<< setw(25) << "Nama" << setw(15) << "Harga" << endl;
                cout <<
                "-----\n";
                for (int i = 0; i < jumlahProduk; i++) {
                    cout << left << setw(10) << idProduk[i] << setw(20)
<< namaProduk[i][0] << setw(25) << namaProduk[i][1] << "Rp " << hargaProduk[i]
<< endl;
                }
            }
        }
    }
}

```

F. Fitur Ubah Laptop

Fitur ini digunakan untuk mengubah produk laptop dan memastikan tidak ada id yang sama

Source Code:

```

        else if (pilihan == "3") {
            if (jumlahProduk == 0) {
                cout << "Tidak ada produk yang dapat diubah!\n";
            } else {

```

```

        cout << "\nDaftar Produk:\n";
        cout << left << setw(10) << "ID" << setw(20) << "Merek"
<< setw(25) << "Nama" << setw(15) << "Harga" << endl;
        cout <<
"-----\n";
        for (int i = 0; i < jumlahProduk; i++) {
            cout << left << setw(10) << idProduk[i] << setw(20)
<< namaProduk[i][0] << setw(25) << namaProduk[i][1] << "Rp " << hargaProduk[i]
<< endl;
        }
        int id, index = -1;
        cout << "\nMasukkan ID produk yang ingin diubah: ";
        cin >> id;
        cin.ignore();

        for (int i = 0; i < jumlahProduk; i++) {
            if (idProduk[i] == id) {
                index = i;
            }
        }
        if (index != -1) {
            int idBaru;
            cout << "Masukkan ID baru: ";
            cin >> idBaru;

            bool idSudahAda = false;
            for (int i = 0; i < jumlahProduk; i++) {
                if (idProduk[i] == idBaru && i != index) {
                    cout << "ID baru sudah digunakan oleh produk
lain!\n";

                    idSudahAda = true;
                    break;
                }
            }

            if (!idSudahAda) {
                idProduk[index] = idBaru;
                cin.ignore();
                cout << "Masukkan Merek baru: ";
                getline(cin, namaProduk[index][0]);
                cout << "Masukkan Nama baru: ";
                getline(cin, namaProduk[index][1]);
                cout << "Masukkan Harga baru: ";
                cin >> hargaProduk[index];
                cin.ignore();
                cout << "Produk berhasil diperbarui!\n";
            }
        } else {
            cout << "Produk tidak ditemukan!\n";
        }
    }
}

```

G. Fitur Hapus Laptop

Fitur ini digunakan untuk menghapus produk laptop yang telah ditambahkan

Source Code:

```
        else if (pilihan == "4") {
            if (jumlahProduk == 0) {
                cout << "Tidak ada produk yang dapat dihapus!\n";
            } else {
                cout << "\nDaftar Produk:\n";
                cout << left << setw(10) << "ID" << setw(20) << "Merek"
<< setw(25) << "Nama" << setw(15) << "Harga" << endl;
                cout <<
                "-----\n";
                for (int i = 0; i < jumlahProduk; i++) {
                    cout << left << setw(10) << idProduk[i] << setw(20)
<< namaProduk[i][0] << setw(25) << namaProduk[i][1] << "Rp " << hargaProduk[i]
<< endl;
                }

                int id, index = -1;
                cout << "\nMasukkan ID produk yang ingin dihapus: ";
                cin >> id;
                cin.ignore();

                for (int i = 0; i < jumlahProduk; i++) {
                    if (idProduk[i] == id) {
                        index = i;
                        break;
                    }
                }

                if (index != -1) {
                    for (int i = index; i < jumlahProduk - 1; i++) {
                        idProduk[i] = idProduk[i + 1];
                        namaProduk[i][0] = namaProduk[i + 1][0];
                        namaProduk[i][1] = namaProduk[i + 1][1];
                        hargaProduk[i] = hargaProduk[i + 1];
                    }

                    jumlahProduk--;
                    cout << "Produk berhasil dihapus!\n";
                }
            }
        }
```

H. Fitur Keluar

Fitur ini digunakan untuk keluar program

Source Code:

```
        else if (pilihan == "5") {
            login_success = false;
        } else {
            cout << "Pilihan tidak valid!\n";
        }
    }
} else if (pilih == "2") {
    cout << "Program selesai." << endl;
    break;
} else {
    cout << "Pilihan tidak valid!" << endl;
}
}
return 0;
}
```

4. Uji Coba dan Hasil Output

4.1 Uji Coba

1. Skenario 1: Jika Login Berhasil

Saat user memasukkan username “zeydanfazlemawla” dan password “2409106010”, maka program akan memvalidasi, Jika sesuai maka lanjut dan masuk ke menu berikutnya

2. Skenario 2 : Jika Login Gagal 3 Kali

Saat user memasukkan username selain “zeydanfazlemawla” dan password selain “2409106010” sampai 3 kali, maka program akan berhenti karena percobaan login sudah 3 kali

3. Skenario 3 : Jika Pengguna Memilih Pilihan 1

Pengguna akan diminta menginput id, nama, merek, dan harga laptop, lalu sistem akan menyimpannya didalam array

4. Skenario 4: Jika Pengguna Memilih Pilihan 2

Sistem akan menampilkan produk dengan format id, nama, merek, dan harga dalam bentuk tabel

5. Skenario 5 : Jika Pengguna Memilih Pilihan 3

Sistem akan menampilkan produk dengan format id, nama, merek, dan harga dalam bentuk tabel, lalu pengguna diminta menginput id yang ingin diubah, selanjutnya menginput id baru, nama baru, merek baru, harga baru, dan menyimpannya dalam array

6. Skenario 6 : Jika Pengguna Memilih Pilihan 4

Sistem akan menampilkan produk dengan format id, nama, merek, dan harga dalam bentuk tabel, lalu pengguna diminta menginput id yang ingin dihapus, selanjutnya produk dengan id yang diinput akan terhapus dari array

7. Skenario 7 : Jika Pengguna Memilih Pilihan 1 dan Menginput id yang Sudah Ada

Pengguna akan diminta menginput id, tetapi program akan balik ke menu karena id telah digunakan

8. Skenario 8: Jika Pengguna Memilih Pilihan 2 dan Produk Masih Kosong

Sistem akan menampilkan pesan tidak ada produk

9. Skenario 9 : Jika Pengguna Memilih Pilihan 3 dan Menginput id yang Tidak Ada

Sistem akan menampilkan produk dengan format id, nama, merek, dan harga dalam bentuk tabel, lalu pengguna diminta menginput id yang ingin diubah, selanjutnya program akan balik ke menu karena tidak ada id yang sesuai

10. Skenario 10 : Jika Pengguna Memilih Pilihan 4 dan Menginput id yang Tidak Ada

Sistem akan menampilkan produk dengan format id, nama, merek, dan harga dalam bentuk tabel, lalu pengguna diminta menginput id yang ingin dihapus, selanjutnya program akan balik ke menu karena tidak ada id yang sesuai

11. Skenario 11 : Jika Pengguna Memilih Pilihan 3 dan Menginput id baru yang Sudah Ada

Sistem akan menampilkan produk dengan format id, nama, merek, dan harga dalam bentuk tabel, lalu pengguna diminta menginput id yang ingin diubah, lalu saat input id baru, pengguna menginput id yang sudah ada, selanjutnya program akan balik ke menu karena id sudah digunakan

12. Skenario 11 : Jika Pengguna Ingin Keluar Program

Pada menu utama pengguna menginput 5 untuk logout dan menginput 2 untuk mengakhiri program

4.2 Hasil Output

```
=====
|   MANAJEMEN PRODUK LAPTOP   |
=====
| 1. Login                    |
| 2. Keluar                   |
=====
Masukkan Pilihan: 1

=====
|               LOGIN               |
=====
Masukkan Username: zeydanfazlemawla
Masukkan Password: 2409106010
Login Berhasil.

=====
| MANAJEMEN PRODUK LAPTOP MAIN MENU |
=====
| 1. Tambah Laptop            |
| 2. Lihat Laptop             |
| 3. Ubah Laptop              |
| 4. Hapus Laptop             |
| 5. Logout                   |
=====
Masukkan Pilihan: █
```

Gambar 4.1 Login Berhasil dan Masuk Menu Utama

```
=====
|   MANAJEMEN PRODUK LAPTOP   |
=====
| 1. Login                    |
| 2. Keluar                   |
=====
Masukkan Pilihan: 1

=====
|           LOGIN            |
=====
Masukkan Username: dwiki
Masukkan Password: 001
Username atau Password salah! Percobaan ke-1 dari 3

=====
|           LOGIN            |
=====
Masukkan Username: kingdwiki
Masukkan Password: 2006
Username atau Password salah! Percobaan ke-2 dari 3

=====
|           LOGIN            |
=====
Masukkan Username: admin
Masukkan Password: admin#1234
Username atau Password salah! Percobaan ke-3 dari 3
Kamu gagal login 3 kali. Program berhenti.
```

Gambar 4.2 Program Terhenti Karena Login Gagal


```

=====
|  MANAJEMEN PRODUK LAPTOP MAIN MENU  |
=====
|  1. Tambah Laptop                    |
|  2. Lihat Laptop                     |
|  3. Ubah Laptop                      |
|  4. Hapus Laptop                     |
|  5. Logout                           |
=====
Masukkan Pilihan: 1
Masukkan ID Produk: 112
Masukkan Merek Produk: ASUS
Masukkan Nama Produk: ROG STRIX
Masukkan Harga Produk: 15000000
Produk berhasil ditambahkan!

```

Gambar 4.3 Menu Tambah Laptop

```

=====
|  MANAJEMEN PRODUK LAPTOP MAIN MENU  |
=====
|  1. Tambah Laptop                    |
|  2. Lihat Laptop                     |
|  3. Ubah Laptop                      |
|  4. Hapus Laptop                     |
|  5. Logout                           |
=====
Masukkan Pilihan: 2

Daftar Produk:
ID          Merek          Nama          Harga
-----
112         ASUS           ROG STRIX     Rp 15000000

```

Gambar 4.4 Menu Lihat Laptop

```
=====
|  MANAJEMEN PRODUK LAPTOP MAIN MENU  |
=====
|  1. Tambah Laptop                    |
|  2. Lihat Laptop                     |
|  3. Ubah Laptop                      |
|  4. Hapus Laptop                     |
|  5. Logout                           |
=====
Masukkan Pilihan: 3

Daftar Produk:
ID      Merek      Nama      Harga
-----
112     ASUS       ROG STRIX  Rp 15000000

Masukkan ID produk yang ingin diubah: 112
Masukkan ID baru: 113
Masukkan Merek baru: LENOVO
Masukkan Nama baru: LEGION
Masukkan Harga baru: 20000000
Produk berhasil diperbarui!
```

Gambar 4.5 Menu Ubah Laptop

```
=====
|  MANAJEMEN PRODUK LAPTOP MAIN MENU  |
=====
|  1. Tambah Laptop                    |
|  2. Lihat Laptop                     |
|  3. Ubah Laptop                      |
|  4. Hapus Laptop                     |
|  5. Logout                           |
=====
Masukkan Pilihan: 4

Daftar Produk:
ID      Merek      Nama      Harga
-----
113     LENOVO     LEGION     Rp 20000000

Masukkan ID produk yang ingin dihapus: 113
Produk berhasil dihapus!
```

Gambar 4.6 Menu Hapus Laptop

```
=====
|  MANAJEMEN PRODUK LAPTOP MAIN MENU  |
=====
|  1. Tambah Laptop                    |
|  2. Lihat Laptop                     |
|  3. Ubah Laptop                      |
|  4. Hapus Laptop                     |
|  5. Logout                           |
=====
Masukkan Pilihan: 1
Masukkan ID Produk: 111
ID Produk sudah ada! Gunakan ID lain.
```

Gambar 4.7 Tambah Laptop dengan ID yang Sama

```
=====
|  MANAJEMEN PRODUK LAPTOP MAIN MENU  |
=====
|  1. Tambah Laptop                    |
|  2. Lihat Laptop                     |
|  3. Ubah Laptop                      |
|  4. Hapus Laptop                     |
|  5. Logout                           |
=====
Masukkan Pilihan: 2
Tidak ada produk.
```

Gambar 4.8 Produk Kosong Ketika Melihat Laptop

```
=====
|  MANAJEMEN PRODUK LAPTOP MAIN MENU  |
=====
| 1. Tambah Laptop |
| 2. Lihat Laptop  |
| 3. Ubah Laptop   |
| 4. Hapus Laptop  |
| 5. Logout        |
=====
Masukkan Pilihan: 3

Daftar Produk:
ID      Merek      Nama      Harga
-----
111     ASUS        TUF        Rp 1200000

Masukkan ID produk yang ingin diubah: 1
Produk tidak ditemukan!
```

Gambar 4.9 Menginput ID yang tidak ada dalam Menu Ubah

```
=====
|  MANAJEMEN PRODUK LAPTOP MAIN MENU  |
=====
| 1. Tambah Laptop |
| 2. Lihat Laptop  |
| 3. Ubah Laptop   |
| 4. Hapus Laptop  |
| 5. Logout        |
=====
Masukkan Pilihan: 4

Daftar Produk:
ID      Merek      Nama      Harga
-----
111     ASUS        TUF        Rp 1200000

Masukkan ID produk yang ingin dihapus: 2
Produk tidak ditemukan!
```

Gambar 4.10 Menginput ID yang Tidak Ada dalam Menu Hapus

```

=====
|  MANAJEMEN PRODUK LAPTOP MAIN MENU  |
=====
|  1. Tambah Laptop                    |
|  2. Lihat Laptop                     |
|  3. Ubah Laptop                      |
|  4. Hapus Laptop                     |
|  5. Logout                           |
=====
Masukkan Pilihan: 3

Daftar Produk:
ID          Merek          Nama          Harga
-----
111         ASUS           TUF           Rp 1200000
221         AXIO            PONGO         Rp 10000000

Masukkan ID produk yang ingin diubah: 111
Masukkan ID baru: 221
ID baru sudah digunakan oleh produk lain!

```

Gambar 4.11 Menginput ID Baru yang Sudah Ada dalam Menu Ubah

```

=====
|  MANAJEMEN PRODUK LAPTOP MAIN MENU  |
=====
|  1. Tambah Laptop                    |
|  2. Lihat Laptop                     |
|  3. Ubah Laptop                      |
|  4. Hapus Laptop                     |
|  5. Logout                           |
=====
Masukkan Pilihan: 5

=====
|  MANAJEMEN PRODUK LAPTOP            |
=====
|  1. Login                            |
|  2. Keluar                           |
=====
Masukkan Pilihan: 2
Program selesai.

```

Gambar 4.12 Keluar Program

5. Git

5.1 Git Init

```
● PS D:\praktikum-apl> git init
Initialized empty Git repository in D:/praktikum-apl/.git/
```

Gambar 5.1 Git Init

Command ini berfungsi untuk menginisiasi repository yang ada pada file lokal dan berlokasi di folder .git

5.2 Git Add

```
● PS D:\praktikum-apl> git add .
```

Gambar 5.2 Git Add

Command ini berfungsi untuk menambah file yang ingin dicommit

5.3 Git Commit

```
● PS D:\praktikum-apl> git commit -m "don"
[main e011b22] don
2 files changed, 213 insertions(+)
create mode 100644 post-test/post-test-2/2409106010-ZeydanFazleMawla-PT-2.cpp
create mode 100644 post-test/post-test-2/2409106010-ZeydanFazleMawla-PT-2.exe
```

Gambar 5.3 Git Commit

Command ini berfungsi untuk menyimpan perubahan ke repository lokal git dengan pesan

5.4 Git Remote

```
● PS D:\praktikum-apl> git remote add origin https://github.com/wahyugod/praktikum-apl.git
```

Gambar 5.4 Git Remote

Command ini berfungsi untuk menghubungkan repository yang ada di lokal komputer dengan repository cloud pada github

5.5 Git Push

```
● PS D:\praktikum-apl> git push -u origin main
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), 48.09 KiB | 9.62 MiB/s, done.
Total 6 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/wahyugod/praktikum-apl.git
   87d13da..e011b22  main -> main
branch 'main' set up to track 'origin/main'.
```

Gambar 5.5 Git Push

Command ini berfungsi untuk mengupload semua hal yang ada di repository lokal ke github