

LAPORAN PRAKTIKUM
POSTTEST 3
ALGORITMA PEMROGRAMAN LANJUT

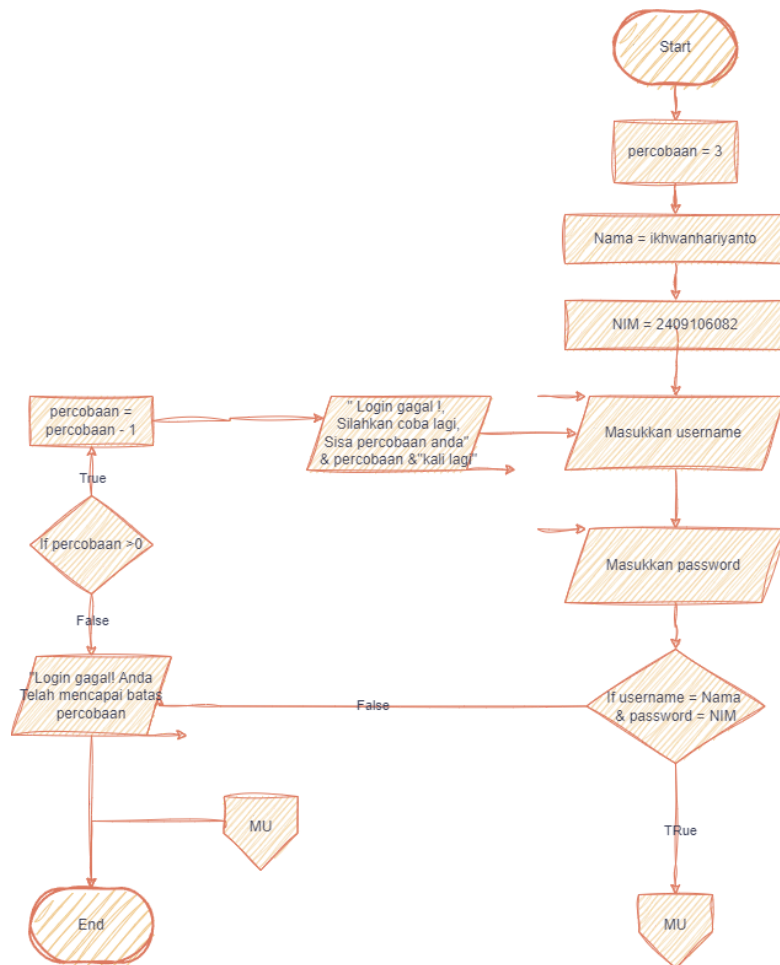


Disusun oleh:
IKHWAN HARIYANTO (2409106082)
Kelas (B2'24)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

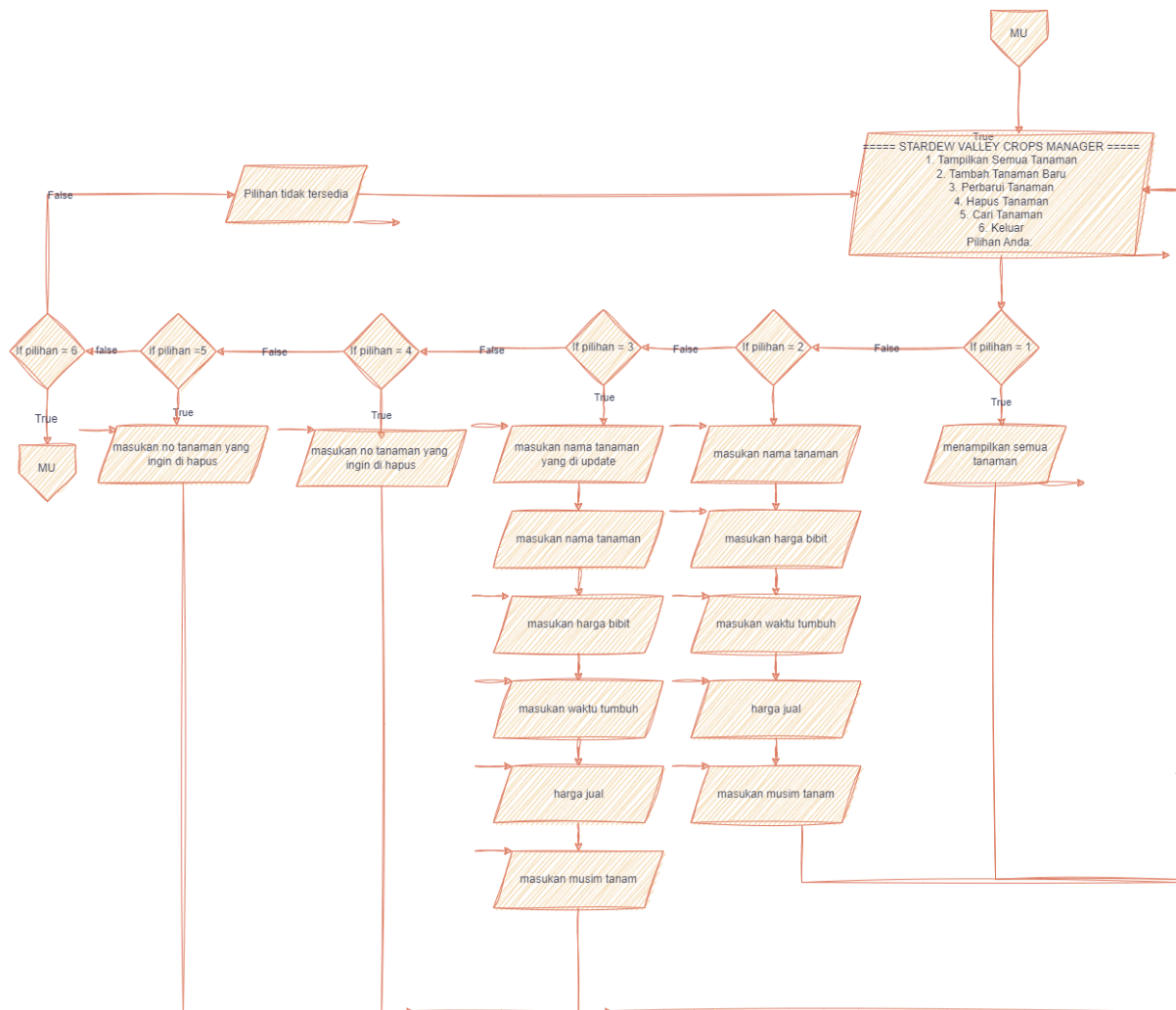
1. Flowchart

1.1 Menu Login



1.1 Flowchat Menu Login

1.2 Menu utama



1.2 Flowchart Menu utama

2. Analisis Program

2.1 Deskripsi Singkat Program

Program ini adalah aplikasi yang dimana dapat membantu pemain stardew valley untuk melihat harga beli dan harga jual dari sebuah tanaman yang ada dalam game tersebut, dan dapat menambahkan dan menghapus tanaman yang tidak relevan

2.2 Penjelasan Alur & Algoritma

Inisialisasi Program

Program dimulai dengan `main()` yang membersihkan layar konsol dan menampilkan header program konversi. Pengguna diarahkan untuk login terlebih dahulu sebelum dapat mengakses fitur utama.

2.1. Inisialisasi Program

Program dimulai dengan:

1. Mengimpor library yang diperlukan (iostream, string, vector, iomanip, limits)
2. Menetapkan kredensial login (username dan NIM)
3. Mendeklarasikan variabel untuk login, data, dan struktur data untuk menyimpan informasi tanaman

2.2. Sistem Login

Alur login:

1. Menampilkan sistem login
2. Meminta input nama dan NIM
3. Memeriksa apakah input sesuai dengan kredensial yang benar
4. Jika benar, login berhasil
5. Jika salah, pengguna diberi kesempatan maksimal 3 kali percobaan
6. Jika gagal setelah 3 percobaan, program berhenti

2.3. Inisialisasi Data Tanaman

Setelah login berhasil, program menginisialisasi data tanaman dengan:

Mendefinisikan berbagai jenis tanaman di Stardew Valley

Menyimpan informasi seperti nama, harga bibit, waktu tumbuh, harga jual, musim tanam, dan tipe panen

Mengelompokkan tanaman berdasarkan musim (Spring, Summer, Fall)

Menghitung jumlah tanaman per musim

2.4. Menu Utama

Program menampilkan menu utama dengan opsi:

1. Tampilkan Semua Tanaman
2. Tambah Tanaman Baru
3. Perbarui Tanaman
4. Hapus Tanaman
5. Cari Tanaman
6. Keluar

2.5. Fungsi-fungsi Utama

2.5.1 Tampilkan Semua Tanaman (Opsi 1)

Membersihkan layar

Menampilkan daftar semua tanaman dalam format tabel

Menampilkan statistik jumlah tanaman per musim

2.5.2 Tambah Tanaman Baru (Opsi 2)

Meminta input untuk detail tanaman baru (nama, harga bibit, waktu tumbuh, harga jual, musim, tipe panen) Menambahkan data ke dalam vector yang sesuai

Memperbarui statistik jumlah tanaman per musim

2.5.3 Perbarui Tanaman (Opsi 3)

Menampilkan daftar tanaman untuk referensi
Meminta nomor tanaman yang akan diperbarui
Memperbarui data tanaman yang dipilih
Memperbarui statistik jumlah tanaman per musim

2.5.4 Hapus Tanaman (Opsi 4)

Menampilkan daftar tanaman untuk referensi
Meminta nomor tanaman yang akan dihapus
Meminta konfirmasi penghapusan
Jika dikonfirmasi, menghapus tanaman dari semua vector
Memperbarui statistik jumlah tanaman per musim

2.5.5 Cari Tanaman (Opsi 5)

Menyediakan opsi pencarian berdasarkan:

1. Nama tanaman
2. Musim tanam
3. Tipe panen

Menampilkan hasil pencarian dalam format tabel

2.5.6 Keluar (Opsi 6)

Menampilkan pesan terima kasih
Mengakhiri program

2.6. Validasi Input

Program melakukan validasi input untuk:
Memastikan input berupa angka saat memilih menu
Memastikan indeks tanaman valid saat memperbarui atau menghapus
Memastikan format data yang dimasukkan sesuai

3. Source Code

3.1 Menu Login

menu ini adalah menu awal saat pengguna memulai program pengguna akan memasukan username dan password yang telah di tentukan jika pengguna salah 3 kali dalam memasukan password dan username maka program akan terhenti dan jika berhasil maka pengguna akan memasuki menu utama.

```
// Login system
cout << "==== LOGIN SYSTEM =====" << endl;

while (loginAttempts < 3 && !isLoggedIn) {
    cout << "Masukkan Nama: ";
    cin >> inputName;
    cout << "Masukkan NIM: ";
    cin >> inputNim;

    // Check login against all users
    for (const User& user : users) {
        if (inputName == user.name && inputNim == user.nim) {
            cout << "Login berhasil!" << endl;
            isLoggedIn = true;
            break;
        }
    }

    if (!isLoggedIn) {
        loginAttempts++;
#ifdef _WIN32
        system("cls");
#else
        system("clear");
#endif

        cout << "Login gagal! Percobaan ke-" << loginAttempts << " dari
3" << endl;
    }
}
```

gambar 3.1 menu login

3.2 Menu utama

menu ini adalah menu yang bisa pengguna pilih dan menampilkan menu dan pengguna dapat meninput sesuai angka yang di input.

```
while (running) {  
    // Tampilkan menu  
    cout << "\n===== STARDEW VALLEY CROPS MANAGER =====" << endl;  
    cout << "1. Tampilkan Semua Tanaman" << endl;  
    cout << "2. Tambah Tanaman Baru" << endl;  
    cout << "3. Perbarui Tanaman" << endl;  
    cout << "4. Hapus Tanaman" << endl;  
    cout << "5. Cari Tanaman" << endl;  
    cout << "6. Keluar" << endl;  
    cout << "Pilihan Anda: ";  
  
    // Validasi input  
    if (!(cin >> choice)) {  
        cin.clear();  
        cin.ignore(numeric_limits<streamsize>::max(), '\n');  
        cout << "Input tidak valid! Masukkan angka." << endl;  
        continue;  
    }  
}
```

gambar 3.2 Menu utama

3.3 menu penampilkkan semua tanaman

menu ini adalah menu yang menampilkan seluruh data tanaman yang ada dalam program.


```

    cout << string(36, '=') << " DAFTAR SEMUA TANAMAN " << string(36, '=') << endl;

    cout << left << setw(5) << "No" << setw(20) << "Nama"
        << setw(15) << "Harga Bibit" << setw(15) << "Waktu Tumbuh"
        << setw(15) << "Harga Jual" << setw(15) << "Musim"
        << setw(10) << "Tipe" << endl;
    cout << string(94, '-') << endl;

    for (size_t i = 0; i < cropNames.size(); i++) {
        cout << left << setw(5) << (i + 1)
            << setw(20) << cropNames[i]
            << setw(15) << seedPrices[i]
            << setw(15) << growthTimes[i]
            << setw(15) << sellingPrices[i]
            << setw(15) << growSeasons[i]
            << setw(10) << harvestTypes[i] << endl;
    }

```

gambar 3.3

3.4 penambahan item tanaman

menu ini adalah menu penambahan tanaman baru pengguna dapat menginput nama tanaman yang ingin di tambahkan dan pengguna akan di beri arahan untuk mengisi data data yang lain

```

else if (choice == 2) {
    // Add new crop
    Crop newCrop;

    cin.ignore();
    cout << "\n==== TAMBAH TANAMAN BARU =====" << endl;
    cout << "Nama Tanaman: ";
    getline(cin, newCrop.name);

    cout << "Harga Bibit: ";
    cin >> newCrop.seedPrice;

    cout << "Waktu Tumbuh (hari): ";
    cin >> newCrop.growthTime;

```

```

        cout << "Harga Jual: ";
        cin >> newCrop.sellingPrice;

        cin.ignore();
        cout << "Musim Tanam (Spring/Summer/Fall): ";
        getline(cin, newCrop.growSeason);

        cout << "Tipe Harvest (single/multiple): ";
        getline(cin, newCrop.harvestType);

        // Add data to array
        crops.push_back(newCrop);

        // Update statistics
        if (newCrop.growSeason == "Spring") {
            cropsPerSeason[0]++;
        } else if (newCrop.growSeason == "Summer") {
            cropsPerSeason[1]++;
        } else if (newCrop.growSeason == "Fall") {
            cropsPerSeason[2]++;
        } else if (newCrop.growSeason == "Summer, Fall") {
            cropsPerSeason[1]++;
            cropsPerSeason[2]++;
        }

        cout << "Tanaman berhasil ditambahkan!" << endl;
    }
}

```

gambar 3.4

3.5 perbarui tanaman

menu ini adalah menu pemberbaruan data tanaman yang ada dengan data yang ingin di perbarui.

```

cout << "\n==== PERBARUI TANAMAN =====" << endl;
cout << "Daftar Tanaman:" << endl;
for (size_t i = 0; i < cropNames.size(); i++) {
    cout << (i + 1) << ". " << cropNames[i] << endl;
}

cout << "Masukkan nomor tanaman yang ingin diperbarui: ";
cin >> index;

```

```

if (index < 1 || index > static_cast<int>(cropNames.size())) {
    cout << "Nomor tanaman tidak valid!" << endl;
    continue;
}

index--;

if (growSeasons[index] == "Spring") {
    cropsPerSeason[0]--;
} else if (growSeasons[index] == "Summer") {
    cropsPerSeason[1]--;
} else if (growSeasons[index] == "Fall") {
    cropsPerSeason[2]--;
} else if (growSeasons[index] == "Summer, Fall") {
    cropsPerSeason[1]--;
    cropsPerSeason[2]--;
}

```

gambar 3.5

3.6 hapus tanaman

menu ini adalah menu penghapusan data tanaman yang dimana pengguna dapat menghapus data tanaman yang bisa di hapus.

```

else if (choice == 4) {
    // Hapus tanaman
    int index;

    cout << "\n===== HAPUS TANAMAN =====" << endl;

    // Tampilkan daftar singkat untuk referensi
    cout << "Daftar Tanaman:" << endl;
    for (size_t i = 0; i < cropNames.size(); i++) {
        cout << (i + 1) << ". " << cropNames[i] << endl;
    }

    cout << "Masukkan nomor tanaman yang ingin dihapus: ";
    cin >> index;

    if (index < 1 || index > static_cast<int>(cropNames.size())) {
        cout << "Nomor tanaman tidak valid!" << endl;
        continue;
    }
}

```

```

        index--; // Konversi ke indeks array (0-based)

        // Hapus dari statistik
        if (growSeasons[index] == "Spring") {
            cropsPerSeason[0]--;
        } else if (growSeasons[index] == "Summer") {
            cropsPerSeason[1]--;
        } else if (growSeasons[index] == "Fall") {
            cropsPerSeason[2]--;
        } else if (growSeasons[index] == "Summer, Fall") {
            cropsPerSeason[1]--;
            cropsPerSeason[2]--;
        }
    }

    cout << "Tanaman " << cropNames[index] << " akan dihapus. Lanjutkan?
(y/n): ";
    char confirmation;
    cin >> confirmation;

    if (confirmation == 'y' || confirmation == 'Y') {

```

gambar 3.6

3.7 pencarian nama tanaman

menu pencarian tanaman, pengguna dapat mencari tanaman berdasarkan nama ataupun musim.

```

    cout << "\n===== CARI TANAMAN =====" << endl;
    cout << "1. Cari berdasarkan nama" << endl;
    cout << "2. Cari berdasarkan musim" << endl;
    cout << "3. Cari berdasarkan tipe harvest" << endl;
    cout << "Pilihan: ";

    int searchChoice;
    cin >> searchChoice;
    cin.ignore();

    if (searchChoice == 1) {
        cout << "Masukkan nama tanaman: ";
        getline(cin, searchTerm);

        cout << "\nHasil Pencarian untuk '" << searchTerm << "':" <<
endl;

        cout << left << setw(5) << "No" << setw(20) << "Nama"
            << setw(15) << "Harga Bibit" << setw(15) << "Waktu
Tumbuh"
            << setw(15) << "Harga Jual" << setw(15) << "Musim"

```

```
<< setw(10) << "Tipe" << endl;  
cout << string(95, '-') << endl;
```

gambar 3.7

3.7 data tanaman

menu pencarian tanaman, pengguna dapat mencari tanaman berdasarkan nama ataupun musim.

```
crops.push_back({"Blue Jazz", 30, 7, 50, "Spring", "single"});  
cropsPerSeason[0]++;  
  
crops.push_back({"Cauliflower", 80, 12, 175, "Spring", "single"});  
cropsPerSeason[0]++;  
  
crops.push_back({"Garlic", 40, 4, 60, "Spring", "single"});  
cropsPerSeason[0]++;  
  
crops.push_back({"Kale", 70, 6, 110, "Spring", "single"});  
cropsPerSeason[0]++;  
  
crops.push_back({"Parsnip", 20, 4, 35, "Spring", "single"});  
cropsPerSeason[0]++;  
  
crops.push_back({"Potato", 50, 6, 80, "Spring", "single"});  
cropsPerSeason[0]++;
```

gambar 3.8

A. Fitur Tambah Data

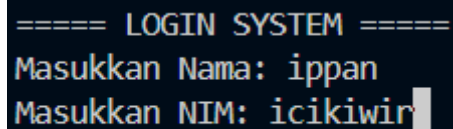
Fitur ini digunakan untuk membersihkan output sebelum program ini berjalan

```
#ifdef _WIN32
system("cls");
#else
system("clear");
#endif
```

4.1 Uji Coba

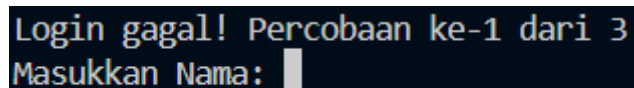
(Jelaskan skenario yang digunakan untuk menguji program, misalnya dengan berbagai jenis input.)

4.2 Hasil Output



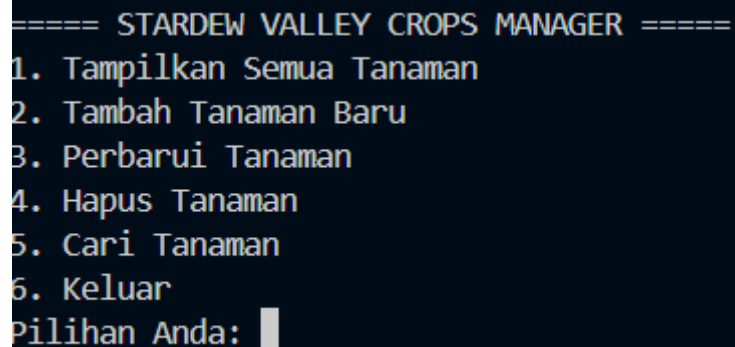
```
==== LOGIN SYSTEM ====
Masukkan Nama: ippan
Masukkan NIM: icikiwir
```

gambar 4.2 menu login salah



```
Login gagal! Percobaan ke-1 dari 3
Masukkan Nama:
```

gambar 4.2 output password salah



```
==== STARDEW VALLEY CROPS MANAGER ====
1. Tampilkan Semua Tanaman
2. Tambah Tanaman Baru
3. Perbarui Tanaman
4. Hapus Tanaman
5. Cari Tanaman
6. Keluar
Pilihan Anda:
```

gambar 4.3 menu utama

===== DAFTAR SEMUA TANAMAN =====						
No	Nama	Harga Bibit	Waktu Tumbuh	Harga Jual	Musim	Tipe
1	Blue Jazz	30	7	50	Spring	single
2	Cauliflower	80	12	175	Spring	single
3	Garlic	40	4	60	Spring	single
4	Kale	70	6	110	Spring	single
5	Parsnip	20	4	35	Spring	single
6	Potato	50	6	80	Spring	single
7	Rhubarb	100	13	220	Spring	single
8	Tulip	20	6	30	Spring	single
9	Unmilled Rice	40	6	30	Spring	single
10	Carrot	0	3	35	Spring	single
11	Coffee Bean	2500	10	60	Spring, Summer	multiple
12	Green Bean	60	10	40	Spring	multiple
13	Strawberry	100	8	120	Spring	multiple
14	Melon	80	12	250	Summer	single
15	Poppy	100	7	140	Summer	single
16	Radish	40	6	90	Summer	single
17	Red Cabbage	100	9	260	Summer	single
18	Starfruit	400	13	750	Summer	single
19	Summer Spangle	50	8	90	Summer	single
20	Wheat	10	4	25	Summer, Fall	single
21	Sunflower	200	8	80	Summer, Fall	single
22	Blueberry	80	13	150	Summer	multiple
23	Corn	150	14	50	Summer, Fall	multiple
24	Hops	50	11	25	Summer	multiple
25	Hot Pepper	40	5	40	Summer	multiple
26	Tomato	50	11	60	Summer	multiple
27	Summer Squash	0	6	45	Summer	multiple

gambar 4.4 menampilkan daftar tumbuhan

```

===== TAMBAH TANAMAN BARU =====
Nama Tanaman: spiderlily
Harga Bibit: 2000
Waktu Tumbuh (hari): 19
Harga Jual: 99999
Musim Tanam (Spring/Summer/Fall): fall
Tipe Harvest (single/multiple): single
Tanaman berhasil ditambahkan!

```

gambar 4.5 penambahan tanaman

```

Masukkan nomor tanaman yang ingin diperbarui: 38
Nama Tanaman [Grape]: anggur merah di tepi sungai
Harga Bibit [60]: 150
Waktu Tumbuh [10]: 11
Harga Jual [80]: 80
Musim Tanam [Fall]: fall
Tipe Harvest [multiple]: multiple
Tanaman berhasil diperbarui!

```

gambar 4.6 memperbarui tanaman

```
38. anggur merah di tepi sungai
39. spiderlily
Masukkan nomor tanaman yang ingin dihapus: 38
Tanaman anggur merah di tepi sungai akan dihapus. Lanjutkan? (y/n): y
Tanaman berhasil dihapus!
```

gambar 4.6 penghapusan tanaman

```
===== CARI TANAMAN =====
1. Cari berdasarkan nama
2. Cari berdasarkan musim
3. Cari berdasarkan tipe harvest
Pilihan: 1
Masukkan nama tanaman: Tomato

Hasil Pencarian untuk 'Tomato':
No  Nama          Harga Bibit  Waktu Tumbuh  Harga Jual  Musim  Tipe
-----
1   Tomato        50          11           60          Summer  multiple
```

gambar 4.6 pencarian tanaman

```
===== STARDEW VALLEY CROPS MANAGER =====
1. Tampilkan Semua Tanaman
2. Tambah Tanaman Baru
3. Perbarui Tanaman
4. Hapus Tanaman
5. Cari Tanaman
6. Keluar
Pilihan Anda: 6
Terima kasih telah menggunakan program Stardew Valley Crops Manager!
```

gambar 4.6 logout

5. Langkah Langkah Git

5.1 Git Init

Perintah `git init` memungkinkan pengguna untuk menginisialisasi repository baru di dalam suatu direktori kerja. Setelah perintah ini dijalankan, pengguna dapat mulai melacak perubahan yang terjadi pada berbagai file.

```
Lenovo-GK@ippan MINGW64 ~/Desktop/praktikum-APL/posttest/post-test-apl-1 (main)
$ git init
Initialized empty Git repository in C:/Users/Lenovo-GK/Desktop/praktikum-APL/posttest/post-test-apl-1/.git/
```

Gambar 5.1 Git Init

5.2 Git Add

Perintah "git add" digunakan untuk menambahkan file yang ingin Anda komit.

```
Lenovo-GK@ippan MINGW64 ~/Desktop/praktikum-APL/posttest/post-test-apl-1 (main)  
$ git add .
```

5.2 Git Add

5.3 Git Commit

Commit adalah tindakan menyimpan perubahan kode ke dalam repositori lokal.

```
Lenovo-GK@ippan MINGW64 ~/Desktop/praktikum-APL/posttest/post-test-apl-1 (main)  
$ git commit -m "ganti nama file"
```

5.3 Git Commit

5.4 Git Remote

Git Remote adalah perintah yang digunakan untuk menghubungkan repository lokal dengan repository yang ada di GitHub.

```
Lenovo-GK@ippan MINGW64 ~/Desktop/praktikum-APL/posttest/post-test-apl-1 (main)  
$ git remote add origin https://github.com/ippanicikiwir/KELAS-B24/tree/main/praktikum-apl/posttest/post-test-apl-1
```

5.4 Git Remote

5.5 Git Push

Push adalah perintah yang digunakan untuk mengirim commit dari repositori lokal ke repositori jarak jauh (server).

```
Lenovo-GK@ippan MINGW64 ~/Desktop/praktikum-APL/posttest/post-test-apl-1 (main)  
$ git push -u origin main
```

5.5 Git Push