

## **PROYEK AKHIR SEMESTER GASAL**



NAMA : VINA SUKASIH  
KELAS : X PPLG 1  
NIS : 258741

JUDUL PROYEK : APLIKASI SMARTFINANCE TRACKER  
(Aplikasi Manajemen Keuangan Sederhana)

PENGEMBANGAN PERANGKAT LUNAK DAN GIM  
**SMK NEGERI 1 KANDEMAN**  
TAHUN PELAJARAN 2025/2026

## DESKRIPSI PROYEK

SmartFinance Tracker adalah sebuah aplikasi console berbasis C# yang berfungsi sebagai alat bantu pencatatan keuangan sederhana. Melalui aplikasi ini, pengguna dapat mencatat pemasukan serta pengeluaran, melihat daftar transaksi yang telah dimasukkan, dan menampilkan ringkasan saldo akhir secara otomatis. Semua transaksi disimpan secara lokal selama program berjalan tanpa memerlukan database atau koneksi internet.

Melalui aplikasi ini, pengguna dapat:

1. Menambahkan transaksi pemasukan dan pengeluaran.
2. Melihat riwayat transaksi yang telah dicatat.
3. Melihat laporan keuangan berupa total pemasukan, total pengeluaran, dan saldo akhir.
4. Menggunakan navigasi menu interaktif berbasis teks.

Proyek ini mengimplementasikan **konsep dasar pemrograman** seperti:

1. **Variabel dan Tipe Data** Digunakan untuk menyimpan data transaksi seperti jenis, nominal, dan keterangan. Contoh: int, double, string.
2. **Operator** Penggunaan operator membantu proses perhitungan dan logika secara efisien.
3. **Perulangan (looping)** Program menggunakan for untuk menampilkan dan menghitung transaksi, serta do-while untuk menjalankan menu secara berulang hingga pengguna memilih keluar.
4. **Percabangan** Penggunaan percabangan memungkinkan program merespon input yang bebeda secara dinamis.
5. **Array atau List** Digunakan sebagai struktur data, data transaksi disimpan dalam array seperti jenisTransaksi[], nominalTransaksi[], dan keterangan[].
6. **Fungsi/Method** Program menggunakan pemecahan fungsi seperti: Loading() untuk animasi loading, Tampilkan Transaksi() untuk menampilkan daftar transaksi, TampilkanRingkasan() untuk menghitung dan menampilkan total keuangan.
7. **Input dan Output Console** Input diterima menggunakan Console.ReadLine() dan output ditampilkan dengan Console.WriteLine() untuk interaksi pengguna.

## RINGKASAN TEORI C# YANG DIGUNAKAN

### 1. Variabel dan Tipe Data

Digunakan untuk menyimpan data transaksi seperti jenis, nominal, dan keterangan. Contoh: int, double, string.

```
string[] jenisTransaksi = new string[100];
double[] nominalTransaksi = new double[100];
string[] keterangan = new string[100];
int jumlahTransaksi = 0;
int menu;
```

- string → menyimpan teks (jenis transaksi & keterangan).
- int → menyimpan bilangan bulat (jumlah transaksi, pilihan menu).
- double → menyimpan angka desimal (nominal uang).

### 2. Operator

Penggunaan operator membantu proses perhitungan dan logika secara efisien.

Contoh:

```
totalPemasukan += transaksi[i];
jumlahTransaksi++;
```

Jenis operator yang digunakan:

- **Aritmatika:** +, -
- **Perbandingan:** ==
- **Penugasan:** =
- **Logika:** !
- **Increment:** ++

### 3. Percabangan (Kondisi)

Penggunaan percabangan memungkinkan program merespon input yang berbeda secara dinamis.

Contoh:

```
if(jenis[i].ToLower() == "pemasukan")
    totalPemasukan += transaksi[i];
```

```
switch(menu)
{
    case 1:
        // Tambah transaksi
        break;
}
```

### 4. Perulangan (Looping)

Program menggunakan for untuk menampilkan dan menghitung transaksi, serta do-while untuk menjalankan menu secara berulang hingga pengguna memilih keluar.

Contoh for:

```
for (int i = 0; i < jumlah; i++)
{
    Console.WriteLine($"{i + 1}. [{jenis[i]}] Rp {nominal[i]} - {keterangan[i]}");
```

Contoh do-while:

```
do
{
    // menu program
} while (menu != 4);
```

### 5. Array atau List

Data transaksi disimpan dalam array seperti jenisTransaksi[], nominalTransaksi[], dan keterangan[].

Contoh:

```
string[] jenisTransaksi = new string[100];
double[] nominalTransaksi = new double[100];
string[] keterangan = new string[100];
```

## 6. Fungsi atau Method

Program menggunakan pemecahan fungsi seperti: Loading() untuk animasi loading, TampilkanRingkasan() untuk menghitung dan menampilkan total keuangan. Contoh:

```
static void Loading(string teks)
{
    Console.WriteLine(teks);
    for (int i = 0; i < 3; i++)
    {
        Console.Write(".");
        Thread.Sleep(300);
    }
}

static void TampilkanRingkasan(double[] transaksi, string[]
jenis, int jumlah)
```

## 7. Input dan Output

Input dan Output Console input diterima menggunakan Console.ReadLine() dan output ditampilkan dengan Console.WriteLine() untuk interaksi pengguna. Contoh Input:

```
Console.WriteLine("Nominal (Rp) : ");
nominalTransaksi[jumlahTransaksi] = double.Parse(Console.ReadLine());
```

Contoh Output:

```
Console.WriteLine("==== RINGKASAN KEUANGAN ===");
```

## PERANCANGAN PROGRAM

Rancangan Tampilan Aplikasi SmartFinance Tracker

```
WELCOME TO SMARTFINANCE TRACKER
=====
 
==== MENU UTAMA ====
1. Tambah Transaksi
2. Lihat Riwayat Transaksi
3. Lihat Ringkasan Saldo
4. Keluar
Pilih menu (1-4): 1

==== TAMBAH TRANSAKSI ====
Jenis (Pemasukan/Pengeluaran): Pemasukan
Nominal (Rp): 1.500.000
Keterangan: gaji bulan november
Menyimpan data...
Transaksi berhasil disimpan!

==== MENU UTAMA ====
1. Tambah Transaksi
2. Lihat Riwayat Transaksi
3. Lihat Ringkasan Saldo
4. Keluar
Pilih menu (1-4): 1

==== TAMBAH TRANSAKSI ====
Jenis (Pemasukan/Pengeluaran): Pengeluaran
Nominal (Rp): 245.000
Keterangan: untuk membeli beras dan buah
Menyimpan data...
Transaksi berhasil disimpan!

==== MENU UTAMA ====
1. Tambah Transaksi
2. Lihat Riwayat Transaksi
3. Lihat Ringkasan Saldo
4. Keluar
Pilih menu (1-4): 3

==== RINGKASAN KEUANGAN ====
Total Pemasukan : Rp 1500000
Total Pengeluaran : Rp 245000
Saldo Akhir : Rp 1255000

==== MENU UTAMA ====
1. Tambah Transaksi
2. Lihat Riwayat Transaksi
3. Lihat Ringkasan Saldo
4. Keluar
Pilih menu (1-4): 4

Menutup aplikasi...
Terima kasih telah menggunakan SmartFinance!
```

## IMPLEMENTASI PROGRAM

Kode program utama  
using System;

```
namespace Projek_Akhir_SmartFinance_vinasukasih_xpplg1
{
    internal class Program
    {
        // Fungsi untuk menampilkan teks dengan efek loading
        static void Loading(string teks)
        {
            Console.WriteLine(teks);
            for (int i = 0; i < 3; i++)
            {
                Console.Write(".");
                Thread.Sleep(300);
            }
            Console.WriteLine();
        }

        // Fungsi untuk menampilkan ringkasan pemasukan, pengeluaran, dan saldo
        static void TampilanRingkasan(double[] transaksi, string[] jenis, int jumlah)
        {
            double totalPemasukan = 0, totalPengeluaran = 0;

            // Menghitung total pemasukan & total total pengeluaran
            for (int i = 0; i < jumlah; i++)
            {
                if (jenis[i].ToLower() == "pemasukan")
                    totalPemasukan += transaksi[i];

                else if (jenis[i].ToLower() == "pengeluaran")
                    totalPengeluaran += transaksi[i];
            }

            double saldoAkhir = totalPemasukan - totalPengeluaran;

            // Menampilkan hasil ringkasan
            Console.ForegroundColor = ConsoleColor.Yellow;
            Console.WriteLine("\n==== RINGKASAN KEUANGAN ====");
            Console.ResetColor();
            Console.WriteLine($"Total Pemasukan : Rp {totalPemasukan}");
            Console.WriteLine($"Total Pengeluaran : Rp {totalPengeluaran}");
            Console.WriteLine($"Saldo Akhir : Rp {saldoAkhir}");
        }
}
```

```

// Fungsi untuk menampilkan semua transaksi
static void TampilkanTransaksi(string[] jenis, double[] nominal, string[]
keterangan, int jumlah)
{
    Console.ForegroundColor = ConsoleColor.Cyan;
    Console.WriteLine("\n==== RIWAYAT TRANSAKSI ====");
    Console.ResetColor();

    if (jumlah == 0) // Jika belum ada transaksi
    {
        Console.WriteLine("Belum ada transaksi yang dicatat.");
        return;
    }

    // Loop untuk menampilkan semua transaksi
    for (int i = 0; i < jumlah; i++)
    {
        Console.WriteLine($"{i + 1}. [{jenis[i]}] Rp {nominal[i]} - 
{keterangan[i]}");
    }
}

static void Main(string[] args)
{
    // Array sebagai penyimpanan data transaksi
    string[] jenisTransaksi = new string[100];
    double[] nominalTransaksi = new double[100];
    string[] keterangan = new string[100];
    int jumlahTransaksi = 0;

    int menu;

    // Tampilan awal aplikasi
    Console.ForegroundColor = ConsoleColor.Blue;

    Console.WriteLine("=====");
    Console.WriteLine(" WELCOME TO SMARTFINANCE TRACKER
");

    Console.WriteLine("=====");
    Console.ResetColor();

    // Perulangan utama menu program
    do
    {
        // Menampilkan menu
        Console.WriteLine("\n==== MENU UTAMA ===");
        Console.WriteLine("1. Tambah Transaksi");
        Console.WriteLine("2. Lihat Riwayat Transaksi");
        Console.WriteLine("3. Lihat Ringasan Saldo");

```

```

Console.WriteLine("4. Keluar");
Console.Write("Pilih menu (1-4): ");

// Validasi input menu
if (!int.TryParse(Console.ReadLine(), out menu))
{
    Console.WriteLine("Input tidak valid!");
    continue;
}

switch (menu)
{
    case 1:
        // Input data transaksi
        Console.ForegroundColor = ConsoleColor.Green;
        Console.WriteLine("\n==== TAMBAH");
        Console.ResetColor();

        Console.Write("Jenis (Pemasukan/Pengeluaran): ");
        jenisTransaksi[jumlahTransaksi] =
        Console.ReadLine();

        Console.Write("Nominal (Rp): ");
        nominalTransaksi[jumlahTransaksi] =
        double.Parse(Console.ReadLine());

        Console.WriteLine("Keterangan: ");
        keterangan[jumlahTransaksi] =
        Console.ReadLine();

        jumlahTransaksi++; // Menambah jumlah data
        yang tersimpan

        Loading("Menyimpan data");
        Console.ForegroundColor = ConsoleColor.Green;
        Console.WriteLine("Transaksi berhasil
        disimpan!");
        Console.ResetColor();
        break;

    case 2:
        // Menampilkan riwayat transaksi
        TampilanTransaksi(jenisTransaksi,
        nominalTransaksi, keterangan, jumlahTransaksi);
        break;
}

```

```

        case 3:
            // Menampilkan ringkasan saldo
            TampilanRingkasan(nominalTransaksi,
                jenisTransaksi, jumlahTransaksi);
            break;

        case 4:
            // Keluar dari program
            Console.ForegroundColor =
                ConsoleColor.Yellow;
            Loading("\nMenutup aplikasi");
            Console.WriteLine("Terima kasih telah
                menggunakan SmartFinance!");
            Console.ResetColor();
            break;

        default:
            Console.WriteLine("Pilihan tidak valid!");
            break;
    }

} while (menu != 4); // Program berjalan terus sampai memilih keluar
}
}
}

```

### Penjelasan Kode Program

1. Deklarasi Namespace dan Class  
using System;

```

namespace Projek_Akhir_SmartFinance_vinasukasih_xpplg1
{
    internal class Program
}

```

- using System; → digunakan agar kita bisa memakai perintah dasar C# seperti Console.WriteLine() dan Console.ReadLine().
- namespace Projek\_Akhir\_SmartFinance → ruang lingkup (wadah) program agar tidak bentrok dengan program lain.
- class Program → kelas utama tempat kode program dijalankan.

2. Method Loading()

```
static void Loading(string teks)
```

fungsi ini menampilkan animasi loading dengan menambahkan titik tiga secara bertahap.

Fungsinya:

- Agar tampilan lebih menarik dan tidak terlalu kaku.
- Memberikan efek seolah-olah sedang diproses.

### 3. Method TampilkanRingkasan()

```
static void TampilkanRingkasan(double[] transaksi, string[] jenis, int jumlah)
```

- `transaksi[]` → Array yang menyimpan nominal uang.
- `jenis[]` → Array menyimpan jenis transaksi (pemasukan/pengeluaran).
- `jumlah` → jumlah data transaksi yang tersimpan.

Fungsi:

- Menghitung total pemasukan
- Menghitung total pengeluaran
- Menampilkan saldo akhir

### 4. Method TampilkanTransaksi()

```
static void TampilkanRingkasan(double[] transaksi, string[] jenis, int jumlah)
```

Fungsinya: Menampilkan daftar transaksi yang tersimpan, dan Menampilkan nomor urut, jenis, nominal, dan keterangan.

Parameter:

- `jenis[]` → menyimpan tulisan “Pemasukan” / “Pengeluaran”
- `nominal[]` → menyimpan angka uang
- `keterangan[]` → catatan transaksi
- `jumlah` → berapa data yang sudah dimasukan

### 5. Method Main() – Inti Program

#### 5.1 Deklarasi Variabel dan Array

```
string[] jenisTransaksi = new string[100];
double[] nominalTransaksi = new double[100];
string[] keterangan = new string[100];
int jumlahTransaksi = 0;
int menu;
```

Penjelasan:

- `jenisTransaksi[]` menggunakan tipe `string` – Untuk menyimpan jenis transaksi (“pemasukan” / “pengeluaran”).
- `nominalTransaksi[]` menggunakan tipe `double` – Untuk menyimpan nilai uang.
- `keterangan[]` menggunakan `string` – Untuk menyimpan deskripsi transaksi.
- `jumlahTransaksi[]` menggunakan tipe `int` – Untuk penghitung data, menentukan index array
- `menu` menggunakan tipe `int` – untuk menyimpan pilihan menu dari user.

#### 5.2 Tampilan Awal

```
Console.WriteLine("WELCOME TO SMARTFINANCE TRACKER");
```

- Muncul hanya 1x di awal program

- Ditampilkan dalam warna untuk memperindah tampilan

```
5.3 Perulangan Menu Utama
do
{
    ...
} while (menu != 4);
```

- ❖ Untuk menjaga agar program terus berjalan sampai user memilih keluar

#### 5.4 Validasi Input Menu

```
if (!int.TryParse(Console.ReadLine(), out menu))
{
    Console.WriteLine("Input tidak valid!");
    continue;
}
```

- ❖ Untuk mencegah error jika input bukan angka.
- ❖ Untuk membuat program lebih aman.

#### 5.5 Switch Menu

```
switch(menu)
```

Penjelasan:

Case    Fungsi

1. Tambah transaksi
  2. Lihat riwayat transaksi
  3. Tampilkan ringkasan saldo
  4. Keluar program
- default Menangani input salah

#### 5.6 Penjelasan Tambahan Transaksi

```
Console.Write("Jenis:");
jenisTransaksi[jumlahTransaksi] = Console.ReadLine();
```

- ❖ Proses:

- Data disimpan ke array sesuai urutan
- Index ditentukan oleh jumlahTransaksi
- Setelah input selesai → index ditambah

Tampilan Output Hasil Program

```
WELCOME TO SMARTFINANCE TRACKER
=====
 
==== MENU UTAMA ====
1. Tambah Transaksi
2. Lihat Riwayat Transaksi
3. Lihat Ringkasan Saldo
4. Keluar
Pilih menu (1-4): 1

==== TAMBAH TRANSAKSI ====
Jenis (Pemasukan/Pengeluaran): Pemasukan
Nominal (Rp): 1.500.000
Keterangan: gaji bulan november
Menyimpan data...
Transaksi berhasil disimpan!

==== MENU UTAMA ====
1. Tambah Transaksi
2. Lihat Riwayat Transaksi
3. Lihat Ringkasan Saldo
4. Keluar
Pilih menu (1-4): 1

==== TAMBAH TRANSAKSI ====
Jenis (Pemasukan/Pengeluaran): Pengeluaran
Nominal (Rp): 245.000
Keterangan: untuk membeli beras dan buah
Menyimpan data...
Transaksi berhasil disimpan!

==== MENU UTAMA ====
1. Tambah Transaksi
2. Lihat Riwayat Transaksi
3. Lihat Ringkasan Saldo
4. Keluar
Pilih menu (1-4): 3

==== RINGKASAN KEUANGAN ====
Total Pemasukan : Rp 1500000
Total Pengeluaran : Rp 245000
Saldo Akhir : Rp 1255000

==== MENU UTAMA ====
1. Tambah Transaksi
2. Lihat Riwayat Transaksi
3. Lihat Ringkasan Saldo
4. Keluar
Pilih menu (1-4): 4

Menutup aplikasi...
Terima kasih telah menggunakan SmartFinance!
```

## LAMPIRAN

### Kode Program Lengkap

```
8     namespace Projek_Akhir_SmartFinance_vinasukasih_xppgl1
9     {
10        0 references
11        internal class Program
12        {
13            // Fungsi untuk menampilkan teks dengan efek loading
14            2 references
15            static void Loading(string teks)
16            {
17                Console.WriteLine(teks);
18                for (int i = 0; i < 3; i++)
19                {
20                    Console.Write(".");
21                    Thread.Sleep(300);
22                }
23            }
24
25            // Fungsi untuk menampilkan ringkasan pemasukan, pengeluaran, dan saldo
26            1 reference
27            static void TampilkanRingkasan(double[] transaksi, string[] jenis, int jumlah)
28            {
29                double totalPemasukan = 0, totalPengeluaran = 0;
30
31                // Menghitung total pemasukan & total total pengeluaran
32                for (int i = 0; i < jumlah; i++)
33                {
34                    if (jenis[i].ToLower() == "pemasukan")
35                        totalPemasukan += transaksi[i];
36
37                    else if (jenis[i].ToLower() == "pengeluaran")
38                        totalPengeluaran += transaksi[i];
39
40                double saldoAkhir = totalPemasukan - totalPengeluaran;
41
42                // Menampilkan hasil ringkasan
43                Console.ForegroundColor = ConsoleColor.Yellow;
44                Console.WriteLine("\n==== RINGKASAN KEUANGAN ===");
45                Console.ResetColor();
46                Console.WriteLine($"Total Pemasukan : Rp {totalPemasukan}");
47                Console.WriteLine($"Total Pengeluaran : Rp {totalPengeluaran}");
48                Console.WriteLine($"Saldo Akhir : Rp {saldoAkhir}");
49
50                // Fungsi untuk menampilkan semua transaksi
51                1 reference
52                static void TampilkanTransaksi(string[] jenis, double[] nominal, string[] keterangan, int jumlah)
53                {
54                    Console.ForegroundColor = ConsoleColor.Cyan;
55                    Console.WriteLine("\n==== RIWAYAT TRANSAKSI ===");
56                    Console.ResetColor();
57
58                    if (jumlah == 0) // Jika belum ada transaksi
59                    {
60                        Console.WriteLine("Belum ada transaksi yang dicatat.");
61                        return;
62                    }
63
64                    // Loop untuk menampilkan semua transaksi
```

```

64      // Loop menulis setiap transaksi ke layar
65      for (int i = 0; i < jumlah; i++)
66      {
67          Console.WriteLine($"{i + 1}. [{jenis[i]}] Rp {nominal[i]} - {keterangan[i]}");
68      }
69  }
70
71  static void Main(string[] args)
72  {
73      // Array sebagai penyimpanan data transaksi
74      string[] jenisTransaksi = new string[100];
75      double[] nominalTransaksi = new double[100];
76      string[] keterangan = new string[100];
77      int jumlahTransaksi = 0;
78
79      int menu;
80
81      // Tampilan awal aplikasi
82      Console.ForegroundColor = ConsoleColor.Blue;
83      Console.WriteLine("===== WELCOME TO SMARTFINANCE TRACKER =====");
84      Console.ResetColor();
85
86      // Perulangan utama menu program
87      do
88      {
89          // Menampilkan menu
90          Console.WriteLine("\n==== MENU UTAMA ====");
91          Console.WriteLine("1. Tambah Transaksi");
92
93          Console.WriteLine("2. Lihat Riwayat Transaksi");
94          Console.WriteLine("3. Lihat Ringkasan Saldo");
95          Console.WriteLine("4. Keluar");
96          Console.Write("Pilih menu (1-4): ");
97
98          // Validasi input menu
99          if (!int.TryParse(Console.ReadLine(), out menu))
100         {
101             Console.WriteLine("Input tidak valid!");
102             continue;
103         }
104
105         switch (menu)
106         {
107             case 1:
108                 // Input data transaksi
109                 Console.ForegroundColor = ConsoleColor.Green;
110                 Console.WriteLine("\n==== TAMBAH TRANSAKSI ====");
111                 Console.ResetColor();
112
113                 Console.Write("Jenis (Pemasukan/Pengeluaran): ");
114                 jenisTransaksi[jumlahTransaksi] = Console.ReadLine();
115
116                 Console.Write("Nominal (Rp): ");
117                 nominalTransaksi[jumlahTransaksi] = double.Parse(Console.ReadLine());
118
119                 Console.Write("Keterangan: ");
120                 keterangan[jumlahTransaksi] = Console.ReadLine();
121
122                 jumlahTransaksi++; // Menambah jumlah data yang tersimpan

```

```
124     Loading("Menyimpan data");
125     Console.ForegroundColor = ConsoleColor.Green;
126     Console.WriteLine("Transaksi berhasil disimpan!");
127     Console.ResetColor();
128     break;
129
130 case 2:
131     // Menampilkan riwayat transaksi
132     TampilkanTransaksi(jenisTransaksi, nominalTransaksi, keterangan, jumlahTransaksi);
133     break;
134
135 case 3:
136     // Menampilkan ringkasan saldo
137     TampilkanRingkasan(nominalTransaksi, jenisTransaksi, jumlahTransaksi);
138     break;
139
140 case 4:
141     // Keluar dari program
142     ConsoleColor ConsoleColor = ConsoleColor.Yellow;
143     Loading("\nMenutup aplikasi");
144     Console.WriteLine("Terima kasih telah menggunakan SmartFinance!");
145     Console.ResetColor();
146     break;
147
148 default:
149     Console.WriteLine("Pilihan tidak valid!");
150     break;
151 }
152 } while (menu != 4); // Program berjalan terus sampai memilih keluar
153
154 }
155 }
```

ScreenShot Tampilan Program

```
WELCOME TO SMARTFINANCE TRACKER
=====
 
==== MENU UTAMA ====
1. Tambah Transaksi
2. Lihat Riwayat Transaksi
3. Lihat Ringkasan Saldo
4. Keluar
Pilih menu (1-4): 1

==== TAMBAH TRANSAKSI ====
Jenis (Pemasukan/Pengeluaran): Pemasukan
Nominal (Rp): 1.500.000
Keterangan: gaji bulan november
Menyimpan data...
Transaksi berhasil disimpan!

==== MENU UTAMA ====
1. Tambah Transaksi
2. Lihat Riwayat Transaksi
3. Lihat Ringkasan Saldo
4. Keluar
Pilih menu (1-4): 1

==== TAMBAH TRANSAKSI ====
Jenis (Pemasukan/Pengeluaran): Pengeluaran
Nominal (Rp): 245.000
Keterangan: untuk membeli beras dan buah
Menyimpan data...
Transaksi berhasil disimpan!

==== MENU UTAMA ====
1. Tambah Transaksi
2. Lihat Riwayat Transaksi
3. Lihat Ringkasan Saldo
4. Keluar
Pilih menu (1-4): 3

==== RINGKASAN KEUANGAN ====
Total Pemasukan : Rp 1500000
Total Pengeluaran : Rp 245000
Saldo Akhir : Rp 1255000

==== MENU UTAMA ====
1. Tambah Transaksi
2. Lihat Riwayat Transaksi
3. Lihat Ringkasan Saldo
4. Keluar
Pilih menu (1-4): 4

Menutup aplikasi...
Terima kasih telah menggunakan SmartFinance!
```